Package 'rgugik'

May 9, 2026

Type Package

Title Search and Retrieve Spatial Data from 'GUGiK'

Version 0.4.2

Description Automatic open data acquisition from resources of Polish Head Office of Geodesy and Cartography ('Główny Urząd Geodezji i Kartografii') (<https://www.gov.pl/web/gugik>).
Available datasets include various types of numeric, raster and vector data, such as orthophotomaps, digital elevation models (digital terrain models, digital surface model, point clouds), state register of borders, spatial databases, geometries of cadastral parcels, 3D models of buildings, and more.
It is also possible to geocode addresses or objects using the geocodePL_get() function.

License MIT + file LICENSE

Depends R (>= 3.5)

Imports sf, jsonlite

Suggests curl, knitr, rmarkdown, stars, terra, testthat, tibble

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

URL https://kadyb.github.io/rgugik/, https://github.com/kadyb/rgugik

BugReports https://github.com/kadyb/rgugik/issues

VignetteBuilder knitr

NeedsCompilation no

Author Krzysztof Dyba [aut, cre] (ORCID:
<https://orcid.org/0000-0002-8614-3816>),
Jakub Nowosad [aut] (ORCID: <https://orcid.org/0000-0002-1057-3721>),
Maciej Beręsewicz [ctb] (ORCID:
<https://orcid.org/0000-0002-8281-4301>),
Grzegorz Sapijaszko [ctb],
GUGiK [ctb] (source of the data)

Maintainer Krzysztof Dyba <adres7@gmail.com>

Repository CRAN

Date/Publication 2025-03-04 14:30:02 UTC

Contents

`borders_download` *Download State Register of Borders*

Description

Download State Register of Borders

Usage

```
borders_download(type, outdir = ".", unzip = TRUE, ...)
```

Arguments

`type`	"administrative units", "special units" or "addresses"
`outdir`	(optional) name of the output directory; by default, files are saved in the working directory

unzip	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
...	additional argument for utils::download.file()

Value

a selected data type in SHP format

Examples

```
## Not run:
borders_download("administrative units") # 366 MB

## End(Not run)
```

---

borders_get *Get the boundaries of administrative units*

Description

Get the boundaries of administrative units

Usage

```
borders_get(voivodeship = NULL, county = NULL, commune = NULL, TERYT = NULL)
```

Arguments

voivodeship	selected voivodeships in Polish. Check voivodeship_names() function
county	county names in Polish. Check county_names() function
commune	commune names in Polish. Check commune_names() function
TERYT	voivodeships, counties or communes (2, 4 or 7 characters)

Details

If all arguments are NULL (default), the boundary of Poland will be returned.

Value

a sf data.frame (EPSG: 2180)

Examples

```
## Not run:
voivodeship_geom = borders_get(voivodeship = "lubuskie") # 494 KB
county_geom = borders_get(county = "Sopot") # 18 KB
commune_geom = borders_get(commune = c("Hel", "Krynica Morska")) # 11 KB
poland_geom = borders_get() # 1124.3 KB

## End(Not run)
```

commune_names *Communes in Poland*

Description

The data frame contains names of communes, and their identifiers (TERC, 7 characters).

Usage

```
commune_names
```

Format

An object of class `data.frame` with 2479 rows and 2 columns.

Details

Last update: 6 January 2025

Examples

```
commune_names
```

county_names *Counties in Poland*

Description

The data frame contains the names of counties, their identifiers (TERYT, 4 characters) and the availability of building models in the LOD2 standard (logical value).

Usage

```
county_names
```

Format

An object of class `data.frame` with 380 rows and 3 columns.

Examples

```
county_names
```

DEM_request *Get metadata and links to available digital elevation models*

Description

Get metadata and links to available digital elevation models

Usage

```
DEM_request(x)
```

Arguments

x	an sf, sfc or SpatVector object with one or more features (requests are based on the bounding boxes of the provided features)

Details

The server can return a maximum of 1000 records in a single query. If your area of interest exceeds this limit, you can generate a grid of smaller polygons (sf::st_make_grid()) or a regular grid of points (sf::st_sample()).

Value

a data frame with metadata and links to the digital elevation models (different formats of digital terrain model, digital surface model and point clouds)

See Also

tile_download()

Examples

```
## Not run:
library(sf)
polygon_path = system.file("datasets/search_area.gpkg", package = "rgugik")
polygon = read_sf(polygon_path)
req_df = DEM_request(polygon)

# simple filtering by attributes
req_df = req_df[req_df$year > 2018, ]
req_df = req_df[req_df$product == "PointCloud" & req_df$format == "LAS", ]

## End(Not run)
```

egib_download *Download Land and Building Register (EGiB) layers*

Description

Download Land and Building Register (EGiB) layers

Usage

```
egib_download(
  county = NULL,
  TERYT = NULL,
  layer = "parcels",
  outdir = ".",
  ...
)
```

Arguments

county	County name in Polish. Check `county_names()` function.
TERYT	County ID (4 characters).
layer	Requested layer, `parcels` default. Other common layer is `buildings`. You can check available layers by `egib_layers()` dataset.
outdir	name of the output directory where the data has to be stored, current directory by default. If you don't want to download data, pass a `NULL` value.
...	any other parameter passed to `sf::read_sf()` function.

Value

If the `outdir` argument is specified, the data will be downloaded to disk in geopackage format. If the `outdir` argument is `NULL`, simple feature data frame is returned, but only for the first object.

References

The EGiB data (*Ewidencja Gruntów i Budynków*) consist of 2 primary layers: cadastral data (parcels) and buildings footprints. The data is maintained on county level, which results in 380 units (different sources of data). It may contain additional layers like points of detailed horizontal and vertical geodetic control network ("*osnowa_pozioma*" and "*osnowa_pionowa*" respectively).

https://www.geoportal.gov.pl/en/data/land-and-building-register-egib/

https://www.geoportal.gov.pl/en/data/detailed-control-network-database-bdsog/

Examples

```
## Not run:
egib_download(TERYT = c("2476", "2264"), layer = "buildings", outdir = ".") # 2.2 + 2.6 MB
parcels = egib_download(county = "Świętochłowice", layer = "parcels", outdir = NULL) # 3.9 MB

## End(Not run)
```

---

egib_layers *Available URLs and layers of Land and Building Register (EGiB)*

Description

The data frame contains the names of counties, their identifiers (TERYT, 4 characters), the URL to the EGiB data (as Web Feature Service) and the names of available layers.

Usage

```
egib_layers
```

Format

An object of class `data.frame` with 380 rows and 4 columns.

Examples

```
egib_layers
```

---

emuia_download *Download Register of Towns, Streets and Addresses for communes*

Description

Download Register of Towns, Streets and Addresses for communes

Usage

```
emuia_download(commune = NULL, TERYT = NULL, outdir = ".", unzip = TRUE, ...)
```

Arguments

commune	commune name in Polish. Check `commune_names()` function.
TERYT	county ID (7 characters)
outdir	(optional) name of the output directory; by default, files are saved in the working directory
unzip	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
...	additional argument for `utils::download.file()`

Value

a register in SHP format

Examples

```
## Not run:
emuia_download(commune = "Kotla") # 38 KB
emuia_download(TERYT = c("0203042", "2412032")) # 75 KB

## End(Not run)
```

`geocodePL_get` *Convert addresses and objects to geographic coordinates*

Description

Convert addresses and objects to geographic coordinates

Usage

```
geocodePL_get(
  address = NULL,
  road = NULL,
  rail_crossing = NULL,
  geoname = NULL
)
```

Arguments

`address`	place with or without street and house number
`road`	road number with or without mileage
`rail_crossing`	rail crossing identifier (11 characters including 2 spaces, format: "XXX XXX XXX")
`geoname`	name of the geographical object from State Register of Geographical Names (function `geonames_download()`)

Value

a sf data.frame (EPSG: 2180) with metadata

Examples

```
## Not run:
geocodePL_get(address = "Marki") # place
geocodePL_get(address = "Marki, Andersa") # place and street
geocodePL_get(address = "Marki, Andersa 1") # place, street and house number
geocodePL_get(address = "Królewskie Brzeziny 13") # place and house number

geocodePL_get(road = "632") # road number
geocodePL_get(road = "632 55") # road number and mileage

geocodePL_get(rail_crossing = "001 018 478")

geocodePL_get(geoname = "Las Mierzei") # physiographic object

## End(Not run)
```

---

geodb_download *Download General Geographic Databases for entire voivodeships*

Description

Download General Geographic Databases for entire voivodeships

Usage

```
geodb_download(voivodeships, outdir = ".", unzip = TRUE, ...)
```

Arguments

voivodeships	selected voivodeships in Polish or English, or TERC (object voivodeship_names can by helpful)
outdir	(optional) name of the output directory; by default, files are saved in the working directory
unzip	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
...	additional argument for utils::download.file()

Value

a database in Geography Markup Language format (.GML), the content and detail level corresponds to the general geographic map in the scale of 1:250000

References

description of topographical and general geographical databases, and technical standards for making maps (in Polish): https://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU20210001412/O/D20211412.pdf

brief description of categories and layer names (in English and Polish): https://kadyb.github.io/rgugik/articles/articles/spatialdb_description.html

Examples

```
## Not run:
geodb_download(c("opolskie", "lubuskie")) # 12.7 MB
geodb_download(c("Opole", "Lubusz")) # 12.7 MB
geodb_download(c("16", "08")) # 12.7 MB

## End(Not run)
```

`geonames_download` *Download State Register of Geographical Names*

Description

Download State Register of Geographical Names

Usage

```
geonames_download(type, format = "SHP", outdir = ".", unzip = TRUE, ...)
```

Arguments

`type`	names of places ("place") and/or physiographic objects ("object")
`format`	data format ("GML", "SHP" (default) and/or "XLSX")
`outdir`	(optional) name of the output directory; by default, files are saved in the working directory
`unzip`	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
`...`	additional argument for `utils::download.file()`

Value

a selected data type in the specified format

References

https://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU20150000219/O/D20150219.pdf

Examples

```
## Not run:
geonames_download(type = "place", format = "SHP") # 18.2 MB

## End(Not run)
```

minmaxDTM_get *Get minimum and maximum elevation for a given polygon*

Description

Get minimum and maximum elevation for a given polygon

Usage

```
minmaxDTM_get(polygon)
```

Arguments

polygon	the polygon layer with only one object (area less than 10 ha), the larger the polygon area, the lower DTM resolution, the input coordinate system must be EPSG:2180

Value

a data frame with vector points and min/max terrain elevation (EPSG:2180)

Examples

```
## Not run:
library(sf)
polygon_path = system.file("datasets/search_area.gpkg", package = "rgugik")
polygon = read_sf(polygon_path)
minmax = minmaxDTM_get(polygon)

## End(Not run)
```

models3D_download *Download 3D models of buildings for counties*

Description

Download 3D models of buildings for counties

Usage

```
models3D_download(
  county = NULL,
  TERYT = NULL,
  LOD = "LOD1",
  outdir = ".",
  unzip = TRUE,
  ...
)
```

Arguments

county	county name in Polish. Check county_names() function.
TERYT	county ID (4 characters)
LOD	level of detail for building models ("LOD1" or "LOD2"). "LOD1" is default. "LOD2" is only available for ten voivodeships (TERC: "04", "06", "12", "14", "16", "18", "20", "24", "26", "28"). Check voivodeship_names() function.
outdir	(optional) name of the output directory; by default, files are saved in the working directory
unzip	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
...	additional argument for utils::download.file()

Value

models of buildings in Geography Markup Language format (.GML)

Examples

```
## Not run:
models3D_download(TERYT = c("2476", "2264")) # 3.6 MB
models3D_download(county = "sejneński", LOD = "LOD2") # 7.0 MB

## End(Not run)
```

---

ortho_request *Get metadata and links to available orthoimages*

Description

Get metadata and links to available orthoimages

Usage

```
ortho_request(x)

orto_request(x)
```

Arguments

x	an sf, sfc or SpatVector object with one or more features (requests are based on the bounding boxes of the provided features)

Details

The server can return a maximum of 1000 records in a single query. If your area of interest exceeds this limit, you can generate a grid of smaller polygons (sf::st_make_grid()) or a regular grid of points (sf::st_sample()).

Value

a data frame with metadata and links to the orthoimages

See Also

tile_download()

Examples

```
## Not run:
library(sf)
polygon_path = system.file("datasets/search_area.gpkg", package = "rgugik")
polygon = read_sf(polygon_path)
req_df = ortho_request(polygon)

# simple filtering by attributes
req_df = req_df[req_df$composition == "CIR", ]
req_df = req_df[req_df$resolution <= 0.25 & req_df$year >= 2016, ]

## End(Not run)
```

---

parcel_get *Get the geometry of cadastral parcels*

Description

Get the geometry of cadastral parcels

Usage

```
parcel_get(TERYT = NULL, X = NULL, Y = NULL)
```

Arguments

TERYT	parcel ID (18 characters, e.g. "141201_1.0001.6509")
X	longitude (EPSG: 2180)
Y	latitude (EPSG: 2180)

Value

a simple feature geometry (in case of TERYT) or data frame with simple feature geometry and TERYT (in case of coordinates)

Examples

```
## Not run:
parcel = parcel_get(TERYT = "141201_1.0001.6509")
parcel = parcel_get(X = 313380.5, Y = 460166.4)

## End(Not run)
```

pointDTM100_download *Download digital terrain models for voivodeships (100 m resolution)*

Description

Download digital terrain models for voivodeships (100 m resolution)

Usage

```
pointDTM100_download(voivodeships, outdir = ".", unzip = TRUE, ...)
```

Arguments

voivodeships	selected voivodeships in Polish or English, or TERC (function voivodeship_names() can by helpful)
outdir	(optional) name of the output directory; by default, files are saved in the working directory
unzip	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
...	additional argument for utils::download.file()

Value

text files with X, Y, Z columns (EPSG:2180)

Examples

```
## Not run:
pointDTM100_download(c("opolskie", "świętokrzyskie")) # 8.5 MB
pointDTM100_download(c("Opole", "Swietokrzyskie")) # 8.5 MB
pointDTM100_download(c("16", "26")) # 8.5 MB

## End(Not run)
```

pointDTM_get *Get terrain elevation for a given polygon*

Description

Get terrain elevation for a given polygon

Usage

```
pointDTM_get(polygon, distance = 1, print_iter = TRUE)
```

Arguments

polygon	the polygon layer with only one object (its area is limited to the 20 ha * distance parameter), the input coordinate system must be EPSG:2180
distance	distance between points in meters (must be integer and greater than 1)
print_iter	print the current iteration of all (logical, TRUE default)

Value

a data frame with vector points and terrain elevation (EPSG:2180, Vertical Reference System:PL-KRON86-NH)

Examples

```
## Not run:
library(sf)
polygon_path = system.file("datasets/search_area.gpkg", package = "rgugik")
polygon = read_sf(polygon_path)
DTM = pointDTM_get(polygon, distance = 2)

## End(Not run)
```

---

tile_download *Download requested tiles*

Description

Download requested tiles

Usage

```
tile_download(df_req, outdir = ".", unzip = TRUE, print_iter = TRUE, ...)
```

Arguments

df_req	a data frame obtained using the ortho_request() and DEM_request() functions
outdir	(optional) name of the output directory; by default, files are saved in the working directory
unzip	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed; only suitable for certain elevation data
print_iter	print the current iteration of all (logical, TRUE default)
...	additional argument for utils::download.file()

Value

georeferenced tiles with properties (resolution, year, etc.) as specified in the input data frame

Examples

```
## Not run:
library(sf)
options(timeout = 600)
polygon_path = system.file("datasets/search_area.gpkg", package = "rgugik")
polygon = read_sf(polygon_path)

req_df = ortho_request(polygon)
tile_download(req_df[1, ]) # download the first image only

req_df = DEM_request(polygon)
tile_download(req_df[1, ]) # download the first DEM only

## End(Not run)
```

`topodb_download` *Download Topographic Databases for counties*

Description

Download Topographic Databases for counties

Usage

```
topodb_download(county = NULL, TERYT = NULL, outdir = ".", unzip = TRUE, ...)
```

Arguments

`county`	county name in Polish. Check `county_names()` function.
`TERYT`	county ID (4 characters)
`outdir`	(optional) name of the output directory; by default, files are saved in the working directory
`unzip`	TRUE (default) or FALSE, when TRUE the downloaded archive will be extracted and removed
`...`	additional argument for `utils::download.file()`

Value

a database in Geography Markup Language format (.GML), the content and detail level corresponds to the topographic map in the scale of 1:10000

References

description of topographical and general geographical databases, and technical standards for making maps (in Polish): https://isap.sejm.gov.pl/isap.nsf/download.xsp/WDU20210001412/O/D20211412.pdf

brief description of categories and layer names (in English and Polish): https://kadyb.github.io/rgugik/articles/articles/spatialdb_description.html

Examples

```
## Not run:
topodb_download(county = "Świętochłowice") # 2.4 MB
topodb_download(TERYT = c("2476", "2264")) # 4.8 MB

## End(Not run)
```

`voivodeship_names` *Voivodeships in Poland*

Description

The data frame contains Polish and English names of voivodeships, and their identifiers (TERC, 2 characters).

Usage

```
voivodeship_names
```

Format

An object of class `data.frame` with 16 rows and 3 columns.

Examples

```
voivodeship_names
```

Index