

Manual for Package: gis

Revision 5M

Karl Kästner

July 5, 2023

Contents

1	@GeoImg	5
1.1	GeoImg	5
1.2	crop	5
1.3	read	5
1.4	write	5
2	gis	5
2.1	GPX	5
2.2	batavia_zero	5
3	centreline/@Centreline	5
3.1	Centreline	5
3.2	channel_planimetry	6
3.3	clip	6
3.4	connect_graph	6
3.5	curvature	6
3.6	cut	6
3.7	determine_width	6
3.8	distance	6
3.9	export_cross_section	6
3.10	export_node	6
3.11	export_shp	6
3.12	find_nearest_segment	7
3.13	from_polygon	7
3.14	from_shp	7
3.15	get	7
3.16	init	7
3.17	init_connect	7
3.18	init_node_D	7

3.19	link_centreline	7
3.20	plot	7
3.21	plot_connection	7
3.22	prune	8
3.23	prune_leaves	8
3.24	prune_manually	8
3.25	reachable	8
3.26	remove_duplicate_points	8
3.27	resample	8
3.28	routing	8
3.29	routing2	8
3.30	shp_resample_simple	8
3.31	snmesh	8
3.32	squeeze	9
3.33	trim_ends	9
3.34	weighed_connection_matrix	9
3.35	xy2sn	9
4	centreline/@Segment	9
4.1	Segment	9
4.2	build_inverse_index	9
4.3	connectivity_matrix	9
4.4	init_seg_id	9
5	centreline	9
5.1	sn2xy_quadratic	9
5.2	thalweg	10
5.3	xy2sn_quadratic	10
6	gis	10
6.1	gpx_export_csv	10
6.2	hgt_plot	10
6.3	hgt_read	10
6.4	hgt_read_all	10
6.5	hgt_resample	10
6.6	nmeatime	10
6.7	read_xyz	10
7	shapefile/@Shp	11
7.1	Shp	11
7.2	area	11
7.3	buffer	11
7.4	centroid	11
7.5	clip	11

7.6	clip_rect	11
7.7	close_polygon	11
7.8	concat	11
7.9	connect_network	12
7.10	contour	12
7.11	copy_attribute	12
7.12	cp	12
7.13	create	12
7.14	curvature	12
7.15	cut	12
7.16	diameter	12
7.17	edges	13
7.18	export_geo	13
7.19	export_gpx	13
7.20	export_gpx_track	13
7.21	export_ldb	13
7.22	export_poly	13
7.23	export_sdf	13
7.24	export_spline	13
7.25	extract_coastline	14
7.26	first_point	14
7.27	flat	14
7.28	generate_four_colour_index	14
7.29	generate_rectangle	14
7.30	import_geo	14
7.31	import_poly	14
7.32	inpolygon	14
7.33	join_lines	14
7.34	last_point	15
7.35	latlon2utm	15
7.36	length	15
7.37	length2	15
7.38	line2point	15
7.39	link_lines	15
7.40	make_clockwise	15
7.41	merge	15
7.42	merge2	16
7.43	nan2zero	16
7.44	padd_nan	16
7.45	plot	16
7.46	points	16
7.47	polygon_boundary	16
7.48	read	16
7.49	readZ	16

7.50	reassign_id	17
7.51	remove_duplicate_points	17
7.52	remove_leaves	17
7.53	remove_nan	17
7.54	remove_polygon_closure	17
7.55	remove_short_elements	17
7.56	renumber	17
7.57	resample	17
7.58	resample_2	18
7.59	resample_min	18
7.60	resample_quick	18
7.61	scale	18
7.62	segment	18
7.63	select_for_refinement	18
7.64	set_geometry	18
7.65	set_resolution	18
7.66	singlepart_to_multipart	19
7.67	skip	19
7.68	smooth	19
7.69	split_jump	19
7.70	split_line	19
7.71	split_nan	19
7.72	swap_hemisphere	19
7.73	translate	19
7.74	write	20
8	shapefile	20
8.1	astar_multi	20
8.2	astar_recursive	20
8.3	edge_chain	20
8.4	edge_from_bnd	20
8.5	preload_shp	20
8.6	read_gpx	20
8.7	shapewrite__	20
8.8	shapewrite_man	23
8.9	shp2geo	23
8.10	shp2kml	23
8.11	shp_plot_attribute	23
8.12	split_section	23
8.13	write_polygon	23
9	gis	23
9.1	shp2csv	23
9.2	write_xyz	23

1 @GeoImg

1.1 GeoImg

loading, manipulation and writing of geospatial images with world file

1.2 crop

cropt the image and modify the pgw accordingly

1.3 read

read geospatial image and pgw from disk and

1.4 write

write geospatial image and pgw to disk

2 gis

2.1 GPX

2.2 batavia_zero

3 centreline/@Centreline

3.1 Centreline

3.2 `channel_planimetry`

3.3 `clip`

3.4 `connect_graph`

3.5 `curvature`

3.6 `cut`

3.7 `determine_width`

3.8 `distance`

3.9 `export_cross_section`

3.10 `export_node`

3.11 `export_shp`

3.12 `find_nearest_segment`

3.13 `from_polygon`

3.14 `from_shp`

3.15 `get`

3.16 `init`

```
obj.seg_S(id(end)) = NaN;
```

3.17 `init_connect`

3.18 `init_node_D`

3.19 `link_centreline`

3.20 `plot`

3.21 `plot_connection`

3.22 `prune`

3.23 `prune_leaves`

3.24 `prune_manually`

3.25 `reachable`

3.26 `remove_duplicate_points`

3.27 `resample`

3.28 `routing`

3.29 `routing2`

3.30 `shp_resample_simple`

3.31 `snmesh`

3.32 squeeze

3.33 trim_ends

3.34 weighed_connection_matrix

3.35 xy2sn

4 centreline/@Segment

4.1 Segment

4.2 build_inverse_index

4.3 connectivity_matrix

4.4 init_seg_id

5 centreline

5.1 sn2xy_quadratic

5.2 thalweg

5.3 xy2sn_quadratic

6 gis

6.1 gpx_export_csv

6.2 hgt_plot

6.3 hgt_read

```
% [ floor(median(z(kk))) meannan(z(kk)) min(z(kk)) max(z(kk)) ]
```

6.4 hgt_read_all

6.5 hgt_resample

6.6 nmeatime

6.7 read_xyz

7 shapefile/@Shp

7.1 Shp

shape file processing

7.2 area

area of polygon shapes

7.3 buffer

buffer or shrink a polygon by a fixed distance

7.4 centroid

7.5 clip

crop input shape file to specified polygon

7.6 clip_rect

rectrangular crop of the shapefile

7.7 close_polygon

close polygon, i.e. make the first point identical to the last

7.8 concat

concatenate two shapefiles

7.9 connect_network

```
TODO make unique
attach segments to
XY = [cvec(shp.X),shp.;
knnsearch for nearest n neighbours
for each segment
```

7.10 contour

7.11 copy_attribute

copy attributes from one shapefile to the other

7.12 cp

copy a shapefile on disk

7.13 create

create a new shapefile with given geometry

7.14 curvature

curvature of line segments

7.15 cut

7.16 diameter

determine diameter of polygon of every element

7.17 edges

edges of polygons line loops loops

7.18 export_geo

export geometry file understood by SLIM

7.19 export_gpx

export data into a gpx file

7.20 export_gpx_track

export a data into a gpx track file

7.21 export_ldb

export Delft3D-4 land-boundary

7.22 export_poly

export poly-file understood by SLIM

7.23 export_sdf

7.24 export_spline

export splines (for D3D?)

7.25 extract_coastline

7.26 first_point

extract first point of all shapefile features

7.27 flat

7.28 generate_four_colour_index

unique colour-indices fpr poligons

7.29 generate_rectangle

generate rectangular polygon

7.30 import_geo

7.31 import_poly

import poly file

7.32 inpolygon

test if point is in any of the polygons

7.33 join_lines

join line segments

7.34 last_point

return last point of features

7.35 latlon2utm

convert latitude and longitude to utm

7.36 length

number of points of each feature

7.37 length2

length of line segments

7.38 line2point

convert lines to points

7.39 link_lines

link lines with same endpoints

7.40 make_clockwise

make polygons clockwise

7.41 merge

7.42 merge2

7.43 nan2zero

replace not a number values with zeros, for writing to disk
NAN values are not allowed according to the spec

7.44 padd_nan

padd NaN at end of features

7.45 plot

display the shapefile

7.46 points

returns points of the features

7.47 polygon_boundary

7.48 read

read shapefile from file

7.49 readZ

read shapefile with z-data from file
this is a workaround, as matlab cannot read files with z-data

7.50 reassign_id

assign a unique identifier to each feature

7.51 remove_duplicate_points

remove duplicate points from features

7.52 remove_leaves

7.53 remove_nan

remove NaN points from features

7.54 remove_polygon_closure

remove last points of polygon if they are identical to the first

7.55 remove_short_elements

remove features with few points

7.56 renumber

generate a new index

7.57 resample

resample coordinates

7.58 resample_2

resample coordinates

7.59 resample_min

resample coordinates

7.60 resample_quick

resample coordinates

7.61 scale

7.62 segment

separate disjoint sections of polygons and lines

7.63 select_for_refinement

select elements for refinement

7.64 set_geometry

set feature geometry

7.65 set_resolution

set resolution for mesh generation

7.66 singlepart_to_multipart

concatenate line segments (parts) of shp data files into one
same as single part to multipart in qgis
returns also indices into the original file

7.67 skip

quick resampling of features by leaving out points

7.68 smooth

smooth the features

7.69 split_jump

split features where distance between points exceeds a threshold

7.70 split_line

split line features into single segments

7.71 split_nan

splits shp line and polygons at NaN into two different groups

7.72 swap_hemisphere

swap northern and southern hemisphere for UTM coordinates

7.73 translate

translate coordinates

7.74 write

write the shapefile to disk

8 shapefile

8.1 astar_multi

8.2 astar_recursive

astar path finding algorithm

8.3 edge_chain

8.4 edge_from_bnd

8.5 preload_shp

8.6 read_gpx

8.7 shapewrite__

Copyright (C) 2014,2015 Philip Nienhuis

This program is free software; you can redistribute it and/or
modify it
under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 3 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

-*- texinfo -*-

@deftypefn {Function File} {@var{status} =} shapewrite (@var{shpstr}, @var{fname})

Write contents of map- or geostruct to a GIS shape file.

@var{shpstr} must be a valid mapstruct or geostruct, a struct array with an entry for each shape feature, with fields Geometry, BoundingBox, and X and Y (mapstruct) or Lat and Lon (geostruct). For geostructs, Lat and Lon field data will be written as X and Y data. Field Geometry can have data values of 'Point', 'MultiPoint', 'Line', or 'Polygon', all case-insensitive. For each shape feature, field BoundingBox should contain the minimum and maximum (X,Y) coordinates in a 2x2 array [minX, minY; maxX, maxY]. The X and Y fields should contain X (or Latitude) and Y (or Longitude) coordinates for each point or vertex as row vectors; for polylines and polygons vertices of each subfeature (if present) should be separated by NaN entries.

@var{fname} should be a valid shape file name, optionally with a '.shp' suffix.

shapewrite produces 2 or 3 files, i.e. a .shp file (the actual shape file), a .shx file (index file), and if @var{shpstr} contained additional fields, a .dbf file (dBase type 3) with the contents of those additional fields.

@var{status} is 1 if the shape file set was written successfully, 0 otherwise.

```

@seealso{shaperead, shapeinfo}
@end deftpfn
Author: Philip Nienhuis <prnienhuis@users.sf.net>
Created: 2014-12-30
Input validation
Assess shape variable type (oct or ml/geo ml/map)
Yep. Find out what type
Assume it is an Octave-style struct read by shaperead
Assume it is a Matlab-style mapstruct
Assume it is a Matlab-style geostruct
Not a supported struct type
Check file name
Later on bname.shx and bname.dbf will be read
Prepare a few things
Change Lat/Lon fields into X/Y
Only now (after input checks) open .shp and .shx files & rewind
    just to be sure
Write headers in .shp & .shx (identical). First magic number 9994 +
    5 zeros
In between here = filelength in 16-bit words (single). For .shx it's
    known
Next, shp file version
Shape feature type
Bounding box. Can be run later for ML type shape structs. Fill with
    zeros
Skip to start of first record position
Write shape features one by one
Write record start pos to .shx file
Write record contents
Point
Record index number
Record length (fixed)
Shape type
Simply write XY coordinates
MultiPoint
Record index number
Record length
Shape type
Bounding box
Nr of points
Polyline/-gon
Record index number
Prepare multipart polygons
Augment idx for later on, & this trick eliminates trailing NaN rows
Record length
Shape type
Bounding box
Number of parts, number of points, part pointers
Write file length into .shp header

```

```
Close files
Check for dbfwrite function
Write rest of attributes
Attributes + shp data in mapstruct
Attributes + shp data in geostruct
```

8.8 shapewrite_man

8.9 shp2geo

8.10 shp2kml

8.11 shp_plot_attribute

8.12 split_section

8.13 write_polygon

9 gis

9.1 shp2csv

9.2 write_xyz