

# Package ‘whitebox’

March 29, 2023

**Type** Package

**Title** 'WhiteboxTools' R Frontend

**Version** 2.3.0

**Description** An R frontend for the 'WhiteboxTools' library, which is an advanced geospatial data analysis platform developed by Prof. John Lindsay at the University of Guelph's Geomorphometry and Hydrogeomatics Research Group. 'WhiteboxTools' can be used to perform common geographical information systems (GIS) analysis operations, such as cost-distance analysis, distance buffering, and raster reclassification. Remote sensing and image processing tasks include image enhancement (e.g. panchromatic sharpening, contrast adjustments), image mosaicing, numerous filtering operations, simple classification (k-means), and common image transformations. 'WhiteboxTools' also contains advanced tooling for spatial hydrological analysis (e.g. flow-accumulation, watershed delineation, stream network analysis, sink removal), terrain analysis (e.g. common terrain indices such as slope, curvatures, wetness index, hillshading; hypsometric analysis; multi-scale topographic position analysis), and LiDAR data processing. Suggested citation: Lindsay (2016) <[doi:10.1016/j.cageo.2016.07.003](https://doi.org/10.1016/j.cageo.2016.07.003)>.

**Maintainer** Andrew Brown <[brown.andrewg@gmail.com](mailto:brown.andrewg@gmail.com)>

**License** MIT + file LICENSE

**SystemRequirements** WhiteboxTools  
(<https://github.com/jblindsey/whitebox-tools/releases/latest>)

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.2.3

**URL** <https://github.com/giswqs/whiteboxR>

**BugReports** <https://github.com/giswqs/whiteboxR/issues>

**Suggests** knitr, rmarkdown, testthat, terra, sf, raster

**VignetteBuilder** knitr

**Depends** R (>= 3.0.0)

**LazyData** true

**NeedsCompilation** no

**Author** Qiusheng Wu [aut],  
Andrew Brown [ctb, cre]

Repository CRAN

Date/Publication 2023-03-29 13:30:02 UTC

## R topics documented:

check_whitebox_binary . . . . .	14
sample_dem_data . . . . .	14
wbt . . . . .	15
wbttoolparameters . . . . .	17
wbttools . . . . .	18
wbt_absolute_value . . . . .	19
wbt_accumulation_curvature . . . . .	20
wbt_activate . . . . .	21
wbt_adaptive_filter . . . . .	21
wbt_add . . . . .	22
wbt_add_point_coordinates_to_table . . . . .	23
wbt_aggregate_raster . . . . .	24
wbt_and . . . . .	25
wbt_anova . . . . .	26
wbt_arcosh . . . . .	27
wbt_arc_cos . . . . .	27
wbt_arc_sin . . . . .	28
wbt_arc_tan . . . . .	29
wbt_arsinh . . . . .	30
wbt_artanh . . . . .	31
wbt_ascii_to_las . . . . .	31
wbt_aspect . . . . .	32
wbt_assess_route . . . . .	33
wbt_atan2 . . . . .	34
wbt_attribute_correlation . . . . .	35
wbt_attribute_correlation_neighbourhood_analysis . . . . .	35
wbt_attribute_histogram . . . . .	36
wbt_attribute_scattergram . . . . .	37
wbt_average_flowpath_slope . . . . .	38
wbt_average_normal_vector_angular_deviation . . . . .	39
wbt_average_overlay . . . . .	40
wbt_average_upslope_flowpath_length . . . . .	41
wbt_balance_contrast_enhancement . . . . .	41
wbt_basins . . . . .	42
wbt_bilateral_filter . . . . .	43
wbt_block_maximum_gridding . . . . .	44
wbt_block_minimum_gridding . . . . .	45
wbt_boundary_shape_complexity . . . . .	46
wbt_breach_depressions . . . . .	47
wbt_breach_depressions_least_cost . . . . .	48
wbt_breach_single_cell_pits . . . . .	49
wbt_breakline_mapping . . . . .	49

wbt_buffer_raster . . . . .	50
wbt_burn_streams_at_roads . . . . .	51
wbt_canny_edge_detection . . . . .	52
wbt_ceil . . . . .	53
wbt_centroid . . . . .	54
wbt_centroid_vector . . . . .	55
wbt_change_vector_analysis . . . . .	56
wbt_circular_variance_of_aspect . . . . .	57
wbt_classify_buildings_in_lidar . . . . .	58
wbt_classify_lidar . . . . .	59
wbt_classify_overlap_points . . . . .	60
wbt_clean_vector . . . . .	61
wbt_clip . . . . .	61
wbt_clip_lidar_to_polygon . . . . .	62
wbt_clip_raster_to_polygon . . . . .	63
wbt_closing . . . . .	64
wbt_clump . . . . .	65
wbt_colourize_based_on_class . . . . .	66
wbt_colourize_based_on_point_returns . . . . .	67
wbt_compactness_ratio . . . . .	68
wbt_conditional_evaluation . . . . .	68
wbt_conditioned_latin_hypercube . . . . .	69
wbt_conservative_smoothing_filter . . . . .	71
wbt_construct_vector_tin . . . . .	72
wbt_contours_from_points . . . . .	73
wbt_contours_from_raster . . . . .	74
wbt_convert_nodata_to_zero . . . . .	75
wbt_convert_raster_format . . . . .	75
wbt_corner_detection . . . . .	76
wbt_correct_vignetting . . . . .	77
wbt_cos . . . . .	78
wbt_cosh . . . . .	79
wbt_cost_allocation . . . . .	79
wbt_cost_distance . . . . .	80
wbt_cost_pathway . . . . .	81
wbt_count_if . . . . .	82
wbt_create_colour_composite . . . . .	83
wbt_create_hexagonal_vector_grid . . . . .	84
wbt_create_plane . . . . .	85
wbt_create_rectangular_vector_grid . . . . .	86
wbt_crispness_index . . . . .	87
wbt_cross_tabulation . . . . .	87
wbt_csv_points_to_vector . . . . .	88
wbt_cumulative_distribution . . . . .	89
wbt_curvedness . . . . .	90
wbt_d8_flow_accumulation . . . . .	91
wbt_d8_mass_flux . . . . .	92
wbt_d8_pointer . . . . .	93

wbt_dbscan . . . . .	94
wbt_decrement . . . . .	95
wbt_dem_void_filling . . . . .	95
wbt_depth_in_sink . . . . .	97
wbt_depth_to_water . . . . .	98
wbt_dev_from_mean_elev . . . . .	99
wbt_difference . . . . .	100
wbt_difference_curvature . . . . .	101
wbt_diff_from_mean_elev . . . . .	102
wbt_diff_of_gaussian_filter . . . . .	103
wbt_directional_relief . . . . .	104
wbt_direct_decorrelation_stretch . . . . .	105
wbt_dissolve . . . . .	106
wbt_distance_to_outlet . . . . .	107
wbt_diversity_filter . . . . .	108
wbt_divide . . . . .	109
wbt_downslope_distance_to_stream . . . . .	110
wbt_downslope_flowpath_length . . . . .	111
wbt_downslope_index . . . . .	112
wbt_d_inf_flow_accumulation . . . . .	113
wbt_d_inf_mass_flux . . . . .	114
wbt_d_inf_pointer . . . . .	115
wbt_edge_contamination . . . . .	115
wbt_edge_density . . . . .	116
wbt_edge_preserving_mean_filter . . . . .	117
wbt_edge_proportion . . . . .	118
wbt_elevation_above_stream . . . . .	119
wbt_elevation_above_stream_euclidean . . . . .	120
wbt_elev_above_pit . . . . .	121
wbt_elev_percentile . . . . .	121
wbt_elev_relative_to_min_max . . . . .	122
wbt_elev_relative_to_watershed_min_max . . . . .	123
wbt_eliminate_coincident_points . . . . .	124
wbt_elongation_ratio . . . . .	125
wbt_embankment_mapping . . . . .	126
wbt_emboss_filter . . . . .	127
wbt_equal_to . . . . .	128
wbt_erase . . . . .	129
wbt_erase_polygon_from_lidar . . . . .	130
wbt_erase_polygon_from_raster . . . . .	131
wbt_euclidean_allocation . . . . .	132
wbt_euclidean_distance . . . . .	132
wbt_evaluate_training_sites . . . . .	133
wbt_exp . . . . .	134
wbt_exp2 . . . . .	135
wbt_export_table_to_csv . . . . .	136
wbt_exposure_towards_wind_flux . . . . .	137
wbt_extend_vector_lines . . . . .	138

wbt_extract_nodes . . . . .	139
wbt_extract_raster_values_at_points . . . . .	139
wbt_extract_streams . . . . .	140
wbt_extract_valleys . . . . .	141
wbt_farthest_channel_head . . . . .	142
wbt_fast_almost_gaussian_filter . . . . .	143
wbt_fd8_flow_accumulation . . . . .	144
wbt_fd8_pointer . . . . .	145
wbt_feature_preserving_smoothing . . . . .	146
wbt_fetch_analysis . . . . .	147
wbt_fill_burn . . . . .	148
wbt_fill_depressions . . . . .	149
wbt_fill_depressions_planchon_and_darboux . . . . .	150
wbt_fill_depressions_wang_and_liu . . . . .	151
wbt_fill_missing_data . . . . .	152
wbt_fill_single_cell_pits . . . . .	153
wbt_filter_lidar . . . . .	153
wbt_filter_lidar_classes . . . . .	154
wbt_filter_lidar_scan_angles . . . . .	155
wbt_filter_raster_features_by_area . . . . .	156
wbt_find_flightline_edge_points . . . . .	157
wbt_find_lowest_or_highest_points . . . . .	158
wbt_find_main_stem . . . . .	159
wbt_find_no_flow_cells . . . . .	160
wbt_find_parallel_flow . . . . .	160
wbt_find_patch_or_class_edge_cells . . . . .	161
wbt_find_ridges . . . . .	162
wbt_fix_dangling_arcs . . . . .	163
wbt_flatten_lakes . . . . .	164
wbt_flightline_overlap . . . . .	165
wbt_flip_image . . . . .	166
wbt_flood_order . . . . .	167
wbt_floor . . . . .	167
wbt_flow_accumulation_full_workflow . . . . .	168
wbt_flow_length_diff . . . . .	169
wbt_gamma_correction . . . . .	170
wbt_gaussian_contrast_stretch . . . . .	171
wbt_gaussian_curvature . . . . .	172
wbt_gaussian_filter . . . . .	173
wbt_gaussian_scale_space . . . . .	174
wbt_generalize_classified_raster . . . . .	175
wbt_generalize_with_similarity . . . . .	176
wbt_generating_function . . . . .	177
wbt_geomorphons . . . . .	178
wbt_greater_than . . . . .	179
wbt_hack_stream_order . . . . .	180
wbt_heat_map . . . . .	181
wbt_height_above_ground . . . . .	182

wbt_help	182
wbt_highest_position	183
wbt_high_pass_bilateral_filter	184
wbt_high_pass_filter	185
wbt_high_pass_median_filter	186
wbt_hillshade	187
wbt_hillslopes	188
wbt_histogram_equalization	189
wbt_histogram_matching	190
wbt_histogram_matching_two_images	191
wbt_hole_proportion	192
wbt_horizontal_excess_curvature	192
wbt_horizon_angle	193
wbt_horton_stream_order	194
wbt_hydrologic_connectivity	195
wbt_hypsometrically_tinted_hillshade	196
wbt_hypsometric_analysis	197
wbt_idw_interpolation	198
wbt_ihs_to_rgb	199
wbt_image_autocorrelation	200
wbt_image_correlation	201
wbt_image_correlation_neighbourhood_analysis	202
wbt_image_regression	203
wbt_image_segmentation	204
wbt_image_slider	205
wbt_image_stack_profile	206
wbt_impoundment_size_index	207
wbt_increment	208
wbt_individual_tree_detection	208
wbt_init	210
wbt_insert_dams	213
wbt_install	214
wbt_install_wb_extension	215
wbt_integer_division	216
wbt_integral_image	217
wbt_intersect	217
wbt_inverse_principal_component_analysis	218
wbt_in_place_add	219
wbt_in_place_divide	220
wbt_in_place_multiply	221
wbt_in_place_subtract	221
wbt_isobasins	222
wbt_is_no_data	223
wbt_jenson_snap_pour_points	224
wbt_join_tables	225
wbt_kappa_index	226
wbt_knn_classification	227
wbt_knn_regression	228

wbt_ks_test_for_normality . . . . .	229
wbt_k_means_clustering . . . . .	230
wbt_k_nearest_mean_filter . . . . .	231
wbt_laplacian_filter . . . . .	232
wbt_laplacian_of_gaussian_filter . . . . .	233
wbt_las_to_ascii . . . . .	234
wbt_las_to_laz . . . . .	234
wbt_las_to_multipoint_shapefile . . . . .	235
wbt_las_to_shapefile . . . . .	236
wbt_las_to_zlidar . . . . .	237
wbt_launch_wb_runner . . . . .	238
wbt_layer_footprint . . . . .	238
wbt_laz_to_las . . . . .	239
wbt_lee_sigma_filter . . . . .	240
wbt_length_of_upstream_channels . . . . .	241
wbt_less_than . . . . .	242
wbt_license . . . . .	243
wbt_lidar_block_maximum . . . . .	243
wbt_lidar_block_minimum . . . . .	244
wbt_lidar_classify_subset . . . . .	245
wbt_lidar_colourize . . . . .	246
wbt_lidar_contour . . . . .	247
wbt_lidar_digital_surface_model . . . . .	248
wbt_lidar_eigenvalue_features . . . . .	249
wbt_lidar_elevation_slice . . . . .	250
wbt_lidar_ground_point_filter . . . . .	251
wbt_lidar_hex_binning . . . . .	252
wbt_lidar_hillshade . . . . .	253
wbt_lidar_histogram . . . . .	254
wbt_lidar_idw_interpolation . . . . .	255
wbt_lidar_info . . . . .	256
wbt_lidar_join . . . . .	257
wbt_lidar_kappa_index . . . . .	258
wbt_lidar_nearest_neighbour_gridding . . . . .	259
wbt_lidar_point_density . . . . .	260
wbt_lidar_point_return_analysis . . . . .	261
wbt_lidar_point_stats . . . . .	262
wbt_lidar_ransac_planes . . . . .	263
wbt_lidar_rbf_interpolation . . . . .	264
wbt_lidar_remove_duplicates . . . . .	265
wbt_lidar_remove_outliers . . . . .	266
wbt_lidar_rooftop_analysis . . . . .	267
wbt_lidar_segmentation . . . . .	269
wbt_lidar_segmentation_based_filter . . . . .	270
wbt_lidar_shift . . . . .	271
wbt_lidar_sibson_interpolation . . . . .	272
wbt_lidar_sort_by_time . . . . .	273
wbt_lidar_thin . . . . .	274

wbt_lidar_thin_high_density . . . . .	275
wbt_lidar_tile . . . . .	276
wbt_lidar_tile_footprint . . . . .	277
wbt_lidar_tin_gridding . . . . .	278
wbt_lidar_tophat_transform . . . . .	279
wbt_linearity_index . . . . .	280
wbt_lines_to_polygons . . . . .	280
wbt_line_detection_filter . . . . .	281
wbt_line_intersections . . . . .	282
wbt_line_thinning . . . . .	283
wbt_list_tools . . . . .	284
wbt_list_unique_values . . . . .	284
wbt_list_unique_values_raster . . . . .	285
wbt_ln . . . . .	286
wbt_local_hypsometric_analysis . . . . .	287
wbt_local_quadratic_regression . . . . .	288
wbt_log10 . . . . .	289
wbt_log2 . . . . .	289
wbt_logistic_regression . . . . .	290
wbt_longest_flowpath . . . . .	291
wbt_long_profile . . . . .	292
wbt_long_profile_from_points . . . . .	293
wbt_lowest_position . . . . .	294
wbt_low_points_on_headwater_divides . . . . .	295
wbt_majority_filter . . . . .	296
wbt_map_off_terrain_objects . . . . .	297
wbt_max . . . . .	298
wbt_maximal_curvature . . . . .	299
wbt_maximum_filter . . . . .	300
wbt_max_absolute_overlay . . . . .	301
wbt_max_anisotropy_dev . . . . .	301
wbt_max_anisotropy_dev_signature . . . . .	302
wbt_max_branch_length . . . . .	303
wbt_max_difference_from_mean . . . . .	304
wbt_max_downslope_elev_change . . . . .	305
wbt_max_elevation_deviation . . . . .	306
wbt_max_elev_dev_signature . . . . .	307
wbt_max_overlay . . . . .	308
wbt_max_upslope_elev_change . . . . .	309
wbt_max_upslope_flowpath_length . . . . .	309
wbt_max_upslope_value . . . . .	310
wbt_md_inf_flow_accumulation . . . . .	311
wbt_mean_curvature . . . . .	312
wbt_mean_filter . . . . .	313
wbt_median_filter . . . . .	314
wbt_medoid . . . . .	315
wbt_merge_line_segments . . . . .	316
wbt_merge_table_with_csv . . . . .	317



wbt_merge_vectors . . . . .	318
wbt_min . . . . .	318
wbt_minimal_curvature . . . . .	319
wbt_minimum_bounding_box . . . . .	320
wbt_minimum_bounding_circle . . . . .	321
wbt_minimum_bounding_envelope . . . . .	322
wbt_minimum_convex_hull . . . . .	323
wbt_minimum_filter . . . . .	324
wbt_min_absolute_overlay . . . . .	325
wbt_min_dist_classification . . . . .	325
wbt_min_downslope_elev_change . . . . .	326
wbt_min_max_contrast_stretch . . . . .	327
wbt_min_overlay . . . . .	328
wbt_modified_k_means_clustering . . . . .	329
wbt_modify_lidar . . . . .	330
wbt_modify_no_data_value . . . . .	331
wbt_modulo . . . . .	331
wbt_mosaic . . . . .	332
wbt_mosaic_with_feathering . . . . .	333
wbt_multidirectional_hillshade . . . . .	334
wbt_multiply . . . . .	335
wbt_multiply_overlay . . . . .	336
wbt_multiscale_curvatures . . . . .	337
wbt_multiscale_elevation_percentile . . . . .	338
wbt_multiscale_roughness . . . . .	339
wbt_multiscale_roughness_signature . . . . .	340
wbt_multiscale_std_dev_normals . . . . .	341
wbt_multiscale_std_dev_normals_signature . . . . .	342
wbt_multiscale_topographic_position_image . . . . .	343
wbt_multi_part_to_single_part . . . . .	344
wbt_narrowness_index . . . . .	345
wbt_natural_neighbour_interpolation . . . . .	346
wbt_nearest_neighbour_gridding . . . . .	347
wbt_negate . . . . .	348
wbt_new_raster_from_base . . . . .	348
wbt_normalized_difference_index . . . . .	349
wbt_normalize_lidar . . . . .	350
wbt_normal_vectors . . . . .	351
wbt_not . . . . .	352
wbt_not_equal_to . . . . .	353
wbt_num_downslope_neighbours . . . . .	354
wbt_num_inflowing_neighbours . . . . .	354
wbt_num_upslope_neighbours . . . . .	355
wbt_olympic_filter . . . . .	356
wbt_opening . . . . .	357
wbt_openness . . . . .	358
wbt_or . . . . .	359
wbt_paired_sample_t_test . . . . .	360

wbt_panchromatic_sharpening . . . . .	361
wbt_parallelepiped_classification . . . . .	362
wbt_patch_orientation . . . . .	363
wbt_pennock_landform_class . . . . .	363
wbt_percentage_contrast_stretch . . . . .	364
wbt_percentile_filter . . . . .	365
wbt_percent_elev_range . . . . .	366
wbt_percent_equal_to . . . . .	367
wbt_percent_greater_than . . . . .	368
wbt_percent_less_than . . . . .	369
wbt_perimeter_area_ratio . . . . .	370
wbt_phi_coefficient . . . . .	371
wbt_pick_from_list . . . . .	372
wbt_piecewise_contrast_stretch . . . . .	373
wbt_plan_curvature . . . . .	374
wbt_polygonize . . . . .	375
wbt_polygons_to_lines . . . . .	375
wbt_polygon_area . . . . .	376
wbt_polygon_long_axis . . . . .	377
wbt_polygon_perimeter . . . . .	378
wbt_polygon_short_axis . . . . .	378
wbt_power . . . . .	379
wbt_prewitt_filter . . . . .	380
wbt_principal_component_analysis . . . . .	381
wbt_print_geo_tiff_tags . . . . .	382
wbt_profile . . . . .	382
wbt_profile_curvature . . . . .	383
wbt_qin_flow_accumulation . . . . .	384
wbt_quantiles . . . . .	385
wbt_quinn_flow_accumulation . . . . .	386
wbt_radial_basis_function_interpolation . . . . .	387
wbt_radius_of_gyration . . . . .	388
wbt_raise_walls . . . . .	389
wbt_random_field . . . . .	390
wbt_random_forest_classification . . . . .	391
wbt_random_forest_regression . . . . .	392
wbt_random_sample . . . . .	393
wbt_range_filter . . . . .	394
wbt_rasterize_streams . . . . .	395
wbt_raster_area . . . . .	396
wbt_raster_calculator . . . . .	397
wbt_raster_cell_assignment . . . . .	398
wbt_raster_histogram . . . . .	399
wbt_raster_perimeter . . . . .	400
wbt_raster_streams_to_vector . . . . .	401
wbt_raster_summary_stats . . . . .	402
wbt_raster_to_vector_lines . . . . .	402
wbt_raster_to_vector_points . . . . .	403

wbt_raster_to_vector_polygons . . . . .	404
wbt_reciprocal . . . . .	405
wbt_reclass . . . . .	406
wbt_reclass_equal_interval . . . . .	407
wbt_reclass_from_file . . . . .	408
wbt_reconcile_multiple_headers . . . . .	409
wbt_recover_flightline_info . . . . .	410
wbt_recreate_pass_lines . . . . .	411
wbt_reinitialize_attribute_table . . . . .	412
wbt_related_circumscribing_circle . . . . .	412
wbt_relative_aspect . . . . .	413
wbt_relative_topographic_position . . . . .	414
wbt_remove_field_edge_points . . . . .	415
wbt_remove_off_terrain_objects . . . . .	416
wbt_remove_polygon_holes . . . . .	417
wbt_remove_raster_polygon_holes . . . . .	418
wbt_remove_short_streams . . . . .	419
wbt_remove_spurs . . . . .	420
wbt_repair_stream_vector_topology . . . . .	421
wbt_resample . . . . .	422
wbt_rescale_value_range . . . . .	423
wbt_rgb_to_ihs . . . . .	424
wbt_rho8_flow_accumulation . . . . .	425
wbt_rho8_pointer . . . . .	426
wbt_ring_curvature . . . . .	427
wbt_river_centerlines . . . . .	428
wbt_roberts_cross_filter . . . . .	429
wbt_root_mean_square_error . . . . .	430
wbt_rotor . . . . .	430
wbt_round . . . . .	431
wbt_ruggedness_index . . . . .	432
wbt_run_tool . . . . .	433
wbt_rust_backtrace . . . . .	434
wbt_scharr_filter . . . . .	434
wbt_sediment_transport_index . . . . .	435
wbt_select_tiles_by_polygon . . . . .	436
wbt_set_nodata_value . . . . .	437
wbt_shadow_animation . . . . .	438
wbt_shadow_image . . . . .	439
wbt_shape_complexity_index . . . . .	440
wbt_shape_complexity_index_raster . . . . .	441
wbt_shape_index . . . . .	441
wbt_shreve_stream_magnitude . . . . .	442
wbt_sigmoidal_contrast_stretch . . . . .	443
wbt_sin . . . . .	444
wbt_single_part_to_multi_part . . . . .	445
wbt_sinh . . . . .	446
wbt_sink . . . . .	446

wbt_slope . . . . .	447
wbt_slope_vs_aspect_plot . . . . .	448
wbt_slope_vs_elevation_plot . . . . .	449
wbt_smooth_vectors . . . . .	450
wbt_smooth_vegetation_residual . . . . .	451
wbt_snap_pour_points . . . . .	452
wbt_sobel_filter . . . . .	453
wbt_sort_lidar . . . . .	454
wbt_source . . . . .	455
wbt_spherical_std_dev_of_normals . . . . .	455
wbt_split_colour_composite . . . . .	456
wbt_split_lidar . . . . .	457
wbt_split_vector_lines . . . . .	458
wbt_split_with_lines . . . . .	459
wbt_square . . . . .	460
wbt_square_root . . . . .	460
wbt_standard_deviation_contrast_stretch . . . . .	461
wbt_standard_deviation_filter . . . . .	462
wbt_standard_deviation_of_slope . . . . .	463
wbt_stochastic_depression_analysis . . . . .	464
wbt_strahler_order_basins . . . . .	465
wbt_strahler_stream_order . . . . .	466
wbt_stream_link_class . . . . .	467
wbt_stream_link_identifier . . . . .	468
wbt_stream_link_length . . . . .	469
wbt_stream_link_slope . . . . .	470
wbt_stream_power_index . . . . .	471
wbt_stream_slope_continuous . . . . .	472
wbt_subbasins . . . . .	473
wbt_subtract . . . . .	474
wbt_sum_overlay . . . . .	475
wbt_surface_area_ratio . . . . .	475
wbt_svm_classification . . . . .	476
wbt_svm_regression . . . . .	477
wbt_symmetrical_difference . . . . .	479
wbt_tan . . . . .	480
wbt_tangential_curvature . . . . .	480
wbt_tanh . . . . .	481
wbt_thicken_raster_line . . . . .	482
wbt_time_in_daylight . . . . .	483
wbt_tin_gridding . . . . .	484
wbt_toolbox . . . . .	485
wbt_tool_help . . . . .	486
wbt_tool_parameters . . . . .	486
wbt_tophat_transform . . . . .	487
wbt_topographic_position_animation . . . . .	488
wbt_topological_stream_order . . . . .	489
wbt_topo_render . . . . .	490

wbt_total_curvature . . . . .	491
wbt_total_filter . . . . .	492
wbt_to_degrees . . . . .	493
wbt_to_radians . . . . .	494
wbt_trace_downslope_flowpaths . . . . .	495
wbt_travelling_salesman_problem . . . . .	496
wbt_trend_surface . . . . .	497
wbt_trend_surface_vector_points . . . . .	498
wbt_tributary_identifier . . . . .	499
wbt_truncate . . . . .	500
wbt_turning_bands_simulation . . . . .	501
wbt_two_sample_ks_test . . . . .	502
wbt_union . . . . .	503
wbt_unnest_basins . . . . .	504
wbt_unsharp_masking . . . . .	505
wbt_unsphericity . . . . .	506
wbt_update_nodata_cells . . . . .	507
wbt_upslope_depression_storage . . . . .	508
wbt_user_defined_weights_filter . . . . .	508
wbt_vector_hex_binning . . . . .	509
wbt_vector_lines_to_raster . . . . .	510
wbt_vector_points_to_raster . . . . .	511
wbt_vector_polygons_to_raster . . . . .	512
wbt_vector_stream_network_analysis . . . . .	513
wbt_version . . . . .	514
wbt_vertical_excess_curvature . . . . .	515
wbt_viewshed . . . . .	516
wbt_view_code . . . . .	517
wbt_visibility_index . . . . .	517
wbt_voronoi_diagram . . . . .	518
wbt_watershed . . . . .	519
wbt_weighted_overlay . . . . .	520
wbt_weighted_sum . . . . .	521
wbt_wetness_index . . . . .	522
wbt_wilcoxon_signed_rank_test . . . . .	523
wbt_write_function_memory_insertion . . . . .	524
wbt_xor . . . . .	525
wbt_yield_filter . . . . .	526
wbt_yield_map . . . . .	527
wbt_yield_normalization . . . . .	528
wbt_zlidar_to_las . . . . .	529
wbt_zonal_statistics . . . . .	529
wbt_z_scores . . . . .	530

---

check\_whitebox\_binary *Check for 'WhiteboxTools' executable path*

---

### Description

Check for 'WhiteboxTools' executable path

### Usage

```
check_whitebox_binary(silent = TRUE)
```

### Arguments

silent            logical. Print help on installation/setting path. Default TRUE.

### Value

logical if 'WhiteboxTools' executable file exists.

### See Also

[wbt\\_exe\\_path\(\)](#)

---

sample\_dem\_data            *Convenience method for path to sample DEM*

---

### Description

Get a file path to DEM.tif stored in extdata subfolder of whitebox package installation directory. If needed, download the TIFF file from GitHub.

### Usage

```
sample_dem_data(
  destfile = file.path(system.file("extdata", package = "whitebox"), "DEM.tif"),
  ...
)
```

### Arguments

destfile            Path to target location of sample data. Will be downloaded if does not exist. Defaults to file path of extdata subfolder of whitebox package installation directory.

...                 additional arguments to download.file()

**Value**

character.

**Examples**

```
if (check_whitebox_binary()) {
  wbt_slope(sample_dem_data(), output = "slope.tif")
}
unlink(c('slope.tif', 'settings.json'))
```

---

wbt

*Run WhiteboxTools by Tool Name*


---

**Description**

You are required to specify all required arguments as either paths to files, or R object types that can be associated with a file.

`wbt_result()`: return a combined list of results from either the history of a `wbt_result` (if present and `history=TRUE`), or the result of a `wbt_result`

**Usage**

```
wbt(
  result,
  tool_name,
  ...,
  crs = NULL,
  verbose_mode = FALSE,
  command_only = FALSE
)

## S3 method for class 'wbt_result'
wbt(
  result,
  tool_name,
  ...,
  crs = NULL,
  verbose_mode = FALSE,
  command_only = FALSE
)

wbt_result(result, i = NULL, history = TRUE, attribute = "output")

## S3 method for class 'character'
wbt(
  result,
```

```

    tool_name,
    ...,
    crs = NULL,
    verbose_mode = FALSE,
    command_only = FALSE
)

## S3 method for class ``function``
wbt(
  result,
  tool_name,
  ...,
  crs = NULL,
  verbose_mode = FALSE,
  command_only = FALSE
)

## S3 method for class 'missing'
wbt(
  result,
  tool_name,
  ...,
  crs = NULL,
  verbose_mode = FALSE,
  command_only = FALSE
)

```

### Arguments

result	an object of class <code>wbt_result</code>
tool_name	character. name of the tool to run. Or a tool/function name (i.e. a symbol) that is non-standard evaluated as a character.
...	arguments to tool
crs	character Optional: a WKT Coordinate Reference System string, or other identifier such as EPSG code or PROJ string
verbose_mode	passed to <code>wbt_run_tool()</code>
command_only	Return command that would be run with <code>system()</code> ? Default: FALSE
i	Optional index of result list element to return as result. Default is whole list.
history	Default: TRUE returns a list of all history results
attribute	Default: "output"

### Details

Supports `SpatRaster` / `RasterLayer` input / output. Arguments are transformed from their source class and passed to `WhiteboxTools` executable as standard character string arguments involving file paths.



To print help for any tool, see `wbt_tool_help()`

`tool_name` may be specified with or without quotes or `wbt_` prefix. e.g. `"wbt_slope"`, `wbt_slope`, `slope`, and `"slope"` are identical.

### Value

a list with class `"wbt_result"` containing elements:

- `tool` - the tool name
- `args` - arguments passed to executable
- `stdout` - console output (result of `wbt_run_tool()`)
- `crs` - Coordinate Reference System string (WKT or PROJ)
- `result` - any 'result' parameters (`--output`) that can be converted to R objects after run. A list of RasterLayer or character. May be a try-error if arguments are specified incorrectly.
- `history` - history of 'result' when `wbt_result` was passed as input, most recent output at end

list of result in attribute if `"history"` is present, otherwise the result in attribute. If `i` is specified, just the `i`th element of the list.

### See Also

[wbt\\_tool\\_help\(\)](#)

---

wbttoolparameters	<i>'WhiteboxTools' Tool Parameters</i>
-------------------	--

---

### Description

This data set is a `data.frame` containing tool parameters and associated metadata

### Usage

```
wbttoolparameters
```

### Format

An object of class `data.frame` with 2220 rows and 13 columns.

### Variables

- `"function_name"` - R function name
- `"tool_name"` - 'WhiteboxTools' tool name
- `"name"` - parameter name
- `"flags"` - flags used to specify parameter on command line; comma separated
- `"description"` - parameter description

- "parameter\_class" - parameter type
- "parameter\_detail" - parameter details; character: data type followed by colon and more specifics, For OptionList possible values, comma-separated (if defined)
- "default\_value" - parameter default value, if any
- "optional" - parameter "optional" flag; note that some combination of optional parameters may be required for certain conditions
- "label" - labels for selected subset of "flags" **used as R function argument names** for wbt\_ functions
- "is\_input" - logical. Classification of 'input' parameters
- "is\_output" - logical. Classification of 'output' parameters

### Source

[WhiteboxTools](#)

### See Also

[wbtttools wbt\\_tool\\_parameters\(\)](#)

---

wbtttools

*'WhiteboxTools' Tool List*

---

### Description

This data set is a `data.frame` containing tools by name and associated R function name

### Usage

```
wbtttools
```

### Format

An object of class `data.frame` with 545 rows and 8 columns.

### Variables

- "tool\_name" - 'WhiteboxTools' tool name
- "function\_name" - R function name
- "toolbox\_name" - 'WhiteboxTools' toolbox name
- "label" - 'WhiteboxTools' tool label
- "description" - Brief description
- "github" - Link to related code on 'GitHub'
- "book" - Link to 'WhiteboxTools' Manual
- "is\_extension" - Tool is part of 'General Toolset Extension' (GTE), as opposed to the "open core"

**Source**

WhiteboxTools

**See Also**

[wbttoolparameters wbt\\_list\\_tools\(\)](#)

---

wbt_absolute_value	<i>Absolute value</i>
--------------------	-----------------------

---

**Description**

Calculates the absolute value of every cell in a raster.

**Usage**

```
wbt_absolute_value(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_accumulation_curvature`*Accumulation curvature*

---

## Description

This tool calculates accumulation curvature from an input DEM.

## Usage

```
wbt_accumulation_curvature(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

<code>dem</code>	Name of the input raster DEM file.
<code>output</code>	Name of the output raster image file.
<code>log</code>	Display output values using a log-scale.
<code>zfactor</code>	Z conversion factor.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

## Value

Returns the tool text outputs.

---

wbt_activate	<i>Activate 'WhiteboxTools' Extensions</i>
--------------	--

---

**Description**

Activate 'WhiteboxTools' Extensions

**Usage**

```
wbt_activate(
    email,
    activation_key,
    seat = 1,
    destdir = dirname(wbt_exe_path(shell_quote = FALSE))
)
```

**Arguments**

email	Email Address
activation_key	Activation Key
seat	Seat Number (Default 1)
destdir	Directory containing whitebox_tools and /plugins/ folder.

**Value**

0 for success (invisibly). Try-error on error.

---

wbt_adaptive_filter	<i>Adaptive filter</i>
---------------------	------------------------

---

**Description**

Performs an adaptive filter on an image.

**Usage**

```
wbt_adaptive_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    threshold = 2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
threshold	Difference from mean threshold, in standard deviations.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_add

*Add*


---

**Description**

Performs an addition operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_add(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_add\_point\_coordinates\_to\_table

*Add point coordinates to table*

---

**Description**

Modifies the attribute table of a point vector by adding fields containing each point's X and Y coordinates.

**Usage**

```
wbt_add_point_coordinates_to_table(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector Points file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_aggregate\_raster *Aggregate raster*

---

## Description

Aggregates a raster to a lower resolution.

## Usage

```
wbt_aggregate_raster(  
    input,  
    output,  
    agg_factor = 2,  
    type = "mean",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input	Input raster file.
output	Output raster file.
agg_factor	Aggregation factor, in pixels.
type	Statistic used to fill output pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.



---

wbt_and	<i>And</i>
---------	------------

---

### Description

Performs a logical AND operator on two Boolean raster images.

### Usage

```
wbt_and(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input raster file.
input2	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_anova	<i>Anova</i>
-----------	--------------

---

### Description

Performs an analysis of variance (ANOVA) test on a raster dataset.

### Usage

```
wbt_anova(  
  input,  
  features,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

### Arguments

input	Input raster file.
features	Feature definition (or class) raster.
output	Output HTML file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_arcosh	<i>Arcosh</i>
------------	---------------

---

### Description

Returns the inverse hyperbolic cosine (arcosh) of each values in a raster.

### Usage

```
wbt_arcosh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_arc_cos	<i>Arc cos</i>
-------------	----------------

---

### Description

Returns the inverse cosine (arccos) of each values in a raster.

**Usage**

```
wbt_arc_cos(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_arc_sin	<i>Arc sin</i>
-------------	----------------

---

**Description**

Returns the inverse sine (arcsin) of each values in a raster.

**Usage**

```
wbt_arc_sin(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_arc_tan	<i>Arc tan</i>
-------------	----------------

---

**Description**

Returns the inverse tangent (arctan) of each values in a raster.

**Usage**

```
wbt_arc_tan(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_arsinh	<i>Arsinh</i>
------------	---------------

---

**Description**

Returns the inverse hyperbolic sine (arsinh) of each values in a raster.

**Usage**

```
wbt_arsinh(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_artanh	<i>Artanh</i>
------------	---------------

---

### Description

Returns the inverse hyperbolic tangent (arctanh) of each values in a raster.

### Usage

```
wbt_artanh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_ascii_to_las	<i>Ascii to las</i>
------------------	---------------------

---

### Description

Converts one or more ASCII files containing LiDAR points into LAS files.

**Usage**

```
wbt_ascii_to_las(
    inputs,
    pattern,
    proj = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

inputs	Input LiDAR ASCII files (.csv).
pattern	Input field pattern.
proj	Well-known-text string or EPSG code describing projection.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_aspect	<i>Aspect</i>
------------	---------------

---

**Description**

Calculates an aspect raster from an input DEM.

**Usage**

```
wbt_aspect(
    dem,
    output,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```



**Arguments**

dem	Input raster DEM file.
output	Output raster file.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_assess_route	<i>Assess route</i>
------------------	---------------------

---

**Description**

This tool assesses a route for slope, elevation, and visibility variation.

**Usage**

```
wbt_assess_route(
  routes,
  dem,
  output,
  length = "",
  dist = 20,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

routes	Name of the input routes vector file.
dem	Name of the input DEM raster file.
output	Name of the output lines shapefile.
length	Maximum segment length (m).
dist	Search distance, in grid cells, used in visibility analysis.

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_atan2	<i>Atan2</i>
-----------	--------------

---

**Description**

Returns the 2-argument inverse tangent (atan2).

**Usage**

```
wbt_atan2(
  input_y,
  input_x,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input_y	Input y raster file or constant value (rise).
input_x	Input x raster file or constant value (run).
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_attribute\_correlation  
*Attribute correlation*

---

### Description

Performs a correlation analysis on attribute fields from a vector database.

### Usage

```
wbt_attribute_correlation(  
  input,  
  output = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

### Arguments

input	Input vector file.
output	Output HTML file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_attribute\_correlation\_neighbourhood\_analysis  
*Attribute correlation neighbourhood analysis*

---

### Description

Performs a correlation on two input vector attributes within a neighbourhood search windows.

**Usage**

```
wbt_attribute_correlation_neighbourhood_analysis(
    input,
    field1,
    field2,
    radius = NULL,
    min_points = NULL,
    stat = "pearson",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector file.
field1	First input field name (dependent variable) in attribute table.
field2	Second input field name (independent variable) in attribute table.
radius	Search Radius (in map units).
min_points	Minimum number of points.
stat	Correlation type; one of 'pearson' (default) and 'spearman'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_attribute\_histogram

*Attribute histogram*

---

**Description**

Creates a histogram for the field values of a vector's attribute table.

**Usage**

```
wbt_attribute_histogram(
    input,
    field,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector file.
field	Input field name in attribute table.
output	Output HTML file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_attribute\_scattergram  
*Attribute scattergram*

---

**Description**

Creates a scattergram for two field values of a vector's attribute table.

**Usage**

```
wbt_attribute_scattergram(
    input,
    fieldx,
    fieldy,
    output,
    trendline = FALSE,
    wd = NULL,
```

```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input raster file.
fieldx	Input field name in attribute table for the x-axis.
fieldy	Input field name in attribute table for the y-axis.
output	Output HTML file (default name will be based on input file if unspecified).
trendline	Draw the trendline.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_average_flowpath_slope
    Average flowpath slope

```

---

### Description

Measures the average slope gradient from each grid cell to all upslope divide cells.

### Usage

```

wbt_average_flowpath_slope(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_average\_normal\_vector\_angular\_deviation  
*Average normal vector angular deviation*

---

**Description**

Calculates the circular variance of aspect at a scale for a DEM.

**Usage**

```
wbt_average_normal_vector_angular_deviation(
  dem,
  output,
  filter = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
filter	Size of the filter kernel.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_average_overlay	<i>Average overlay</i>
---------------------	------------------------

---

**Description**

Calculates the average for each grid cell from a group of raster images.

**Usage**

```
wbt_average_overlay(  
    inputs,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt\_average\_upslope\_flowpath\_length  
*Average upslope flowpath length*

---

**Description**

Measures the average length of all upslope flowpaths draining each grid cell.

**Usage**

```
wbt_average_upslope_flowpath_length(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_balance\_contrast\_enhancement  
*Balance contrast enhancement*

---

**Description**

Performs a balance contrast enhancement on a colour-composite image of multispectral data.

**Usage**

```
wbt_balance_contrast_enhancement(
    input,
    output,
    band_mean = 100,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input colour composite image file.
output	Output raster file.
band_mean	Band mean value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_basins	<i>Basins</i>
------------	---------------

---

**Description**

Identifies drainage basins that drain to the DEM edge.

**Usage**

```
wbt_basins(
    d8_pntr,
    output,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_bilateral\_filter *Bilateral filter*

---

**Description**

A bilateral filter is an edge-preserving smoothing filter introduced by Tomasi and Manduchi (1998).

**Usage**

```
wbt_bilateral_filter(
  input,
  output,
  sigma_dist = 0.75,
  sigma_int = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
sigma_dist	Standard deviation in distance in pixels.
sigma_int	Standard deviation in intensity in pixels.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_block\_maximum\_gridding  
*Block maximum gridding*

---

**Description**

Creates a raster grid based on a set of vector points and assigns grid values using a block maximum scheme.

**Usage**

```
wbt_block_maximum_gridding(
  input,
  field,
  output,
  use_z = FALSE,
  cell_size = NULL,
  base = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector Points file.
field	Input field name in attribute table.
output	Output raster file.
use_z	Use z-coordinate instead of field?.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_block\_minimum\_gridding  
*Block minimum gridding*

---

**Description**

Creates a raster grid based on a set of vector points and assigns grid values using a block minimum scheme.

**Usage**

```
wbt_block_minimum_gridding(
  input,
  field,
  output,
  use_z = FALSE,
  cell_size = NULL,
  base = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector Points file.
field	Input field name in attribute table.
output	Output raster file.
use_z	Use z-coordinate instead of field?.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_boundary_shape_complexity`  
*Boundary shape complexity*

---

**Description**

Calculates the complexity of the boundaries of raster polygons.

**Usage**

```
wbt_boundary_shape_complexity(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<code>input</code>	Input raster file.
<code>output</code>	Output raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_breach\_depressions  
*Breach depressions*

---

### Description

Breaches all of the depressions in a DEM using Lindsay's (2016) algorithm. This should be preferred over depression filling in most cases.

### Usage

```
wbt_breach_depressions(
    dem,
    output,
    max_depth = NULL,
    max_length = NULL,
    flat_increment = NULL,
    fill_pits = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
max_depth	Optional maximum breach depth (default is Inf).
max_length	Optional maximum breach channel length (in grid cells; default is Inf).
flat_increment	Optional elevation increment applied to flat areas.
fill_pits	Optional flag indicating whether to fill single-cell pits.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_breach\_depressions\_least\_cost  
*Breach depressions least cost*

---

### Description

Breaches the depressions in a DEM using a least-cost pathway method.

### Usage

```
wbt_breach_depressions_least_cost(  
    dem,  
    output,  
    dist,  
    max_cost = NULL,  
    min_dist = TRUE,  
    flat_increment = NULL,  
    fill = TRUE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
dist	Maximum search distance for breach paths in cells.
max_cost	Optional maximum breach cost (default is Inf).
min_dist	Optional flag indicating whether to minimize breach distances.
flat_increment	Optional elevation increment applied to flat areas.
fill	Optional flag indicating whether to fill any remaining unbreached depressions.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_breach\_single\_cell\_pits  
*Breach single cell pits*

---

**Description**

Removes single-cell pits from an input DEM by breaching.

**Usage**

```
wbt_breach_single_cell_pits(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_breakline\_mapping *Breakline mapping*

---

**Description**

This tool maps breaklines from an input DEM.

**Usage**

```
wbt_breakline_mapping(
    dem,
    output,
    threshold = 2,
    min_length = 3,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster image file.
output	Name of the output vector lines file.
threshold	Threshold value (0 - infinity but typically 1 to 5 works well).
min_length	Minimum line length, in grid cells.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_buffer_raster	<i>Buffer raster</i>
-------------------	----------------------

---

**Description**

Maps a distance-based buffer around each non-background (non-zero/non-nodata) grid cell in an input image.

**Usage**

```
wbt_buffer_raster(
    input,
    output,
    size,
    gridcells = FALSE,
```

```

    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input raster file.
output	Output raster file.
size	Buffer size.
gridcells	Optional flag to indicate that the 'size' threshold should be measured in grid cells instead of the default map units.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_burn_streams_at_roads
    Burn streams at roads

```

---

### Description

Burns-in streams at the sites of road embankments.

### Usage

```

wbt_burn_streams_at_roads(
  dem,
  streams,
  roads,
  output,
  width = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

dem	Input raster digital elevation model (DEM) file.
streams	Input vector streams file.
roads	Input vector roads file.
output	Output raster file.
width	Maximum road embankment width, in map units.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_canny\_edge\_detection  
*Canny edge detection*

---

**Description**

This tool performs a Canny edge-detection filter on an input image.

**Usage**

```
wbt_canny_edge_detection(
  input,
  output,
  sigma = 0.5,
  low = 0.05,
  high = 0.15,
  add_back = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Name of the input raster image file.
output	Name of the output raster image file.
sigma	Sigma value used in Gaussian filtering, default = 0.5.
low	Low threshold, default = 0.05.
high	High threshold, default = 0.15.
add_back	Add the edge cells back to the input image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_ceil

*Ceil*


---

**Description**

Returns the smallest (closest to negative infinity) value that is greater than or equal to the values in a raster.

**Usage**

```
wbt_ceil(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_centroid	<i>Centroid</i>
--------------	-----------------

---

**Description**

Calculates the centroid, or average location, of raster polygon objects.

**Usage**

```
wbt_centroid(
  input,
  output,
  text_output = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
text_output	Optional text output.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_centroid_vector	<i>Centroid vector</i>
---------------------	------------------------

---

**Description**

Identifies the centroid point of a vector polyline or polygon feature or a group of vector points.

**Usage**

```
wbt_centroid_vector(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input vector file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_change_vector_analysis`*Change vector analysis*

---

### Description

Performs a change vector analysis on a two-date multi-spectral dataset.

### Usage

```
wbt_change_vector_analysis(  
    date1,  
    date2,  
    magnitude,  
    direction,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>date1</code>	Input raster files for the earlier date.
<code>date2</code>	Input raster files for the later date.
<code>magnitude</code>	Output vector magnitude raster file.
<code>direction</code>	Output vector Direction raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.



---

```
wbt_circular_variance_of_aspect
    Circular variance of aspect
```

---

### Description

Calculates the circular variance of aspect at a scale for a DEM.

### Usage

```
wbt_circular_variance_of_aspect(
    dem,
    output,
    filter = 11,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
filter	Size of the filter kernel.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_classify\_buildings\_in\_lidar  
*Classify buildings in lidar*

---

### Description

Reclassifies a LiDAR points that lie within vector building footprints.

### Usage

```
wbt_classify_buildings_in_lidar(  
    input,  
    buildings,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
buildings	Input vector polygons file.
output	Output LiDAR file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_classify\_lidar      *Classify lidar*

---

### Description

Classify points within a LiDAR point cloud based on point properties.

### Usage

```
wbt_classify_lidar(
    input,
    output = NULL,
    radius = 1.5,
    grd_threshold = 0.1,
    oto_threshold = 2,
    planarity_threshold = 0.85,
    linearity_threshold = 0.7,
    iterations = 30,
    facade_threshold = 0.5,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
radius	Search distance used in neighbourhood search (metres).
grd_threshold	Ground threshold (metres).
oto_threshold	Off-terrain object threshold (metres).
planarity_threshold	Planarity threshold (0-1).
linearity_threshold	Linearity threshold (0-1).
iterations	Number of iterations.
facade_threshold	Facade threshold (metres).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_classify_overlap_points
    Classify overlap points
```

---

**Description**

Classifies or filters LAS points in regions of overlapping flight lines.

**Usage**

```
wbt_classify_overlap_points(
    input,
    output,
    resolution = 2,
    criterion = "max scan angle",
    filter = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
resolution	The size of the square area used to evaluate nearby points in the LiDAR data.
criterion	Criterion used to identify overlapping points; options are 'max scan angle', 'not min point source ID', 'not min time', 'multiple point source IDs'.
filter	Filter out points from overlapping flightlines? If false, overlaps will simply be classified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_clean_vector	<i>Clean vector</i>
------------------	---------------------

---

### Description

Removes null features and lines/polygons with fewer than the required number of vertices.

### Usage

```
wbt_clean_vector(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input vector file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_clip	<i>Clip</i>
----------	-------------

---

### Description

Extract all the features, or parts of features, that overlap with the features of the clip vector.

**Usage**

```
wbt_clip(
  input,
  clip,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector file.
clip	Input clip polygon vector file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_clip_lidar_to_polygon
  Clip lidar to polygon
```

---

**Description**

Clips a LiDAR point cloud to a vector polygon or polygons.

**Usage**

```
wbt_clip_lidar_to_polygon(
  input,
  polygons,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
polygons	Input vector polygons file.
output	Output LiDAR file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_clip\_raster\_to\_polygon  
*Clip raster to polygon*

---

**Description**

Clips a raster to a vector polygon.

**Usage**

```
wbt_clip_raster_to_polygon(
  input,
  polygons,
  output,
  maintain_dimensions = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
polygons	Input vector polygons file.
output	Output raster file.
maintain_dimensions	Maintain input raster dimensions?.

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_closing	<i>Closing</i>
-------------	----------------

---

**Description**

A closing is a mathematical morphology operation involving an erosion (min filter) of a dilation (max filter) set.

**Usage**

```
wbt_closing(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.



**Value**

Returns the tool text outputs.

---

wbt\_clump

*Clump*


---

**Description**

Groups cells that form discrete areas, assigning them unique identifiers.

**Usage**

```
wbt_clump(
  input,
  output,
  diag = TRUE,
  zero_back = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
diag	Flag indicating whether diagonal connections should be considered.
zero_back	Flag indicating whether zero values should be treated as a background.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_colourize\_based\_on\_class  
*Colourize based on class*

---

**Description**

Sets the RGB values of a LiDAR point cloud based on the point classification values.

**Usage**

```
wbt_colourize_based_on_class(
    input,
    output = NULL,
    intensity_blending = 50,
    clr_str = "",
    use_unique_clr_for_buildings = FALSE,
    radius = "",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
intensity_blending	Intensity blending amount (0-100 percent).
clr_str	Colour values, e.g. 2: (184, 167, 108); 5: #9ab86c.
use_unique_clr_for_buildings	Use unique colours for each building?.
radius	Search distance used in neighbourhood search.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_colourize\_based\_on\_point\_returns  
*Colourize based on point returns*

---

### Description

Sets the RGB values of a LiDAR point cloud based on the point returns.

### Usage

```
wbt_colourize_based_on_point_returns(
    input,
    output = NULL,
    intensity_blending = 50,
    only = "(230,214,170)",
    first = "(0,140,0)",
    intermediate = "(255,0,255)",
    last = "(0,0,255)",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
intensity_blending	Intensity blending amount (0-100 percent).
only	Only return colour, e.g. (230,214,170), #e6d6aa, or 0xe6d6aa.
first	First return colour, e.g. (230,214,170), #e6d6aa, or 0xe6d6aa.
intermediate	Intermediate return colour, e.g. (230,214,170), #e6d6aa, or 0xe6d6aa.
last	Last return colour, e.g. (230,214,170), #e6d6aa, or 0xe6d6aa.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_compactness\_ratio *Compactness ratio*

---

### Description

Calculates the compactness ratio ( $A/P$ ), a measure of shape complexity, for vector polygons.

### Usage

```
wbt_compactness_ratio(  
    input,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_conditional\_evaluation  
*Conditional evaluation*

---

### Description

Performs a conditional evaluation (if-then-else) operation on a raster.

**Usage**

```
wbt_conditional_evaluation(
  input,
  output,
  statement = "",
  true = NULL,
  false = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Name of the input raster file.
output	Name of the output raster file.
statement	Conditional statement e.g. value > 35.0. This statement must be a valid Rust statement.
true	Value where condition evaluates TRUE (input raster or constant value).
false	Value where condition evaluates FALSE (input raster or constant value).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_conditioned\_latin\_hypercube  
*Conditioned latin hypercube*

---

**Description**

Implements conditioned Latin Hypercube sampling.

**Usage**

```
wbt_conditioned_latin_hypercube(
  inputs,
  output,
  samples = 500,
  iterations = 25000,
  seed = NULL,
  prob = 0.5,
  threshold = NULL,
  temp = 1,
  temp_decay = 0.05,
  cycle = 10,
  average = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Name of the input raster file.
output	Output shapefile.
samples	Number of sample sites returned.
iterations	Maximum iterations (if stopping criteria not reached).
seed	Seed for RNG consistency.
prob	Probability of random resample or resampling worst strata between $[0, 1]$ .
threshold	Objective function values below the threshold stop the resampling iterations.
temp	Initial annealing temperature between $[0, 1]$ .
temp_decay	Annealing temperature decay proportion between $[0, 1]$ . Reduce temperature by this proportion each annealing cycle.
cycle	Number of iterations before decaying annealing temperature.
average	Weight the continuous objective function by the $1/N$ contributing strata.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_conservative\_smoothing\_filter  
*Conservative smoothing filter*

---

## Description

Performs a conservative-smoothing filter on an image.

## Usage

```
wbt_conservative_smoothing_filter(  
    input,  
    output,  
    filterx = 3,  
    filtery = 3,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.

---

wbt\_construct\_vector\_tin

*Construct vector tin*


---

### Description

Creates a vector triangular irregular network (TIN) for a set of vector points.

### Usage

```
wbt_construct_vector_tin(
    input,
    output,
    field = NULL,
    use_z = FALSE,
    max_triangle_edge_length = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input vector points file.
output	Output vector polygon file.
field	Input field name in attribute table.
use_z	Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?.
max_triangle_edge_length	Optional maximum triangle edge length; triangles larger than this size will not be gridded.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_contours\_from\_points  
*Contours from points*

---

## Description

Creates a contour coverage from a set of input points.

## Usage

```
wbt_contours_from_points(
  input,
  output,
  field = NULL,
  use_z = FALSE,
  max_triangle_edge_length = NULL,
  interval = 10,
  base = 0,
  smooth = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

## Arguments

input	Input vector points file.
output	Output vector lines file.
field	Input field name in attribute table.
use_z	Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?.
max_triangle_edge_length	Optional maximum triangle edge length; triangles larger than this size will not be gridded.
interval	Contour interval.
base	Base contour height.
smooth	Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_contours\_from\_raster  
*Contours from raster*

---

**Description**

Derives a vector contour coverage from a raster surface.

**Usage**

```
wbt_contours_from_raster(
    input,
    output,
    interval = 10,
    base = 0,
    smooth = 9,
    tolerance = 10,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input surface raster file.
output	Output vector contour file.
interval	Contour interval.
base	Base contour height.
smooth	Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
tolerance	Tolerance factor, in degrees (0-45); determines generalization level.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_convert\_nodata\_to\_zero  
*Convert nodata to zero*

---

**Description**

Converts nodata values in a raster to zero.

**Usage**

```
wbt_convert_nodata_to_zero(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_convert\_raster\_format  
*Convert raster format*

---

**Description**

Converts raster data from one format to another.

**Usage**

```
wbt_convert_raster_format(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_corner\_detection *Corner detection*

---

**Description**

Identifies corner patterns in boolean images using hit-and-miss pattern matching.

**Usage**

```
wbt_corner_detection(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input boolean image.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_correct\_vignetting  
*Correct vignetting*

---

**Description**

Corrects the darkening of images towards corners.

**Usage**

```
wbt_correct_vignetting(
  input,
  pp,
  output,
  focal_length = 304.8,
  image_width = 228.6,
  n = 4,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
pp	Input principal point file.
output	Output raster file.
focal_length	Camera focal length, in millimeters.

image_width	Distance between photograph edges, in millimeters.
n	The 'n' parameter.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_cos	<i>Cos</i>
---------	------------

---

**Description**

Returns the cosine (cos) of each values in a raster.

**Usage**

```
wbt_cos(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_cosh

*Cosh*


---

**Description**

Returns the hyperbolic cosine (cosh) of each values in a raster.

**Usage**

```
wbt_cosh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_cost\_allocation

*Cost allocation*


---

**Description**

Identifies the source cell to which each grid cell is connected by a least-cost pathway in a cost-distance analysis.

**Usage**

```
wbt_cost_allocation(
    source,
    backlink,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

source	Input source raster file.
backlink	Input backlink raster file generated by the cost-distance tool.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_cost_distance	<i>Cost distance</i>
-------------------	----------------------

---

**Description**

Performs cost-distance accumulation on a cost surface and a group of source cells.

**Usage**

```
wbt_cost_distance(
    source,
    cost,
    out_accum,
    out_backlink,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```



**Arguments**

source	Input source raster file.
cost	Input cost (friction) raster file.
out_accum	Output cost accumulation raster file.
out_backlink	Output backlink raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_cost_pathway	<i>Cost pathway</i>
------------------	---------------------

---

**Description**

Performs cost-distance pathway analysis using a series of destination grid cells.

**Usage**

```
wbt_cost_pathway(
  destination,
  backlink,
  output,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

destination	Input destination raster file.
backlink	Input backlink raster file generated by the cost-distance tool.
output	Output cost pathway raster file.
zero_background	Flag indicating whether zero values should be treated as a background.

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_count_if	<i>Count if</i>
--------------	-----------------

---

**Description**

Counts the number of occurrences of a specified value in a cell-stack of rasters.

**Usage**

```
wbt_count_if(
  inputs,
  output,
  value,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
value	Search value (e.g. countif value = 5.0).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_create_colour_composite
    Create colour composite
```

---

**Description**

Creates a colour-composite image from three bands of multispectral imagery.

**Usage**

```
wbt_create_colour_composite(
    red,
    green,
    blue,
    output,
    opacity = NULL,
    enhance = TRUE,
    zeros = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

red	Input red band image file.
green	Input green band image file.
blue	Input blue band image file.
output	Output colour composite file.
opacity	Input opacity band image file (optional).
enhance	Optional flag indicating whether a balance contrast enhancement is performed.
zeros	Optional flag to indicate if zeros are nodata values.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_create_hexagonal_vector_grid`*Create hexagonal vector grid*

---

**Description**

Creates a hexagonal vector grid.

**Usage**

```
wbt_create_hexagonal_vector_grid(  
    input,  
    output,  
    width,  
    orientation = "horizontal",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input base file.
<code>output</code>	Output vector polygon file.
<code>width</code>	The grid cell width.
<code>orientation</code>	Grid Orientation, 'horizontal' or 'vertical'.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_create_plane	<i>Create plane</i>
------------------	---------------------

---

**Description**

Creates a raster image based on the equation for a simple plane.

**Usage**

```
wbt_create_plane(  
    base,  
    output,  
    gradient = 15,  
    aspect = 90,  
    constant = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

base	Input base raster file.
output	Output raster file.
gradient	Slope gradient in degrees (-85.0 to 85.0).
aspect	Aspect (direction) in degrees clockwise from north (0.0-360.0).
constant	Constant value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_create\_rectangular\_vector\_grid  
*Create rectangular vector grid*

---

### Description

Creates a rectangular vector grid.

### Usage

```
wbt_create_rectangular_vector_grid(  
    input,  
    output,  
    width,  
    height,  
    xorig = 0,  
    yorig = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input base file.
output	Output vector polygon file.
width	The grid cell width.
height	The grid cell height.
xorig	The grid origin x-coordinate.
yorig	The grid origin y-coordinate.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_crispness\_index    *Crispness index*

---

### Description

Calculates the Crispness Index, which is used to quantify how crisp (or conversely how fuzzy) a probability image is.

### Usage

```
wbt_crispness_index(
  input,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Optional output html file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_cross\_tabulation    *Cross tabulation*

---

### Description

Performs a cross-tabulation on two categorical images.

**Usage**

```
wbt_cross_tabulation(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file 1.
input2	Input raster file 1.
output	Output HTML file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_csv\_points\_to\_vector

*Csv points to vector*

---

**Description**

Converts a CSV text file to vector points.

**Usage**

```
wbt_csv_points_to_vector(
    input,
    output,
    xfield = 0,
    yfield = 1,
    epsg = NULL,
    wd = NULL,
```



```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input CSV file (i.e. source of data to be imported).
output	Output vector file.
xfield	X field number (e.g. 0 for first field).
yfield	Y field number (e.g. 1 for second field).
epsg	EPSG projection (e.g. 2958).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_cumulative_distribution
      Cumulative distribution

```

---

### Description

Converts a raster image to its cumulative distribution function.

### Usage

```

wbt_cumulative_distribution(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_curvedness	<i>Curvedness</i>
----------------	-------------------

---

**Description**

This tool calculates curvedness from an input DEM.

**Usage**

```
wbt_curvedness(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
log	Display output values using a log-scale.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_d8\_flow\_accumulation  
*D8 flow accumulation*

---

**Description**

Calculates a D8 flow accumulation raster from an input DEM or flow pointer.

**Usage**

```
wbt_d8_flow_accumulation(
    input,
    output,
    out_type = "cells",
    log = FALSE,
    clip = FALSE,
    pntr = FALSE,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster DEM or D8 pointer file.
output	Output raster file.
out_type	Output type; one of 'cells' (default), 'catchment area', and 'specific contributing area'.
log	Optional flag to request the output be log-transformed.
clip	Optional flag to request clipping the display max by 1 percent.
pntr	Is the input raster a D8 flow pointer rather than a DEM?.
esri_pntr	Input D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_d8_mass_flux	<i>D8 mass flux</i>
------------------	---------------------

---

**Description**

Performs a D8 mass flux calculation.

**Usage**

```
wbt_d8_mass_flux(
  dem,
  loading,
  efficiency,
  absorption,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
loading	Input loading raster file.
efficiency	Input efficiency raster file.
absorption	Input absorption raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_d8_pointer	<i>D8 pointer</i>
----------------	-------------------

---

**Description**

Calculates a D8 flow pointer raster from an input DEM.

**Usage**

```
wbt_d8_pointer(
    dem,
    output,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_dbscan

*Dbscan*


---

### Description

Performs a DBSCAN-based unsupervised clustering operation.

### Usage

```
wbt_dbscan(
    inputs,
    output,
    scaling = "Normalize",
    search_dist = 0.01,
    min_points = 5,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

inputs	Names of the input rasters.
output	Name of the output raster file.
scaling	Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
search_dist	Search-distance parameter.
min_points	Minimum point density needed to define 'core' point in cluster.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_decrement	<i>Decrement</i>
---------------	------------------

---

### Description

Decreases the values of each grid cell in an input raster by 1.0 (see also InPlaceSubtract).

### Usage

```
wbt_decrement(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_dem_void_filling	<i>Dem void filling</i>
----------------------	-------------------------

---

### Description

This tool can be used to fill the void areas of a DEM using another fill DEM data set.

**Usage**

```
wbt_dem_void_filling(
    dem,
    fill,
    output,
    mean_plane_dist = 20,
    edge_treatment = "use DEM",
    weight_value = 2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster DEM file, containing the void areas.
fill	Name of the input fill DEM file, containing the values used to fill the void areas in the other DEM.
output	Name of the output void-filled DEM file.
mean_plane_dist	Distance to void edge at which the mean-plane value is used as an offset, measured in grid cells.
edge_treatment	How should void-edge cells be treated? Options include 'use DEM' (default), 'use Fill', 'average'.
weight_value	Weight value used for IDW interpolation (default is 2.0).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt_depth_in_sink	<i>Depth in sink</i>
-------------------	----------------------

---

### Description

Measures the depth of sinks (depressions) in a DEM.

### Usage

```
wbt_depth_in_sink(  
    dem,  
    output,  
    zero_background = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
zero_background	Flag indicating whether the background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_depth\_to\_water      *Depth to water*

---

### Description

This tool calculates cartographic depth-to-water (DTW) index.

### Usage

```
wbt_depth_to_water(  
    dem,  
    output,  
    streams = NULL,  
    lakes = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
streams	Name of the input streams vector (optional).
lakes	Name of the input lakes vector (optional).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_dev_from_mean_elev`*Dev from mean elev*

---

**Description**

Calculates deviation from mean elevation.

**Usage**

```
wbt_dev_from_mean_elev(  
    dem,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>filterx</code>	Size of the filter kernel in the x-direction.
<code>filtery</code>	Size of the filter kernel in the y-direction.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_difference	<i>Difference</i>
----------------	-------------------

---

### Description

Outputs the features that occur in one of the two vector inputs but not both, i.e. no overlapping features.

### Usage

```
wbt_difference(  
    input,  
    overlay,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input vector file.
overlay	Input overlay vector file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_difference_curvature`*Difference curvature*

---

## Description

This tool calculates difference curvature from an input DEM.

## Usage

```
wbt_difference_curvature(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

<code>dem</code>	Name of the input raster DEM file.
<code>output</code>	Name of the output raster image file.
<code>log</code>	Display output values using a log-scale.
<code>zfactor</code>	Z conversion factor.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

## Value

Returns the tool text outputs.

---

`wbt_diff_from_mean_elev`*Diff from mean elev*

---

### Description

Calculates difference from mean elevation (equivalent to a high-pass filter).

### Usage

```
wbt_diff_from_mean_elev(  
    dem,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>filterx</code>	Size of the filter kernel in the x-direction.
<code>filtery</code>	Size of the filter kernel in the y-direction.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_diff\_of\_gaussian\_filter  
*Diff of gaussian filter*

---

### Description

Performs a Difference of Gaussian (DoG) filter on an image.

### Usage

```
wbt_diff_of_gaussian_filter(  
    input,  
    output,  
    sigma1 = 2,  
    sigma2 = 4,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
sigma1	Standard deviation distance in pixels.
sigma2	Standard deviation distance in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_directional_relief`*Directional relief*

---

**Description**

Calculates relief for cells in an input DEM for a specified direction.

**Usage**

```
wbt_directional_relief(  
    dem,  
    output,  
    azimuth = 0,  
    max_dist = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>azimuth</code>	Wind azimuth in degrees.
<code>max_dist</code>	Optional maximum search distance (unspecified if none; in xy units).
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt\_direct\_decorrelation\_stretch

*Direct decorrelation stretch*


---

### Description

Performs a direct decorrelation stretch enhancement on a colour-composite image of multispectral data.

### Usage

```
wbt_direct_decorrelation_stretch(
    input,
    output,
    k = 0.5,
    clip = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input colour composite image file.
output	Output raster file.
k	Achromatic factor (k) ranges between 0 (no effect) and 1 (full saturation stretch), although typical values range from 0.3 to 0.7.
clip	Optional percent to clip the upper tail by during the stretch.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_dissolve	<i>Dissolve</i>
--------------	-----------------

---

### Description

Removes the interior, or shared, boundaries within a vector polygon coverage.

### Usage

```
wbt_dissolve(
  input,
  output,
  field = NULL,
  snap = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Input vector file.
output	Output vector file.
field	Dissolve field attribute (optional).
snap	Snap tolerance.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_distance\_to\_outlet  
*Distance to outlet*

---

**Description**

Calculates the distance of stream grid cells to the channel network outlet cell.

**Usage**

```
wbt_distance_to_outlet(  
    d8_pntr,  
    streams,  
    output,  
    esri_pntr = FALSE,  
    zero_background = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_diversity\_filter *Diversity filter*

---

### Description

Assigns each cell in the output grid the number of different values in a moving window centred on each grid cell in the input raster.

### Usage

```
wbt_diversity_filter(  
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_divide	<i>Divide</i>
------------	---------------

---

### Description

Performs a division operation on two rasters or a raster and a constant value.

### Usage

```
wbt_divide(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_downslope\_distance\_to\_stream  
*Downslope distance to stream*

---

### Description

Measures distance to the nearest downslope stream cell.

### Usage

```
wbt_downslope_distance_to_stream(  
    dem,  
    streams,  
    output,  
    dinf = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
streams	Input raster streams file.
output	Output raster file.
dinf	Use the D-infinity flow algorithm instead of D8?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_downslope\_flowpath\_length  
*Downslope flowpath length*

---

### Description

Calculates the downslope flowpath length from each cell to basin outlet.

### Usage

```
wbt_downslope_flowpath_length(  
    d8_pntr,  
    output,  
    watersheds = NULL,  
    weights = NULL,  
    esri_pntr = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input D8 pointer raster file.
output	Output raster file.
watersheds	Optional input watershed raster file.
weights	Optional input weights raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_downslope\_index     *Downslope index*

---

### Description

Calculates the Hjerdt et al. (2004) downslope index.

### Usage

```
wbt_downslope_index(  
    dem,  
    output,  
    drop = 2,  
    out_type = "tangent",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
drop	Vertical drop value (default is 2.0).
out_type	Output type, options include 'tangent', 'degrees', 'radians', 'distance' (default is 'tangent').
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

```
wbt_d_inf_flow_accumulation
    D inf flow accumulation
```

---

**Description**

Calculates a D-infinity flow accumulation raster from an input DEM.

**Usage**

```
wbt_d_inf_flow_accumulation(
    input,
    output,
    out_type = "Specific Contributing Area",
    threshold = NULL,
    log = FALSE,
    clip = FALSE,
    pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster DEM or D-infinity pointer file.
output	Output raster file.
out_type	Output type; one of 'cells', 'sca' (default), and 'ca'.
threshold	Optional convergence threshold parameter, in grid cells; default is infinity.
log	Optional flag to request the output be log-transformed.
clip	Optional flag to request clipping the display max by 1 percent.
pntr	Is the input raster a D-infinity flow pointer rather than a DEM?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_d\_inf\_mass\_flux     *D inf mass flux*

---

### Description

Performs a D-infinity mass flux calculation.

### Usage

```
wbt_d_inf_mass_flux(  
    dem,  
    loading,  
    efficiency,  
    absorption,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
loading	Input loading raster file.
efficiency	Input efficiency raster file.
absorption	Input absorption raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_d\_inf\_pointer      *D inf pointer*

---

### Description

Calculates a D-infinity flow pointer (flow direction) raster from an input DEM.

### Usage

```
wbt_d_inf_pointer(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_edge\_contamination      *Edge contamination*

---

### Description

Identifies grid cells within an input DEM that may be impacted by edge contamination for hydrological applications.

**Usage**

```
wbt_edge_contamination(
    dem,
    output,
    flow_type = "mfd",
    zfactor = "",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Name of the input DEM raster file; must be depressionless.
output	Name of the output raster file.
flow_type	Flow algorithm type, one of 'd8', 'mfd', or 'dinf'.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_edge_density	<i>Edge density</i>
------------------	---------------------

---

**Description**

Calculates the density of edges, or breaks-in-slope within DEMs.

**Usage**

```
wbt_edge_density(
    dem,
    output,
    filter = 11,
    norm_diff = 5,
    zfactor = NULL,
```

```

    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
filter	Size of the filter kernel.
norm_diff	Maximum difference in normal vectors, in degrees.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_edge_preserving_mean_filter
    Edge preserving mean filter

```

---

### Description

Performs a simple edge-preserving mean filter on an input image.

### Usage

```

wbt_edge_preserving_mean_filter(
  input,
  output,
  threshold,
  filter = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

input	Input raster file.
output	Output raster file.
threshold	Maximum difference in values.
filter	Size of the filter kernel.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_edge\_proportion    *Edge proportion*

---

**Description**

Calculate the proportion of cells in a raster polygon that are edge cells.

**Usage**

```
wbt_edge_proportion(
  input,
  output,
  output_text = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
output_text	flag indicating whether a text report should also be output.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_elevation\_above\_stream  
*Elevation above stream*

---

**Description**

Calculates the elevation of cells above the nearest downslope stream cell.

**Usage**

```
wbt_elevation_above_stream(  
  dem,  
  streams,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem              Input raster DEM file.

streams          Input raster streams file.

output           Output raster file.

wd               Changes the working directory.

verbose\_mode    Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only    Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_elevation\_above\_stream\_euclidean  
*Elevation above stream euclidean*

---

### Description

Calculates the elevation of cells above the nearest (Euclidean distance) stream cell.

### Usage

```
wbt_elevation_above_stream_euclidean(  
    dem,  
    streams,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
streams	Input raster streams file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_elev\_above\_pit     *Elev above pit*

---

### Description

Calculate the elevation of each grid cell above the nearest downstream pit cell or grid edge cell.

### Usage

```
wbt_elev_above_pit(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_elev\_percentile     *Elev percentile*

---

### Description

Calculates the elevation percentile raster from a DEM.

**Usage**

```
wbt_elev_percentile(
    dem,
    output,
    filterx = 11,
    filtery = 11,
    sig_digits = 2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
sig_digits	Number of significant digits.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_elev\_relative\_to\_min\_max  
*Elev relative to min max*

---

**Description**

Calculates the elevation of a location relative to the minimum and maximum elevations in a DEM.

**Usage**

```
wbt_elev_relative_to_min_max(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_elev_relative_to_watershed_min_max
    Elev relative to watershed min max
```

---

**Description**

Calculates the elevation of a location relative to the minimum and maximum elevations in a watershed.

**Usage**

```
wbt_elev_relative_to_watershed_min_max(
    dem,
    watersheds,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
watersheds	Input raster watersheds file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_eliminate\_coincident\_points  
*Eliminate coincident points*

---

**Description**

Removes any coincident, or nearly coincident, points from a vector points file.

**Usage**

```
wbt_eliminate_coincident_points(  
  input,  
  output,  
  tolerance,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input vector file.
output	Output vector points file.
tolerance	The distance tolerance for points.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_elongation\_ratio *Elongation ratio*

---

**Description**

Calculates the elongation ratio for vector polygons.

**Usage**

```
wbt_elongation_ratio(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_embankment\_mapping

*Embankment mapping*


---

### Description

Maps and/or removes road embankments from an input fine-resolution DEM.

### Usage

```
wbt_embankment_mapping(
    dem,
    road_vec,
    output,
    search_dist = 2.5,
    min_road_width = 6,
    typical_width = 30,
    max_height = 2,
    max_width = 60,
    max_increment = 0.05,
    spillout_slope = 4,
    remove_embankments = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

dem	Input raster DEM file.
road_vec	Input vector polygons file.
output	Output raster file.
search_dist	Search distance used to reposition transportation vectors onto road embankments (in map units).
min_road_width	Minimum road width; this is the width of the paved road surface (in map units).
typical_width	Typical embankment width; this is the maximum width of an embankment with roadside ditches (in map units).
max_height	Typical embankment maximum height; this is the height a typical embankment with roadside ditches (in map units).
max_width	Maximum embankment width, typically where embankments traverse steep-sided valleys (in map units).
max_increment	Maximum upwards increment between neighbouring cells on an embankment (in elevation units).

spillout_slope	Spillout slope (in degrees).
remove_embankments	Optional flag indicating whether to output a DEM with embankments removed (true) or an embankment raster map (false).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_emboss_filter	<i>Emboss filter</i>
-------------------	----------------------

---

**Description**

Performs an emboss filter on an image, similar to a hillshade operation.

**Usage**

```
wbt_emboss_filter(
    input,
    output,
    direction = "n",
    clip = 0,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
direction	Direction of reflection; options include 'n', 's', 'e', 'w', 'ne', 'se', 'nw', 'sw'.
clip	Optional amount to clip the distribution tails by, in percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_equal_to	<i>Equal to</i>
--------------	-----------------

---

**Description**

Performs a equal-to comparison operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_equal_to(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1      Input raster file or constant value.

input2      Input raster file or constant value.

output      Output raster file.

wd      Changes the working directory.

verbose\_mode      Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt_erase	<i>Erase</i>
-----------	--------------

---

### Description

Removes all the features, or parts of features, that overlap with the features of the erase vector polygon.

### Usage

```
wbt_erase(  
    input,  
    erase,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input vector file.
erase	Input erase polygon vector file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```
wbt_erase_polygon_from_lidar
    Erase polygon from lidar
```

---

### Description

Erases (cuts out) a vector polygon or polygons from a LiDAR point cloud.

### Usage

```
wbt_erase_polygon_from_lidar(  
    input,  
    polygons,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
polygons	Input vector polygons file.
output	Output LiDAR file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```
wbt_erase_polygon_from_raster
    Erase polygon from raster
```

---

### Description

Erases (cuts out) a vector polygon from a raster.

### Usage

```
wbt_erase_polygon_from_raster(  
    input,  
    polygons,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
polygons	Input vector polygons file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_euclidean\_allocation

*Euclidean allocation*

---

### Description

Assigns grid cells in the output raster the value of the nearest target cell in the input image, measured by the Shih and Wu (2004) Euclidean distance transform.

### Usage

```
wbt_euclidean_allocation(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_euclidean\_distance

*Euclidean distance*

---

### Description

Calculates the Shih and Wu (2004) Euclidean distance transform.

**Usage**

```
wbt_euclidean_distance(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_evaluate_training_sites
    Evaluate training sites
```

---

**Description**

This tool can be used to inspect the overlap in spectral signatures of training sites for various classes.

**Usage**

```
wbt_evaluate_training_sites(
    inputs,
    polys,
    field,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

inputs	Name of the input band images.
polys	Name of the input training site polygons shapefile.
field	Name of the attribute containing class name data.
output	Name of the output report file (*.html).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_exp	<i>Exp</i>
---------	------------

---

**Description**

Returns the exponential (base e) of values in a raster.

**Usage**

```
wbt_exp(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_exp2	<i>Exp2</i>
----------	-------------

---

**Description**

Returns the exponential (base 2) of values in a raster.

**Usage**

```
wbt_exp2(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_export\_table\_to\_csv  
*Export table to csv*

---

### Description

Exports an attribute table to a CSV text file.

### Usage

```
wbt_export_table_to_csv(  
    input,  
    output,  
    headers = TRUE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input vector file.
output	Output csv file.
headers	Export field names as file header?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_exposure\_towards\_wind\_flux  
*Exposure towards wind flux*

---

## Description

Evaluates hydrologic connectivity within a DEM.

## Usage

```
wbt_exposure_towards_wind_flux(  
    dem,  
    output,  
    azimuth = "",  
    max_dist = "",  
    zfactor = "",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

dem	Name of the input DEM raster file.
output	Name of the output raster file.
azimuth	Wind azimuth, in degrees.
max_dist	Optional maximum search distance. Minimum value is 5 x cell size.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.

---

`wbt_extend_vector_lines`*Extend vector lines*

---

### Description

Extends vector lines by a specified distance.

### Usage

```
wbt_extend_vector_lines(  
    input,  
    output,  
    dist,  
    extend = "both ends",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>input</code>	Input vector polyline file.
<code>output</code>	Output vector polyline file.
<code>dist</code>	The distance to extend.
<code>extend</code>	Extend direction, 'both ends' (default), 'line start', 'line end'.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_extract\_nodes      *Extract nodes*

---

### Description

Converts vector lines or polygons into vertex points.

### Usage

```
wbt_extract_nodes(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input vector lines or polygon file.
output	Output vector points file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_extract\_raster\_values\_at\_points  
*Extract raster values at points*

---

### Description

Extracts the values of raster(s) at vector point locations.

**Usage**

```
wbt_extract_raster_values_at_points(
    inputs,
    points,
    out_text = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
points	Input vector points file.
out_text	Output point values as text? Otherwise, the only output is to the points file's attribute table.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_extract\_streams     *Extract streams*

---

**Description**

Extracts stream grid cells from a flow accumulation raster.

**Usage**

```
wbt_extract_streams(
    flow_accum,
    output,
    threshold,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

flow_accum	Input raster D8 flow accumulation file.
output	Output raster file.
threshold	Threshold in flow accumulation values for channelization.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_extract\_valleys    *Extract valleys*

---

### Description

Identifies potential valley bottom grid cells based on local topography alone.

### Usage

```

wbt_extract_valleys(
  dem,
  output,
  variant = "LQ",
  line_thin = TRUE,
  filter = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
variant	Options include 'LQ' (lower quartile), 'JandR' (Johnston and Rosenfeld), and 'PandD' (Peucker and Douglas); default is 'LQ'.
line_thin	Optional flag indicating whether post-processing line-thinning should be performed.
filter	Optional argument (only used when variant='lq') providing the filter size, in grid cells, used for lq-filtering (default is 5).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_farthest\_channel\_head  
*Farthest channel head*

---

**Description**

Calculates the distance to the furthest upstream channel head for each stream cell.

**Usage**

```
wbt_farthest_channel_head(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_fast\_almost\_gaussian\_filter  
*Fast almost gaussian filter*

---

**Description**

Performs a fast approximate Gaussian filter on an image.

**Usage**

```
wbt_fast_almost_gaussian_filter(  
    input,  
    output,  
    sigma = 1.8,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
sigma	Standard deviation distance in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_fd8\_flow\_accumulation  
*Fd8 flow accumulation*

---

**Description**

Calculates an FD8 flow accumulation raster from an input DEM.

**Usage**

```
wbt_fd8_flow_accumulation(
  dem,
  output,
  out_type = "specific contributing area",
  exponent = 1.1,
  threshold = NULL,
  log = FALSE,
  clip = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```



**Arguments**

dem	Input raster DEM file.
output	Output raster file.
out_type	Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
exponent	Optional exponent parameter; default is 1.1.
threshold	Optional convergence threshold parameter, in grid cells; default is infinity.
log	Optional flag to request the output be log-transformed.
clip	Optional flag to request clipping the display max by 1 percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_fd8_pointer	<i>Fd8 pointer</i>
-----------------	--------------------

---

**Description**

Calculates an FD8 flow pointer raster from an input DEM.

**Usage**

```
wbt_fd8_pointer(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_feature\_preserving\_smoothing  
*Feature preserving smoothing*

---

**Description**

Reduces short-scale variation in an input DEM using a modified Sun et al. (2007) algorithm.

**Usage**

```
wbt_feature_preserving_smoothing(  
  dem,  
  output,  
  filter = 11,  
  norm_diff = 15,  
  num_iter = 3,  
  max_diff = 0.5,  
  zfactor = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
filter	Size of the filter kernel.

norm_diff	Maximum difference in normal vectors, in degrees.
num_iter	Number of iterations.
max_diff	Maximum allowable absolute elevation change (optional).
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_fetch\_analysis      *Fetch analysis*

---

**Description**

Performs an analysis of fetch or upwind distance to an obstacle.

**Usage**

```
wbt_fetch_analysis(
  dem,
  output,
  azimuth = 0,
  hgt_inc = 0.05,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
azimuth	Wind azimuth in degrees in degrees.
hgt_inc	Height increment value.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_fill_burn	<i>Fill burn</i>
---------------	------------------

---

**Description**

Burns streams into a DEM using the FillBurn (Saunders, 1999) method.

**Usage**

```
wbt_fill_burn(
  dem,
  streams,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
streams	Input vector streams file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_fill\_depressions *Fill depressions*

---

### Description

Fills all of the depressions in a DEM. Depression breaching should be preferred in most cases.

### Usage

```
wbt_fill_depressions(  
    dem,  
    output,  
    fix_flats = TRUE,  
    flat_increment = NULL,  
    max_depth = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
fix_flats	Optional flag indicating whether flat areas should have a small gradient applied.
flat_increment	Optional elevation increment applied to flat areas.
max_depth	Optional maximum depression depth to fill.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_fill_depressions_planchon_and_darboux`*Fill depressions planchon and darboux*

---

**Description**

Fills all of the depressions in a DEM using the Planchon and Darboux (2002) method.

**Usage**

```
wbt_fill_depressions_planchon_and_darboux(  
    dem,  
    output,  
    fix_flats = TRUE,  
    flat_increment = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>fix_flats</code>	Optional flag indicating whether flat areas should have a small gradient applied.
<code>flat_increment</code>	Optional elevation increment applied to flat areas.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_fill_depressions_wang_and_liu`*Fill depressions wang and liu*

---

**Description**

Fills all of the depressions in a DEM using the Wang and Liu (2006) method. Depression breaching should be preferred in most cases.

**Usage**

```
wbt_fill_depressions_wang_and_liu(  
    dem,  
    output,  
    fix_flats = TRUE,  
    flat_increment = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>fix_flats</code>	Optional flag indicating whether flat areas should have a small gradient applied.
<code>flat_increment</code>	Optional elevation increment applied to flat areas.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_fill\_missing\_data *Fill missing data*

---

### Description

Fills NoData holes in a DEM.

### Usage

```
wbt_fill_missing_data(  
    input,  
    output,  
    filter = 11,  
    weight = 2,  
    no_edges = TRUE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
filter	Filter size (cells).
weight	IDW weight value.
no_edges	Optional flag indicating whether to exclude NoData cells in edge regions.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

```
wbt_fill_single_cell_pits
    Fill single cell pits
```

---

**Description**

Raises pit cells to the elevation of their lowest neighbour.

**Usage**

```
wbt_fill_single_cell_pits(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_filter_lidar    Filter lidar
```

---

**Description**

Filters points within a LiDAR point cloud based on point properties.

**Usage**

```
wbt_filter_lidar(
  input,
  output = NULL,
  statement = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
statement	Filter statement e.g. <code>x &lt; 5000.0 &amp;&amp; y &gt; 100.0 &amp;&amp; is_late &amp;&amp; !is_noise</code> . This statement must be a valid Rust statement.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_filter\_lidar\_classes

*Filter lidar classes*

---

**Description**

Removes points in a LAS file with certain specified class values.

**Usage**

```
wbt_filter_lidar_classes(
  input,
  output,
  exclude_cls = NULL,
  wd = NULL,
  verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input LiDAR file.
output	Output LiDAR file.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_filter_lidar_scan_angles
    Filter lidar scan angles

```

---

### Description

Removes points in a LAS file with scan angles greater than a threshold.

### Usage

```

wbt_filter_lidar_scan_angles(
  input,
  output,
  threshold,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
threshold	Scan angle threshold.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_filter\_raster\_features\_by\_area  
*Filter raster features by area*

---

**Description**

Removes small-area features from a raster.

**Usage**

```
wbt_filter_raster_features_by_area(  
  input,  
  output,  
  threshold,  
  background = "zero",  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
threshold	Remove features with fewer grid cells than this threshold value.
background	Background value.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_find\_flightline\_edge\_points  
*Find flightline edge points*

---

**Description**

Identifies points along a flightline's edge in a LAS file.

**Usage**

```
wbt_find_flightline_edge_points(  
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input LiDAR file.
output	Output file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_find\_lowest\_or\_highest\_points  
*Find lowest or highest points*

---

### Description

Locates the lowest and/or highest valued cells in a raster.

### Usage

```
wbt_find_lowest_or_highest_points(  
    input,  
    output,  
    out_type = "lowest",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output vector points file.
out_type	Output type; one of 'area' (default) and 'volume'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_find\_main\_stem      *Find main stem*

---

### Description

Finds the main stem, based on stream lengths, of each stream network.

### Usage

```
wbt_find_main_stem(  
    d8_pntr,  
    streams,  
    output,  
    esri_pntr = FALSE,  
    zero_background = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```
wbt_find_no_flow_cells
    Find no flow cells
```

---

**Description**

Finds grid cells with no downslope neighbours.

**Usage**

```
wbt_find_no_flow_cells(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_find_parallel_flow
    Find parallel flow
```

---

**Description**

Finds areas of parallel flow in D8 flow direction rasters.



**Usage**

```
wbt_find_parallel_flow(
    d8_pntr,
    streams,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

d8_pntr	Input D8 pointer raster file.
streams	Input raster streams file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_find\_patch\_or\_class\_edge\_cells  
*Find patch or class edge cells*

---

**Description**

Finds all cells located on the edge of patch or class features.

**Usage**

```
wbt_find_patch_or_class_edge_cells(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_find_ridges	<i>Find ridges</i>
-----------------	--------------------

---

**Description**

Identifies potential ridge and peak grid cells.

**Usage**

```
wbt_find_ridges(
  dem,
  output,
  line_thin = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
line_thin	Optional flag indicating whether post-processing line-thinning should be performed.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_fix\_dangling\_arcs *Fix dangling arcs*

---

**Description**

This tool fixes undershot and overshoot arcs, two common topological errors, in an input vector lines file.

**Usage**

```
wbt_fix_dangling_arcs(
    input,
    output,
    dist = "",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input lines vector file.
output	Name of the output lines vector file.
dist	Snap distance, in xy units (metres).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_flatten_lakes	<i>Flatten lakes</i>
-------------------	----------------------

---

### Description

Flattens lake polygons in a raster DEM.

### Usage

```
wbt_flatten_lakes(  
    dem,  
    lakes,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
lakes	Input lakes vector polygons file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_flightline\_overlap  
*Flightline overlap*

---

### Description

Reads a LiDAR (LAS) point file and outputs a raster containing the number of overlapping flight-lines in each grid cell.

### Usage

```
wbt_flightline_overlap(  
    input,  
    output = NULL,  
    resolution = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
output	Output file.
resolution	Output raster's grid resolution.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_flip_image	<i>Flip image</i>
----------------	-------------------

---

### Description

Reflects an image in the vertical or horizontal axis.

### Usage

```
wbt_flip_image(
    input,
    output,
    direction = "vertical",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
direction	Direction of reflection; options include 'v' (vertical), 'h' (horizontal), and 'b' (both).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_flood_order	<i>Flood order</i>
-----------------	--------------------

---

### Description

Assigns each DEM grid cell its order in the sequence of inundations that are encountered during a search starting from the edges, moving inward at increasing elevations.

### Usage

```
wbt_flood_order(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_floor	<i>Floor</i>
-----------	--------------

---

### Description

Returns the largest (closest to positive infinity) value that is less than or equal to the values in a raster.

**Usage**

```
wbt_floor(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_flow\_accumulation\_full\_workflow  
*Flow accumulation full workflow*

---

**Description**

Resolves all of the depressions in a DEM, outputting a breached DEM, an aspect-aligned non-divergent flow pointer, and a flow accumulation raster.

**Usage**

```
wbt_flow_accumulation_full_workflow(
    dem,
    out_dem,
    out_pntr,
    out_accum,
    out_type = "Specific Contributing Area",
    log = FALSE,
    clip = FALSE,
```



```

    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

dem	Input raster DEM file.
out_dem	Output raster DEM file.
out_pntr	Output raster flow pointer file.
out_accum	Output raster flow accumulation file.
out_type	Output type; one of 'cells', 'sca' (default), and 'ca'.
log	Optional flag to request the output be log-transformed.
clip	Optional flag to request clipping the display max by 1 percent.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_flow\_length\_diff *Flow length diff*

---

### Description

Calculates the local maximum absolute difference in downslope flowpath length, useful in mapping drainage divides and ridges.

### Usage

```

wbt_flow_length_diff(
  d8_pntr,
  output,
  esri_pntr = FALSE,
  wd = NULL,

```

```
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
  )
```

### Arguments

d8_pntr	Input D8 pointer raster file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_gamma\_correction *Gamma correction*

---

### Description

Performs a gamma correction on an input images.

### Usage

```
wbt_gamma_correction(  
  input,  
  output,  
  gamma = 0.5,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
gamma	Gamma value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_gaussian\_contrast\_stretch  
*Gaussian contrast stretch*

---

**Description**

Performs a Gaussian contrast stretch on input images.

**Usage**

```
wbt_gaussian_contrast_stretch(
  input,
  output,
  num_tones = 256,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
num_tones	Number of tones in the output image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

wbt\_gaussian\_curvature  
*Gaussian curvature*

**Description**

Calculates a mean curvature raster from an input DEM.

**Usage**

```
wbt_gaussian_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_gaussian\_filter     *Gaussian filter*

---

### Description

Performs a Gaussian filter on an image.

### Usage

```
wbt_gaussian_filter(  
    input,  
    output,  
    sigma = 0.75,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
sigma	Standard deviation distance in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_gaussian\_scale\_space  
*Gaussian scale space*

---

### Description

Uses the fast Gaussian approximation algorithm to produce scaled land-surface parameter measurements from an input DEM.

### Usage

```
wbt_gaussian_scale_space(
    dem,
    output,
    output_zscore,
    output_scale,
    points = NULL,
    sigma = 0.5,
    step = 0.5,
    num_steps = 10,
    lsp = "Slope",
    z_factor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

dem	Name of the input DEM raster file.
output	Name of the output land-surface parameter raster file.
output_zscore	Name of the output z-score raster file.
output_scale	Name of the output scale raster file.
points	Name of the input vector points shapefile.
sigma	Initial sigma value (cells).
step	Step size as any positive non-zero integer.
num_steps	Number of steps.
lsp	Output land-surface parameter; one of 'AnisotropyLTP', 'Aspect', 'DiffMeanElev', 'Eastness', 'Elevation', 'Hillshade', 'MeanCurvature', 'Northness', 'PlanCurvature', 'ProfileCurvature', 'Ruggedness', 'Slope', 'TanCurvature', 'TotalCurvature'.
z_factor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_generalize\_classified\_raster  
*Generalize classified raster*

---

**Description**

Generalizes a raster containing class or object features by removing small features.

**Usage**

```
wbt_generalize_classified_raster(
  input,
  output,
  min_size = 4,
  method = "longest",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Name of the input raster image file.
output	Name of the output raster file.
min_size	Minimum feature size, in grid cells.
method	Grouping method; one of 'longest' (default), 'largest', and 'nearest'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_generalize\_with\_similarity  
*Generalize with similarity*

---

**Description**

Generalizes a raster containing class or object features by removing small features using similarity criteria of neighbouring features.

**Usage**

```
wbt_generalize_with_similarity(  
    input,  
    similarity,  
    output,  
    min_size = 4,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Name of the input raster image file.
similarity	Names of the input similarity images.
output	Name of the output raster file.
min_size	Minimum feature size, in grid cells.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

`wbt_generating_function`*Generating function*

---

**Description**

This tool calculates generating function from an input DEM.

**Usage**

```
wbt_generating_function(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Name of the input raster DEM file.
<code>output</code>	Name of the output raster image file.
<code>log</code>	Display output values using a log-scale.
<code>zfactor</code>	Z conversion factor.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_geomorphons      *Geomorphons*

---

### Description

Computes geomorphon patterns.

### Usage

```
wbt_geomorphons(
  dem,
  output,
  search = 50,
  threshold = 0,
  fdist = 0,
  skip = 0,
  forms = TRUE,
  residuals = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
search	Look up distance (in cells).
threshold	Flatness threshold for the classification function (in degrees).
fdist	Distance (in cells) to begin reducing the flatness threshold to avoid problems with pseudo-flat lines-of-sight.
skip	Distance (in cells) to begin calculating lines-of-sight.
forms	Classify geomorphons into 10 common land morphologies, else output ternary pattern.
residuals	Convert elevation to residuals of a linear model.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_greater_than	<i>Greater than</i>
------------------	---------------------

---

**Description**

Performs a greater-than comparison operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_greater_than(
    input1,
    input2,
    output,
    incl_equals = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
incl_equals	Perform a greater-than-or-equal-to operation.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_hack\_stream\_order *Hack stream order*

---

### Description

Assigns the Hack stream order to each tributary in a stream network.

### Usage

```
wbt_hack_stream_order(
    d8_pntr,
    streams,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_heat_map	<i>Heat map</i>
--------------	-----------------

---

### Description

Calculates a heat map, or kernel density estimation (KDE), for an input point set.

### Usage

```
wbt_heat_map(
  input,
  output,
  weight_field = NULL,
  bandwidth = "",
  kernel = "quartic",
  cell_size = "",
  base = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Name of the input points shapefile.
output	Name of the output raster image file.
weight_field	Optional name of the attribute containing point weight.
bandwidth	Bandwidth (metres).
kernel	Kernel type; one of 'uniform', 'triangular', 'epanechnikov', 'quartic', 'triweight', 'tricube', 'gaussian', 'cosine', 'logistic', 'sigmoid', 'silverman'.
cell_size	Optionally specified cell size of output raster, in metres. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_height\_above\_ground  
*Height above ground*

---

**Description**

Normalizes a LiDAR point cloud, providing the height above the nearest ground-classified point.

**Usage**

```
wbt_height_above_ground(  
    input,  
    output = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input LiDAR file (including extension).
output	Output lidar file (including extension).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_help *Help description for 'WhiteboxTools'*

---

**Description**

Help description for 'WhiteboxTools'

**Usage**

```
wbt_help()
```

**Value**

Returns the help description for 'WhiteboxTools' as an R character vector.

**Examples**

```
## Not run:
wbt_help()

## End(Not run)
```

---

wbt\_highest\_position *Highest position*

---

**Description**

Identifies the stack position of the maximum value within a raster stack on a cell-by-cell basis.

**Usage**

```
wbt_highest_position(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_high\_pass\_bilateral\_filter  
*High pass bilateral filter*

---

### Description

Performs a high-pass bilateral filter, by differencing an input image by the bilateral filter by Tomasi and Manduchi (1998).

### Usage

```
wbt_high_pass_bilateral_filter(  
    input,  
    output,  
    sigma_dist = 0.75,  
    sigma_int = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
sigma_dist	Standard deviation in distance in pixels.
sigma_int	Standard deviation in intensity in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_high\_pass\_filter *High pass filter*

---

### Description

Performs a high-pass filter on an input image.

### Usage

```
wbt_high_pass_filter(  
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_high_pass_median_filter`*High pass median filter*

---

### Description

Performs a high pass median filter on an input image.

### Usage

```
wbt_high_pass_median_filter(  
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    sig_digits = 2,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>input</code>	Input raster file.
<code>output</code>	Output raster file.
<code>filterx</code>	Size of the filter kernel in the x-direction.
<code>filtery</code>	Size of the filter kernel in the y-direction.
<code>sig_digits</code>	Number of significant digits.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_hillshade`*Hillshade*

---

### Description

Calculates a hillshade raster from an input DEM.

### Usage

```
wbt_hillshade(  
  dem,  
  output,  
  azimuth = 315,  
  altitude = 30,  
  zfactor = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

### Arguments

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>azimuth</code>	Illumination source azimuth in degrees.
<code>altitude</code>	Illumination source altitude in degrees.
<code>zfactor</code>	Optional multiplier for when the vertical and horizontal units are not the same.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_hillslopes	<i>Hillslopes</i>
----------------	-------------------

---

### Description

Identifies the individual hillslopes draining to each link in a stream network.

### Usage

```
wbt_hillslopes(  
    d8_pntr,  
    streams,  
    output,  
    esri_pntr = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_histogram\_equalization  
*Histogram equalization*

---

### Description

Performs a histogram equalization contrast enhancement on an image.

### Usage

```
wbt_histogram_equalization(  
    input,  
    output,  
    num_tones = 256,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
num_tones	Number of tones in the output image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_histogram_matching`*Histogram matching*

---

**Description**

Alters the statistical distribution of a raster image matching it to a specified PDF.

**Usage**

```
wbt_histogram_matching(  
    input,  
    histo_file,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input raster file.
<code>histo_file</code>	Input reference probability distribution function (pdf) text file.
<code>output</code>	Output raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_histogram\_matching\_two\_images  
*Histogram matching two images*

---

**Description**

Alters the cumulative distribution function of a raster image to that of another image.

**Usage**

```
wbt_histogram_matching_two_images(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input1	Input raster file to modify.
input2	Input reference raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_hole\_proportion     *Hole proportion*

---

### Description

Calculates the proportion of the total area of a polygon's holes relative to the area of the polygon's hull.

### Usage

```
wbt_hole_proportion(
    input,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_horizontal\_excess\_curvature  
*Horizontal excess curvature*

---

### Description

This tool calculates horizontal excess curvature from an input DEM.



**Usage**

```
wbt_horizontal_excess_curvature(
    dem,
    output,
    log = FALSE,
    zfactor = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
log	Display output values using a log-scale.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_horizon_angle	<i>Horizon angle</i>
-------------------	----------------------

---

**Description**

Calculates horizon angle (maximum upwind slope) for each grid cell in an input DEM.

**Usage**

```
wbt_horizon_angle(
    dem,
    output,
    azimuth = 0,
    max_dist = 100,
    wd = NULL,
```

```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
azimuth	Azimuth, in degrees.
max_dist	Optional maximum search distance (unspecified if none; in xy units). Minimum value is 5 x cell size.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_horton\_stream\_order  
*Horton stream order*

---

### Description

Assigns the Horton stream order to each tributary in a stream network.

### Usage

```

wbt_horton_stream_order(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_hydrologic\_connectivity  
*Hydrologic connectivity*

---

**Description**

This tool evaluates hydrologic connectivity within a DEM.

**Usage**

```
wbt_hydrologic_connectivity(
  dem,
  output1,
  output2,
  exponent = 1,
  threshold = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Name of the input DEM raster file; must be depressionless.
output1	Name of the output downslope unsaturated length (DUL) file.
output2	Name of the output upslope disconnected saturated area (UDSA) file.
exponent	Optional exponent parameter; default is 1.0.
threshold	Optional convergence threshold parameter, in grid cells; default is infinity.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_hypsometrically\_tinted\_hillshade  
*Hypsometrically tinted hillshade*

---

**Description**

Creates an colour shaded relief image from an input DEM.

**Usage**

```
wbt_hypsometrically_tinted_hillshade(  
  dem,  
  output,  
  altitude = 45,  
  hs_weight = 0.5,  
  brightness = 0.5,  
  atmospheric = 0,  
  palette = "atlas",  
  reverse = FALSE,  
  zfactor = NULL,  
  full_mode = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
altitude	Illumination source altitude in degrees.
hs_weight	Weight given to hillshade relative to relief (0.0-1.0).
brightness	Brightness factor (0.0-1.0).
atmospheric	Atmospheric effects weight (0.0-1.0).
palette	Options include 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'purple', 'viridis', 'gn_yl', 'pi_y_g', 'bl_yl_rd', and 'deep'.
reverse	Optional flag indicating whether to use reverse the palette.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
full_mode	Optional flag indicating whether to use full 360-degrees of illumination sources.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_hypsometric\_analysis  
*Hypsometric analysis*

---

**Description**

Calculates a hypsometric curve for one or more DEMs.

**Usage**

```
wbt_hypsometric_analysis(  
  inputs,  
  output,  
  watershed = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

inputs	Input DEM files.
output	Output HTML file (default name will be based on input file if unspecified).
watershed	Input watershed files (optional).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_idw\_interpolation *Idw interpolation*

---

**Description**

Interpolates vector points into a raster surface using an inverse-distance weighted scheme.

**Usage**

```
wbt_idw_interpolation(  
  input,  
  field,  
  output,  
  use_z = FALSE,  
  weight = 2,  
  radius = NULL,  
  min_points = NULL,  
  cell_size = NULL,  
  base = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input vector Points file.
field	Input field name in attribute table.
output	Output raster file.
use_z	Use z-coordinate instead of field?.
weight	IDW weight value.
radius	Search Radius in map units.
min_points	Minimum number of points.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_ihs_to_rgb	<i>Ihs to rgb</i>
----------------	-------------------

---

**Description**

Converts intensity, hue, and saturation (IHS) images into red, green, and blue (RGB) images.

**Usage**

```
wbt_ihs_to_rgb(
  intensity,
  hue,
  saturation,
  red = NULL,
  green = NULL,
  blue = NULL,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

intensity	Input intensity file.
hue	Input hue file.
saturation	Input saturation file.
red	Output red band file. Optionally specified if colour-composite not specified.
green	Output green band file. Optionally specified if colour-composite not specified.
blue	Output blue band file. Optionally specified if colour-composite not specified.
output	Output colour-composite file. Only used if individual bands are not specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_image\_autocorrelation  
*Image autocorrelation*

---

**Description**

Performs Moran's I analysis on two or more input images.

**Usage**

```
wbt_image_autocorrelation(  
  inputs,  
  output,  
  contiguity = "Rook",  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```



**Arguments**

inputs	Input raster files.
output	Output HTML file (default name will be based on input file if unspecified).
contiguity	Contiguity type.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_image\_correlation *Image correlation*

---

**Description**

Performs image correlation on two or more input images.

**Usage**

```
wbt_image_correlation(
  inputs,
  output = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
output	Output HTML file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_image\_correlation\_neighbourhood\_analysis  
*Image correlation neighbourhood analysis*

---

**Description**

Performs image correlation on two input images neighbourhood search windows.

**Usage**

```
wbt_image_correlation_neighbourhood_analysis(  
    input1,  
    input2,  
    output1,  
    output2,  
    filter = 11,  
    stat = "pearson",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input1	Input raster file.
input2	Input raster file.
output1	Output correlation (r-value or rho) raster file.
output2	Output significance (p-value) raster file.
filter	Size of the filter kernel.
stat	Correlation type; one of 'pearson' (default) and 'spearman'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_image\_regression *Image regression*

---

### Description

Performs image regression analysis on two input images.

### Usage

```
wbt_image_regression(  
    input1,  
    input2,  
    output,  
    out_residuals = NULL,  
    standardize = FALSE,  
    scattergram = FALSE,  
    num_samples = 1000,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input raster file (independent variable, X).
input2	Input raster file (dependent variable, Y).
output	Output HTML file for regression summary report.
out_residuals	Output raster regression residual file.
standardize	Optional flag indicating whether to standardize the residuals map.
scattergram	Optional flag indicating whether to output a scattergram.
num_samples	Number of samples used to create scattergram.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_image_segmentation`*Image segmentation*

---

**Description**

Performs a region-growing based segmentation on a set of multi-spectral images.

**Usage**

```
wbt_image_segmentation(  
    inputs,  
    output,  
    threshold = 0.5,  
    steps = 10,  
    min_area = 4,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>inputs</code>	Names of the input band images.
<code>output</code>	Name of the output raster file.
<code>threshold</code>	Distance threshold, in z-scores.
<code>steps</code>	Number of steps.
<code>min_area</code>	Minimum object area, in grid cells (1-8).
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_image_slider	<i>Image slider</i>
------------------	---------------------

---

## Description

This tool creates an image slider from two input images.

## Usage

```
wbt_image_slider(
    input1,
    input2,
    output,
    palette1 = "grey",
    reverse1 = FALSE,
    label1 = "",
    palette2 = "grey",
    reverse2 = FALSE,
    label2 = "",
    height = 600,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

## Arguments

input1	Name of the left input image file.
input2	Name of the right input image file.
output	Name of the output HTML file (*.html).
palette1	Left image palette; options are 'grey', 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'rgb'.
reverse1	Reverse left image palette?.
label1	Left image label (leave blank for none).
palette2	Right image palette; options are 'grey', 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'rgb'.
reverse2	Reverse right image palette?.
label2	Right image label (leave blank for none).
height	Image height, in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_image\_stack\_profile  
*Image stack profile*

---

**Description**

Plots an image stack profile (i.e. signature) for a set of points and multispectral images.

**Usage**

```
wbt_image_stack_profile(  
  inputs,  
  points,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

inputs	Input multispectral image files.
points	Input vector points file.
output	Output HTML file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_impoundment\_size\_index  
*Impoundment size index*

---

### Description

Calculates the impoundment size resulting from damming a DEM.

### Usage

```
wbt_impoundment_size_index(  

    dem,  

    damlength,  

    out_mean = NULL,  

    out_max = NULL,  

    out_volume = NULL,  

    out_area = NULL,  

    out_dam_height = NULL,  

    wd = NULL,  

    verbose_mode = FALSE,  

    compress_rasters = FALSE,  

    command_only = FALSE  

)
```

### Arguments

dem	Input raster DEM file.
damlength	Maximum length of the dam.
out_mean	Output mean flooded depth file.
out_max	Output maximum flooded depth file.
out_volume	Output flooded volume file.
out_area	Output flooded area file.
out_dam_height	Output dam height file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_increment	<i>Increment</i>
---------------	------------------

---

### Description

Increases the values of each grid cell in an input raster by 1.0. (see also InPlaceAdd).

### Usage

```
wbt_increment(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_individual_tree_detection	<i>Individual tree detection</i>
-------------------------------	----------------------------------

---

### Description

Identifies points in a LiDAR point cloud that are associated with the tops of individual trees.



**Usage**

```
wbt_individual_tree_detection(  
    input,  
    output = NULL,  
    min_search_radius = 1,  
    min_height = 0,  
    max_search_radius = "",  
    max_height = "",  
    only_use_veg = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Name of the input LiDAR file.
output	Name of the output vector points file.
min_search_radius	Minimum search radius (m).
min_height	Minimum height (m).
max_search_radius	Maximum search radius (m).
max_height	Maximum height (m).
only_use_veg	Only use veg. class points?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_init	<i>Initialize 'WhiteboxTools'</i>
----------	-----------------------------------

---

### Description

wbt\_init(): Check if a suitable 'WhiteboxTools' executable is present. Search default path in package directory or set it manually with exe\_path.

wbt\_options(): Get/set package options

- whitebox.exe\_path - character. Path to executable file. The default value is the package installation directory, subdirectory "WBT", followed by whitebox\_tools.exe or whitebox\_tools. Set the whitebox.exe\_path option using wbt\_init() exe\_path argument
- whitebox.wd - character. Path to WhiteboxTools working directory. Used as --wd argument for tools that support it when wd is not specified elsewhere.
- whitebox.verbose - logical. Should standard output from calls to executable be cat() out for readability? Default is result of interactive(). Individual tools may have verbose\_mode setting that produce only single-line output when FALSE. These argument values are left as the defaults defined in the package documentation for that function. When whitebox.verbose=FALSE no output is produced. Set the value of whitebox.verbose with wbt\_verbose() verbose argument.
- whitebox.compress\_rasters - logical. Should raster output from WhiteboxTools be compressed? Default: FALSE. Set the value of whitebox.compress\_rasters with wbt\_compress\_rasters() compress\_rasters argument.
- whitebox.max\_procs - integer. Maximum number of processes for tools that run in parallel or partially parallelize. Default: -1 uses all of the available cores.

wbt\_exe\_path(): Get the file path of the 'WhiteboxTools' executable.

wbt\_default\_path(): Get the default file path of the 'WhiteboxTools' executable.

wbt\_data\_dir(): Get the directory where 'WhiteboxTools' data are stored.

wbt\_wd(): Get or set the 'WhiteboxTools' working directory. Default: "" (unset) is your R working directory if no other options are set.

wbt\_verbose(): Check verbose options for 'WhiteboxTools'

wbt\_compress\_rasters(): Check raster compression option for 'WhiteboxTools'. Default: FALSE

wbt\_max\_procs(): Check maximum number of processes for for tools that run in parallel or partially parallelize. Default: -1 uses all of the available cores.

### Usage

```
wbt_init(
  exe_path = wbt_exe_path(shell_quote = FALSE),
  ...,
  check_version = TRUE
)
```

```

wbt_options(
  exe_path = NULL,
  wd = NULL,
  verbose = NULL,
  compress_rasters = NULL,
  max_procs = NULL
)

wbt_exe_path(exe_path = NULL, shell_quote = TRUE)

wbt_default_path()

wbt_data_dir()

wbt_wd(wd = NULL)

wbt_verbose(verbose = NULL)

wbt_compress_rasters(compress_rasters = NULL)

wbt_max_procs(max_procs = NULL)

```

### Arguments

exe_path	Optional: User-supplied path to 'WhiteboxTools' executable. Default: NULL
...	additional arguments to wbt_options()
check_version	Check version of 'WhiteboxTools' installed against version R package was built for? Default: TRUE
wd	character; Default: NULL; if set the package option whitebox.wd is set specified path (if directory exists)
verbose	Default: NULL; if logical, set the package option whitebox.verbose to specified value
compress_rasters	Default: NULL; if logical, set the package option whitebox.compress_rasters to specified value
max_procs	Default: NULL; if integer, set the package option whitebox.max_procs to specified value
shell_quote	Return shQuote() result?

### Details

wbt\_exe\_path(): Checks system environment variable R\_WHITEBOX\_EXE\_PATH and package option whitebox.exe\_path. Set your desired path with either Sys.setenv(R\_WHITEBOX\_EXE\_PATH = "C:/path/to/whitebox\_tools.exe") or options(whitebox.exe\_path = "C:/path/to/whitebox\_tools.exe"). The default, backwards-compatible path is returned by wbt\_default\_path()

wbt\_default\_path(): Returns a path to 'WhiteboxTools' executable including a platform-specific executable (with or without .exe extension)

`wbt_data_dir()`: Uses platform-specific user data directory from `tools::R_user_dir(package = "whitebox", which = "data")` on R 4.0+. On R<4 returns the original default `find.package("whitebox")`.

`wbt_wd()`: Before you set the working directory in a session the default output will be in your current R working directory unless otherwise specified. You can change working directory at any time by setting the `wd` argument to `wbt_wd()` and running a tool. Note that once you have set a working directory, the directory needs to be set somewhere to "replace" the old value; just dropping the flag will not change the working directory back to the R working directory. To "unset" the option in the R package you can use `wbt_wd("")` which is equivalent to `wbt_wd(getwd())`.

### Value

`wbt_init()`: logical; TRUE if binary file is found at `exe_path`

`wbt_options()`: an invisible list containing current `whitebox.exe_path`, `whitebox.verbose`, `whitebox.compress_rasters`, and `whitebox.max_procs` options

Returns the file path of 'WhiteboxTools' executable.

`wbt_wd()`: character; when working directory is unset, will not add `--wd=` arguments to calls and should be the same as using `getwd()`. See Details.

`wbt_verbose()`: logical; defaults to result of `interactive()`

`wbt_compress_rasters()`: logical; defaults to NA

`wbt_max_procs()`: integer; defaults to `NA_integer_`

### See Also

[install\\_whitebox\(\)](#) [whitebox](#)

### Examples

```
## Not run:
## wbt_init():

# set path to binary as an argument
# wbt_init(exe_path = "not/a/valid/path/whitebox_tools.exe")

## End(Not run)
## Not run:

## wbt_options():

# set multiple options (e.g. exe_path and verbose) with wbt_options()
wbt_options(exe_path = "not/a/valid/path/whitebox_tools.exe", verbose = TRUE)

## End(Not run)
## Not run:
wbt_exe_path()

## End(Not run)
## Not run:
```

```
## wbt_wd():  
  
# set WBT working directory to R working directory  
wbt_wd(wd = getwd())  
  
## End(Not run)  
## Not run:  
  
## wbt_verbose():  
  
wbt_verbose(verbose = TRUE)  
  
## End(Not run)  
## Not run:  
  
## wbt_compress_rasters():  
  
wbt_compress_rasters(compress_rasters = TRUE)  
  
## End(Not run)  
## Not run:  
  
## wbt_max_procs():  
  
wbt_max_procs(max_procs = 2)  
  
## End(Not run)
```

---

wbt\_insert\_dams

*Insert dams*

---

## Description

Calculates the impoundment size resulting from damming a DEM.

## Usage

```
wbt_insert_dams(  
  dem,  
  dam_pts,  
  output,  
  damlength,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
dam_pts	Input vector dam points file.
output	Output file.
damlength	Maximum length of the dam.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_install	<i>Download and Install 'WhiteboxTools'</i>
-------------	---

---

**Description**

This function downloads the 'WhiteboxTools' binary if needed. Pre-compiled binaries are only available for download for 64-bit Linux (Ubuntu 20.04), Windows and Mac OS (Intel) platforms. If you need WhiteboxTools for another platform follow the instructions here: <https://github.com/jblindsay/whitebox-tools>

**Usage**

```
wbt_install(pkg_dir = wbt_data_dir(), force = FALSE, remove = FALSE)

install_whitebox(pkg_dir = wbt_data_dir(), force = FALSE, remove = FALSE)

wbt_install_extension(
  extension = c("GeneralToolsetExtension", "AgricultureToolset",
    "DemAndSpatialHydrologyToolset", "LidarAndRemoteSensingToolset"),
  destdir = dirname(wbt_exe_path(shell_quote = FALSE))
)
```

**Arguments**

pkg_dir	default install path is to whitebox package "WBT" folder
force	logical. Force install? Default FALSE. When remove=TRUE passed to unlink() to change permissions to allow removal of files/directories.
remove	logical. Remove contents of "WBT" folder from pkg_dir? Default: FALSE
extension	Extension name
destdir	Directory to create /plugins/ directory for extracting extensions

**Details**

'WhiteboxTools' and all of its extensions can be uninstalled by passing the remove=TRUE argument.

**Value**

Prints out the location of the WhiteboxTools binary, if found. NULL otherwise.

**Examples**

```
## Not run:
install_whitebox()

## End(Not run)
```

---

```
wbt_install_wb_extension
      Install wb extension
```

---

**Description**

Use to install a Whitebox extension product.

**Usage**

```
wbt_install_wb_extension(
  install_extension = "General Toolset Extension",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

install_extension	Name of the extension product to install. Options include: 'General Toolset Extension', 'DEM & Spatial Hydrology Extension', 'Lidar & Remote Sensing Extension', and 'Agriculture Extension'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_integer\_division *Integer division*

---

**Description**

Performs an integer division operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_integer_division(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt\_integral\_image      *Integral image*

---

### Description

Transforms an input image (summed area table) into its integral image equivalent.

### Usage

```
wbt_integral_image(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_intersect      *Intersect*

---

### Description

Identifies the parts of features in common between two input vector layers.

**Usage**

```
wbt_intersect(
  input,
  overlay,
  output,
  snap = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector file.
overlay	Input overlay vector file.
output	Output vector file.
snap	Snap tolerance.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_inverse\_principal\_component\_analysis  
*Inverse principal component analysis*

---

**Description**

This tool performs an inverse principal component analysis on a series of input component images.

**Usage**

```
wbt_inverse_principal_component_analysis(
  inputs,
  report,
  wd = NULL,
  verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

inputs	Name of the input PCA component images.
report	Name of the PCA report file (*.html).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_in_place_add	<i>In place add</i>
------------------	---------------------

---

### Description

Performs an in-place addition operation (input1 += input2).

### Usage

```

wbt_in_place_add(
  input1,
  input2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

### Arguments

input1	Input raster file.
input2	Input raster file or constant value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_in\_place\_divide      *In place divide*

---

**Description**

Performs an in-place division operation (input1 /= input2).

**Usage**

```
wbt_in_place_divide(
    input1,
    input2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1      Input raster file.

input2      Input raster file or constant value.

wd      Changes the working directory.

verbose\_mode      Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_in\_place\_multiply *In place multiply*

---

### Description

Performs an in-place multiplication operation (input1 \*= input2).

### Usage

```
wbt_in_place_multiply(  
    input1,  
    input2,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input raster file.
input2	Input raster file or constant value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_in\_place\_subtract *In place subtract*

---

### Description

Performs an in-place subtraction operation (input1 -= input2).

**Usage**

```
wbt_in_place_subtract(
    input1,
    input2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file.
input2	Input raster file or constant value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_isobasins	<i>Isobasins</i>
---------------	------------------

---

**Description**

Divides a landscape into nearly equal sized drainage basins (i.e. watersheds).

**Usage**

```
wbt_isobasins(
    dem,
    output,
    size,
    connections = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
size	Target basin size, in grid cells.
connections	Output upstream-downstream flow connections among basins?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_is_no_data	<i>Is no data</i>
----------------	-------------------

---

**Description**

Identifies NoData valued pixels in an image.

**Usage**

```
wbt_is_no_data(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_jenson\_snap\_pour\_points  
*Jenson snap pour points*

---

**Description**

Moves outlet points used to specify points of interest in a watershedding operation to the nearest stream cell.

**Usage**

```
wbt_jenson_snap_pour_points(  

    pour_pts,  

    streams,  

    output,  

    snap_dist,  

    wd = NULL,  

    verbose_mode = FALSE,  

    compress_rasters = FALSE,  

    command_only = FALSE  

)
```

**Arguments**

pour_pts	Input vector pour points (outlet) file.
streams	Input raster streams file.
output	Output vector file.
snap_dist	Maximum snap distance in map units.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt_join_tables	<i>Join tables</i>
-----------------	--------------------

---

**Description**

Merge a vector's attribute table with another table based on a common field.

**Usage**

```
wbt_join_tables(  
  input1,  
  pkey,  
  input2,  
  fkey,  
  import_field = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input1	Input primary vector file (i.e. the table to be modified).
pkey	Primary key field.
input2	Input foreign vector file (i.e. source of data to be imported).
fkey	Foreign key field.
import_field	Imported field (all fields will be imported if not specified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_kappa_index	<i>Kappa index</i>
-----------------	--------------------

---

### Description

Performs a kappa index of agreement (KIA) analysis on two categorical raster files.

### Usage

```
wbt_kappa_index(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input classification raster file.
input2	Input reference raster file.
output	Output HTML file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_knn\_classification  
*Knn classification*

---

### Description

Performs a supervised k-nearest neighbour classification using training site polygons/points and predictor rasters.

### Usage

```
wbt_knn_classification(
  inputs,
  training,
  field,
  scaling = "Normalize",
  output = NULL,
  k = 5,
  clip = TRUE,
  test_proportion = 0.2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

inputs	Names of the input predictor rasters.
training	Name of the input training site polygons/points shapefile.
field	Name of the attribute containing class name data.
scaling	Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output	Name of the output raster file.
k	k-parameter, which determines the number of nearest neighbours used.
clip	Perform training data clipping to remove outlier pixels?.
test_proportion	The proportion of the dataset to include in the test split; default is 0.2.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_knn_regression	<i>Knn regression</i>
--------------------	-----------------------

---

**Description**

Performs a supervised k-nearest neighbour regression using training site points and predictor rasters.

**Usage**

```
wbt_knn_regression(
  inputs,
  training,
  field,
  scaling = "Normalize",
  output = NULL,
  k = 5,
  weight = TRUE,
  test_proportion = 0.2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Names of the input predictor rasters.
training	Name of the input training site points Shapefile.
field	Name of the attribute containing response variable name data.
scaling	Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output	Name of the output raster file.
k	k-parameter, which determines the number of nearest neighbours used.
weight	Use distance weighting?.
test_proportion	The proportion of the dataset to include in the test split; default is 0.2.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_ks\_test\_for\_normality  
*Ks test for normality*

---

**Description**

Evaluates whether the values in a raster are normally distributed.

**Usage**

```
wbt_ks_test_for_normality(
    input,
    output,
    num_samples = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output HTML file.
num_samples	Number of samples. Leave blank to use whole image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_k\_means\_clustering

*K means clustering*


---

### Description

Performs a k-means clustering operation on a multi-spectral dataset.

### Usage

```
wbt_k_means_clustering(
    inputs,
    output,
    classes,
    out_html = NULL,
    max_iterations = 10,
    class_change = 2,
    initialize = "diagonal",
    min_class_size = 10,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

inputs	Input raster files.
output	Output raster file.
classes	Number of classes.
out_html	Output HTML report file.
max_iterations	Maximum number of iterations.
class_change	Minimum percent of cells changed between iterations before completion.
initialize	How to initialize cluster centres?.
min_class_size	Minimum class size, in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_k_nearest_mean_filter
      K nearest mean filter
```

---

**Description**

A k-nearest mean filter is a type of edge-preserving smoothing filter.

**Usage**

```
wbt_k_nearest_mean_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    k = 5,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
k	k-value in pixels; this is the number of nearest-valued neighbours to use.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_laplacian\_filter *Laplacian filter*

---

### Description

Performs a Laplacian filter on an image.

### Usage

```
wbt_laplacian_filter(  
    input,  
    output,  
    variant = "3x3(1)",  
    clip = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
variant	Optional variant value. Options include 3x3(1), 3x3(2), 3x3(3), 3x3(4), 5x5(1), and 5x5(2) (default is 3x3(1)).
clip	Optional amount to clip the distribution tails by, in percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_laplacian\_of\_gaussian\_filter  
*Laplacian of gaussian filter*

---

### Description

Performs a Laplacian-of-Gaussian (LoG) filter on an image.

### Usage

```
wbt_laplacian_of_gaussian_filter(  
    input,  
    output,  
    sigma = 0.75,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
sigma	Standard deviation in pixels.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_las\_to\_ascii      *Las to ascii*

---

### Description

Converts one or more LAS files into ASCII text files.

### Usage

```
wbt_las_to_ascii(
    inputs,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

inputs	Input LiDAR files.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_las\_to\_laz      *Las to laz*

---

### Description

This tool converts one or more LAS files into the LAZ format.

**Usage**

```
wbt_las_to_laz(
    input,
    output = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input LAS files (leave blank to use all LAS files in WorkingDirectory).
output	Output LAZ file (including extension).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_las_to_multipoint_shapefile
    Las to multipoint shapefile
```

---

**Description**

Converts one or more LAS files into MultipointZ vector Shapefiles. When the input parameter is not specified, the tool grids all LAS files contained within the working directory.

**Usage**

```
wbt_las_to_multipoint_shapefile(
    input,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_las\_to\_shapefile *Las to shapefile*

---

**Description**

Converts one or more LAS files into a vector Shapefile of POINT ShapeType.

**Usage**

```
wbt_las_to_shapefile(  
  input,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input LiDAR file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_las\_to\_zlidar      *Las to zlidar*

---

### Description

Converts one or more LAS files into the zlidar compressed LiDAR data format.

### Usage

```
wbt_las_to_zlidar(
  inputs = NULL,
  outdir = NULL,
  compress = "brotli",
  level = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

inputs	Input LAS files.
outdir	Output directory into which zlidar files are created. If unspecified, it is assumed to be the same as the inputs.
compress	Compression method, including 'brotli' and 'deflate'.
level	Compression level (1-9).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_launch\_wb\_runner    *Launch wb runner*

---

### Description

Opens the Whitebox Runner application.

### Usage

```
wbt_launch_wb_runner(
    clear_app_state = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

clear_app_state	Clear the application state memory?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_layer\_footprint    *Layer footprint*

---

### Description

Creates a vector polygon footprint of the area covered by a raster grid or vector layer.

**Usage**

```
wbt_layer_footprint(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster or vector file.
output	Output vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_laz_to_las	<i>Laz to las</i>
----------------	-------------------

---

**Description**

This tool converts one or more LAZ files into the LAS format.

**Usage**

```
wbt_laz_to_las(
    input,
    output = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input LAZ files (leave blank to use all LAZ files in WorkingDirectory).
output	Output LAS file (including extension).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_lee_sigma_filter` *Lee sigma filter*

---

**Description**

Performs a Lee (Sigma) smoothing filter on an image.

**Usage**

```
wbt_lee_sigma_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  sigma = 10,
  m = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.



sigma	Sigma value should be related to the standard deviation of the distribution of image speckle noise.
m	M-threshold value the minimum allowable number of pixels within the intensity range.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_length\_of\_upstream\_channels  
*Length of upstream channels*

---

**Description**

Calculates the total length of channels upstream.

**Usage**

```
wbt_length_of_upstream_channels(  
  d8_pntr,  
  streams,  
  output,  
  esri_pntr = FALSE,  
  zero_background = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.

zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_less_than	<i>Less than</i>
---------------	------------------

---

**Description**

Performs a less-than comparison operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_less_than(
  input1,
  input2,
  output,
  incl_equals = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
incl_equals	Perform a less-than-or-equal-to operation.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_license	<i>License information for 'WhiteboxTools'</i>
-------------	--

---

**Description**

License information for 'WhiteboxTools'

**Usage**

```
wbt_license()
```

**Value**

Returns the license information for WhiteboxTools as an R character vector.

**Examples**

```
## Not run:  
wbt_license()  
  
## End(Not run)
```

---

wbt_lidar_block_maximum	<i>Lidar block maximum</i>
-------------------------	----------------------------

---

**Description**

Creates a block-maximum raster from an input LAS file. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

**Usage**

```
wbt_lidar_block_maximum(  
  input,  
  output = NULL,  
  resolution = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input LiDAR file.
output	Output file.
resolution	Output raster's grid resolution.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_block\_minimum  
*Lidar block minimum*

---

**Description**

Creates a block-minimum raster from an input LAS file. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

**Usage**

```
wbt_lidar_block_minimum(  
  input,  
  output = NULL,  
  resolution = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input LiDAR file.
output	Output file.
resolution	Output raster's grid resolution.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_classify\_subset  
*Lidar classify subset*

---

**Description**

Classifies the values in one LiDAR point cloud that correspond with points in a subset cloud.

**Usage**

```
wbt_lidar_classify_subset(
  base,
  subset,
  output,
  subset_class,
  nonsubset_class = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

base	Input base LiDAR file.
subset	Input subset LiDAR file.
output	Output LiDAR file.
subset_class	Subset point class value (must be 0-18; see LAS specifications).
nonsubset_class	Non-subset point class value (must be 0-18; see LAS specifications).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_lidar_colourize` *Lidar colourize*

---

**Description**

Adds the red-green-blue colour fields of a LiDAR (LAS) file based on an input image.

**Usage**

```
wbt_lidar_colourize(
  in_lidar,
  in_image,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

`in_lidar` Input LiDAR file.

`in_image` Input colour image file.

`output` Output LiDAR file.

`wd` Changes the working directory.

`verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_contour      *Lidar contour*

---

## Description

This tool creates a vector contour coverage from an input LiDAR point file.

## Usage

```
wbt_lidar_contour(
  input,
  output = NULL,
  interval = 10,
  base = 0,
  smooth = 5,
  parameter = "elevation",
  returns = "all",
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  max_triangle_edge_length = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

## Arguments

input	Name of the input LiDAR points.
output	Name of the output vector lines file.
interval	Contour interval.
base	Base contour.
smooth	Smoothing filter size (in num. points), e.g. 3, 5, 7, 9, 11.
parameter	Interpolation parameter; options are 'elevation' (default), 'intensity', 'user_data'.
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
minz	Optional minimum elevation for inclusion in interpolation.
maxz	Optional maximum elevation for inclusion in interpolation.
max_triangle_edge_length	Optional maximum triangle edge length; triangles larger than this size will not be gridded.
wd	Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_lidar_digital_surface_model`  
*Lidar digital surface model*

---

**Description**

Creates a top-surface digital surface model (DSM) from a LiDAR point cloud.

**Usage**

```
wbt_lidar_digital_surface_model(  
  input,  
  output = NULL,  
  resolution = 1,  
  radius = 0.5,  
  minz = NULL,  
  maxz = NULL,  
  max_triangle_edge_length = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input LiDAR file (including extension).
<code>output</code>	Output raster file (including extension).
<code>resolution</code>	Output raster's grid resolution.
<code>radius</code>	Search Radius.
<code>minz</code>	Optional minimum elevation for inclusion in interpolation.
<code>maxz</code>	Optional maximum elevation for inclusion in interpolation.
<code>max_triangle_edge_length</code>	Optional maximum triangle edge length; triangles larger than this size will not be gridded.



wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_eigenvalue\_features  
*Lidar eigenvalue features*

---

**Description**

Calculate eigenvalue-based metrics from a LiDAR point cloud.

**Usage**

```
wbt_lidar_eigenvalue_features(  
  input,  
  num_neighbours = NULL,  
  radius = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Name of the input LiDAR points.
num_neighbours	Number of neighbours used in search.
radius	Search distance used in neighbourhood search.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_elevation\_slice

*Lidar elevation slice*

---

**Description**

Outputs all of the points within a LiDAR (LAS) point file that lie between a specified elevation range.

**Usage**

```
wbt_lidar_elevation_slice(
    input,
    output,
    minz = NULL,
    maxz = NULL,
    cls = FALSE,
    inclassval = 2,
    outclassval = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
minz	Minimum elevation value (optional).
maxz	Maximum elevation value (optional).
cls	Optional boolean flag indicating whether points outside the range should be retained in output but reclassified.
inclassval	Optional parameter specifying the class value assigned to points within the slice.
outclassval	Optional parameter specifying the class value assigned to points within the slice.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_lidar_ground_point_filter
    Lidar ground point filter
```

---

**Description**

Identifies ground points within LiDAR dataset using a slope-based method.

**Usage**

```
wbt_lidar_ground_point_filter(
    input,
    output,
    radius = 2,
    min_neighbours = 0,
    slope_threshold = 45,
    height_threshold = 1,
    classify = TRUE,
    slope_norm = TRUE,
    height_above_ground = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
radius	Search Radius.
min_neighbours	The minimum number of neighbouring points within search areas. If fewer points than this threshold are identified during the fixed-radius search, a subsequent kNN search is performed to identify the k number of neighbours.
slope_threshold	Maximum inter-point slope to be considered an off-terrain point.
height_threshold	Inter-point height difference to be considered an off-terrain point.
classify	Classify points as ground (2) or off-ground (1).
slope_norm	Perform initial ground slope normalization?.
height_above_ground	Transform output to height above average ground elevation?.

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_hex\_binning *Lidar hex binning*

---

**Description**

Hex-bins a set of LiDAR points.

**Usage**

```
wbt_lidar_hex_binning(  
  input,  
  output,  
  width,  
  orientation = "horizontal",  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input base file.
output	Output vector polygon file.
width	The grid cell width.
orientation	Grid Orientation, 'horizontal' or 'vertical'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_hillshade    *Lidar hillshade*

---

**Description**

Calculates a hillshade value for points within a LAS file and stores these data in the RGB field.

**Usage**

```
wbt_lidar_hillshade(
    input,
    output,
    azimuth = 315,
    altitude = 30,
    radius = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output file.
azimuth	Illumination source azimuth in degrees.
altitude	Illumination source altitude in degrees.
radius	Search Radius.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_histogram    *Lidar histogram*

---

### Description

Creates a histogram of LiDAR data.

### Usage

```
wbt_lidar_histogram(  
    input,  
    output,  
    parameter = "elevation",  
    clip = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
output	Output HTML file (default name will be based on input file if unspecified).
parameter	Parameter; options are 'elevation' (default), 'intensity', 'scan angle', 'class', 'time'.
clip	Amount to clip distribution tails (in percent).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_lidar\_idw\_interpolation  
*Lidar idw interpolation*

---

## Description

Interpolates LAS files using an inverse-distance weighted (IDW) scheme. When the input/output parameters are not specified, the tool interpolates all LAS files contained within the working directory.

## Usage

```
wbt_lidar_idw_interpolation(
    input,
    output = NULL,
    parameter = "elevation",
    returns = "all",
    resolution = 1,
    weight = 1,
    radius = 2.5,
    exclude_cls = NULL,
    minz = NULL,
    maxz = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

## Arguments

input	Input LiDAR file (including extension).
output	Output raster file (including extension).
parameter	Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
resolution	Output raster's grid resolution.
weight	IDW weight value.
radius	Search Radius.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
minz	Optional minimum elevation for inclusion in interpolation.
maxz	Optional maximum elevation for inclusion in interpolation.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_lidar_info	<i>Lidar info</i>
----------------	-------------------

---

**Description**

Prints information about a LiDAR (LAS) dataset, including header, point return frequency, and classification data and information about the variable length records (VLRs) and geokeys.

**Usage**

```
wbt_lidar_info(
  input,
  output,
  density = TRUE,
  vlr = TRUE,
  geokeys = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output HTML file for summary report.
density	Flag indicating whether or not to calculate the average point density and nominal point spacing.
vlr	Flag indicating whether or not to print the variable length records (VLRs).
geokeys	Flag indicating whether or not to print the geokeys.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.



compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_lidar_join	<i>Lidar join</i>
----------------	-------------------

---

**Description**

Joins multiple LiDAR (LAS) files into a single LAS file.

**Usage**

```
wbt_lidar_join(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input LiDAR files.
output	Output LiDAR file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_kappa\_index *Lidar kappa index*

---

### Description

Performs a kappa index of agreement (KIA) analysis on the classifications of two LAS files.

### Usage

```
wbt_lidar_kappa_index(  
    input1,  
    input2,  
    output,  
    class_accuracy,  
    resolution = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input LiDAR classification file.
input2	Input LiDAR reference file.
output	Output HTML file.
class_accuracy	Output classification accuracy raster file.
resolution	Output raster's grid resolution.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_lidar\_nearest\_neighbour\_gridding  
*Lidar nearest neighbour gridding*

---

## Description

Grids LiDAR files using nearest-neighbour scheme. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

## Usage

```
wbt_lidar_nearest_neighbour_gridding(  

    input,  

    output = NULL,  

    parameter = "elevation",  

    returns = "all",  

    resolution = 1,  

    radius = 2.5,  

    exclude_cls = NULL,  

    minz = NULL,  

    maxz = NULL,  

    wd = NULL,  

    verbose_mode = FALSE,  

    compress_rasters = FALSE,  

    command_only = FALSE  

)
```

## Arguments

input	Input LiDAR file (including extension).
output	Output raster file (including extension).
parameter	Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data', 'time'.
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
resolution	Output raster's grid resolution.
radius	Search Radius.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
minz	Optional minimum elevation for inclusion in interpolation.
maxz	Optional maximum elevation for inclusion in interpolation.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_point\_density  
*Lidar point density*

---

**Description**

Calculates the spatial pattern of point density for a LiDAR data set. When the input/output parameters are not specified, the tool grids all LAS files contained within the working directory.

**Usage**

```
wbt_lidar_point_density(  
  input,  
  output = NULL,  
  returns = "all",  
  resolution = 1,  
  radius = 2.5,  
  exclude_cls = NULL,  
  minz = NULL,  
  maxz = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input LiDAR file (including extension).
output	Output raster file (including extension).
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
resolution	Output raster's grid resolution.
radius	Search radius.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, -exclude_cls='3,4,5,6,7,18'.
minz	Optional minimum elevation for inclusion in interpolation.

maxz	Optional maximum elevation for inclusion in interpolation.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_point\_return\_analysis  
*Lidar point return analysis*

---

**Description**

This tool performs a quality control check on the return values of points in a LiDAR file.

**Usage**

```
wbt_lidar_point_return_analysis(  
  input,  
  output = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_point\_stats *Lidar point stats*

---

### Description

Creates several rasters summarizing the distribution of LAS point data. When the input/output parameters are not specified, the tool works on all LAS files contained within the working directory.

### Usage

```
wbt_lidar_point_stats(  
    input,  
    resolution = 1,  
    num_points = TRUE,  
    num_pulses = FALSE,  
    avg_points_per_pulse = TRUE,  
    z_range = FALSE,  
    intensity_range = FALSE,  
    predom_class = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
resolution	Output raster's grid resolution.
num_points	Flag indicating whether or not to output the number of points (returns) raster.
num_pulses	Flag indicating whether or not to output the number of pulses raster.
avg_points_per_pulse	Flag indicating whether or not to output the average number of points (returns) per pulse raster.
z_range	Flag indicating whether or not to output the elevation range raster.
intensity_range	Flag indicating whether or not to output the intensity range raster.
predom_class	Flag indicating whether or not to output the predominant classification raster.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_ransac\_planes

*Lidar ransac planes*

---

**Description**

Performs a RANSAC analysis to identify points within a LiDAR point cloud that belong to linear planes.

**Usage**

```
wbt_lidar_ransac_planes(
    input,
    output,
    radius = 2,
    num_iter = 50,
    num_samples = 5,
    threshold = 0.35,
    model_size = 8,
    max_slope = 80,
    classify = FALSE,
    last_returns = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
radius	Search Radius.
num_iter	Number of iterations.
num_samples	Number of sample points on which to build the model.
threshold	Threshold used to determine inlier points.
model_size	Acceptable model size.
max_slope	Maximum planar slope.
classify	Classify points as ground (2) or off-ground (1).
last_returns	Only include last- and only-return points.

<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_lidar_rbf_interpolation`  
*Lidar rbf interpolation*

---

**Description**

Interpolates LAS files using a radial basis function (RBF) scheme. When the input/output parameters are not specified, the tool interpolates all LAS files contained within the working directory.

**Usage**

```
wbt_lidar_rbf_interpolation(
  input,
  output = NULL,
  parameter = "elevation",
  returns = "all",
  resolution = 1,
  num_points = 20,
  exclude_cls = NULL,
  minz = NULL,
  maxz = NULL,
  func_type = "ThinPlateSpline",
  poly_order = "none",
  weight = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```



**Arguments**

input	Input LiDAR file (including extension).
output	Output raster file (including extension).
parameter	Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
resolution	Output raster's grid resolution.
num_points	Number of points.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
minz	Optional minimum elevation for inclusion in interpolation.
maxz	Optional maximum elevation for inclusion in interpolation.
func_type	Radial basis function type; options are 'ThinPlateSpline' (default), 'PolyHarmonic', 'Gaussian', 'MultiQuadric', 'InverseMultiQuadric'.
poly_order	Polynomial order; options are 'none' (default), 'constant', 'affine'.
weight	Weight parameter used in basis function.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_remove\_duplicates  
*Lidar remove duplicates*

---

**Description**

Removes duplicate points from a LiDAR data set.

**Usage**

```
wbt_lidar_remove_duplicates(
    input,
    output,
    include_z = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
include_z	Include z-values in point comparison?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_remove\_outliers  
*Lidar remove outliers*

---

**Description**

Removes outliers (high and low points) in a LiDAR point cloud.

**Usage**

```
wbt_lidar_remove_outliers(
    input,
    output,
    radius = 2,
    elev_diff = 50,
    use_median = FALSE,
    classify = TRUE,
```

```

    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input LiDAR file.
output	Output LiDAR file.
radius	Search Radius.
elev_diff	Max. elevation difference.
use_median	Optional flag indicating whether to use the difference from median elevation rather than mean.
classify	Classify points as ground (2) or off-ground (1).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_lidar_rooftop_analysis
      Lidar rooftop analysis

```

---

### Description

Identifies roof segments in a LiDAR point cloud.

### Usage

```

wbt_lidar_rooftop_analysis(
  buildings,
  output,
  input = NULL,
  radius = 2,
  num_iter = 50,
  num_samples = 10,
  threshold = 0.15,

```

```

    model_size = 15,
    max_slope = 65,
    norm_diff = 10,
    azimuth = 180,
    altitude = 30,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

<code>buildings</code>	Input vector build footprint polygons file.
<code>output</code>	Output vector polygon file.
<code>input</code>	Input LiDAR file.
<code>radius</code>	Search Radius.
<code>num_iter</code>	Number of iterations.
<code>num_samples</code>	Number of sample points on which to build the model.
<code>threshold</code>	Threshold used to determine inlier points (in elevation units).
<code>model_size</code>	Acceptable model size, in points.
<code>max_slope</code>	Maximum planar slope, in degrees.
<code>norm_diff</code>	Maximum difference in normal vectors, in degrees.
<code>azimuth</code>	Illumination source azimuth, in degrees.
<code>altitude</code>	Illumination source altitude in degrees.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_lidar_segmentation`*Lidar segmentation*

---

### Description

Segments a LiDAR point cloud based on differences in the orientation of fitted planar surfaces and point proximity.

### Usage

```
wbt_lidar_segmentation(  
    input,  
    output,  
    radius = 2,  
    num_iter = 50,  
    num_samples = 10,  
    threshold = 0.15,  
    model_size = 15,  
    max_slope = 80,  
    norm_diff = 10,  
    maxzdiff = 1,  
    classes = FALSE,  
    ground = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>input</code>	Input LiDAR file.
<code>output</code>	Output LiDAR file.
<code>radius</code>	Search Radius.
<code>num_iter</code>	Number of iterations.
<code>num_samples</code>	Number of sample points on which to build the model.
<code>threshold</code>	Threshold used to determine inlier points.
<code>model_size</code>	Acceptable model size.
<code>max_slope</code>	Maximum planar slope.
<code>norm_diff</code>	Maximum difference in normal vectors, in degrees.
<code>maxzdiff</code>	Maximum difference in elevation (z units) between neighbouring points of the same segment.
<code>classes</code>	Segments don't cross class boundaries.

ground	Classify the largest segment as ground points?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_segmentation\_based\_filter  
*Lidar segmentation based filter*

---

**Description**

Identifies ground points within LiDAR point clouds using a segmentation based approach.

**Usage**

```
wbt_lidar_segmentation_based_filter(
  input,
  output,
  radius = 5,
  norm_diff = 2,
  maxzdiff = 1,
  classify = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output file.
radius	Search Radius.
norm_diff	Maximum difference in normal vectors, in degrees.
maxzdiff	Maximum difference in elevation (z units) between neighbouring points of the same segment.
classify	Classify points as ground (2) or off-ground (1).

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_lidar_shift	<i>Lidar shift</i>
-----------------	--------------------

---

**Description**

Shifts the x,y,z coordinates of a LiDAR file.

**Usage**

```
wbt_lidar_shift(
  input,
  output,
  x_shift = "",
  y_shift = "",
  z_shift = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
x_shift	x-shift value, blank for none.
y_shift	y-shift value, blank for none.
z_shift	z-shift value, blank for none.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_lidar_sibson_interpolation
    Lidar sibson interpolation
```

---

**Description**

This tool interpolates one or more LiDAR tiles using Sibson's natural neighbour method.

**Usage**

```
wbt_lidar_sibson_interpolation(
    input,
    output = NULL,
    parameter = "elevation",
    returns = "all",
    resolution = 1,
    exclude_cls = NULL,
    minz = NULL,
    maxz = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input LiDAR points (leave blank to use all files in WorkingDirectory).
output	Output raster file (including extension).
parameter	Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'user_data'.
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
resolution	Output raster's grid resolution.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
minz	Optional minimum elevation for inclusion in interpolation.
maxz	Optional maximum elevation for inclusion in interpolation.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.



compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_sort\_by\_time  
*Lidar sort by time*

---

**Description**

This tool sorts the points in a LiDAR file by the GPS time.

**Usage**

```
wbt_lidar_sort_by_time(  
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_thin      *Lidar thin*

---

### Description

Thins a LiDAR point cloud, reducing point density.

### Usage

```
wbt_lidar_thin(
    input,
    output,
    resolution = 2,
    method = "lowest",
    save_filtered = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input LiDAR file.
output	Output LiDAR file.
resolution	The size of the square area used to evaluate nearby points in the LiDAR data.
method	Point selection method; options are 'first', 'last', 'lowest' (default), 'highest', 'nearest'.
save_filtered	Save filtered points to separate file?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_lidar\_thin\_high\_density  
*Lidar thin high density*

---

### Description

Thins points from high density areas within a LiDAR point cloud.

### Usage

```
wbt_lidar_thin_high_density(  
    input,  
    output,  
    density,  
    resolution = 1,  
    save_filtered = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
output	Output LiDAR file.
density	Max. point density (points / m <sup>3</sup> ).
resolution	Output raster's grid resolution.
save_filtered	Save filtered points to separate file?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_lidar_tile	<i>Lidar tile</i>
----------------	-------------------

---

### Description

Tiles a LiDAR LAS file into multiple LAS files.

### Usage

```
wbt_lidar_tile(
  input,
  width = 1000,
  height = 1000,
  origin_x = 0,
  origin_y = 0,
  min_points = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Input LiDAR file.
width	Width of tiles in the X dimension; default 1000.0.
height	Height of tiles in the Y dimension.
origin_x	Origin point X coordinate for tile grid.
origin_y	Origin point Y coordinate for tile grid.
min_points	Minimum number of points contained in a tile for it to be saved.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_lidar\_tile\_footprint  
*Lidar tile footprint*

---

### Description

Creates a vector polygon of the convex hull of a LiDAR point cloud. When the input/output parameters are not specified, the tool works with all LAS files contained within the working directory.

### Usage

```
wbt_lidar_tile_footprint(  
    input,  
    output,  
    hull = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input LiDAR file.
output	Output vector polygon file.
hull	Identify the convex hull around points.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_lidar\_tin\_gridding

*Lidar tin gridding*


---

## Description

Creates a raster grid based on a Delaunay triangular irregular network (TIN) fitted to LiDAR points.

## Usage

```
wbt_lidar_tin_gridding(
  input,
  output = NULL,
  parameter = "elevation",
  returns = "all",
  resolution = 1,
  exclude_cls = "7,18",
  minz = NULL,
  maxz = NULL,
  max_triangle_edge_length = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

## Arguments

input	Input LiDAR file (including extension).
output	Output raster file (including extension).
parameter	Interpolation parameter; options are 'elevation' (default), 'intensity', 'class', 'return_number', 'number_of_returns', 'scan angle', 'rgb', 'user data'.
returns	Point return types to include; options are 'all' (default), 'last', 'first'.
resolution	Output raster's grid resolution.
exclude_cls	Optional exclude classes from interpolation; Valid class values range from 0 to 18, based on LAS specifications. Example, <code>-exclude_cls='3,4,5,6,7,18'</code> .
minz	Optional minimum elevation for inclusion in interpolation.
maxz	Optional maximum elevation for inclusion in interpolation.
max_triangle_edge_length	Optional maximum triangle edge length; triangles larger than this size will not be gridded.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lidar\_tophat\_transform  
*Lidar tophat transform*

---

**Description**

Performs a white top-hat transform on a Lidar dataset; as an estimate of height above ground, this is useful for modelling the vegetation canopy.

**Usage**

```
wbt_lidar_tophat_transform(
  input,
  output,
  radius = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
radius	Search Radius.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_linearity\_index    *Linearity index*

---

### Description

Calculates the linearity index for vector polygons.

### Usage

```
wbt_linearity_index(  
    input,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_lines\_to\_polygons    *Lines to polygons*

---

### Description

Converts vector polylines to polygons.



**Usage**

```
wbt_lines_to_polygons(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector line file.
output	Output vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_line_detection_filter
    Line detection filter
```

---

**Description**

Performs a line-detection filter on an image.

**Usage**

```
wbt_line_detection_filter(
    input,
    output,
    variant = "vertical",
    absvals = FALSE,
    clip = 0,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
variant	Optional variant value. Options include 'v' (vertical), 'h' (horizontal), '45', and '135' (default is 'v').
absvals	Optional flag indicating whether outputs should be absolute values.
clip	Optional amount to clip the distribution tails by, in percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_line\_intersections  
*Line intersections*

---

**Description**

Identifies points where the features of two vector line layers intersect.

**Usage**

```
wbt_line_intersections(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1	Input vector polyline file.
input2	Input vector polyline file.
output	Output vector point file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_line_thinning	<i>Line thinning</i>
-------------------	----------------------

---

**Description**

Performs line thinning a on Boolean raster image; intended to be used with the RemoveSpurs tool.

**Usage**

```
wbt_line_thinning(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_list_tools	<i>All available tools in 'WhiteboxTools'</i>
----------------	---

---

**Description**

All available tools in 'WhiteboxTools'

**Usage**

```
wbt_list_tools(keywords = "")
```

**Arguments**

keywords	Keywords may be used to search available tools. Default "" returns all available tools.
----------	---

**Value**

Return all available tools in WhiteboxTools that contain the keywords.

**Examples**

```
## Not run:
wbt_list_tools("lidar")

## End(Not run)
```

---

wbt_list_unique_values	<i>List unique values</i>
------------------------	---------------------------

---

**Description**

Lists the unique values contained in a field within a vector's attribute table.

**Usage**

```
wbt_list_unique_values(
  input,
  field,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector file.
field	Input field name in attribute table.
output	Output HTML file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_list\_unique\_values\_raster  
*List unique values raster*

---

**Description**

Lists the unique values contained in a field within a vector's attribute table.

**Usage**

```
wbt_list_unique_values_raster(  
  input,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_ln

*Ln*


---

**Description**

Returns the natural logarithm of values in a raster.

**Usage**

```
wbt_ln(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_local\_hypsometric\_analysis  
*Local hypsometric analysis*

---

### Description

This tool calculates a local, neighbourhood-based hypsometric integral raster.

### Usage

```
wbt_local_hypsometric_analysis(  
    input,  
    out_mag,  
    out_scale,  
    min_scale = 4,  
    step = 1,  
    num_steps = 10,  
    step_nonlinearity = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Name of the input raster DEM file.
out_mag	Name of the openness output raster file.
out_scale	Name of the openness output raster file.
min_scale	Minimum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
num_steps	Number of steps.
step_nonlinearity	Step nonlinearity factor (1.0-2.0 is typical).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_local\_quadratic\_regression  
*Local quadratic regression*

---

### Description

An implementation of the constrained quadratic regression algorithm using a flexible window size described in Wood (1996).

### Usage

```
wbt_local_quadratic_regression(  
    dem,  
    output,  
    filter = 3,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Name of the input DEM raster file.
output	Name of the output raster file.
filter	Edge length of the filter kernel.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt_log10	<i>Log10</i>
-----------	--------------

---

### Description

Returns the base-10 logarithm of values in a raster.

### Usage

```
wbt_log10(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_log2	<i>Log2</i>
----------	-------------

---

### Description

Returns the base-2 logarithm of values in a raster.

**Usage**

```
wbt_log2(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_logistic\_regression  
*Logistic regression*

---

**Description**

Performs a logistic regression analysis using training site polygons/points and predictor rasters.

**Usage**

```
wbt_logistic_regression(
  inputs,
  training,
  field,
  scaling = "Normalize",
  output = NULL,
  test_proportion = 0.2,
  wd = NULL,
  verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

inputs	Names of the input predictor rasters.
training	Name of the input training site polygons/points shapefile.
field	Name of the attribute containing class data.
scaling	Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output	Name of the output raster file.
test_proportion	The proportion of the dataset to include in the test split; default is 0.2.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_longest\_flowpath *Longest flowpath*

---

### Description

Delineates the longest flowpaths for a group of subbasins or watersheds.

### Usage

```

wbt_longest_flowpath(
  dem,
  basins,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

dem	Input raster DEM file.
basins	Input raster basins file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_long_profile	<i>Long profile</i>
------------------	---------------------

---

**Description**

Plots the stream longitudinal profiles for one or more rivers.

**Usage**

```
wbt_long_profile(
  d8_pntr,
  streams,
  dem,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
dem	Input raster DEM file.
output	Output HTML file.
esri_pntr	D8 pointer uses the ESRI style scheme.

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_long\_profile\_from\_points  
*Long profile from points*

---

**Description**

Plots the longitudinal profiles from flow-paths initiating from a set of vector points.

**Usage**

```
wbt_long_profile_from_points(  
  d8_pntr,  
  points,  
  dem,  
  output,  
  esri_pntr = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
points	Input vector points file.
dem	Input raster DEM file.
output	Output HTML file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_lowest\_position      *Lowest position*

---

**Description**

Identifies the stack position of the minimum value within a raster stack on a cell-by-cell basis.

**Usage**

```
wbt_lowest_position(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs      Input raster files.

output      Output raster file.

wd      Changes the working directory.

verbose\_mode      Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_low\_points\_on\_headwater\_divides  
*Low points on headwater divides*

---

### Description

This tool locates saddle points along ridges within a digital elevation model (DEM).

### Usage

```
wbt_low_points_on_headwater_divides(  
    dem,  
    streams,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Name of the input DEM raster file.
streams	Name of the input stream channel raster file.
output	Name of the output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_majority\_filter     *Majority filter*

---

### Description

Assigns each cell in the output grid the most frequently occurring value (mode) in a moving window centred on each grid cell in the input raster.

### Usage

```
wbt_majority_filter(  
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_map\_off\_terrain\_objects  
*Map off terrain objects*

---

### Description

Maps off-terrain objects in a digital elevation model (DEM).

### Usage

```
wbt_map_off_terrain_objects(  
    dem,  
    output,  
    max_slope = 40,  
    min_size = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
max_slope	Maximum inter-cell absolute slope.
min_size	Minimum feature size, in grid cells.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_max	<i>Max</i>
---------	------------

---

### Description

Performs a MAX operation on two rasters or a raster and a constant value.

### Usage

```
wbt_max(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_maximal\_curvature *Maximal curvature*

---

### Description

Calculates a mean curvature raster from an input DEM.

### Usage

```
wbt_maximal_curvature(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_maximum\_filter      *Maximum filter*

---

### Description

Assigns each cell in the output grid the maximum value in a moving window centred on each grid cell in the input raster.

### Usage

```
wbt_maximum_filter(  
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_max\_absolute\_overlay  
*Max absolute overlay*

---

**Description**

Evaluates the maximum absolute value for each grid cell from a stack of input rasters.

**Usage**

```
wbt_max_absolute_overlay(  
    inputs,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_max\_anisotropy\_dev  
*Max anisotropy dev*

---

**Description**

Calculates the maximum anisotropy (directionality) in elevation deviation over a range of spatial scales.

**Usage**

```
wbt_max_anisotropy_dev(
    dem,
    out_mag,
    out_scale,
    max_scale,
    min_scale = 3,
    step = 2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
out_mag	Output raster DEVmax magnitude file.
out_scale	Output raster DEVmax scale file.
max_scale	Maximum search neighbourhood radius in grid cells.
min_scale	Minimum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_max\_anisotropy\_dev\_signature  
*Max anisotropy dev signature*

---

**Description**

Calculates the anisotropy in deviation from mean for points over a range of spatial scales.

**Usage**

```
wbt_max_anisotropy_dev_signature(
    dem,
    points,
    output,
    max_scale,
    min_scale = 1,
    step = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
points	Input vector points file.
output	Output HTML file.
max_scale	Maximum search neighbourhood radius in grid cells.
min_scale	Minimum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_max\_branch\_length *Max branch length*

---

**Description**

Lindsay and Seibert's (2013) branch length index is used to map drainage divides or ridge lines.

**Usage**

```
wbt_max_branch_length(
  dem,
  output,
  log = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Optional flag to request the output be log-transformed.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_max_difference_from_mean	<i>Max difference from mean</i>
------------------------------	---------------------------------

---

**Description**

Calculates the maximum difference from mean elevation over a range of spatial scales.

**Usage**

```
wbt_max_difference_from_mean(
  dem,
  out_mag,
  out_scale,
  min_scale,
  max_scale,
  step = 1,
```



```

    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

dem	Input raster DEM file.
out_mag	Output raster DIFFmax magnitude file.
out_scale	Output raster DIFFmax scale file.
min_scale	Minimum search neighbourhood radius in grid cells.
max_scale	Maximum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_max_downslope_elev_change
    Max downslope elev change

```

---

### Description

Calculates the maximum downslope change in elevation between a grid cell and its eight downslope neighbors.

### Usage

```

wbt_max_downslope_elev_change(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_max\_elevation\_deviation  
*Max elevation deviation*

---

**Description**

Calculates the maximum elevation deviation over a range of spatial scales.

**Usage**

```
wbt_max_elevation_deviation(  
  dem,  
  out_mag,  
  out_scale,  
  min_scale,  
  max_scale,  
  step = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
out_mag	Output raster DEVmax magnitude file.
out_scale	Output raster DEVmax scale file.
min_scale	Minimum search neighbourhood radius in grid cells.

max_scale	Maximum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_max\_elev\_dev\_signature

*Max elev dev signature*

---

**Description**

Calculates the maximum elevation deviation over a range of spatial scales and for a set of points.

**Usage**

```
wbt_max_elev_dev_signature(
  dem,
  points,
  output,
  min_scale,
  max_scale,
  step = 10,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
points	Input vector points file.
output	Output HTML file.
min_scale	Minimum search neighbourhood radius in grid cells.
max_scale	Maximum search neighbourhood radius in grid cells.

step	Step size as any positive non-zero integer.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_max_overlay	<i>Max overlay</i>
-----------------	--------------------

---

**Description**

Evaluates the maximum value for each grid cell from a stack of input rasters.

**Usage**

```
wbt_max_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_max\_upslope\_elev\_change  
*Max upslope elev change*

---

### Description

Calculates the maximum upslope change in elevation between a grid cell and its eight downslope neighbors.

### Usage

```
wbt_max_upslope_elev_change(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_max\_upslope\_flowpath\_length  
*Max upslope flowpath length*

---

### Description

Measures the maximum length of all upslope flowpaths draining each grid cell.

**Usage**

```
wbt_max_upslope_flowpath_length(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

*wbt\_max\_upslope\_value* *Max upslope value*

---

**Description**

Calculates the maximum upslope value from an input values raster along flowpaths.

**Usage**

```
wbt_max_upslope_value(
    dem,
    values,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input DEM; it must be depressionless.
values	Name of the input values raster file.
output	Name of the output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_md\_inf\_flow\_accumulation  
*Md inf flow accumulation*

---

**Description**

Calculates an FD8 flow accumulation raster from an input DEM.

**Usage**

```
wbt_md_inf_flow_accumulation(
  dem,
  output,
  out_type = "specific contributing area",
  exponent = 1.1,
  threshold = NULL,
  log = FALSE,
  clip = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
out_type	Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
exponent	Optional exponent parameter; default is 1.1.
threshold	Optional convergence threshold parameter, in grid cells; default is infinity.
log	Optional flag to request the output be log-transformed.
clip	Optional flag to request clipping the display max by 1 percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_mean_curvature	<i>Mean curvature</i>
--------------------	-----------------------

---

**Description**

Calculates a mean curvature raster from an input DEM.

**Usage**

```
wbt_mean_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```



**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_mean_filter	<i>Mean filter</i>
-----------------	--------------------

---

**Description**

Performs a mean filter (low-pass filter) on an input image.

**Usage**

```
wbt_mean_filter(
  input,
  output,
  filterx = 3,
  filtery = 3,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

<code>wbt_median_filter</code>	<i>Median filter</i>
--------------------------------	----------------------

---

**Description**

Performs a median filter on an input image.

**Usage**

```
wbt_median_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  sig_digits = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<code>input</code>	Input raster file.
<code>output</code>	Output raster file.
<code>filterx</code>	Size of the filter kernel in the x-direction.
<code>filtery</code>	Size of the filter kernel in the y-direction.
<code>sig_digits</code>	Number of significant digits.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_medoid	<i>Medoid</i>
------------	---------------

---

**Description**

Calculates the medoid for a series of vector features contained in a shapefile.

**Usage**

```
wbt_medoid(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_merge_line_segments`*Merge line segments*

---

**Description**

Merges vector line segments into larger features.

**Usage**

```
wbt_merge_line_segments(  
    input,  
    output,  
    snap = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input vector file.
<code>output</code>	Output vector file.
<code>snap</code>	Snap tolerance.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_merge_table_with_csv`*Merge table with csv*

---

## Description

Merge a vector's attribute table with a table contained within a CSV text file.

## Usage

```
wbt_merge_table_with_csv(  
  input,  
  pkey,  
  csv,  
  fkey,  
  import_field = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

## Arguments

<code>input</code>	Input primary vector file (i.e. the table to be modified).
<code>pkey</code>	Primary key field.
<code>csv</code>	Input CSV file (i.e. source of data to be imported).
<code>fkey</code>	Foreign key field.
<code>import_field</code>	Imported field (all fields will be imported if not specified).
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

## Value

Returns the tool text outputs.

---

wbt\_merge\_vectors      *Merge vectors*

---

### Description

Combines two or more input vectors of the same ShapeType creating a single, new output vector.

### Usage

```
wbt_merge_vectors(
    inputs,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

inputs	Input vector files.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_min      *Min*

---

### Description

Performs a MIN operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_min(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_minimal\_curvature *Minimal curvature*

---

**Description**

Calculates a mean curvature raster from an input DEM.

**Usage**

```
wbt_minimal_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_minimum\_bounding\_box

*Minimum bounding box*

---

**Description**

Creates a vector minimum bounding rectangle around vector features.

**Usage**

```
wbt_minimum_bounding_box(
  input,
  output,
  criterion = "area",
  features = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector file.
output	Output vector polygon file.
criterion	Minimization criterion; options include 'area' (default), 'length', 'width', and 'perimeter'.



features	Find the minimum bounding rectangles around each individual vector feature.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_minimum\_bounding\_circle  
*Minimum bounding circle*

---

**Description**

Delineates the minimum bounding circle (i.e. smallest enclosing circle) for a group of vectors.

**Usage**

```
wbt_minimum_bounding_circle(
  input,
  output,
  features = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector file.
output	Output vector polygon file.
features	Find the minimum bounding circle around each individual vector feature.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_minimum\_bounding\_envelope  
*Minimum bounding envelope*

---

**Description**

Creates a vector axis-aligned minimum bounding rectangle (envelope) around vector features.

**Usage**

```
wbt_minimum_bounding_envelope(
    input,
    output,
    features = TRUE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector file.
output	Output vector polygon file.
features	Find the minimum bounding envelop around each individual vector feature.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_minimum_convex_hull`*Minimum convex hull*

---

**Description**

Creates a vector convex polygon around vector features.

**Usage**

```
wbt_minimum_convex_hull(  
    input,  
    output,  
    features = TRUE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input vector file.
<code>output</code>	Output vector polygon file.
<code>features</code>	Find the hulls around each vector feature.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_minimum\_filter      *Minimum filter*

---

### Description

Assigns each cell in the output grid the minimum value in a moving window centred on each grid cell in the input raster.

### Usage

```
wbt_minimum_filter(  
    input,  
    output,  
    filterx = 11,  
    filtery = 11,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_min\_absolute\_overlay  
*Min absolute overlay*

---

### Description

Evaluates the minimum absolute value for each grid cell from a stack of input rasters.

### Usage

```
wbt_min_absolute_overlay(  
    inputs,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_min\_dist\_classification  
*Min dist classification*

---

### Description

Performs a supervised minimum-distance classification using training site polygons and multi-spectral images.

**Usage**

```
wbt_min_dist_classification(
    inputs,
    polys,
    field,
    output,
    threshold = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

inputs	Names of the input band images.
polys	Name of the input training site polygons shapefile.
field	Name of the attribute containing class name data.
output	Name of the output raster file.
threshold	Distance threshold, in z-scores; blank for none.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_min\_downslope\_elev\_change  
*Min downslope elev change*

---

**Description**

Calculates the minimum downslope change in elevation between a grid cell and its eight downslope neighbors.

**Usage**

```
wbt_min_downslope_elev_change(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_min\_max\_contrast\_stretch

*Min max contrast stretch*

---

**Description**

Performs a min-max contrast stretch on an input greytone image.

**Usage**

```
wbt_min_max_contrast_stretch(
    input,
    output,
    min_val,
    max_val,
    num_tones = 256,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
min_val	Lower tail clip value.
max_val	Upper tail clip value.
num_tones	Number of tones in the output image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_min_overlay	<i>Min overlay</i>
-----------------	--------------------

---

**Description**

Evaluates the minimum value for each grid cell from a stack of input rasters.

**Usage**

```
wbt_min_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.



compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_modified\_k\_means\_clustering  
*Modified k means clustering*

---

**Description**

Performs a modified k-means clustering operation on a multi-spectral dataset.

**Usage**

```
wbt_modified_k_means_clustering(  
  inputs,  
  output,  
  out_html = NULL,  
  start_clusters = 1000,  
  merge_dist = NULL,  
  max_iterations = 10,  
  class_change = 2,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
out_html	Output HTML report file.
start_clusters	Initial number of clusters.
merge_dist	Cluster merger distance.
max_iterations	Maximum number of iterations.
class_change	Minimum percent of cells changed between iterations before completion.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_modify_lidar`      *Modify lidar*

---

**Description**

Modify points within a LiDAR point cloud based on point properties.

**Usage**

```
wbt_modify_lidar(
  input,
  output = NULL,
  statement = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

`input` Name of the input LiDAR points.

`output` Name of the output LiDAR points.

`statement` Modify statement e.g. `x += 5000.0`.

`wd` Changes the working directory.

`verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_modify\_no\_data\_value  
*Modify no data value*

---

**Description**

Modifies nodata values in a raster.

**Usage**

```
wbt_modify_no_data_value(  
    input,  
    new_value = "-32768.0",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
new_value	New NoData value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_modulo                      *Modulo*

---

**Description**

Performs a modulo operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_modulo(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_mosaic	<i>Mosaic</i>
------------	---------------

---

**Description**

Mosaics two or more images together.

**Usage**

```
wbt_mosaic(
    output,
    inputs = NULL,
    method = "nn",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

output	Output raster file.
inputs	Input raster files.
method	Resampling method; options include 'nn' (nearest neighbour), 'bilinear', and 'cc' (cubic convolution).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_mosaic\_with\_feathering  
*Mosaic with feathering*

---

**Description**

Mosaics two images together using a feathering technique in overlapping areas to reduce edge-effects.

**Usage**

```
wbt_mosaic_with_feathering(  
  input1,  
  input2,  
  output,  
  method = "cc",  
  weight = 4,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input1	Input raster file to modify.
input2	Input reference raster file.
output	Output raster file.
method	Resampling method; options include 'nn' (nearest neighbour), 'bilinear', and 'cc' (cubic convolution).
weight	.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_multidirectional\_hillshade  
*Multidirectional hillshade*

---

**Description**

Calculates a multi-direction hillshade raster from an input DEM.

**Usage**

```
wbt_multidirectional_hillshade(  
  dem,  
  output,  
  altitude = 45,  
  zfactor = NULL,  
  full_mode = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
altitude	Illumination source altitude in degrees.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
full_mode	Optional flag indicating whether to use full 360-degrees of illumination sources.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_multiply	<i>Multiply</i>
--------------	-----------------

---

**Description**

Performs a multiplication operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_multiply(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_multiply_overlay`    *Multiply overlay*

---

**Description**

Calculates the sum for each grid cell from a group of raster images.

**Usage**

```
wbt_multiply_overlay(
  inputs,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<code>inputs</code>	Input raster files.
<code>output</code>	Output raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.



---

```
wbt_multiscale_curvatures
    Multiscale curvatures
```

---

## Description

This tool calculates several multiscale curvatures and curvature-based indices from an input DEM.

## Usage

```
wbt_multiscale_curvatures(
    dem,
    out_mag,
    curv_type = "ProfileCurv",
    out_scale = NULL,
    min_scale = 0,
    step = 1,
    num_steps = 1,
    step_nonlinearity = 1,
    log = TRUE,
    standardize = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

## Arguments

dem	Name of the input raster DEM file.
out_mag	Output raster magnitude file.
curv_type	Curvature type.
out_scale	Output raster scale file.
min_scale	Minimum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
num_steps	Number of steps.
step_nonlinearity	Step nonlinearity factor (1.0-2.0 is typical).
log	Display output values using a log-scale.
standardize	Should each scale be standardized to z-scores?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_multiscale_elevation_percentile`  
*Multiscale elevation percentile*

---

**Description**

Calculates surface roughness over a range of spatial scales.

**Usage**

```
wbt_multiscale_elevation_percentile(  
  dem,  
  out_mag,  
  out_scale,  
  sig_digits = 3,  
  min_scale = 4,  
  step = 1,  
  num_steps = 10,  
  step_nonlinearity = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

`dem` Input raster DEM file.

`out_mag` Output raster roughness magnitude file.

`out_scale` Output raster roughness scale file.

`sig_digits` Number of significant digits.

`min_scale` Minimum search neighbourhood radius in grid cells.

`step` Step size as any positive non-zero integer.

`num_steps` Number of steps.

`step_nonlinearity` Step nonlinearity factor (1.0-2.0 is typical).

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_multiscale\_roughness  
*Multiscale roughness*

---

**Description**

Calculates surface roughness over a range of spatial scales.

**Usage**

```
wbt_multiscale_roughness(
  dem,
  out_mag,
  out_scale,
  max_scale,
  min_scale = 1,
  step = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
out_mag	Output raster roughness magnitude file.
out_scale	Output raster roughness scale file.
max_scale	Maximum search neighbourhood radius in grid cells.
min_scale	Minimum search neighbourhood radius in grid cells.
step	Step size as any positive non-zero integer.
wd	Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_multiscale_roughness_signature`  
*Multiscale roughness signature*

---

**Description**

Calculates the surface roughness for points over a range of spatial scales.

**Usage**

```
wbt_multiscale_roughness_signature(  
  dem,  
  points,  
  output,  
  max_scale,  
  min_scale = 1,  
  step = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>points</code>	Input vector points file.
<code>output</code>	Output HTML file.
<code>max_scale</code>	Maximum search neighbourhood radius in grid cells.
<code>min_scale</code>	Minimum search neighbourhood radius in grid cells.
<code>step</code>	Step size as any positive non-zero integer.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_multiscale\_std\_dev\_normals  
*Multiscale std dev normals*

---

**Description**

Calculates surface roughness over a range of spatial scales.

**Usage**

```
wbt_multiscale_std_dev_normals(  
  dem,  
  out_mag,  
  out_scale,  
  min_scale = 1,  
  step = 1,  
  num_steps = 10,  
  step_nonlinearity = 1,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem                    Input raster DEM file.

out\_mag                Output raster roughness magnitude file.

out\_scale              Output raster roughness scale file.

min\_scale              Minimum search neighbourhood radius in grid cells.

step                    Step size as any positive non-zero integer.

num\_steps              Number of steps.

step\_nonlinearity      Step nonlinearity factor (1.0-2.0 is typical).

wd                      Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_multiscale_std_dev_normals_signature`  
*Multiscale std dev normals signature*

---

**Description**

Calculates the surface roughness for points over a range of spatial scales.

**Usage**

```
wbt_multiscale_std_dev_normals_signature(
  dem,
  points,
  output,
  min_scale = 1,
  step = 1,
  num_steps = 10,
  step_nonlinearity = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>points</code>	Input vector points file.
<code>output</code>	Output HTML file.
<code>min_scale</code>	Minimum search neighbourhood radius in grid cells.
<code>step</code>	Step size as any positive non-zero integer.
<code>num_steps</code>	Number of steps.
<code>step_nonlinearity</code>	Step nonlinearity factor (1.0-2.0 is typical).

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_multiscale\_topographic\_position\_image  
*Multiscale topographic position image*

---

**Description**

Creates a multiscale topographic position image from three DEVmax rasters of differing spatial scale ranges.

**Usage**

```
wbt_multiscale_topographic_position_image(
  local,
  meso,
  broad,
  output,
  hillshade = NULL,
  lightness = 1.2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

local	Input local-scale topographic position (DEVmax) raster file.
meso	Input meso-scale topographic position (DEVmax) raster file.
broad	Input broad-scale topographic position (DEVmax) raster file.
output	Output raster file.
hillshade	Input optional hillshade raster file. Note: a multi-directional (360-degree option) hillshade tends to work best in this application.
lightness	Image lightness value (default is 1.2).

<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_multi_part_to_single_part`  
*Multi part to single part*

---

**Description**

Converts a vector file containing multi-part features into a vector containing only single-part features.

**Usage**

```
wbt_multi_part_to_single_part(
  input,
  output,
  exclude_holes = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<code>input</code>	Input vector line or polygon file.
<code>output</code>	Output vector line or polygon file.
<code>exclude_holes</code>	Exclude hole parts from the feature splitting? (holes will continue to belong to their features in output.).
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.



**Value**

Returns the tool text outputs.

---

wbt\_narrowness\_index *Narrowness index*

---

**Description**

Calculates the narrowness of raster polygons.

**Usage**

```
wbt_narrowness_index(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_natural\_neighbour\_interpolation  
*Natural neighbour interpolation*

---

### Description

Creates a raster grid based on Sibson's natural neighbour method.

### Usage

```
wbt_natural_neighbour_interpolation(
  input,
  output,
  field = NULL,
  use_z = FALSE,
  cell_size = NULL,
  base = NULL,
  clip = TRUE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Input vector points file.
output	Output raster file.
field	Input field name in attribute table.
use_z	Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
clip	Clip the data to the convex hull of the points?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_nearest\_neighbour\_gridding  
*Nearest neighbour gridding*

---

### Description

Creates a raster grid based on a set of vector points and assigns grid values using the nearest neighbour.

### Usage

```
wbt_nearest_neighbour_gridding(  
  input,  
  field,  
  output,  
  use_z = FALSE,  
  cell_size = NULL,  
  base = NULL,  
  max_dist = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

### Arguments

input	Input vector Points file.
field	Input field name in attribute table.
output	Output raster file.
use_z	Use z-coordinate instead of field?.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
max_dist	Maximum search distance (optional).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_negate	<i>Negate</i>
------------	---------------

---

### Description

Changes the sign of values in a raster or the 0-1 values of a Boolean raster.

### Usage

```
wbt_negate(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_new_raster_from_base	<i>New raster from base</i>
--------------------------	-----------------------------

---

### Description

Creates a new raster using a base image.

**Usage**

```
wbt_new_raster_from_base(
    base,
    output,
    value = "nodata",
    data_type = "float",
    cell_size = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

base	Input base raster file.
output	Output raster file.
value	Constant value to fill raster with; either 'nodata' or numeric value.
data_type	Output raster data type; options include 'double' (64-bit), 'float' (32-bit), and 'integer' (signed 16-bit) (default is 'float').
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_normalized\_difference\_index  
*Normalized difference index*

---

**Description**

Calculate a normalized-difference index (NDI) from two bands of multispectral image data.

**Usage**

```
wbt_normalized_difference_index(
    input1,
    input2,
    output,
    clip = 0,
    correction = 0,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input image 1 (e.g. near-infrared band).
input2	Input image 2 (e.g. red band).
output	Output raster file.
clip	Optional amount to clip the distribution tails by, in percent.
correction	Optional adjustment value (e.g. 1, or 0.16 for the optimal soil adjusted vegetation index, OSAVI).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_normalize\_lidar    *Normalize lidar*

---

**Description**

Normalizes a LiDAR point cloud.

**Usage**

```
wbt_normalize_lidar(
    input,
    output,
    dtm,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input LiDAR file.
output	Name of the output LiDAR file.
dtm	Name of the input digital terrain model (DTM) raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_normal_vectors	<i>Normal vectors</i>
--------------------	-----------------------

---

**Description**

Calculates normal vectors for points within a LAS file and stores these data (XYZ vector components) in the RGB field.

**Usage**

```
wbt_normal_vectors(
    input,
    output,
    radius = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input LiDAR file.
output	Output LiDAR file.
radius	Search Radius.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_not	<i>Not</i>
---------	------------

---

**Description**

Performs a logical NOT operator on two Boolean raster images.

**Usage**

```
wbt_not(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input1	Input raster file.
input2	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.



**Value**

Returns the tool text outputs.

---

wbt_not_equal_to	<i>Not equal to</i>
------------------	---------------------

---

**Description**

Performs a not-equal-to comparison operation on two rasters or a raster and a constant value.

**Usage**

```
wbt_not_equal_to(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_num\_downslope\_neighbours  
*Num downslope neighbours*

---

### Description

Calculates the number of downslope neighbours to each grid cell in a DEM.

### Usage

```
wbt_num_downslope_neighbours(  
  dem,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_num\_inflowing\_neighbours  
*Num inflowing neighbours*

---

### Description

Computes the number of inflowing neighbours to each cell in an input DEM based on the D8 algorithm.

**Usage**

```
wbt_num_inflowing_neighbours(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_num\_upslope\_neighbours  
*Num upslope neighbours*

---

**Description**

Calculates the number of upslope neighbours to each grid cell in a DEM.

**Usage**

```
wbt_num_upslope_neighbours(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_olympic\_filter     *Olympic filter*

---

**Description**

Performs an olympic smoothing filter on an image.

**Usage**

```
wbt_olympic_filter(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_opening	<i>Opening</i>
-------------	----------------

---

**Description**

An opening is a mathematical morphology operation involving a dilation (max filter) of an erosion (min filter) set.

**Usage**

```
wbt_opening(
  input,
  output,
  filterx = 11,
  filtery = 11,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input      Input raster file.

output      Output raster file.

filterx      Size of the filter kernel in the x-direction.

filtery      Size of the filter kernel in the y-direction.

wd      Changes the working directory.

verbose\_mode      Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_openness	<i>Openness</i>
--------------	-----------------

---

### Description

This tool calculates the topographic openness index from an input DEM.

### Usage

```
wbt_openness(
    input,
    pos_output,
    neg_output,
    dist = 20,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input raster DEM file.
pos_output	Name of the positive openness output raster file.
neg_output	Name of the negative openness output raster file.
dist	Search distance, in grid cells.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_or	<i>Or</i>
--------	-----------

---

### Description

Performs a logical OR operator on two Boolean raster images.

### Usage

```
wbt_or(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input1	Input raster file.
input2	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_paired_sample_t_test`*Paired sample t test*

---

### Description

Performs a 2-sample K-S test for significant differences on two input rasters.

### Usage

```
wbt_paired_sample_t_test(  
    input1,  
    input2,  
    output,  
    num_samples = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>input1</code>	First input raster file.
<code>input2</code>	Second input raster file.
<code>output</code>	Output HTML file.
<code>num_samples</code>	Number of samples. Leave blank to use whole image.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_panchromatic\_sharpening  
*Panchromatic sharpening*

---

### Description

Increases the spatial resolution of image data by combining multispectral bands with panchromatic data.

### Usage

```
wbt_panchromatic_sharpening(
    pan,
    output,
    red = NULL,
    green = NULL,
    blue = NULL,
    composite = NULL,
    method = "brovey",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

pan	Input panchromatic band file.
output	Output colour composite file.
red	Input red band image file. Optionally specified if colour-composite not specified.
green	Input green band image file. Optionally specified if colour-composite not specified.
blue	Input blue band image file. Optionally specified if colour-composite not specified.
composite	Input colour-composite image file. Only used if individual bands are not specified.
method	Options include 'brovey' (default) and 'ihs'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_parallelepiped\_classification  
*Parallelepiped classification*

---

**Description**

Performs a supervised parallelepiped classification using training site polygons and multi-spectral images.

**Usage**

```
wbt_parallelepiped_classification(  
  inputs,  
  polys,  
  field,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

inputs	Name of the input band images.
polys	Name of the input training site polygons shapefile.
field	Name of the attribute containing class name data.
output	Name of the output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_patch\_orientation *Patch orientation*

---

**Description**

Calculates the orientation of vector polygons.

**Usage**

```
wbt_patch_orientation(  
    input,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_pennock\_landform\_class  
*Pennock landform class*

---

**Description**

Classifies hillslope zones based on slope, profile curvature, and plan curvature.

**Usage**

```
wbt_pennock_landform_class(
    dem,
    output,
    slope = 3,
    prof = 0.1,
    plan = 0,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
slope	Slope threshold value, in degrees (default is 3.0).
prof	Profile curvature threshold value (default is 0.1).
plan	Plan curvature threshold value (default is 0.0).
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_percentage\_contrast\_stretch  
*Percentage contrast stretch*

---

**Description**

Performs a percentage linear contrast stretch on input images.

**Usage**

```
wbt_percentage_contrast_stretch(
    input,
    output,
    clip = 1,
    tail = "both",
    num_tones = 256,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
clip	Optional amount to clip the distribution tails by, in percent.
tail	Specified which tails to clip; options include 'upper', 'lower', and 'both' (default is 'both').
num_tones	Number of tones in the output image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_percentile\_filter *Percentile filter*

---

**Description**

Performs a percentile filter on an input image.

**Usage**

```
wbt_percentile_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    sig_digits = 2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
sig_digits	Number of significant digits.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_percent\_elev\_range  
*Percent elev range*

---

**Description**

Calculates percent of elevation range from a DEM.

**Usage**

```
wbt_percent_elev_range(
    dem,
    output,
    filterx = 3,
    filtery = 3,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_percent\_equal\_to *Percent equal to*

---

**Description**

Calculates the percentage of a raster stack that have cell values equal to an input on a cell-by-cell basis.

**Usage**

```
wbt_percent_equal_to(
    inputs,
    comparison,
    output,
    wd = NULL,
```

```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

<code>inputs</code>	Input raster files.
<code>comparison</code>	Input comparison raster file.
<code>output</code>	Output raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_percent_greater_than`  
*Percent greater than*

---

### Description

Calculates the percentage of a raster stack that have cell values greater than an input on a cell-by-cell basis.

### Usage

```

wbt_percent_greater_than(
  inputs,
  comparison,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```



**Arguments**

inputs	Input raster files.
comparison	Input comparison raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_percent\_less\_than *Percent less than*

---

**Description**

Calculates the percentage of a raster stack that have cell values less than an input on a cell-by-cell basis.

**Usage**

```
wbt_percent_less_than(
  inputs,
  comparison,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
comparison	Input comparison raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_perimeter_area_ratio`  
*Perimeter area ratio*

---

**Description**

Calculates the perimeter-area ratio of vector polygons.

**Usage**

```
wbt_perimeter_area_ratio(
  input,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

`input` Input vector polygon file.

`wd` Changes the working directory.

`verbose_mode` Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

`compress_rasters` Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

`command_only` Return command that would be executed by `system()` rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_phi\_coefficient    *Phi coefficient*

---

## Description

This tool performs a binary classification accuracy assessment.

## Usage

```
wbt_phi_coefficient(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input1	Name of the first input raster image file.
input2	Name of the second input raster image file.
output	Name of the output HTML file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.

---

wbt\_pick\_from\_list      *Pick from list*

---

### Description

Outputs the value from a raster stack specified by a position raster.

### Usage

```
wbt_pick_from_list(  
    inputs,  
    pos_input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

inputs	Input raster files.
pos_input	Input position raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_piecewise_contrast_stretch`*Piecewise contrast stretch*

---

**Description**

Performs a piecewise contrast stretch on an input image.

**Usage**

```
wbt_piecewise_contrast_stretch(  
    input,  
    output,  
    FUN = "",  
    greytone = 1024,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Name of the input raster image file.
<code>output</code>	Name of the output raster image file.
<code>FUN</code>	Piecewise break-points e.g. '(50, 0.1); (150, 0.8); (255; 1.0)'. The first value is the input value, the second is the output value.
<code>greytone</code>	Number of greytone in the output image.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_plan\_curvature      *Plan curvature*

---

### Description

Calculates a plan (contour) curvature raster from an input DEM.

### Usage

```
wbt_plan_curvature(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_polygonize	<i>Polygonize</i>
----------------	-------------------

---

### Description

Creates a polygon layer from two or more intersecting line features contained in one or more input vector line files.

### Usage

```
wbt_polygonize(
    inputs,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

inputs	Input vector polyline file.
output	Output vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_polygons_to_lines	<i>Polygons to lines</i>
-----------------------	--------------------------

---

### Description

Converts vector polygons to polylines.

**Usage**

```
wbt_polygons_to_lines(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector polygon file.
output	Output vector lines file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_polygon_area	<i>Polygon area</i>
------------------	---------------------

---

**Description**

Calculates the area of vector polygons.

**Usage**

```
wbt_polygon_area(
    input,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```



**Arguments**

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_polygon\_long\_axis *Polygon long axis*

---

**Description**

Used to map the long axis of polygon features.

**Usage**

```
wbt_polygon_long_axis(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector polygons file.
output	Output vector polyline file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_polygon\_perimeter *Polygon perimeter*

---

### Description

Calculates the perimeter of vector polygons.

### Usage

```
wbt_polygon_perimeter(
    input,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_polygon\_short\_axis  
*Polygon short axis*

---

### Description

Used to map the short axis of polygon features.

**Usage**

```
wbt_polygon_short_axis(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector polygons file.
output	Output vector polyline file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_power	<i>Power</i>
-----------	--------------

---

**Description**

Raises the values in grid cells of one rasters, or a constant value, by values in another raster or constant value.

**Usage**

```
wbt_power(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_prewitt\_filter      *Prewitt filter*

---

**Description**

Performs a Prewitt edge-detection filter on an image.

**Usage**

```
wbt_prewitt_filter(
  input,
  output,
  clip = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
clip	Optional amount to clip the distribution tails by, in percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_principal\_component\_analysis  
*Principal component analysis*

---

**Description**

Performs a principal component analysis (PCA) on a multi-spectral dataset.

**Usage**

```
wbt_principal_component_analysis(  
    inputs,  
    output,  
    num_comp = NULL,  
    standardized = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

inputs	Input raster files.
output	Output HTML report file.
num_comp	Number of component images to output; <= to num. input images.
standardized	Perform standardized PCA?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_print\_geo\_tiff\_tags

*Print geo tiff tags*

---

### Description

Prints the tags within a GeoTIFF.

### Usage

```
wbt_print_geo_tiff_tags(  
    input,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input GeoTIFF file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_profile

*Profile*

---

### Description

Plots profiles from digital surface models.

**Usage**

```
wbt_profile(
  lines,
  surface,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

lines	Input vector line file.
surface	Input raster surface file.
output	Output HTML file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_profile\_curvature *Profile curvature*

---

**Description**

Calculates a profile curvature raster from an input DEM.

**Usage**

```
wbt_profile_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_qin\_flow\_accumulation  
*Qin flow accumulation*

---

**Description**

Calculates Qin et al. (2007) flow accumulation.

**Usage**

```
wbt_qin_flow_accumulation(  
  dem,  
  output,  
  out_type = "specific contributing area",  
  exponent = 10,  
  max_slope = 45,  
  threshold = NULL,  
  log = FALSE,  
  clip = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```



**Arguments**

dem	Name of the input DEM raster file; must be depressionless.
output	Name of the output raster file.
out_type	Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
exponent	Optional upper-bound exponent parameter; default is 10.0.
max_slope	Optional upper-bound slope parameter, in degrees (0-90); default is 45.0.
threshold	Optional convergence threshold parameter, in grid cells; default is infinity.
log	Log-transform the output values?.
clip	Optional flag to request clipping the display max by 1 percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_quantiles	<i>Quantiles</i>
---------------	------------------

---

**Description**

Transforms raster values into quantiles.

**Usage**

```
wbt_quantiles(
  input,
  output,
  num_quantiles = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
num_quantiles	Number of quantiles.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_quinn\_flow\_accumulation  
*Quinn flow accumulation*

---

**Description**

Calculates Quinn et al. (1995) flow accumulation.

**Usage**

```
wbt_quinn_flow_accumulation(
  dem,
  output,
  out_type = "specific contributing area",
  exponent = 1,
  threshold = NULL,
  log = FALSE,
  clip = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Name of the input DEM raster file; must be depressionless.
output	Name of the output raster file.
out_type	Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
exponent	Optional exponent parameter; default is 1.0.
threshold	Optional convergence threshold parameter, in grid cells; default is infinity.
log	Log-transform the output values?.
clip	Optional flag to request clipping the display max by 1 percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_radial\_basis\_function\_interpolation  
*Radial basis function interpolation*

---

**Description**

Interpolates vector points into a raster surface using a radial basis function scheme.

**Usage**

```
wbt_radial_basis_function_interpolation(
  input,
  field,
  output,
  use_z = FALSE,
  radius = NULL,
  min_points = NULL,
  func_type = "ThinPlateSpline",
  poly_order = "none",
  weight = 0.1,
  cell_size = NULL,
  base = NULL,
  wd = NULL,
```

```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input vector points file.
field	Input field name in attribute table.
output	Output raster file.
use_z	Use z-coordinate instead of field?.
radius	Search Radius (in map units).
min_points	Minimum number of points.
func_type	Radial basis function type; options are 'ThinPlateSpline' (default), 'PolyHarmonic', 'Gaussian', 'MultiQuadric', 'InverseMultiQuadric'.
poly_order	Polynomial order; options are 'none' (default), 'constant', 'affine'.
weight	Weight parameter used in basis function.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_radius\_of\_gyration  
*Radius of gyration*

---

### Description

Calculates the distance of cells from their polygon's centroid.

**Usage**

```
wbt_radius_of_gyration(
    input,
    output,
    text_output = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
text_output	Optional text output.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_raise_walls	<i>Raise walls</i>
-----------------	--------------------

---

**Description**

Raises walls in a DEM along a line or around a polygon, e.g. a watershed.

**Usage**

```
wbt_raise_walls(
    input,
    dem,
    output,
    breach = NULL,
    height = 100,
    wd = NULL,
    verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input vector lines or polygons file.
dem	Input raster DEM file.
output	Output raster file.
breach	Optional input vector breach lines.
height	Wall height.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_random_field	<i>Random field</i>
------------------	---------------------

---

### Description

Creates an image containing random values.

### Usage

```

wbt_random_field(
  base,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

base	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_random\_forest\_classification  
*Random forest classification*

---

**Description**

Performs a supervised random forest classification using training site polygons/points and predictor rasters.

**Usage**

```
wbt_random_forest_classification(  
  inputs,  
  training,  
  field,  
  output = NULL,  
  split_criterion = "Gini",  
  n_trees = 500,  
  min_samples_leaf = 1,  
  min_samples_split = 2,  
  test_proportion = 0.2,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

inputs	Names of the input predictor rasters.
training	Name of the input training site polygons/points shapefile.
field	Name of the attribute containing class data.
output	Name of the output raster file.
split_criterion	Split criterion to use when building a tree. Options include 'Gini', 'Entropy', and 'ClassificationError'.
n_trees	The number of trees in the forest.
min_samples_leaf	The minimum number of samples required to be at a leaf node.
min_samples_split	The minimum number of samples required to split an internal node.
test_proportion	The proportion of the dataset to include in the test split; default is 0.2.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_random\_forest\_regression  
*Random forest regression*

---

**Description**

Performs a random forest regression analysis using training site data and predictor rasters.

**Usage**

```
wbt_random_forest_regression(  
  inputs,  
  training,  
  field,  
  output = NULL,  
  n_trees = 100,
```



```

    min_samples_leaf = 1,
    min_samples_split = 2,
    test_proportion = 0.2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

inputs	Names of the input predictor rasters.
training	Name of the input training site points shapefile.
field	Name of the attribute containing response variable name data.
output	Name of the output raster file. This parameter is optional. When unspecified, the tool will only build the model. When specified, the tool will use the built model and predictor rasters to perform a spatial prediction.
n_trees	The number of trees in the forest.
min_samples_leaf	The minimum number of samples required to be at a leaf node.
min_samples_split	The minimum number of samples required to split an internal node.
test_proportion	The proportion of the dataset to include in the test split; default is 0.2.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_random_sample	<i>Random sample</i>
-------------------	----------------------

---

### Description

Creates an image containing randomly located sample grid cells with unique IDs.

**Usage**

```
wbt_random_sample(
    base,
    output,
    num_samples = 1000,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

base	Input raster file.
output	Output raster file.
num_samples	Number of samples.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_range_filter	<i>Range filter</i>
------------------	---------------------

---

**Description**

Assigns each cell in the output grid the range of values in a moving window centred on each grid cell in the input raster.

**Usage**

```
wbt_range_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    wd = NULL,
    verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_rasterize\_streams *Rasterize streams*

---

### Description

Rasterizes vector streams based on Lindsay (2016) method.

### Usage

```

wbt_rasterize_streams(
  streams,
  base,
  output,
  nodata = TRUE,
  feature_id = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

streams	Input vector streams file.
base	Input base raster file.
output	Output raster file.
nodata	Use NoData value for background?.
feature_id	Use feature number as output value?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_raster_area	<i>Raster area</i>
-----------------	--------------------

---

**Description**

Calculates the area of polygons or classes within a raster image.

**Usage**

```
wbt_raster_area(
  input,
  output = NULL,
  out_text = FALSE,
  units = "grid cells",
  zero_back = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
out_text	Would you like to output polygon areas to text?.
units	Area units; options include 'grid cells' and 'map units'.
zero_back	Flag indicating whether zero values should be treated as a background.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_raster\_calculator *Raster calculator*

---

**Description**

Performs a complex mathematical operations on one or more input raster images on a cell-to-cell basis.

**Usage**

```
wbt_raster_calculator(
  output,
  statement = "",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

output	Name of the output raster file.
statement	Statement e.g. cos("raster1") * 35.0 + "raster2". This statement must be a valid Rust statement.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_raster\_cell\_assignment  
*Raster cell assignment*

---

**Description**

Assign row or column number to cells.

**Usage**

```
wbt_raster_cell_assignment(  
  input,  
  output,  
  assign = "column",  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
assign	Which variable would you like to assign to grid cells? Options include 'column', 'row', 'x', and 'y'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_raster\_histogram *Raster histogram*

---

**Description**

Creates a histogram from raster values.

**Usage**

```
wbt_raster_histogram(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output HTML file (default name will be based on input file if unspecified).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_raster\_perimeter *Raster perimeter*

---

### Description

Calculates the perimeters of polygons or classes within a raster image.

### Usage

```
wbt_raster_perimeter(  
    input,  
    output = NULL,  
    out_text = FALSE,  
    units = "grid cells",  
    zero_back = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
out_text	Would you like to output polygon areas to text?.
units	Area units; options include 'grid cells' and 'map units'.
zero_back	Flag indicating whether zero values should be treated as a background.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

`wbt_raster_streams_to_vector`*Raster streams to vector*

---

### Description

Converts a raster stream file into a vector file.

### Usage

```
wbt_raster_streams_to_vector(  
    streams,  
    d8_pntr,  
    output,  
    esri_pntr = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>streams</code>	Input raster streams file.
<code>d8_pntr</code>	Input raster D8 pointer file.
<code>output</code>	Output vector file.
<code>esri_pntr</code>	D8 pointer uses the ESRI style scheme.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_raster_summary_stats`*Raster summary stats*

---

**Description**

Measures a rasters min, max, average, standard deviation, num. non-nodata cells, and total.

**Usage**

```
wbt_raster_summary_stats(  
    input,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_raster_to_vector_lines`*Raster to vector lines*

---

**Description**

Converts a raster lines features into a vector of the POLYLINE shapetype.

**Usage**

```
wbt_raster_to_vector_lines(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster lines file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_raster\_to\_vector\_points  
*Raster to vector points*

---

**Description**

Converts a raster dataset to a vector of the POINT shapetype.

**Usage**

```
wbt_raster_to_vector_points(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output vector points file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_raster\_to\_vector\_polygons  
*Raster to vector polygons*

---

**Description**

Converts a raster dataset to a vector of the POLYGON shapetype.

**Usage**

```
wbt_raster_to_vector_polygons(  
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output vector polygons file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_reciprocal	<i>Reciprocal</i>
----------------	-------------------

---

**Description**

Returns the reciprocal (i.e.  $1 / z$ ) of values in a raster.

**Usage**

```
wbt_reciprocal(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_reclass	<i>Reclass</i>
-------------	----------------

---

### Description

Reclassifies the values in a raster image.

### Usage

```
wbt_reclass(
    input,
    output,
    reclass_vals,
    assign_mode = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
reclass_vals	Reclassification triplet values (new value; from value; to less than), e.g. '0.0;0.0;1.0;1.0;1.0;2.0'.
assign_mode	Optional Boolean flag indicating whether to operate in assign mode, reclass_vals values are interpreted as new value; old value pairs.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_reclass\_equal\_interval  
*Reclass equal interval*

---

### Description

Reclassifies the values in a raster image based on equal-ranges.

### Usage

```
wbt_reclass_equal_interval(  
    input,  
    output,  
    interval = 10,  
    start_val = NULL,  
    end_val = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
interval	Class interval size.
start_val	Optional starting value (default is input minimum value).
end_val	Optional ending value (default is input maximum value).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_reclass\_from\_file *Reclass from file*

---

## Description

Reclassifies the values in a raster image using reclass ranges in a text file.

## Usage

```
wbt_reclass_from_file(  
    input,  
    reclass_file,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input	Input raster file.
reclass_file	Input text file containing reclass ranges.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.



---

```
wbt_reconcile_multiple_headers
    Reconcile multiple headers
```

---

### Description

This tool adjusts the crop yield values for data sets collected with multiple headers or combines.

### Usage

```
wbt_reconcile_multiple_headers(
    input,
    region_field,
    yield_field,
    output,
    radius = NULL,
    min_yield = NULL,
    max_yield = NULL,
    mean_tonnage = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input points shapefile.
region_field	Name of the attribute containing region data.
yield_field	Name of the attribute containing yield data.
output	Name of the output points shapefile.
radius	Optional search radius, in metres. Only specify this value if you want to calculate locally normalized yield.
min_yield	Minimum yield value in output.
max_yield	Maximum yield value in output.
mean_tonnage	Use this optional parameter to force the output to have a certain overall average tonnage.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

```
wbt_recover_flightline_info
    Recover flightline info
```

---

**Description**

Associates LiDAR points by their flightlines.

**Usage**

```
wbt_recover_flightline_info(
    input,
    output,
    max_time_diff = 5,
    pt_src_id = FALSE,
    user_data = FALSE,
    rgb = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
max_time_diff	Maximum in-flightline time difference (seconds).
pt_src_id	Add flightline information to the point source ID.
user_data	Add flightline information to the user data.
rgb	Add flightline information to the RGB colour data.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_recreate\_pass\_lines  
*Recreate pass lines*

---

### Description

This tool can be used to approximate the harvester pass lines from yield points.

### Usage

```
wbt_recreate_pass_lines(
    input,
    yield_field_name,
    output_lines,
    output_points,
    max_change_in_heading = 25,
    ignore_zeros = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input points shapefile.
yield_field_name	Name of the attribute containing yield data.
output_lines	Name of the output pass lines shapefile.
output_points	Name of the output points shapefile.
max_change_in_heading	Max change in heading.
ignore_zeros	Ignore zero-valued yield points?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_reinitialize\_attribute\_table  
*Reinitialize attribute table*

---

### Description

Reinitializes a vector's attribute table deleting all fields but the feature ID (FID).

### Usage

```
wbt_reinitialize_attribute_table(  

    input,  

    wd = NULL,  

    verbose_mode = FALSE,  

    compress_rasters = FALSE,  

    command_only = FALSE  

)
```

### Arguments

input	Input vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_related\_circumscribing\_circle  
*Related circumscribing circle*

---

### Description

Calculates the related circumscribing circle of vector polygons.

**Usage**

```
wbt_related_circumscribing_circle(
    input,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_relative\_aspect    *Relative aspect*

---

**Description**

Calculates relative aspect (relative to a user-specified direction) from an input DEM.

**Usage**

```
wbt_relative_aspect(
    dem,
    output,
    azimuth = 0,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
azimuth	Illumination source azimuth.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_relative\_topographic\_position  
*Relative topographic position*

---

**Description**

Calculates the relative topographic position index from a DEM.

**Usage**

```
wbt_relative_topographic_position(  
  dem,  
  output,  
  filterx = 11,  
  filtery = 11,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_remove\_field\_edge\_points  
*Remove field edge points*

---

**Description**

This tool can be used to remove, or flag, most of the points along the edges from a crop yield data set.

**Usage**

```
wbt_remove_field_edge_points(  
  input,  
  output,  
  dist = NULL,  
  max_change_in_heading = 25,  
  flag_edges = FALSE,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Name of the input points shapefile.
output	Name of the output points shapefile.
dist	Average distance between passes, in meters.
max_change_in_heading	Max change in heading.
flag_edges	Don't remove edge points, just flag them in the attribute table?.
wd	Changes the working directory.

<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_remove_off_terrain_objects`  
*Remove off terrain objects*

---

**Description**

Removes off-terrain objects from a raster digital elevation model (DEM).

**Usage**

```
wbt_remove_off_terrain_objects(  
  dem,  
  output,  
  filter = 11,  
  slope = 15,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

<code>dem</code>	Input raster DEM file.
<code>output</code>	Output raster file.
<code>filter</code>	Filter size (cells).
<code>slope</code>	Slope threshold value.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.



**Value**

Returns the tool text outputs.

---

wbt\_remove\_polygon\_holes  
*Remove polygon holes*

---

**Description**

Removes holes within the features of a vector polygon file.

**Usage**

```
wbt_remove_polygon_holes(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input vector polygon file.
output	Output vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_remove\_raster\_polygon\_holes  
*Remove raster polygon holes*

---

### Description

Removes polygon holes, or 'donut-holes', from raster polygons.

### Usage

```
wbt_remove_raster_polygon_holes(  
    input,  
    output,  
    threshold = 3,  
    use_diagonals = TRUE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Name of the input raster image file.
output	Name of the output raster file.
threshold	Maximum size of removed holes, in grid cells. Blank for no threshold, i.e. remove all holes.
use_diagonals	Use diagonal neighbours during clumping operation.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_remove_short_streams`*Remove short streams*

---

**Description**

Removes short first-order streams from a stream network.

**Usage**

```
wbt_remove_short_streams(  
    d8_pntr,  
    streams,  
    output,  
    min_length,  
    esri_pntr = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>d8_pntr</code>	Input raster D8 pointer file.
<code>streams</code>	Input raster streams file.
<code>output</code>	Output raster file.
<code>min_length</code>	Minimum tributary length (in map units) used for network pruning.
<code>esri_pntr</code>	D8 pointer uses the ESRI style scheme.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_remove_spurs	<i>Remove spurs</i>
------------------	---------------------

---

### Description

Removes the spurs (pruning operation) from a Boolean line image; intended to be used on the output of the LineThinning tool.

### Usage

```
wbt_remove_spurs(  
    input,  
    output,  
    iterations = 10,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
iterations	Maximum number of iterations.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_repair\_stream\_vector\_topology  
*Repair stream vector topology*

---

## Description

This tool resolves topological errors and inconsistencies associated with digitized vector streams.

## Usage

```
wbt_repair_stream_vector_topology(  
    input,  
    output,  
    dist = "",  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input	Name of the input lines vector file.
output	Name of the output lines vector file.
dist	Snap distance, in xy units (metres).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.

---

wbt\_resample

*Resample*


---

### Description

Resamples one or more input images into a destination image.

### Usage

```
wbt_resample(
    inputs,
    output,
    cell_size = NULL,
    base = NULL,
    method = "cc",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

inputs	Input raster files.
output	Output raster file.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
method	Resampling method; options include 'nn' (nearest neighbour), 'bilinear', and 'cc' (cubic convolution).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_rescale_value_range`*Rescale value range*

---

**Description**

Performs a min-max contrast stretch on an input greytone image.

**Usage**

```
wbt_rescale_value_range(  
    input,  
    output,  
    out_min_val,  
    out_max_val,  
    clip_min = NULL,  
    clip_max = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input raster file.
<code>output</code>	Output raster file.
<code>out_min_val</code>	New minimum value in output image.
<code>out_max_val</code>	New maximum value in output image.
<code>clip_min</code>	Optional lower tail clip value.
<code>clip_max</code>	Optional upper tail clip value.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_rgb\_to\_ihs            *Rgb to ihs*

---

### Description

Converts red, green, and blue (RGB) images into intensity, hue, and saturation (IHS) images.

### Usage

```
wbt_rgb_to_ihs(
  intensity,
  hue,
  saturation,
  red = NULL,
  green = NULL,
  blue = NULL,
  composite = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

intensity	Output intensity raster file.
hue	Output hue raster file.
saturation	Output saturation raster file.
red	Input red band image file. Optionally specified if colour-composite not specified.
green	Input green band image file. Optionally specified if colour-composite not specified.
blue	Input blue band image file. Optionally specified if colour-composite not specified.
composite	Input colour-composite image file. Only used if individual bands are not specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.



**Value**

Returns the tool text outputs.

---

wbt\_rho8\_flow\_accumulation  
*Rho8 flow accumulation*

---

**Description**

Calculates Fairfield and Leymarie (1991) flow accumulation.

**Usage**

```
wbt_rho8_flow_accumulation(
    input,
    output,
    out_type = "specific contributing area",
    log = FALSE,
    clip = FALSE,
    pntr = FALSE,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input DEM or Rho8 pointer file; if a DEM is used, it must be depressionless.
output	Name of the output raster file.
out_type	Output type; one of 'cells', 'specific contributing area' (default), and 'catchment area'.
log	Log-transform the output values?.
clip	Optional flag to request clipping the display max by 1 percent.
pntr	Is the input raster a Rho8 flow pointer rather than a DEM?.
esri_pntr	Does the input Rho8 pointer use the ESRI style scheme?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_rho8_pointer	<i>Rho8 pointer</i>
------------------	---------------------

---

**Description**

Calculates a stochastic Rho8 flow pointer raster from an input DEM.

**Usage**

```
wbt_rho8_pointer(
    dem,
    output,
    esri_pntr = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_ring_curvature	<i>Ring curvature</i>
--------------------	-----------------------

---

### Description

This tool calculates ring curvature from an input DEM.

### Usage

```
wbt_ring_curvature(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
log	Display output values using a log-scale.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_river\_centerlines *River centerlines*

---

### Description

Maps river centerlines from an input water raster.

### Usage

```
wbt_river_centerlines(  
    input,  
    output,  
    min_length = 3,  
    radius = 4,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Name of the input raster image file.
output	Name of the output vector lines file.
min_length	Minimum line length, in grid cells.
radius	Search radius for joining distant endnodes, in grid cells.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_roberts\_cross\_filter  
*Roberts cross filter*

---

### Description

Performs a Robert's cross edge-detection filter on an image.

### Usage

```
wbt_roberts_cross_filter(  
    input,  
    output,  
    clip = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
clip	Optional amount to clip the distribution tails by, in percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_root\_mean\_square\_error  
*Root mean square error*

---

### Description

Calculates the RMSE and other accuracy statistics.

### Usage

```
wbt_root_mean_square_error(  
    input,  
    base,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
base	Input base raster file used for comparison.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_rotor                      *Rotor*

---

### Description

This tool calculates rotor from an input DEM.

**Usage**

```
wbt_rotor(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
log	Display output values using a log-scale.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_round	<i>Round</i>
-----------	--------------

---

**Description**

Rounds the values in an input raster to the nearest integer value.

**Usage**

```
wbt_round(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_ruggedness\_index *Ruggedness index*

---

**Description**

Calculates the Riley et al.'s (1999) terrain ruggedness index from an input DEM.

**Usage**

```
wbt_ruggedness_index(
  dem,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.



**Value**

Returns the tool text outputs.

---

wbt_run_tool	<i>Run a tool in 'WhiteboxTools' by name</i>
--------------	--

---

**Description**

Runs a tool and specifies tool arguments. If the prefix "whitebox:." or "wbt\_" is in tool\_name it is removed to match the definitions in wbt\_list\_tools()

**Usage**

```
wbt_run_tool(tool_name, args, verbose_mode = FALSE, command_only = FALSE)
```

**Arguments**

tool_name	The name of the tool to run.
args	Tool arguments.
verbose_mode	Verbose mode. Without this flag, tool outputs will not be printed.
command_only	Return command that would be run with system()? Default: FALSE

**Value**

Returns the (character) output of the tool.

**See Also**

[wbt\\_list\\_tools](#)

**Examples**

```
## Not run:
tool_name <- "breach_depressions"
dem <- system.file("extdata", "DEM.tif", package="whitebox")
output <- "./output.tif"
arg1 <- paste0("--dem=", dem)
arg2 <- paste0("--output=", output)
args <- paste(arg1, arg2)
wbt_run_tool(tool_name, args)

## End(Not run)
```

---

wbt_rust_backtrace	<i>Convenience method for setting RUST_BACKTRACE options for debugging</i>
--------------------	--

---

**Description**

Convenience method for setting RUST\_BACKTRACE options for debugging

**Usage**

```
wbt_rust_backtrace(RUST_BACKTRACE = c("0", "1", "full"))
```

**Arguments**

RUST\_BACKTRACE One of "0", "1", "full", Logical values are converted to integer and then character.

**Value**

value of system environment variable RUST\_BACKTRACE

**Examples**

```
## Not run:
wbt_rust_backtrace(TRUE)

## End(Not run)
```

---

wbt_scharr_filter	<i>Scharr filter</i>
-------------------	----------------------

---

**Description**

Performs a Scharr edge-detection filter on an image.

**Usage**

```
wbt_scharr_filter(
  input,
  output,
  clip = 0,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
clip	Optional amount to clip the distribution tails by, in percent.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_sediment\_transport\_index  
*Sediment transport index*

---

**Description**

Calculates the sediment transport index.

**Usage**

```
wbt_sediment_transport_index(  
  sca,  
  slope,  
  output,  
  sca_exponent = 0.4,  
  slope_exponent = 1.3,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

sca	Input raster specific contributing area (SCA) file.
slope	Input raster slope file.
output	Output raster file.
sca_exponent	SCA exponent value.

slope_exponent	Slope exponent value.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_select\_tiles\_by\_polygon  
*Select tiles by polygon*

---

**Description**

Copies LiDAR tiles overlapping with a polygon into an output directory.

**Usage**

```
wbt_select_tiles_by_polygon(  
  indir,  
  outdir,  
  polygons,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

indir	Input LAS file source directory.
outdir	Output directory into which LAS files within the polygon are copied.
polygons	Input vector polygons file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_set\_nodata\_value *Set nodata value*

---

**Description**

Assign the NoData value for an input image.

**Usage**

```
wbt_set_nodata_value(
    input,
    output,
    back_value = 0,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
back_value	Background value to set to nodata.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_shadow\_animation    *Shadow animation*

---

## Description

This tool creates an animated GIF of shadows based on an input DEM.

## Usage

```
wbt_shadow_animation(
    input,
    output,
    palette = "atlas",
    max_dist = "",
    date = "21/06/2021",
    interval = 15,
    location = "43.5448/-80.2482/-4",
    height = 600,
    delay = 250,
    label = "",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

## Arguments

input	Name of the input digital surface model (DSM) raster file.
output	Name of the output HTML file (*.html).
palette	DSM image palette; options are 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'light_quant', 'purple', 'viridis', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'none'.
max_dist	Optional maximum search distance, in xy units. Minimum value is 5 x cell size.
date	Date in format DD/MM/YYYY.
interval	Time interval, in minutes (1-60).
location	Location, defined as Lat/Long/UTC-offset (e.g. 43.5448/-80.2482/-4).
height	Image height, in pixels.
delay	GIF time delay in milliseconds.
label	Label text (leave blank for none).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_shadow_image	<i>Shadow image</i>
------------------	---------------------

---

**Description**

This tool creates a raster of shadow areas based on an input DEM.

**Usage**

```
wbt_shadow_image(
    input,
    output,
    palette = "soft",
    max_dist = "",
    date = "21/06/2021",
    time = "13:00",
    location = "43.5448/-80.2482/-4",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Name of the input digital surface model (DSM) raster file.
output	Name of the output raster file.
palette	DSM image palette; options are 'atlas', 'high_relief', 'arid', 'soft', 'muted', 'light_quant', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', and 'none'.
max_dist	Optional maximum search distance, in xy unites. Minimum value is 5 x cell size.
date	Date in format DD/MM/YYYY.
time	Time in format HH::MM, e.g. 03:15AM or 14:30.
location	Location, defined as Lat/Long/UTC-offset (e.g. 43.5448/-80.2482/-4).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_shape\_complexity\_index  
*Shape complexity index*

---

**Description**

Calculates overall polygon shape complexity or irregularity.

**Usage**

```
wbt_shape_complexity_index(  
    input,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.



---

wbt\_shape\_complexity\_index\_raster  
*Shape complexity index raster*

---

**Description**

Calculates the complexity of raster polygons or classes.

**Usage**

```
wbt_shape_complexity_index_raster(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_shape\_index      *Shape index*

---

**Description**

This tool calculates the shape index from an input DEM.

**Usage**

```
wbt_shape_index(
    dem,
    output,
    zfactor = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_shreve\_stream\_magnitude  
*Shreve stream magnitude*

---

**Description**

Assigns the Shreve stream magnitude to each link in a stream network.

**Usage**

```
wbt_shreve_stream_magnitude(
    d8_pntr,
    streams,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
```

```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_sigmoidal_contrast_stretch
    Sigmoidal contrast stretch

```

---

### Description

Performs a sigmoidal contrast stretch on input images.

### Usage

```

wbt_sigmoidal_contrast_stretch(
  input,
  output,
  cutoff = 0,
  gain = 1,
  num_tones = 256,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

input	Input raster file.
output	Output raster file.
cutoff	Cutoff value between 0.0 and 0.95.
gain	Gain value.
num_tones	Number of tones in the output image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_sin

*Sin*


---

**Description**

Returns the sine (sin) of each values in a raster.

**Usage**

```
wbt_sin(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_single\_part\_to\_multi\_part  
*Single part to multi part*

---

**Description**

Converts a vector file containing multi-part features into a vector containing only single-part features.

**Usage**

```
wbt_single_part_to_multi_part(
    input,
    output,
    field = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector line or polygon file.
output	Output vector line or polygon file.
field	Grouping ID field name in attribute table.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_sinh

*Sinh*


---

### Description

Returns the hyperbolic sine (sinh) of each values in a raster.

### Usage

```
wbt_sinh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_sink

*Sink*


---

### Description

Identifies the depressions in a DEM, giving each feature a unique identifier.

**Usage**

```
wbt_sink(
  input,
  output,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster DEM file.
output	Output raster file.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_slope	<i>Slope</i>
-----------	--------------

---

**Description**

Calculates a slope raster from an input DEM.

**Usage**

```
wbt_slope(
  dem,
  output,
  zfactor = NULL,
  units = "degrees",
  wd = NULL,
  verbose_mode = FALSE,
```

```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
units	Units of output raster; options include 'degrees', 'radians', 'percent'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_slope_vs_aspect_plot
    Slope vs aspect plot

```

---

### Description

This tool creates a slope-aspect relation plot from an input DEM.

### Usage

```

wbt_slope_vs_aspect_plot(
  input,
  output,
  bin_size = 2,
  min_slope = 0.1,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```



**Arguments**

input	Name of the input raster image file.
output	Name of the output report file (*.html).
bin_size	Aspect bin size, in degrees.
min_slope	Minimum slope, in degrees.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_slope\_vs\_elevation\_plot  
*Slope vs elevation plot*

---

**Description**

Creates a slope vs. elevation plot for one or more DEMs.

**Usage**

```
wbt_slope_vs_elevation_plot(
  inputs,
  output,
  watershed = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

inputs	Input DEM files.
output	Output HTML file (default name will be based on input file if unspecified).
watershed	Input watershed files (optional).

wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_smooth\_vectors     *Smooth vectors*

---

**Description**

Smooths a vector coverage of either a POLYLINE or POLYGON base ShapeType.

**Usage**

```
wbt_smooth_vectors(
  input,
  output,
  filter = 3,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector POLYLINE or POLYGON file.
output	Output vector file.
filter	The filter size, any odd integer greater than or equal to 3; default is 3.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_smooth\_vegetation\_residual  
*Smooth vegetation residual*

---

### Description

This tool can smooth the residual roughness due to vegetation cover in LiDAR DEMs.

### Usage

```
wbt_smooth_vegetation_residual(  
    input,  
    output,  
    max_scale = 30,  
    dev_threshold = 1,  
    scale_threshold = 5,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Name of the input digital elevation model (DEM) raster file.
output	Name of the output raster file.
max_scale	Maximum search neighbourhood radius in grid cells.
dev_threshold	DEVmax Threshold.
scale_threshold	DEVmax scale threshold.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_snap\_pour\_points    *Snap pour points*

---

### Description

Moves outlet points used to specify points of interest in a watershedding operation to the cell with the highest flow accumulation in its neighbourhood.

### Usage

```
wbt_snap_pour_points(  
    pour_pts,  
    flow_accum,  
    output,  
    snap_dist,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>pour_pts</code>	Input vector pour points (outlet) file.
<code>flow_accum</code>	Input raster D8 flow accumulation file.
<code>output</code>	Output vector file.
<code>snap_dist</code>	Maximum snap distance in map units.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_sobel\_filter      *Sobel filter*

---

### Description

Performs a Sobel edge-detection filter on an image.

### Usage

```
wbt_sobel_filter(  
    input,  
    output,  
    variant = "3x3",  
    clip = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
variant	Optional variant value. Options include 3x3 and 5x5 (default is 3x3).
clip	Optional amount to clip the distribution tails by, in percent (default is 0.0).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_sort_lidar	<i>Sort lidar</i>
----------------	-------------------

---

### Description

Sorts LiDAR points based on their properties.

### Usage

```
wbt_sort_lidar(
    input,
    output = NULL,
    criteria = "",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input LiDAR points.
output	Name of the output LiDAR points.
criteria	Sort criteria e.g. 'x 50.0, y 50.0, z'; criteria may include x, y, z, intensity, class, user_data, point_source_id, and scan_angle.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_source	<i>Initialize an R object containing spatial data for use by WhiteboxTools</i>
------------	--

---

### Description

Initialize an R object containing spatial data for use by WhiteboxTools

### Usage

```
wbt_source(  
  x,  
  dsn = NULL,  
  layer = NULL,  
  force = FALSE,  
  verbose = wbt_verbose(),  
  ...  
)
```

### Arguments

x	A terra SpatVector or sf object, or a path to a file that can be read as a SpatVectorProxy
dsn	Data source path / file name
layer	Data layer
force	Force write of vector data to file? Default: FALSE (only write if file does not exist)
verbose	Print information about data source and contents?
...	Additional arguments passed to terra::writeVector() or sf::st_write()

### Value

An R object with attributes wbt\_dsn and wbt\_layer set as needed to support reading and writing R objects from file by WhiteboxTools.

---

wbt_spherical_std_dev_of_normals	<i>Spherical std dev of normals</i>
----------------------------------	-------------------------------------

---

### Description

Calculates the spherical standard deviation of surface normals for a DEM.

**Usage**

```
wbt_spherical_std_dev_of_normals(
    dem,
    output,
    filter = 11,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
filter	Size of the filter kernel.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_split\_colour\_composite  
*Split colour composite*

---

**Description**

Splits an RGB colour composite image into separate multispectral images.

**Usage**

```
wbt_split_colour_composite(
    input,
    red = NULL,
    green = NULL,
    blue = NULL,
    wd = NULL,
    verbose_mode = FALSE,
```



```

    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input colour composite image file.
red	Output red band file.
green	Output green band file.
blue	Output blue band file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_split_lidar	<i>Split lidar</i>
-----------------	--------------------

---

### Description

Splits LiDAR points up into a series of new files based on their properties.

### Usage

```

wbt_split_lidar(
  input,
  criterion = "num_pts",
  interval = "",
  min_pts = 5,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

input	Name of the input LiDAR points.
criterion	Criterion on which to base the split of the input file. Options include 'num_pts', 'x', 'y', 'z', 'intensity', 'class', 'user_data', 'point_source_id', 'scan_angle', 'time'.
interval	Interval.
min_pts	Minimum number of points in an output file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_split\_vector\_lines  
*Split vector lines*

---

**Description**

Used to split a vector line coverage into even-lengthed segments.

**Usage**

```
wbt_split_vector_lines(  
  input,  
  output,  
  length = NULL,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Name of the input lines shapefile.
output	Name of the output lines shapefile.
length	Maximum segment length (m).
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_split\_with\_lines *Split with lines*

---

**Description**

Splits the lines or polygons in one layer using the lines in another layer.

**Usage**

```
wbt_split_with_lines(
  input,
  split,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector line or polygon file.
split	Input vector polyline file.
output	Output vector file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_square	<i>Square</i>
------------	---------------

---

### Description

Squares the values in a raster.

### Usage

```
wbt_square(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_square_root	<i>Square root</i>
-----------------	--------------------

---

### Description

Returns the square root of the values in a raster.

**Usage**

```
wbt_square_root(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_standard\_deviation\_contrast\_stretch  
*Standard deviation contrast stretch*

---

**Description**

Performs a standard-deviation contrast stretch on input images.

**Usage**

```
wbt_standard_deviation_contrast_stretch(  
    input,  
    output,  
    stdev = 2,  
    num_tones = 256,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
stdev	Standard deviation clip value.
num_tones	Number of tones in the output image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_standard\_deviation\_filter  
*Standard deviation filter*

---

**Description**

Assigns each cell in the output grid the standard deviation of values in a moving window centred on each grid cell in the input raster.

**Usage**

```
wbt_standard_deviation_filter(  
  input,  
  output,  
  filterx = 11,  
  filtery = 11,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_standard\_deviation\_of\_slope  
*Standard deviation of slope*

---

**Description**

Calculates the standard deviation of slope from an input DEM.

**Usage**

```
wbt_standard_deviation_of_slope(  
  input,  
  output,  
  zfactor = NULL,  
  filterx = 11,  
  filtery = 11,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input raster DEM file.
output	Output raster DEM file.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_stochastic\_depression\_analysis  
*Stochastic depression analysis*

---

**Description**

Performs a stochastic analysis of depressions within a DEM.

**Usage**

```
wbt_stochastic_depression_analysis(  
  dem,  
  output,  
  rmse,  
  range,  
  iterations = 100,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```



**Arguments**

dem	Input raster DEM file.
output	Output file.
rmse	The DEM's root-mean-square-error (RMSE), in z units. This determines error magnitude.
range	The error field's correlation length, in xy-units.
iterations	The number of iterations.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_strahler\_order\_basins

*Strahler order basins*

---

**Description**

Identifies Strahler-order basins from an input stream network.

**Usage**

```
wbt_strahler_order_basins(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_strahler\_stream\_order  
*Strahler stream order*

---

**Description**

Assigns the Strahler stream order to each link in a stream network.

**Usage**

```
wbt_strahler_stream_order(
    d8_pntr,
    streams,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.

esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_stream\_link\_class *Stream link class*

---

**Description**

Identifies the exterior/interior links and nodes in a stream network.

**Usage**

```
wbt_stream_link_class(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.

verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_stream\_link\_identifier  
*Stream link identifier*

---

**Description**

Assigns a unique identifier to each link in a stream network.

**Usage**

```
wbt_stream_link_identifier(
    d8_pntr,
    streams,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_stream\_link\_length  
*Stream link length*

---

**Description**

Estimates the length of each link (or tributary) in a stream network.

**Usage**

```
wbt_stream_link_length(
    d8_pntr,
    linkid,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
linkid	Input raster streams link ID (or tributary ID) file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_stream\_link\_slope *Stream link slope*

---

**Description**

Estimates the average slope of each link (or tributary) in a stream network.

**Usage**

```
wbt_stream_link_slope(
    d8_pntr,
    linkid,
    dem,
    output,
    esri_pntr = FALSE,
    zero_background = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
linkid	Input raster streams link ID (or tributary ID) file.
dem	Input raster DEM file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_stream_power_index`*Stream power index*

---

## Description

Calculates the relative stream power index.

## Usage

```
wbt_stream_power_index(  
    sca,  
    slope,  
    output,  
    exponent = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

<code>sca</code>	Input raster specific contributing area (SCA) file.
<code>slope</code>	Input raster slope file.
<code>output</code>	Output raster file.
<code>exponent</code>	SCA exponent value.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

## Value

Returns the tool text outputs.

---

wbt\_stream\_slope\_continuous  
*Stream slope continuous*

---

### Description

Estimates the slope of each grid cell in a stream network.

### Usage

```
wbt_stream_slope_continuous(  
    d8_pntr,  
    streams,  
    dem,  
    output,  
    esri_pntr = FALSE,  
    zero_background = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
dem	Input raster DEM file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt_subbasins	<i>Subbasins</i>
---------------	------------------

---

### Description

Identifies the catchments, or sub-basin, draining to each link in a stream network.

### Usage

```
wbt_subbasins(  
    d8_pntr,  
    streams,  
    output,  
    esri_pntr = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input D8 pointer raster file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_subtract	<i>Subtract</i>
--------------	-----------------

---

### Description

Performs a differencing operation on two rasters or a raster and a constant value.

### Usage

```
wbt_subtract(
  input1,
  input2,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input1	Input raster file or constant value.
input2	Input raster file or constant value.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_sum_overlay	<i>Sum overlay</i>
-----------------	--------------------

---

**Description**

Calculates the sum for each grid cell from a group of raster images.

**Usage**

```
wbt_sum_overlay(
    inputs,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

inputs	Input raster files.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_surface_area_ratio	<i>Surface area ratio</i>
------------------------	---------------------------

---

**Description**

Calculates a the surface area ratio of each grid cell in an input DEM.

**Usage**

```
wbt_surface_area_ratio(
    dem,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_svm\_classification  
*Svm classification*

---

**Description**

Performs an SVM binary classification using training site polygons/points and multiple input images.

**Usage**

```
wbt_svm_classification(
    inputs,
    training,
    field,
    scaling = "Normalize",
    output = NULL,
    c = 200,
    gamma = 50,
```

```

    tolerance = 0.1,
    test_proportion = 0.2,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

inputs	Names of the input predictor rasters.
training	Name of the input training site polygons/points Shapefile.
field	Name of the attribute containing class data.
scaling	Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
output	Name of the output raster file.
c	c-value, the regularization parameter.
gamma	Gamma parameter used in setting the RBF (Gaussian) kernel function.
tolerance	The tolerance parameter used in determining the stopping condition.
test_proportion	The proportion of the dataset to include in the test split; default is 0.2.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_svm\_regression      *Svm regression*

---

### Description

Performs a supervised SVM regression analysis using training site points and predictor rasters.

**Usage**

```
wbt_svm_regression(
  inputs,
  training,
  field,
  scaling = "Normalize",
  output = NULL,
  c = 50,
  eps = 10,
  gamma = 0.5,
  test_proportion = 0.2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

<code>inputs</code>	Names of the input predictor rasters.
<code>training</code>	Name of the input training site points Shapefile.
<code>field</code>	Name of the attribute containing class data.
<code>scaling</code>	Scaling method for predictors. Options include 'None', 'Normalize', and 'Standardize'.
<code>output</code>	Name of the output raster file.
<code>c</code>	c-value, the regularization parameter.
<code>eps</code>	Epsilon in the epsilon-SVR model.
<code>gamma</code>	Gamma parameter used in setting the RBF (Gaussian) kernel function.
<code>test_proportion</code>	The proportion of the dataset to include in the test split; default is 0.2.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_symmetrical_difference`*Symmetrical difference*

---

**Description**

Outputs the features that occur in one of the two vector inputs but not both, i.e. no overlapping features.

**Usage**

```
wbt_symmetrical_difference(  
    input,  
    overlay,  
    output,  
    snap = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input</code>	Input vector file.
<code>overlay</code>	Input overlay vector file.
<code>output</code>	Output vector file.
<code>snap</code>	Snap tolerance.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_tan	<i>Tan</i>
---------	------------

---

**Description**

Returns the tangent (tan) of each values in a raster.

**Usage**

```
wbt_tan(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_tangential_curvature	<i>Tangential curvature</i>
--------------------------	-----------------------------

---

**Description**

Calculates a tangential curvature raster from an input DEM.



**Usage**

```
wbt_tangential_curvature(
    dem,
    output,
    log = FALSE,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_tanh

*Tanh*


---

**Description**

Returns the hyperbolic tangent (tanh) of each values in a raster.

**Usage**

```
wbt_tanh(
    input,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_thicken\_raster\_line

*Thicken raster line*

---

**Description**

Thickens single-cell wide lines within a raster image.

**Usage**

```
wbt_thicken_raster_line(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_time\_in\_daylight *Time in daylight*

---

**Description**

Calculates the proportion of time a location is not within an area of shadow.

**Usage**

```
wbt_time_in_daylight(
  dem,
  output,
  lat,
  long,
  az_fraction = 10,
  max_dist = 100,
  utc_offset = "00:00",
  start_day = 1,
  end_day = 365,
  start_time = "00:00:00",
  end_time = "23:59:59",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
lat	Centre point latitude.
long	Centre point longitude.
az_fraction	Azimuth fraction in degrees.
max_dist	Optional maximum search distance. Minimum value is 5 x cell size.
utc_offset	UTC time offset, in hours (e.g. -04:00, +06:00).
start_day	Start day of the year (1-365).
end_day	End day of the year (1-365).
start_time	Starting hour to track shadows (e.g. 5, 5:00, 05:00:00). Assumes 24-hour time: HH:MM:SS. 'sunrise' is also a valid time.

end_time	Ending hour to track shadows (e.g. 21, 21:00, 21:00:00). Assumes 24-hour time: HH:MM:SS. 'sunset' is also a valid time.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_tin_gridding	<i>Tin gridding</i>
------------------	---------------------

---

**Description**

Creates a raster grid based on a triangular irregular network (TIN) fitted to vector points.

**Usage**

```
wbt_tin_gridding(
  input,
  output,
  field = NULL,
  use_z = FALSE,
  resolution = NULL,
  base = NULL,
  max_triangle_edge_length = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input vector points file.
output	Output raster file.
field	Input field name in attribute table.
use_z	Use the 'z' dimension of the Shapefile's geometry instead of an attribute field?.
resolution	Output raster's grid resolution.
base	Optionally specified input base raster file. Not used when a cell size is specified.

max_triangle_edge_length	Optional maximum triangle edge length; triangles larger than this size will not be gridded.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_toolbox	<i>The toolbox for a specific tool in WhiteboxTools</i>
-------------	---

---

**Description**

Retrieve the toolbox for a specific tool.

**Usage**

```
wbt_toolbox(tool_name = NULL)
```

**Arguments**

tool_name	The name of the tool.
-----------	-----------------------

**Details**

Leaving tool\_name as default NULL returns results for all tools, but does not work on Windows.

**Value**

Returns the toolbox for a specific tool.

**Examples**

```
## Not run:
wbt_toolbox("breach_depressions")

## End(Not run)
```

---

wbt_tool_help	<i>Help description for a specific tool in 'WhiteboxTools'</i>
---------------	--

---

**Description**

Retrieves the help description for a specific tool.

**Usage**

```
wbt_tool_help(tool_name = NULL)
```

**Arguments**

tool_name	The name of the tool.
-----------	-----------------------

**Details**

Leaving tool\_name as default NULL returns results for all tools, but does not work on Windows.

**Value**

Returns the help description for a specific tool.

**Examples**

```
## Not run:  
wbt_tool_help("lidar_info")  
  
## End(Not run)
```

---

wbt_tool_parameters	<i>Tool parameter descriptions for a specific tool in 'WhiteboxTools'</i>
---------------------	---

---

**Description**

Retrieves the tool parameter descriptions for a specific tool.

**Usage**

```
wbt_tool_parameters(tool_name, quiet = FALSE)
```

**Arguments**

tool_name	The name of the tool.
quiet	Prevent tool output being printed. Default: FALSE

**Details**

quiet argument can be set to TRUE to allow for "quiet" internal use within other functions.

**Value**

Returns the tool parameter descriptions for a specific tool.

**Examples**

```
## Not run:
wbt_tool_parameters("lidar_info")

## End(Not run)
```

---

wbt\_tophat\_transform *Tophat transform*

---

**Description**

Performs either a white or black top-hat transform on an input image.

**Usage**

```
wbt_tophat_transform(
  input,
  output,
  filterx = 11,
  filtery = 11,
  variant = "white",
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
variant	Optional variant value. Options include 'white' and 'black'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.

compress\_rasters      Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.

command\_only      Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_topographic\_position\_animation  
*Topographic position animation*

---

**Description**

This tool creates an animated GIF of multi-scale local topographic position (elevation deviation).

**Usage**

```
wbt_topographic_position_animation(
    input,
    output,
    palette = "bl_yl_rd",
    min_scale = 1,
    num_steps = 100,
    step_nonlinearity = 1.5,
    height = 600,
    delay = 250,
    label = "",
    dev_max = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input              Name of the input digital elevation model (DEM) raster file.

output             Name of the output HTML file (\*.html).

palette            Image palette; options are 'bl\_yl\_rd', 'bl\_w\_rd', 'purple', 'gn\_yl', 'pi\_y\_g', and 'viridis'.

min\_scale         Minimum search neighbourhood radius in grid cells.

num\_steps         Number of steps.

step\_nonlinearity      Step nonlinearity factor (1.0-2.0 is typical).



height	Image height, in pixels.
delay	GIF time delay in milliseconds.
label	Label text (leave blank for none).
dev_max	Do you want to use DEVmax instead of DEV for measuring local topographic position?.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_topological\_stream\_order  
*Topological stream order*

---

**Description**

Assigns each link in a stream network its topological order.

**Usage**

```
wbt_topological_stream_order(
  d8_pntr,
  streams,
  output,
  esri_pntr = FALSE,
  zero_background = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.

esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_topo_render	<i>Topo render</i>
-----------------	--------------------

---

**Description**

This tool creates a pseudo-3D rendering from an input DEM, for the purpose of effective topographic visualization.

**Usage**

```
wbt_topo_render(
  dem,
  output,
  palette = "soft",
  rev_palette = FALSE,
  az = 315,
  alt = 30,
  background_hgt_offset = 10,
  polygon = NULL,
  background_clr = "[255, 255, 255]",
  attenuation = 0.6,
  ambient_light = 0.2,
  z_factor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Name of the input digital elevation model (DEM) raster file.
output	Name of the output raster file.
palette	Palette name; options are 'atlas', 'high_relief', 'arid', 'soft', 'earthtones', 'muted', 'light_quant', 'purple', 'viridi', 'gn_yl', 'pi_y_g', 'bl_yl_rd', 'deep', 'imhof', and 'white'.
rev_palette	Reverse the palette?.
az	Light source azimuth direction (degrees, 0-360).
alt	Light source altitude (degrees, 0-90).
background_hgt_offset	Offset height of background, in z-units.
polygon	Clipping polygon vector file (optional).
background_clr	Background red-green-blue (RGB) or red-green-blue-alpha (RGBA) colour, e.g. '[255, 255, 245]', '[255, 255, 245, 200]'.
attenuation	Attenuation parameter. Range is 0-4. Zero means no attenuation.
ambient_light	Ambient light parameter. Range is 0.0-0.7. Zero means no ambient light.
z_factor	Elevation multiplier, or a vertical exaggeration.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_total\_curvature    *Total curvature*

---

**Description**

Calculates a total curvature raster from an input DEM.

**Usage**

```
wbt_total_curvature(
    dem,
    output,
    log = FALSE,
    zfactor = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
log	Display output values using a log-scale.
zfactor	Optional multiplier for when the vertical and horizontal units are not the same.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_total_filter	<i>Total filter</i>
------------------	---------------------

---

**Description**

Performs a total filter on an input image.

**Usage**

```
wbt_total_filter(
    input,
    output,
    filterx = 11,
    filtery = 11,
    wd = NULL,
```

```

    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input raster file.
output	Output raster file.
filterx	Size of the filter kernel in the x-direction.
filtery	Size of the filter kernel in the y-direction.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_to_degrees	<i>To degrees</i>
----------------	-------------------

---

### Description

Converts a raster from radians to degrees.

### Usage

```

wbt_to_degrees(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)

```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_to_radians	<i>To radians</i>
----------------	-------------------

---

**Description**

Converts a raster from degrees to radians.

**Usage**

```
wbt_to_radians(
  input,
  output,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_trace\_downslope\_flowpaths  
*Trace downslope flowpaths*

---

**Description**

Traces downslope flowpaths from one or more target sites (i.e. seed points).

**Usage**

```
wbt_trace_downslope_flowpaths(  
    seed_pts,  
    d8_pntr,  
    output,  
    esri_pntr = FALSE,  
    zero_background = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

seed_pts	Input vector seed points file.
d8_pntr	Input D8 pointer raster file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_travelling\_salesman\_problem  
*Travelling salesman problem*

---

### Description

Finds approximate solutions to travelling salesman problems, the goal of which is to identify the shortest route connecting a set of locations.

### Usage

```
wbt_travelling_salesman_problem(  
    input,  
    output,  
    duration = 60,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Name of the input points shapefile.
output	Name of the output lines shapefile.
duration	Maximum duration, in seconds.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt_trend_surface	<i>Trend surface</i>
-------------------	----------------------

---

## Description

Estimates the trend surface of an input raster file.

## Usage

```
wbt_trend_surface(  
    input,  
    output,  
    order = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input	Input raster file.
output	Output raster file.
order	Polynomial order (1 to 10).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.

---

wbt\_trend\_surface\_vector\_points  
*Trend surface vector points*

---

### Description

Estimates a trend surface from vector points.

### Usage

```
wbt_trend_surface_vector_points(
    input,
    field,
    output,
    cell_size,
    order = 1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Input vector Points file.
field	Input field name in attribute table.
output	Output raster file.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
order	Polynomial order (1 to 10).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_tributary\_identifier  
*Tributary identifier*

---

### Description

Assigns a unique identifier to each tributary in a stream network.

### Usage

```
wbt_tributary_identifier(  
    d8_pntr,  
    streams,  
    output,  
    esri_pntr = FALSE,  
    zero_background = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input raster D8 pointer file.
streams	Input raster streams file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
zero_background	Flag indicating whether a background value of zero should be used.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_truncate	<i>Truncate</i>
--------------	-----------------

---

### Description

Truncates the values in a raster to the desired number of decimal places.

### Usage

```
wbt_truncate(  
    input,  
    output,  
    num_decimals = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
num_decimals	Number of decimals left after truncation (default is zero).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_turning_bands_simulation`*Turning bands simulation*

---

**Description**

Creates an image containing random values based on a turning-bands simulation.

**Usage**

```
wbt_turning_bands_simulation(  
    base,  
    output,  
    range,  
    iterations = 1000,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>base</code>	Input base raster file.
<code>output</code>	Output file.
<code>range</code>	The field's range, in xy-units, related to the extent of spatial autocorrelation.
<code>iterations</code>	The number of iterations.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

`wbt_two_sample_ks_test`*Two sample ks test*

---

### Description

Performs a 2-sample K-S test for significant differences on two input rasters.

### Usage

```
wbt_two_sample_ks_test(  
    input1,  
    input2,  
    output,  
    num_samples = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

<code>input1</code>	First input raster file.
<code>input2</code>	Second input raster file.
<code>output</code>	Output HTML file.
<code>num_samples</code>	Number of samples. Leave blank to use whole image.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_union
*Union***Description**

Splits vector layers at their overlaps, creating a layer containing all the portions from both input and overlay layers.

**Usage**

```
wbt_union(
    input,
    overlay,
    output,
    snap = 0,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector file.
overlay	Input overlay vector file.
output	Output vector file.
snap	Snap tolerance.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_unnest\_basins      *Unnest basins*

---

### Description

Extract whole watersheds for a set of outlet points.

### Usage

```
wbt_unnest_basins(  
    d8_pntr,  
    pour_pts,  
    output,  
    esri_pntr = FALSE,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

d8_pntr	Input D8 pointer raster file.
pour_pts	Input vector pour points (outlet) file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_unsharp\_masking     *Unsharp masking*

---

### Description

An image sharpening technique that enhances edges.

### Usage

```
wbt_unsharp_masking(  
    input,  
    output,  
    sigma = 0.75,  
    amount = 100,  
    threshold = 0,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

input	Input raster file.
output	Output raster file.
sigma	Standard deviation distance in pixels.
amount	A percentage and controls the magnitude of each overshoot.
threshold	Controls the minimal brightness change that will be sharpened.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_unsphericity	<i>Unsphericity</i>
------------------	---------------------

---

### Description

This tool calculates the unsphericity curvature from an input DEM.

### Usage

```
wbt_unsphericity(  
    dem,  
    output,  
    log = FALSE,  
    zfactor = 1,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
log	Display output values using a log-scale.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

`wbt_update_nodata_cells`*Update nodata cells*

---

**Description**

Replaces the NoData values in an input raster with the corresponding values contained in a second update layer.

**Usage**

```
wbt_update_nodata_cells(  
    input1,  
    input2,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input1</code>	Input raster file 1.
<code>input2</code>	Input raster file 2; update layer.
<code>output</code>	Output raster file.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_upslope\_depression\_storage  
*Upslope depression storage*

---

### Description

Estimates the average upslope depression storage depth.

### Usage

```
wbt_upslope_depression_storage(  
    dem,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

dem	Input raster DEM file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_user\_defined\_weights\_filter  
*User ined weights filter*

---

### Description

Performs a user-defined weights filter on an image.

**Usage**

```
wbt_user_defined_weights_filter(
    input,
    weights,
    output,
    center = "center",
    normalize = FALSE,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input raster file.
weights	Input weights file.
output	Output raster file.
center	Kernel center cell; options include 'center', 'upper-left', 'upper-right', 'lower-left', 'lower-right'.
normalize	Normalize kernel weights? This can reduce edge effects and lessen the impact of data gaps (nodata) but is not suited when the kernel weights sum to zero.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_vector\_hex\_binning

*Vector hex binning*

---

**Description**

Hex-bins a set of vector points.

**Usage**

```
wbt_vector_hex_binning(
    input,
    output,
    width,
    orientation = "horizontal",
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input base file.
output	Output vector polygon file.
width	The grid cell width.
orientation	Grid Orientation, 'horizontal' or 'vertical'.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_vector\_lines\_to\_raster  
*Vector lines to raster*

---

**Description**

Converts a vector containing polylines into a raster.

**Usage**

```
wbt_vector_lines_to_raster(
    input,
    output,
    field = "FID",
    noata = TRUE,
```

```

    cell_size = NULL,
    base = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input vector lines file.
output	Output raster file.
field	Input field name in attribute table.
nodata	Background value to set to NoData. Without this flag, it will be set to 0.0.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

```

wbt_vector_points_to_raster
    Vector points to raster

```

---

### Description

Converts a vector containing points into a raster.

### Usage

```

wbt_vector_points_to_raster(
  input,
  output,
  field = "FID",
  assign = "last",
  nodata = TRUE,

```

```

    cell_size = NULL,
    base = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)

```

### Arguments

input	Input vector Points file.
output	Output raster file.
field	Input field name in attribute table.
assign	Assignment operation, where multiple points are in the same grid cell; options include 'first', 'last' (default), 'min', 'max', 'sum', 'number'.
nodata	Background value to set to NoData. Without this flag, it will be set to 0.0.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_vector\_polygons\_to\_raster  
*Vector polygons to raster*

---

### Description

Converts a vector containing polygons into a raster.



**Usage**

```
wbt_vector_polygons_to_raster(
    input,
    output,
    field = "FID",
    nodata = TRUE,
    cell_size = NULL,
    base = NULL,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input	Input vector polygons file.
output	Output raster file.
field	Input field name in attribute table.
nodata	Background value to set to NoData. Without this flag, it will be set to 0.0.
cell_size	Optionally specified cell size of output raster. Not used when base raster is specified.
base	Optionally specified input base raster file. Not used when a cell size is specified.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_vector\_stream\_network\_analysis

*Vector stream network analysis*

---

**Description**

This tool performs common stream network analysis operations on an input vector stream file.

**Usage**

```
wbt_vector_stream_network_analysis(
    streams,
    dem,
    output,
    cutting_height = 10,
    snap = 0.1,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

streams	Name of the input streams vector file.
dem	Name of the input DEM raster file.
output	Name of the output lines shapefile.
cutting_height	Maximum ridge-cutting height (z units).
snap	Snap distance, in xy units (metres).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_version	<i>Version information for 'WhiteboxTools'</i>
-------------	--

---

**Description**

Version information for 'WhiteboxTools'

**Usage**

```
wbt_version(extract = FALSE)
```

**Arguments**

extract	Extract semantic version number from first line of result? Default: FALSE
---------	---

**Value**

Returns the version information for 'WhiteboxTools' as an R character vector.

**Examples**

```
## Not run:
wbt_version()

## End(Not run)
```

---

```
wbt_vertical_excess_curvature
      Vertical excess curvature
```

---

**Description**

This tool calculates vertical excess curvature from an input DEM.

**Usage**

```
wbt_vertical_excess_curvature(
  dem,
  output,
  log = FALSE,
  zfactor = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Name of the input raster DEM file.
output	Name of the output raster image file.
log	Display output values using a log-scale.
zfactor	Z conversion factor.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_viewshed	<i>Viewshed</i>
--------------	-----------------

---

**Description**

Identifies the viewshed for a point or set of points.

**Usage**

```
wbt_viewshed(
  dem,
  stations,
  output,
  height = 2,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

dem	Input raster DEM file.
stations	Input viewing station vector file.
output	Output raster file.
height	Viewing station height, in z units.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_view\_code      *Source code for a specific tool in 'WhiteboxTools'*

---

### Description

Opens a web browser to view the source code for a specific tool on the projects source code repository.

### Usage

```
wbt_view_code(tool_name, viewer = FALSE)
```

### Arguments

tool_name	Name of the tool.
viewer	Show source code in browser? default: TRUE

### Value

Returns a GitHub URL to view the source code of the tool.

### Examples

```
## Not run:  
wbt_view_code("breach_depressions")  
  
## End(Not run)
```

---

wbt\_visibility\_index      *Visibility index*

---

### Description

Estimates the relative visibility of sites in a DEM.

### Usage

```
wbt_visibility_index(  
  dem,  
  output,  
  height = 2,  
  res_factor = 2,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

dem	Input raster DEM file.
output	Output raster file.
height	Viewing station height, in z units.
res_factor	The resolution factor determines the density of measured viewsheds.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_voronoi\_diagram    *Voronoi diagram*

---

**Description**

Creates a vector Voronoi diagram for a set of vector points.

**Usage**

```
wbt_voronoi_diagram(  
  input,  
  output,  
  wd = NULL,  
  verbose_mode = FALSE,  
  compress_rasters = FALSE,  
  command_only = FALSE  
)
```

**Arguments**

input	Input vector points file.
output	Output vector polygon file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_watershed	<i>Watershed</i>
---------------	------------------

---

**Description**

Identifies the watershed, or drainage basin, draining to a set of target cells.

**Usage**

```
wbt_watershed(
  d8_pntr,
  pour_pts,
  output,
  esri_pntr = FALSE,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

d8_pntr	Input D8 pointer raster file.
pour_pts	Input pour points (outlet) file.
output	Output raster file.
esri_pntr	D8 pointer uses the ESRI style scheme.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_weighted\_overlay *Weighted overlay*

---

### Description

Performs a weighted sum on multiple input rasters after converting each image to a common scale. The tool performs a multi-criteria evaluation (MCE).

### Usage

```
wbt_weighted_overlay(
  factors,
  weights,
  output,
  cost = NULL,
  constraints = NULL,
  scale_max = 1,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

factors	Input factor raster files.
weights	Weight values, contained in quotes and separated by commas or semicolons. Must have the same number as factors.
output	Output raster file.
cost	Boolean array indicating which factors are cost factors, contained in quotes and separated by commas or semicolons. Must have the same number as factors.
constraints	Input constraints raster files.
scale_max	Suitability scale maximum value (common values are 1.0, 100.0, and 255.0).
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt_weighted_sum	<i>Weighted sum</i>
------------------	---------------------

---

### Description

Performs a weighted-sum overlay on multiple input raster images.

### Usage

```
wbt_weighted_sum(  
    inputs,  
    weights,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

inputs	Input raster files.
weights	Weight values, contained in quotes and separated by commas or semicolons.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_wetness_index	<i>Wetness index</i>
-------------------	----------------------

---

### Description

Calculates the topographic wetness index,  $\text{Ln}(A / \tan(\text{slope}))$ .

### Usage

```
wbt_wetness_index(  
    sca,  
    slope,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

### Arguments

sca	Input raster specific contributing area (SCA) file.
slope	Input raster slope file (in degrees).
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_wilcoxon\_signed\_rank\_test  
*Wilcoxon signed rank test*

---

## Description

Performs a 2-sample K-S test for significant differences on two input rasters.

## Usage

```
wbt_wilcoxon_signed_rank_test(  
    input1,  
    input2,  
    output,  
    num_samples = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

## Arguments

input1	First input raster file.
input2	Second input raster file.
output	Output HTML file.
num_samples	Number of samples. Leave blank to use whole image.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

## Value

Returns the tool text outputs.

---

`wbt_write_function_memory_insertion`*Write function memory insertion*

---

**Description**

Performs a write function memory insertion for single-band multi-date change detection.

**Usage**

```
wbt_write_function_memory_insertion(  
    input1,  
    input2,  
    output,  
    input3 = NULL,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

<code>input1</code>	Input raster file associated with the first date.
<code>input2</code>	Input raster file associated with the second date.
<code>output</code>	Output raster file.
<code>input3</code>	Optional input raster file associated with the third date.
<code>wd</code>	Changes the working directory.
<code>verbose_mode</code>	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
<code>compress_rasters</code>	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
<code>command_only</code>	Return command that would be executed by <code>system()</code> rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_xor
*Xor***Description**

Performs a logical XOR operator on two Boolean raster images.

**Usage**

```
wbt_xor(
    input1,
    input2,
    output,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

**Arguments**

input1	Input raster file.
input2	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt\_yield\_filter      *Yield filter*

---

### Description

Filters crop yield values of point data derived from combine harvester yield monitors.

### Usage

```
wbt_yield_filter(
    input,
    yield_field,
    pass_field,
    output,
    width = 6.096,
    z_score_threshold = 2.5,
    min_yield = 0,
    max_yield = 99999.9,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input points shapefile.
yield_field	Name of the attribute containing yield data.
pass_field	Name of the attribute containing pass line ID.
output	Name of the output points shapefile.
width	Pass swath width (m).
z_score_threshold	Z-score threshold value (default=2.5).
min_yield	Minimum yield value in output.
max_yield	Maximum yield value in output.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt_yield_map	<i>Yield map</i>
---------------	------------------

---

### Description

This tool can be used to create a segmented-vector polygon yield map from a set of harvester points.

### Usage

```
wbt_yield_map(
  input,
  pass_field_name,
  output,
  width = 6.096,
  max_change_in_heading = 25,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

input	Name of the input points shapefile.
pass_field_name	Name of the attribute containing pass line ID.
output	Name of the output polygon shapefile.
width	Pass swath width (m).
max_change_in_heading	Max change in heading.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_yield\_normalization

*Yield normalization*


---

### Description

This tool can be used to normalize the yield points for a field.

### Usage

```
wbt_yield_normalization(
    input,
    yield_field,
    output,
    standardize = FALSE,
    radius = NULL,
    min_yield = 0,
    max_yield = 99999.9,
    wd = NULL,
    verbose_mode = FALSE,
    compress_rasters = FALSE,
    command_only = FALSE
)
```

### Arguments

input	Name of the input points shapefile.
yield_field	Name of the attribute containing yield data.
output	Name of the output points shapefile.
standardize	Should the yield values be standardized (converted to z-scores) rather than normalized?.
radius	Optional search radius, in metres. Only specify this value if you want to calculate locally normalized yield.
min_yield	Minimum yield value in output.
max_yield	Maximum yield value in output.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.



---

wbt\_zlidar\_to\_las      *Zlidar to las*

---

### Description

Converts one or more zlidar files into the LAS data format.

### Usage

```
wbt_zlidar_to_las(
  inputs = NULL,
  outdir = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

### Arguments

inputs	Input ZLidar files.
outdir	Output directory into which zlidar files are created. If unspecified, it is assumed to be the same as the inputs.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

### Value

Returns the tool text outputs.

---

wbt\_zonal\_statistics      *Zonal statistics*

---

### Description

Extracts descriptive statistics for a group of patches in a raster.

**Usage**

```
wbt_zonal_statistics(
  input,
  features,
  output = NULL,
  stat = "mean",
  out_table = NULL,
  wd = NULL,
  verbose_mode = FALSE,
  compress_rasters = FALSE,
  command_only = FALSE
)
```

**Arguments**

input	Input data raster file.
features	Input feature definition raster file.
output	Output raster file.
stat	Statistic to extract, including 'mean', 'median', 'minimum', 'maximum', 'range', 'standard deviation', and 'total'.
out_table	Output HTML Table file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

---

wbt_z_scores	<i>Z scores</i>
--------------	-----------------

---

**Description**

Standardizes the values in an input raster by converting to z-scores.

**Usage**

```
wbt_z_scores(  
    input,  
    output,  
    wd = NULL,  
    verbose_mode = FALSE,  
    compress_rasters = FALSE,  
    command_only = FALSE  
)
```

**Arguments**

input	Input raster file.
output	Output raster file.
wd	Changes the working directory.
verbose_mode	Sets verbose mode. If verbose mode is FALSE, tools will not print output messages.
compress_rasters	Sets the flag used by 'WhiteboxTools' to determine whether to use compression for output rasters.
command_only	Return command that would be executed by system() rather than running tool.

**Value**

Returns the tool text outputs.

# Index

## \* **DataTools**

- wbt\_add\_point\_coordinates\_to\_table, 23
- wbt\_clean\_vector, 61
- wbt\_convert\_nodata\_to\_zero, 75
- wbt\_convert\_raster\_format, 75
- wbt\_csv\_points\_to\_vector, 88
- wbt\_export\_table\_to\_csv, 136
- wbt\_fix\_dangling\_arcs, 163
- wbt\_join\_tables, 225
- wbt\_lines\_to\_polygons, 280
- wbt\_merge\_table\_with\_csv, 317
- wbt\_merge\_vectors, 318
- wbt\_modify\_no\_data\_value, 331
- wbt\_multi\_part\_to\_single\_part, 344
- wbt\_new\_raster\_from\_base, 348
- wbt\_polygons\_to\_lines, 375
- wbt\_print\_geo\_tiff\_tags, 382
- wbt\_raster\_to\_vector\_lines, 402
- wbt\_raster\_to\_vector\_points, 403
- wbt\_raster\_to\_vector\_polygons, 404
- wbt\_reinitialize\_attribute\_table, 412
- wbt\_remove\_polygon\_holes, 417
- wbt\_remove\_raster\_polygon\_holes, 418
- wbt\_set\_nodata\_value, 437
- wbt\_single\_part\_to\_multi\_part, 445
- wbt\_vector\_lines\_to\_raster, 510
- wbt\_vector\_points\_to\_raster, 511
- wbt\_vector\_polygons\_to\_raster, 512

## \* **GISAnalysisDistanceTools**

- wbt\_buffer\_raster, 50
- wbt\_cost\_allocation, 79
- wbt\_cost\_distance, 80
- wbt\_cost\_pathway, 81
- wbt\_euclidean\_allocation, 132
- wbt\_euclidean\_distance, 132

## \* **GISAnalysisOverlayTools**

- wbt\_average\_overlay, 40
- wbt\_clip, 61
- wbt\_clip\_raster\_to\_polygon, 63
- wbt\_count\_if, 82
- wbt\_difference, 100
- wbt\_erase, 129
- wbt\_erase\_polygon\_from\_raster, 131
- wbt\_highest\_position, 183
- wbt\_intersect, 217
- wbt\_line\_intersections, 282
- wbt\_lowest\_position, 294
- wbt\_max\_absolute\_overlay, 301
- wbt\_max\_overlay, 308
- wbt\_merge\_line\_segments, 316
- wbt\_min\_absolute\_overlay, 325
- wbt\_min\_overlay, 328
- wbt\_multiply\_overlay, 336
- wbt\_percent\_equal\_to, 367
- wbt\_percent\_greater\_than, 368
- wbt\_percent\_less\_than, 369
- wbt\_pick\_from\_list, 372
- wbt\_polygonize, 375
- wbt\_split\_with\_lines, 459
- wbt\_sum\_overlay, 475
- wbt\_symmetrical\_difference, 479
- wbt\_union, 503
- wbt\_update\_nodata\_cells, 507
- wbt\_weighted\_overlay, 520
- wbt\_weighted\_sum, 521

## \* **GISAnalysisPatchShapeTools**

- wbt\_boundary\_shape\_complexity, 46
- wbt\_compactness\_ratio, 68
- wbt\_edge\_proportion, 118
- wbt\_elongation\_ratio, 125
- wbt\_find\_patch\_or\_class\_edge\_cells, 161
- wbt\_hole\_proportion, 192
- wbt\_linearity\_index, 280
- wbt\_narrowness\_index, 345

- wbt\_patch\_orientation, 363
- wbt\_perimeter\_area\_ratio, 370
- wbt\_radius\_of\_gyration, 388
- wbt\_related\_circumscribing\_circle, 412
- wbt\_shape\_complexity\_index, 440
- wbt\_shape\_complexity\_index\_raster, 441
- \* **GISAnalysis**
  - wbt\_aggregate\_raster, 24
  - wbt\_block\_maximum\_gridding, 44
  - wbt\_block\_minimum\_gridding, 45
  - wbt\_centroid, 54
  - wbt\_centroid\_vector, 55
  - wbt\_clump, 65
  - wbt\_construct\_vector\_tin, 72
  - wbt\_create\_hexagonal\_vector\_grid, 84
  - wbt\_create\_plane, 85
  - wbt\_create\_rectangular\_vector\_grid, 86
  - wbt\_dissolve, 106
  - wbt\_eliminate\_coincident\_points, 124
  - wbt\_extend\_vector\_lines, 138
  - wbt\_extract\_nodes, 139
  - wbt\_extract\_raster\_values\_at\_points, 139
  - wbt\_filter\_raster\_features\_by\_area, 156
  - wbt\_find\_lowest\_or\_highest\_points, 158
  - wbt\_heat\_map, 181
  - wbt\_idw\_interpolation, 198
  - wbt\_layer\_footprint, 238
  - wbt\_medoid, 315
  - wbt\_minimum\_bounding\_box, 320
  - wbt\_minimum\_bounding\_circle, 321
  - wbt\_minimum\_bounding\_envelope, 322
  - wbt\_minimum\_convex\_hull, 323
  - wbt\_natural\_neighbour\_interpolation, 346
  - wbt\_nearest\_neighbour\_gridding, 347
  - wbt\_polygon\_area, 376
  - wbt\_polygon\_long\_axis, 377
  - wbt\_polygon\_perimeter, 378
  - wbt\_polygon\_short\_axis, 378
  - wbt\_radial\_basis\_function\_interpolation, 387
  - wbt\_raster\_area, 396
  - wbt\_raster\_cell\_assignment, 398
  - wbt\_raster\_perimeter, 400
  - wbt\_reclass, 406
  - wbt\_reclass\_equal\_interval, 407
  - wbt\_reclass\_from\_file, 408
  - wbt\_smooth\_vectors, 450
  - wbt\_split\_vector\_lines, 458
  - wbt\_tin\_gridding, 484
  - wbt\_travelling\_salesman\_problem, 496
  - wbt\_vector\_hex\_binning, 509
  - wbt\_voronoi\_diagram, 518
- \* **General**
  - check\_whitebox\_binary, 14
  - sample\_dem\_data, 14
  - wbt, 15
  - wbt\_activate, 21
  - wbt\_help, 182
  - wbt\_init, 210
  - wbt\_install, 214
  - wbt\_license, 243
  - wbt\_list\_tools, 284
  - wbt\_run\_tool, 433
  - wbt\_rust\_backtrace, 434
  - wbt\_source, 455
  - wbt\_tool\_help, 486
  - wbt\_tool\_parameters, 486
  - wbt\_toolbox, 485
  - wbt\_version, 514
  - wbt\_view\_code, 517
  - wbttoolparameters, 17
  - wbttools, 18
- \* **GeomorphometricAnalysis**
  - wbt\_accumulation\_curvature, 20
  - wbt\_aspect, 32
  - wbt\_assess\_route, 33
  - wbt\_average\_normal\_vector\_angular\_deviation, 39
  - wbt\_breakline\_mapping, 49
  - wbt\_circular\_variance\_of\_aspect, 57
  - wbt\_contours\_from\_points, 73
  - wbt\_contours\_from\_raster, 74
  - wbt\_curvedness, 90
  - wbt\_dem\_void\_filling, 95

- wbt\_dev\_from\_mean\_elev, [99](#)
- wbt\_diff\_from\_mean\_elev, [102](#)
- wbt\_difference\_curvature, [101](#)
- wbt\_directional\_relief, [104](#)
- wbt\_downslope\_index, [112](#)
- wbt\_edge\_density, [116](#)
- wbt\_elev\_above\_pit, [121](#)
- wbt\_elev\_percentile, [121](#)
- wbt\_elev\_relative\_to\_min\_max, [122](#)
- wbt\_elev\_relative\_to\_watershed\_min\_max, [123](#)
- wbt\_embankment\_mapping, [126](#)
- wbt\_exposure\_towards\_wind\_flux, [137](#)
- wbt\_feature\_preserving\_smoothing, [146](#)
- wbt\_fetch\_analysis, [147](#)
- wbt\_fill\_missing\_data, [152](#)
- wbt\_find\_ridges, [162](#)
- wbt\_gaussian\_curvature, [172](#)
- wbt\_gaussian\_scale\_space, [174](#)
- wbt\_generating\_function, [177](#)
- wbt\_geomorphons, [178](#)
- wbt\_hillshade, [187](#)
- wbt\_horizon\_angle, [193](#)
- wbt\_horizontal\_excess\_curvature, [192](#)
- wbt\_hypsometric\_analysis, [197](#)
- wbt\_hypsometrically\_tinted\_hillshade, [196](#)
- wbt\_local\_hypsometric\_analysis, [287](#)
- wbt\_local\_quadratic\_regression, [288](#)
- wbt\_map\_off\_terrain\_objects, [297](#)
- wbt\_max\_anisotropy\_dev, [301](#)
- wbt\_max\_anisotropy\_dev\_signature, [302](#)
- wbt\_max\_branch\_length, [303](#)
- wbt\_max\_difference\_from\_mean, [304](#)
- wbt\_max\_downslope\_elev\_change, [305](#)
- wbt\_max\_elev\_dev\_signature, [307](#)
- wbt\_max\_elevation\_deviation, [306](#)
- wbt\_max\_upslope\_elev\_change, [309](#)
- wbt\_maximal\_curvature, [299](#)
- wbt\_mean\_curvature, [312](#)
- wbt\_min\_downslope\_elev\_change, [326](#)
- wbt\_minimal\_curvature, [319](#)
- wbt\_multidirectional\_hillshade, [334](#)
- wbt\_multiscale\_curvatures, [337](#)
- wbt\_multiscale\_elevation\_percentile, [338](#)
- wbt\_multiscale\_roughness, [339](#)
- wbt\_multiscale\_roughness\_signature, [340](#)
- wbt\_multiscale\_std\_dev\_normals, [341](#)
- wbt\_multiscale\_std\_dev\_normals\_signature, [342](#)
- wbt\_multiscale\_topographic\_position\_image, [343](#)
- wbt\_num\_downslope\_neighbours, [354](#)
- wbt\_num\_upslope\_neighbours, [355](#)
- wbt\_openness, [358](#)
- wbt\_pennock\_landform\_class, [363](#)
- wbt\_percent\_elev\_range, [366](#)
- wbt\_plan\_curvature, [374](#)
- wbt\_profile, [382](#)
- wbt\_profile\_curvature, [383](#)
- wbt\_relative\_aspect, [413](#)
- wbt\_relative\_topographic\_position, [414](#)
- wbt\_remove\_off\_terrain\_objects, [416](#)
- wbt\_ring\_curvature, [427](#)
- wbt\_rotor, [430](#)
- wbt\_ruggedness\_index, [432](#)
- wbt\_sediment\_transport\_index, [435](#)
- wbt\_shadow\_animation, [438](#)
- wbt\_shadow\_image, [439](#)
- wbt\_shape\_index, [441](#)
- wbt\_slope, [447](#)
- wbt\_slope\_vs\_aspect\_plot, [448](#)
- wbt\_slope\_vs\_elevation\_plot, [449](#)
- wbt\_smooth\_vegetation\_residual, [451](#)
- wbt\_spherical\_std\_dev\_of\_normals, [455](#)
- wbt\_standard\_deviation\_of\_slope, [463](#)
- wbt\_stream\_power\_index, [471](#)
- wbt\_surface\_area\_ratio, [475](#)
- wbt\_tangential\_curvature, [480](#)
- wbt\_time\_in\_daylight, [483](#)
- wbt\_topo\_render, [490](#)

- wbt\_topographic\_position\_animation, 488
- wbt\_total\_curvature, 491
- wbt\_unsphericity, 506
- wbt\_vertical\_excess\_curvature, 515
- wbt\_viewshed, 516
- wbt\_visibility\_index, 517
- wbt\_wetness\_index, 522
- \* **HydrologicalAnalysis**
  - wbt\_average\_flowpath\_slope, 38
  - wbt\_average\_upslope\_flowpath\_length, 41
  - wbt\_basins, 42
  - wbt\_breach\_depressions, 47
  - wbt\_breach\_depressions\_least\_cost, 48
  - wbt\_breach\_single\_cell\_pits, 49
  - wbt\_burn\_streams\_at\_roads, 51
  - wbt\_d8\_flow\_accumulation, 91
  - wbt\_d8\_mass\_flux, 92
  - wbt\_d8\_pointer, 93
  - wbt\_d\_inf\_flow\_accumulation, 113
  - wbt\_d\_inf\_mass\_flux, 114
  - wbt\_d\_inf\_pointer, 115
  - wbt\_depth\_in\_sink, 97
  - wbt\_depth\_to\_water, 98
  - wbt\_downslope\_distance\_to\_stream, 110
  - wbt\_downslope\_flowpath\_length, 111
  - wbt\_edge\_contamination, 115
  - wbt\_elevation\_above\_stream, 119
  - wbt\_elevation\_above\_stream\_euclidean, 120
  - wbt\_fd8\_flow\_accumulation, 144
  - wbt\_fd8\_pointer, 145
  - wbt\_fill\_burn, 148
  - wbt\_fill\_depressions, 149
  - wbt\_fill\_depressions\_planchon\_and\_darboux, 150
  - wbt\_fill\_depressions\_wang\_and\_liu, 151
  - wbt\_fill\_single\_cell\_pits, 153
  - wbt\_find\_no\_flow\_cells, 160
  - wbt\_find\_parallel\_flow, 160
  - wbt\_flatten\_lakes, 164
  - wbt\_flood\_order, 167
  - wbt\_flow\_accumulation\_full\_workflow, 168
  - wbt\_flow\_length\_diff, 169
  - wbt\_hillslopes, 188
  - wbt\_hydrologic\_connectivity, 195
  - wbt\_impoundment\_size\_index, 207
  - wbt\_insert\_dams, 213
  - wbt\_isobasins, 222
  - wbt\_jenson\_snap\_pour\_points, 224
  - wbt\_longest\_flowpath, 291
  - wbt\_low\_points\_on\_headwater\_divides, 295
  - wbt\_max\_upslope\_flowpath\_length, 309
  - wbt\_max\_upslope\_value, 310
  - wbt\_md\_inf\_flow\_accumulation, 311
  - wbt\_num\_inflowing\_neighbours, 354
  - wbt\_qin\_flow\_accumulation, 384
  - wbt\_quinn\_flow\_accumulation, 386
  - wbt\_raise\_walls, 389
  - wbt\_rho8\_flow\_accumulation, 425
  - wbt\_rho8\_pointer, 426
  - wbt\_river\_centerlines, 428
  - wbt\_sink, 446
  - wbt\_snap\_pour\_points, 452
  - wbt\_stochastic\_depression\_analysis, 464
  - wbt\_strahler\_order\_basins, 465
  - wbt\_subbasins, 473
  - wbt\_trace\_downslope\_flowpaths, 495
  - wbt\_unnest\_basins, 504
  - wbt\_upslope\_depression\_storage, 508
  - wbt\_watershed, 519
- \* **ImageProcessingToolsClassification**
  - wbt\_evaluate\_training\_sites, 133
  - wbt\_generalize\_classified\_raster, 175
  - wbt\_generalize\_with\_similarity, 176
  - wbt\_image\_segmentation, 204
  - wbt\_min\_dist\_classification, 325
  - wbt\_parallelepiped\_classification, 362
- \* **ImageProcessingToolsFilters**
  - wbt\_adaptive\_filter, 21
  - wbt\_bilateral\_filter, 43
  - wbt\_canny\_edge\_detection, 52
  - wbt\_conservative\_smoothing\_filter, 71

- wbt\_corner\_detection, 76
- wbt\_diff\_of\_gaussian\_filter, 103
- wbt\_diversity\_filter, 108
- wbt\_edge\_preserving\_mean\_filter, 117
- wbt\_emboss\_filter, 127
- wbt\_fast\_almost\_gaussian\_filter, 143
- wbt\_gaussian\_filter, 173
- wbt\_high\_pass\_bilateral\_filter, 184
- wbt\_high\_pass\_filter, 185
- wbt\_high\_pass\_median\_filter, 186
- wbt\_k\_nearest\_mean\_filter, 231
- wbt\_laplacian\_filter, 232
- wbt\_laplacian\_of\_gaussian\_filter, 233
- wbt\_lee\_sigma\_filter, 240
- wbt\_line\_detection\_filter, 281
- wbt\_majority\_filter, 296
- wbt\_maximum\_filter, 300
- wbt\_mean\_filter, 313
- wbt\_median\_filter, 314
- wbt\_minimum\_filter, 324
- wbt\_olympic\_filter, 356
- wbt\_percentile\_filter, 365
- wbt\_prewitt\_filter, 380
- wbt\_range\_filter, 394
- wbt\_roberts\_cross\_filter, 429
- wbt\_scharr\_filter, 434
- wbt\_sobel\_filter, 453
- wbt\_standard\_deviation\_filter, 462
- wbt\_total\_filter, 492
- wbt\_unsharp\_masking, 505
- wbt\_user\_defined\_weights\_filter, 508
- \* **ImageProcessingToolsImageEnhancement**
  - wbt\_balance\_contrast\_enhancement, 41
  - wbt\_correct\_vignetting, 77
  - wbt\_direct\_decorrelation\_stretch, 105
  - wbt\_gamma\_correction, 170
  - wbt\_gaussian\_contrast\_stretch, 171
  - wbt\_histogram\_equalization, 189
  - wbt\_histogram\_matching, 190
  - wbt\_histogram\_matching\_two\_images, 191
  - wbt\_min\_max\_contrast\_stretch, 327
  - wbt\_panchromatic\_sharpening, 361
  - wbt\_percentage\_contrast\_stretch, 364
  - wbt\_piecewise\_contrast\_stretch, 373
  - wbt\_sigmoidal\_contrast\_stretch, 443
  - wbt\_standard\_deviation\_contrast\_stretch, 461
- \* **ImageProcessingTools**
  - wbt\_change\_vector\_analysis, 56
  - wbt\_closing, 64
  - wbt\_create\_colour\_composite, 83
  - wbt\_flip\_image, 166
  - wbt\_ihs\_to\_rgb, 199
  - wbt\_image\_slider, 205
  - wbt\_image\_stack\_profile, 206
  - wbt\_integral\_image, 217
  - wbt\_line\_thinning, 283
  - wbt\_mosaic, 332
  - wbt\_mosaic\_with\_feathering, 333
  - wbt\_normalized\_difference\_index, 349
  - wbt\_opening, 357
  - wbt\_remove\_spurs, 420
  - wbt\_resample, 422
  - wbt\_rgb\_to\_ihs, 424
  - wbt\_split\_colour\_composite, 456
  - wbt\_thicken\_raster\_line, 482
  - wbt\_tophat\_transform, 487
  - wbt\_write\_function\_memory\_insertion, 524
- \* **LiDARTools**
  - wbt\_ascii\_to\_las, 31
  - wbt\_classify\_buildings\_in\_lidar, 58
  - wbt\_classify\_lidar, 59
  - wbt\_classify\_overlap\_points, 60
  - wbt\_clip\_lidar\_to\_polygon, 62
  - wbt\_colourize\_based\_on\_class, 66
  - wbt\_colourize\_based\_on\_point\_returns, 67
  - wbt\_erase\_polygon\_from\_lidar, 130
  - wbt\_filter\_lidar, 153
  - wbt\_filter\_lidar\_classes, 154
  - wbt\_filter\_lidar\_scan\_angles, 155
  - wbt\_find\_flightline\_edge\_points,



- 157
- wbt\_flightline\_overlap, 165
- wbt\_height\_above\_ground, 182
- wbt\_individual\_tree\_detection, 208
- wbt\_las\_to\_ascii, 234
- wbt\_las\_to\_laz, 234
- wbt\_las\_to\_multipoint\_shapefile, 235
- wbt\_las\_to\_shapefile, 236
- wbt\_las\_to\_zlidar, 237
- wbt\_laz\_to\_las, 239
- wbt\_lidar\_block\_maximum, 243
- wbt\_lidar\_block\_minimum, 244
- wbt\_lidar\_classify\_subset, 245
- wbt\_lidar\_colourize, 246
- wbt\_lidar\_contour, 247
- wbt\_lidar\_digital\_surface\_model, 248
- wbt\_lidar\_eigenvalue\_features, 249
- wbt\_lidar\_elevation\_slice, 250
- wbt\_lidar\_ground\_point\_filter, 251
- wbt\_lidar\_hex\_binning, 252
- wbt\_lidar\_hillshade, 253
- wbt\_lidar\_histogram, 254
- wbt\_lidar\_idw\_interpolation, 255
- wbt\_lidar\_info, 256
- wbt\_lidar\_join, 257
- wbt\_lidar\_kappa\_index, 258
- wbt\_lidar\_nearest\_neighbour\_gridding, 259
- wbt\_lidar\_point\_density, 260
- wbt\_lidar\_point\_return\_analysis, 261
- wbt\_lidar\_point\_stats, 262
- wbt\_lidar\_ransac\_planes, 263
- wbt\_lidar\_rbf\_interpolation, 264
- wbt\_lidar\_remove\_duplicates, 265
- wbt\_lidar\_remove\_outliers, 266
- wbt\_lidar\_rooftop\_analysis, 267
- wbt\_lidar\_segmentation, 269
- wbt\_lidar\_segmentation\_based\_filter, 270
- wbt\_lidar\_shift, 271
- wbt\_lidar\_sibson\_interpolation, 272
- wbt\_lidar\_sort\_by\_time, 273
- wbt\_lidar\_thin, 274
- wbt\_lidar\_thin\_high\_density, 275
- wbt\_lidar\_tile, 276
- wbt\_lidar\_tile\_footprint, 277
- wbt\_lidar\_tin\_gridding, 278
- wbt\_lidar\_tophat\_transform, 279
- wbt\_modify\_lidar, 330
- wbt\_normal\_vectors, 351
- wbt\_normalize\_lidar, 350
- wbt\_recover\_flightline\_info, 410
- wbt\_select\_tiles\_by\_polygon, 436
- wbt\_sort\_lidar, 454
- wbt\_split\_lidar, 457
- wbt\_zlidar\_to\_las, 529
- \* MachineLearning**
  - wbt\_dbscan, 94
  - wbt\_k\_means\_clustering, 230
  - wbt\_knn\_classification, 227
  - wbt\_knn\_regression, 228
  - wbt\_logistic\_regression, 290
  - wbt\_modified\_k\_means\_clustering, 329
  - wbt\_random\_forest\_classification, 391
  - wbt\_random\_forest\_regression, 392
  - wbt\_svm\_classification, 476
  - wbt\_svm\_regression, 477
- \* MathandStatsTools**
  - wbt\_absolute\_value, 19
  - wbt\_add, 22
  - wbt\_and, 25
  - wbt\_anova, 26
  - wbt\_arc\_cos, 27
  - wbt\_arc\_sin, 28
  - wbt\_arc\_tan, 29
  - wbt\_arcosh, 27
  - wbt\_arsinh, 30
  - wbt\_artanh, 31
  - wbt\_atan2, 34
  - wbt\_attribute\_correlation, 35
  - wbt\_attribute\_correlation\_neighbourhood\_analysis, 35
  - wbt\_attribute\_histogram, 36
  - wbt\_attribute\_scattergram, 37
  - wbt\_ceil, 53
  - wbt\_conditional\_evaluation, 68
  - wbt\_conditioned\_latin\_hypercube, 69
  - wbt\_cos, 78
  - wbt\_cosh, 79

- wbt\_crispness\_index, 87
- wbt\_cross\_tabulation, 87
- wbt\_cumulative\_distribution, 89
- wbt\_decrement, 95
- wbt\_divide, 109
- wbt\_equal\_to, 128
- wbt\_exp, 134
- wbt\_exp2, 135
- wbt\_floor, 167
- wbt\_greater\_than, 179
- wbt\_image\_autocorrelation, 200
- wbt\_image\_correlation, 201
- wbt\_image\_correlation\_neighbourhood\_analysis, 202
- wbt\_image\_regression, 203
- wbt\_in\_place\_add, 219
- wbt\_in\_place\_divide, 220
- wbt\_in\_place\_multiply, 221
- wbt\_in\_place\_subtract, 221
- wbt\_increment, 208
- wbt\_integer\_division, 216
- wbt\_inverse\_principal\_component\_analysis, 218
- wbt\_is\_no\_data, 223
- wbt\_kappa\_index, 226
- wbt\_ks\_test\_for\_normality, 229
- wbt\_less\_than, 242
- wbt\_list\_unique\_values, 284
- wbt\_list\_unique\_values\_raster, 285
- wbt\_ln, 286
- wbt\_log10, 289
- wbt\_log2, 289
- wbt\_max, 298
- wbt\_min, 318
- wbt\_modulo, 331
- wbt\_multiply, 335
- wbt\_negate, 348
- wbt\_not, 352
- wbt\_not\_equal\_to, 353
- wbt\_or, 359
- wbt\_paired\_sample\_t\_test, 360
- wbt\_phi\_coefficient, 371
- wbt\_power, 379
- wbt\_principal\_component\_analysis, 381
- wbt\_quantiles, 385
- wbt\_random\_field, 390
- wbt\_random\_sample, 393
- wbt\_raster\_calculator, 397
- wbt\_raster\_histogram, 399
- wbt\_raster\_summary\_stats, 402
- wbt\_reciprocal, 405
- wbt\_rescale\_value\_range, 423
- wbt\_root\_mean\_square\_error, 430
- wbt\_round, 431
- wbt\_sin, 444
- wbt\_sinh, 446
- wbt\_square, 460
- wbt\_square\_root, 460
- wbt\_subtract, 474
- wbt\_tan, 480
- wbt\_tanh, 481
- wbt\_to\_degrees, 493
- wbt\_to\_radians, 494
- wbt\_trend\_surface, 497
- wbt\_trend\_surface\_vector\_points, 498
- wbt\_truncate, 500
- wbt\_turning\_bands\_simulation, 501
- wbt\_two\_sample\_ks\_test, 502
- wbt\_wilcoxon\_signed\_rank\_test, 523
- wbt\_xor, 525
- wbt\_z\_scores, 530
- wbt\_zonal\_statistics, 529
- \* PrecisionAgriculture**
  - wbt\_reconcile\_multiple\_headers, 409
  - wbt\_recreate\_pass\_lines, 411
  - wbt\_remove\_field\_edge\_points, 415
  - wbt\_yield\_filter, 526
  - wbt\_yield\_map, 527
  - wbt\_yield\_normalization, 528
- \* StreamNetworkAnalysis**
  - wbt\_distance\_to\_outlet, 107
  - wbt\_extract\_streams, 140
  - wbt\_extract\_valleys, 141
  - wbt\_farthest\_channel\_head, 142
  - wbt\_find\_main\_stem, 159
  - wbt\_hack\_stream\_order, 180
  - wbt\_horton\_stream\_order, 194
  - wbt\_length\_of\_upstream\_channels, 241
  - wbt\_long\_profile, 292
  - wbt\_long\_profile\_from\_points, 293
  - wbt\_raster\_streams\_to\_vector, 401
  - wbt\_rasterize\_streams, 395

- wbt\_remove\_short\_streams, 419
- wbt\_repair\_stream\_vector\_topology, 421
- wbt\_shreve\_stream\_magnitude, 442
- wbt\_strahler\_stream\_order, 466
- wbt\_stream\_link\_class, 467
- wbt\_stream\_link\_identifier, 468
- wbt\_stream\_link\_length, 469
- wbt\_stream\_link\_slope, 470
- wbt\_stream\_slope\_continuous, 472
- wbt\_topological\_stream\_order, 489
- wbt\_tributary\_identifier, 499
- wbt\_vector\_stream\_network\_analysis, 513
- \* **WhiteboxUtilities**
  - wbt\_install\_wb\_extension, 215
  - wbt\_launch\_wb\_runner, 238
- \* **datasets**
  - sample\_dem\_data, 14
  - wbttoolparameters, 17
  - wbttools, 18
- check\_whitebox\_binary, 14
- install\_whitebox (wbt\_install), 214
- install\_whitebox(), 212
- sample\_dem\_data, 14
- wbt, 15
- wbt\_absolute\_value, 19
- wbt\_accumulation\_curvature, 20
- wbt\_activate, 21
- wbt\_adaptive\_filter, 21
- wbt\_add, 22
- wbt\_add\_point\_coordinates\_to\_table, 23
- wbt\_aggregate\_raster, 24
- wbt\_and, 25
- wbt\_anova, 26
- wbt\_arc\_cos, 27
- wbt\_arc\_sin, 28
- wbt\_arc\_tan, 29
- wbt\_arcosh, 27
- wbt\_arsinh, 30
- wbt\_artanh, 31
- wbt\_ascii\_to\_las, 31
- wbt\_aspect, 32
- wbt\_assess\_route, 33
- wbt\_atan2, 34
- wbt\_attribute\_correlation, 35
- wbt\_attribute\_correlation\_neighbourhood\_analysis, 35
- wbt\_attribute\_histogram, 36
- wbt\_attribute\_scattergram, 37
- wbt\_average\_flowpath\_slope, 38
- wbt\_average\_normal\_vector\_angular\_deviation, 39
- wbt\_average\_overlay, 40
- wbt\_average\_upslope\_flowpath\_length, 41
- wbt\_balance\_contrast\_enhancement, 41
- wbt\_basins, 42
- wbt\_bilateral\_filter, 43
- wbt\_block\_maximum\_gridding, 44
- wbt\_block\_minimum\_gridding, 45
- wbt\_boundary\_shape\_complexity, 46
- wbt\_breach\_depressions, 47
- wbt\_breach\_depressions\_least\_cost, 48
- wbt\_breach\_single\_cell\_pits, 49
- wbt\_breakline\_mapping, 49
- wbt\_buffer\_raster, 50
- wbt\_burn\_streams\_at\_roads, 51
- wbt\_canny\_edge\_detection, 52
- wbt\_ceil, 53
- wbt\_centroid, 54
- wbt\_centroid\_vector, 55
- wbt\_change\_vector\_analysis, 56
- wbt\_circular\_variance\_of\_aspect, 57
- wbt\_classify\_buildings\_in\_lidar, 58
- wbt\_classify\_lidar, 59
- wbt\_classify\_overlap\_points, 60
- wbt\_clean\_vector, 61
- wbt\_clip, 61
- wbt\_clip\_lidar\_to\_polygon, 62
- wbt\_clip\_raster\_to\_polygon, 63
- wbt\_closing, 64
- wbt\_clump, 65
- wbt\_colourize\_based\_on\_class, 66
- wbt\_colourize\_based\_on\_point\_returns, 67
- wbt\_compactness\_ratio, 68
- wbt\_compress\_rasters (wbt\_init), 210
- wbt\_conditional\_evaluation, 68
- wbt\_conditioned\_latin\_hypercube, 69
- wbt\_conservative\_smoothing\_filter, 71
- wbt\_construct\_vector\_tin, 72
- wbt\_contours\_from\_points, 73

- wbt\_contours\_from\_raster, 74
- wbt\_convert\_nodata\_to\_zero, 75
- wbt\_convert\_raster\_format, 75
- wbt\_corner\_detection, 76
- wbt\_correct\_vignetting, 77
- wbt\_cos, 78
- wbt\_cosh, 79
- wbt\_cost\_allocation, 79
- wbt\_cost\_distance, 80
- wbt\_cost\_pathway, 81
- wbt\_count\_if, 82
- wbt\_create\_colour\_composite, 83
- wbt\_create\_hexagonal\_vector\_grid, 84
- wbt\_create\_plane, 85
- wbt\_create\_rectangular\_vector\_grid, 86
- wbt\_crispness\_index, 87
- wbt\_cross\_tabulation, 87
- wbt\_csv\_points\_to\_vector, 88
- wbt\_cumulative\_distribution, 89
- wbt\_curvedness, 90
- wbt\_d8\_flow\_accumulation, 91
- wbt\_d8\_mass\_flux, 92
- wbt\_d8\_pointer, 93
- wbt\_d\_inf\_flow\_accumulation, 113
- wbt\_d\_inf\_mass\_flux, 114
- wbt\_d\_inf\_pointer, 115
- wbt\_data\_dir(wbt\_init), 210
- wbt\_dbscan, 94
- wbt\_decrement, 95
- wbt\_default\_path(wbt\_init), 210
- wbt\_dem\_void\_filling, 95
- wbt\_depth\_in\_sink, 97
- wbt\_depth\_to\_water, 98
- wbt\_dev\_from\_mean\_elev, 99
- wbt\_diff\_from\_mean\_elev, 102
- wbt\_diff\_of\_gaussian\_filter, 103
- wbt\_difference, 100
- wbt\_difference\_curvature, 101
- wbt\_direct\_decorrelation\_stretch, 105
- wbt\_directional\_relief, 104
- wbt\_dissolve, 106
- wbt\_distance\_to\_outlet, 107
- wbt\_diversity\_filter, 108
- wbt\_divide, 109
- wbt\_downslope\_distance\_to\_stream, 110
- wbt\_downslope\_flowpath\_length, 111
- wbt\_downslope\_index, 112
- wbt\_edge\_contamination, 115
- wbt\_edge\_density, 116
- wbt\_edge\_preserving\_mean\_filter, 117
- wbt\_edge\_proportion, 118
- wbt\_elev\_above\_pit, 121
- wbt\_elev\_percentile, 121
- wbt\_elev\_relative\_to\_min\_max, 122
- wbt\_elev\_relative\_to\_watershed\_min\_max, 123
- wbt\_elevation\_above\_stream, 119
- wbt\_elevation\_above\_stream\_euclidean, 120
- wbt\_eliminate\_coincident\_points, 124
- wbt\_elongation\_ratio, 125
- wbt\_embankment\_mapping, 126
- wbt\_emboss\_filter, 127
- wbt\_equal\_to, 128
- wbt\_erase, 129
- wbt\_erase\_polygon\_from\_lidar, 130
- wbt\_erase\_polygon\_from\_raster, 131
- wbt\_euclidean\_allocation, 132
- wbt\_euclidean\_distance, 132
- wbt\_evaluate\_training\_sites, 133
- wbt\_exe\_path(wbt\_init), 210
- wbt\_exe\_path(), 14
- wbt\_exp, 134
- wbt\_exp2, 135
- wbt\_export\_table\_to\_csv, 136
- wbt\_exposure\_towards\_wind\_flux, 137
- wbt\_extend\_vector\_lines, 138
- wbt\_extract\_nodes, 139
- wbt\_extract\_raster\_values\_at\_points, 139
- wbt\_extract\_streams, 140
- wbt\_extract\_valleys, 141
- wbt\_farthest\_channel\_head, 142
- wbt\_fast\_almost\_gaussian\_filter, 143
- wbt\_fd8\_flow\_accumulation, 144
- wbt\_fd8\_pointer, 145
- wbt\_feature\_preserving\_smoothing, 146
- wbt\_fetch\_analysis, 147
- wbt\_fill\_burn, 148
- wbt\_fill\_depressions, 149
- wbt\_fill\_depressions\_planchon\_and\_darboux, 150
- wbt\_fill\_depressions\_wang\_and\_liu, 151
- wbt\_fill\_missing\_data, 152
- wbt\_fill\_single\_cell\_pits, 153
- wbt\_filter\_lidar, 153

- wbt\_filter\_lidar\_classes, 154
- wbt\_filter\_lidar\_scan\_angles, 155
- wbt\_filter\_raster\_features\_by\_area, 156
- wbt\_find\_flightline\_edge\_points, 157
- wbt\_find\_lowest\_or\_highest\_points, 158
- wbt\_find\_main\_stem, 159
- wbt\_find\_no\_flow\_cells, 160
- wbt\_find\_parallel\_flow, 160
- wbt\_find\_patch\_or\_class\_edge\_cells, 161
- wbt\_find\_ridges, 162
- wbt\_fix\_dangling\_arcs, 163
- wbt\_flatten\_lakes, 164
- wbt\_flightline\_overlap, 165
- wbt\_flip\_image, 166
- wbt\_flood\_order, 167
- wbt\_floor, 167
- wbt\_flow\_accumulation\_full\_workflow, 168
- wbt\_flow\_length\_diff, 169
- wbt\_gamma\_correction, 170
- wbt\_gaussian\_contrast\_stretch, 171
- wbt\_gaussian\_curvature, 172
- wbt\_gaussian\_filter, 173
- wbt\_gaussian\_scale\_space, 174
- wbt\_generalize\_classified\_raster, 175
- wbt\_generalize\_with\_similarity, 176
- wbt\_generating\_function, 177
- wbt\_geomorphons, 178
- wbt\_greater\_than, 179
- wbt\_hack\_stream\_order, 180
- wbt\_heat\_map, 181
- wbt\_height\_above\_ground, 182
- wbt\_help, 182
- wbt\_high\_pass\_bilateral\_filter, 184
- wbt\_high\_pass\_filter, 185
- wbt\_high\_pass\_median\_filter, 186
- wbt\_highest\_position, 183
- wbt\_hillshade, 187
- wbt\_hillslopes, 188
- wbt\_histogram\_equalization, 189
- wbt\_histogram\_matching, 190
- wbt\_histogram\_matching\_two\_images, 191
- wbt\_hole\_proportion, 192
- wbt\_horizon\_angle, 193
- wbt\_horizontal\_excess\_curvature, 192
- wbt\_horton\_stream\_order, 194
- wbt\_hydrologic\_connectivity, 195
- wbt\_hypsometric\_analysis, 197
- wbt\_hypsometrically\_tinted\_hillshade, 196
- wbt\_idw\_interpolation, 198
- wbt\_ihs\_to\_rgb, 199
- wbt\_image\_autocorrelation, 200
- wbt\_image\_correlation, 201
- wbt\_image\_correlation\_neighbourhood\_analysis, 202
- wbt\_image\_regression, 203
- wbt\_image\_segmentation, 204
- wbt\_image\_slider, 205
- wbt\_image\_stack\_profile, 206
- wbt\_impoundment\_size\_index, 207
- wbt\_in\_place\_add, 219
- wbt\_in\_place\_divide, 220
- wbt\_in\_place\_multiply, 221
- wbt\_in\_place\_subtract, 221
- wbt\_increment, 208
- wbt\_individual\_tree\_detection, 208
- wbt\_init, 210
- wbt\_insert\_dams, 213
- wbt\_install, 214
- wbt\_install\_extension (wbt\_install), 214
- wbt\_install\_wb\_extension, 215
- wbt\_integer\_division, 216
- wbt\_integral\_image, 217
- wbt\_intersect, 217
- wbt\_inverse\_principal\_component\_analysis, 218
- wbt\_is\_no\_data, 223
- wbt\_isobasins, 222
- wbt\_jenson\_snap\_pour\_points, 224
- wbt\_join\_tables, 225
- wbt\_k\_means\_clustering, 230
- wbt\_k\_nearest\_mean\_filter, 231
- wbt\_kappa\_index, 226
- wbt\_knn\_classification, 227
- wbt\_knn\_regression, 228
- wbt\_ks\_test\_for\_normality, 229
- wbt\_laplacian\_filter, 232
- wbt\_laplacian\_of\_gaussian\_filter, 233
- wbt\_las\_to\_ascii, 234
- wbt\_las\_to\_laz, 234
- wbt\_las\_to\_multipoint\_shapefile, 235
- wbt\_las\_to\_shapefile, 236
- wbt\_las\_to\_zlidar, 237

- wbt\_launch\_wb\_runner, 238
- wbt\_layer\_footprint, 238
- wbt\_laz\_to\_las, 239
- wbt\_lee\_sigma\_filter, 240
- wbt\_length\_of\_upstream\_channels, 241
- wbt\_less\_than, 242
- wbt\_license, 243
- wbt\_lidar\_block\_maximum, 243
- wbt\_lidar\_block\_minimum, 244
- wbt\_lidar\_classify\_subset, 245
- wbt\_lidar\_colourize, 246
- wbt\_lidar\_contour, 247
- wbt\_lidar\_digital\_surface\_model, 248
- wbt\_lidar\_eigenvalue\_features, 249
- wbt\_lidar\_elevation\_slice, 250
- wbt\_lidar\_ground\_point\_filter, 251
- wbt\_lidar\_hex\_binning, 252
- wbt\_lidar\_hillshade, 253
- wbt\_lidar\_histogram, 254
- wbt\_lidar\_idw\_interpolation, 255
- wbt\_lidar\_info, 256
- wbt\_lidar\_join, 257
- wbt\_lidar\_kappa\_index, 258
- wbt\_lidar\_nearest\_neighbour\_gridding, 259
- wbt\_lidar\_point\_density, 260
- wbt\_lidar\_point\_return\_analysis, 261
- wbt\_lidar\_point\_stats, 262
- wbt\_lidar\_ransac\_planes, 263
- wbt\_lidar\_rbf\_interpolation, 264
- wbt\_lidar\_remove\_duplicates, 265
- wbt\_lidar\_remove\_outliers, 266
- wbt\_lidar\_rooftop\_analysis, 267
- wbt\_lidar\_segmentation, 269
- wbt\_lidar\_segmentation\_based\_filter, 270
- wbt\_lidar\_shift, 271
- wbt\_lidar\_sibson\_interpolation, 272
- wbt\_lidar\_sort\_by\_time, 273
- wbt\_lidar\_thin, 274
- wbt\_lidar\_thin\_high\_density, 275
- wbt\_lidar\_tile, 276
- wbt\_lidar\_tile\_footprint, 277
- wbt\_lidar\_tin\_gridding, 278
- wbt\_lidar\_tophat\_transform, 279
- wbt\_line\_detection\_filter, 281
- wbt\_line\_intersections, 282
- wbt\_line\_thinning, 283
- wbt\_linearity\_index, 280
- wbt\_lines\_to\_polygons, 280
- wbt\_list\_tools, 284, 433
- wbt\_list\_tools(), 19
- wbt\_list\_unique\_values, 284
- wbt\_list\_unique\_values\_raster, 285
- wbt\_ln, 286
- wbt\_local\_hypsometric\_analysis, 287
- wbt\_local\_quadratic\_regression, 288
- wbt\_log10, 289
- wbt\_log2, 289
- wbt\_logistic\_regression, 290
- wbt\_long\_profile, 292
- wbt\_long\_profile\_from\_points, 293
- wbt\_longest\_flowpath, 291
- wbt\_low\_points\_on\_headwater\_divides, 295
- wbt\_lowest\_position, 294
- wbt\_majority\_filter, 296
- wbt\_map\_off\_terrain\_objects, 297
- wbt\_max, 298
- wbt\_max\_absolute\_overlay, 301
- wbt\_max\_anisotropy\_dev, 301
- wbt\_max\_anisotropy\_dev\_signature, 302
- wbt\_max\_branch\_length, 303
- wbt\_max\_difference\_from\_mean, 304
- wbt\_max\_downslope\_elev\_change, 305
- wbt\_max\_elev\_dev\_signature, 307
- wbt\_max\_elevation\_deviation, 306
- wbt\_max\_overlay, 308
- wbt\_max\_procs (wbt\_init), 210
- wbt\_max\_upslope\_elev\_change, 309
- wbt\_max\_upslope\_flowpath\_length, 309
- wbt\_max\_upslope\_value, 310
- wbt\_maximal\_curvature, 299
- wbt\_maximum\_filter, 300
- wbt\_md\_inf\_flow\_accumulation, 311
- wbt\_mean\_curvature, 312
- wbt\_mean\_filter, 313
- wbt\_median\_filter, 314
- wbt\_medoid, 315
- wbt\_merge\_line\_segments, 316
- wbt\_merge\_table\_with\_csv, 317
- wbt\_merge\_vectors, 318
- wbt\_min, 318
- wbt\_min\_absolute\_overlay, 325
- wbt\_min\_dist\_classification, 325
- wbt\_min\_downslope\_elev\_change, 326

- wbt\_min\_max\_contrast\_stretch, 327
- wbt\_min\_overlay, 328
- wbt\_minimal\_curvature, 319
- wbt\_minimum\_bounding\_box, 320
- wbt\_minimum\_bounding\_circle, 321
- wbt\_minimum\_bounding\_envelope, 322
- wbt\_minimum\_convex\_hull, 323
- wbt\_minimum\_filter, 324
- wbt\_modified\_k\_means\_clustering, 329
- wbt\_modify\_lidar, 330
- wbt\_modify\_no\_data\_value, 331
- wbt\_modulo, 331
- wbt\_mosaic, 332
- wbt\_mosaic\_with\_feathering, 333
- wbt\_multi\_part\_to\_single\_part, 344
- wbt\_multidirectional\_hillshade, 334
- wbt\_multiply, 335
- wbt\_multiply\_overlay, 336
- wbt\_multiscale\_curvatures, 337
- wbt\_multiscale\_elevation\_percentile, 338
- wbt\_multiscale\_roughness, 339
- wbt\_multiscale\_roughness\_signature, 340
- wbt\_multiscale\_std\_dev\_normals, 341
- wbt\_multiscale\_std\_dev\_normals\_signature, 342
- wbt\_multiscale\_topographic\_position\_image, 343
- wbt\_narrowness\_index, 345
- wbt\_natural\_neighbour\_interpolation, 346
- wbt\_nearest\_neighbour\_gridding, 347
- wbt\_negate, 348
- wbt\_new\_raster\_from\_base, 348
- wbt\_normal\_vectors, 351
- wbt\_normalize\_lidar, 350
- wbt\_normalized\_difference\_index, 349
- wbt\_not, 352
- wbt\_not\_equal\_to, 353
- wbt\_num\_downslope\_neighbours, 354
- wbt\_num\_inflowing\_neighbours, 354
- wbt\_num\_upslope\_neighbours, 355
- wbt\_olympic\_filter, 356
- wbt\_opening, 357
- wbt\_openness, 358
- wbt\_options (wbt\_init), 210
- wbt\_or, 359
- wbt\_paired\_sample\_t\_test, 360
- wbt\_panchromatic\_sharpening, 361
- wbt\_parallelepiped\_classification, 362
- wbt\_patch\_orientation, 363
- wbt\_pennock\_landform\_class, 363
- wbt\_percent\_elev\_range, 366
- wbt\_percent\_equal\_to, 367
- wbt\_percent\_greater\_than, 368
- wbt\_percent\_less\_than, 369
- wbt\_percentage\_contrast\_stretch, 364
- wbt\_percentile\_filter, 365
- wbt\_perimeter\_area\_ratio, 370
- wbt\_phi\_coefficient, 371
- wbt\_pick\_from\_list, 372
- wbt\_piecewise\_contrast\_stretch, 373
- wbt\_plan\_curvature, 374
- wbt\_polygon\_area, 376
- wbt\_polygon\_long\_axis, 377
- wbt\_polygon\_perimeter, 378
- wbt\_polygon\_short\_axis, 378
- wbt\_polygonize, 375
- wbt\_polygons\_to\_lines, 375
- wbt\_power, 379
- wbt\_prewitt\_filter, 380
- wbt\_principal\_component\_analysis, 381
- wbt\_print\_geo\_tiff\_tags, 382
- wbt\_profile, 382
- wbt\_profile\_curvature, 383
- wbt\_qin\_flow\_accumulation, 384
- wbt\_quantiles, 385
- wbt\_quinn\_flow\_accumulation, 386
- wbt\_radial\_basis\_function\_interpolation, 387
- wbt\_radius\_of\_gyration, 388
- wbt\_raise\_walls, 389
- wbt\_random\_field, 390
- wbt\_random\_forest\_classification, 391
- wbt\_random\_forest\_regression, 392
- wbt\_random\_sample, 393
- wbt\_range\_filter, 394
- wbt\_raster\_area, 396
- wbt\_raster\_calculator, 397
- wbt\_raster\_cell\_assignment, 398
- wbt\_raster\_histogram, 399
- wbt\_raster\_perimeter, 400
- wbt\_raster\_streams\_to\_vector, 401
- wbt\_raster\_summary\_stats, 402
- wbt\_raster\_to\_vector\_lines, 402

- wbt\_raster\_to\_vector\_points, 403
- wbt\_raster\_to\_vector\_polygons, 404
- wbt\_rasterize\_streams, 395
- wbt\_reciprocal, 405
- wbt\_reclass, 406
- wbt\_reclass\_equal\_interval, 407
- wbt\_reclass\_from\_file, 408
- wbt\_reconcile\_multiple\_headers, 409
- wbt\_recover\_flightline\_info, 410
- wbt\_recreate\_pass\_lines, 411
- wbt\_reinitialize\_attribute\_table, 412
- wbt\_related\_circumscribing\_circle, 412
- wbt\_relative\_aspect, 413
- wbt\_relative\_topographic\_position, 414
- wbt\_remove\_field\_edge\_points, 415
- wbt\_remove\_off\_terrain\_objects, 416
- wbt\_remove\_polygon\_holes, 417
- wbt\_remove\_raster\_polygon\_holes, 418
- wbt\_remove\_short\_streams, 419
- wbt\_remove\_spurs, 420
- wbt\_repair\_stream\_vector\_topology, 421
- wbt\_resample, 422
- wbt\_rescale\_value\_range, 423
- wbt\_result (wbt), 15
- wbt\_rgb\_to\_ihs, 424
- wbt\_rho8\_flow\_accumulation, 425
- wbt\_rho8\_pointer, 426
- wbt\_ring\_curvature, 427
- wbt\_river\_centerlines, 428
- wbt\_roberts\_cross\_filter, 429
- wbt\_root\_mean\_square\_error, 430
- wbt\_rotor, 430
- wbt\_round, 431
- wbt\_ruggedness\_index, 432
- wbt\_run\_tool, 433
- wbt\_rust\_backtrace, 434
- wbt\_scharr\_filter, 434
- wbt\_sediment\_transport\_index, 435
- wbt\_select\_tiles\_by\_polygon, 436
- wbt\_set\_nodata\_value, 437
- wbt\_shadow\_animation, 438
- wbt\_shadow\_image, 439
- wbt\_shape\_complexity\_index, 440
- wbt\_shape\_complexity\_index\_raster, 441
- wbt\_shape\_index, 441
- wbt\_shreve\_stream\_magnitude, 442
- wbt\_sigmoidal\_contrast\_stretch, 443
- wbt\_sin, 444
- wbt\_single\_part\_to\_multi\_part, 445
- wbt\_sinh, 446
- wbt\_sink, 446
- wbt\_slope, 447
- wbt\_slope\_vs\_aspect\_plot, 448
- wbt\_slope\_vs\_elevation\_plot, 449
- wbt\_smooth\_vectors, 450
- wbt\_smooth\_vegetation\_residual, 451
- wbt\_snap\_pour\_points, 452
- wbt\_sobel\_filter, 453
- wbt\_sort\_lidar, 454
- wbt\_source, 455
- wbt\_spherical\_std\_dev\_of\_normals, 455
- wbt\_split\_colour\_composite, 456
- wbt\_split\_lidar, 457
- wbt\_split\_vector\_lines, 458
- wbt\_split\_with\_lines, 459
- wbt\_square, 460
- wbt\_square\_root, 460
- wbt\_standard\_deviation\_contrast\_stretch, 461
- wbt\_standard\_deviation\_filter, 462
- wbt\_standard\_deviation\_of\_slope, 463
- wbt\_stochastic\_depression\_analysis, 464
- wbt\_strahler\_order\_basins, 465
- wbt\_strahler\_stream\_order, 466
- wbt\_stream\_link\_class, 467
- wbt\_stream\_link\_identifier, 468
- wbt\_stream\_link\_length, 469
- wbt\_stream\_link\_slope, 470
- wbt\_stream\_power\_index, 471
- wbt\_stream\_slope\_continuous, 472
- wbt\_subbasins, 473
- wbt\_subtract, 474
- wbt\_sum\_overlay, 475
- wbt\_surface\_area\_ratio, 475
- wbt\_svm\_classification, 476
- wbt\_svm\_regression, 477
- wbt\_symmetrical\_difference, 479
- wbt\_tan, 480
- wbt\_tangential\_curvature, 480
- wbt\_tanh, 481
- wbt\_thicken\_raster\_line, 482
- wbt\_time\_in\_daylight, 483
- wbt\_tin\_gridding, 484
- wbt\_to\_degrees, 493
- wbt\_to\_radians, 494



- wbt\_tool\_help, 486
- wbt\_tool\_help(), 17
- wbt\_tool\_parameters, 486
- wbt\_tool\_parameters(), 18
- wbt\_toolbox, 485
- wbt\_tophat\_transform, 487
- wbt\_topo\_render, 490
- wbt\_topographic\_position\_animation, 488
- wbt\_topological\_stream\_order, 489
- wbt\_total\_curvature, 491
- wbt\_total\_filter, 492
- wbt\_trace\_downslope\_flowpaths, 495
- wbt\_travelling\_salesman\_problem, 496
- wbt\_trend\_surface, 497
- wbt\_trend\_surface\_vector\_points, 498
- wbt\_tributary\_identifier, 499
- wbt\_truncate, 500
- wbt\_turning\_bands\_simulation, 501
- wbt\_two\_sample\_ks\_test, 502
- wbt\_union, 503
- wbt\_unnest\_basins, 504
- wbt\_unsharp\_masking, 505
- wbt\_unsphericity, 506
- wbt\_update\_nodata\_cells, 507
- wbt\_upslope\_depression\_storage, 508
- wbt\_user\_defined\_weights\_filter, 508
- wbt\_vector\_hex\_binning, 509
- wbt\_vector\_lines\_to\_raster, 510
- wbt\_vector\_points\_to\_raster, 511
- wbt\_vector\_polygons\_to\_raster, 512
- wbt\_vector\_stream\_network\_analysis, 513
- wbt\_verbose (wbt\_init), 210
- wbt\_version, 514
- wbt\_vertical\_excess\_curvature, 515
- wbt\_view\_code, 517
- wbt\_viewshed, 516
- wbt\_visibility\_index, 517
- wbt\_voronoi\_diagram, 518
- wbt\_watershed, 519
- wbt\_wd (wbt\_init), 210
- wbt\_weighted\_overlay, 520
- wbt\_weighted\_sum, 521
- wbt\_wetness\_index, 522
- wbt\_wilcoxon\_signed\_rank\_test, 523
- wbt\_write\_function\_memory\_insertion, 524
- wbt\_xor, 525
- wbt\_yield\_filter, 526
- wbt\_yield\_map, 527
- wbt\_yield\_normalization, 528
- wbt\_z\_scores, 530
- wbt\_zlidar\_to\_las, 529
- wbt\_zonal\_statistics, 529
- wbttoolparameters, 17, 19
- wbttools, 18, 18
- whitebox, 212