Package: imageTCGAutils (via r-universe)

May 30, 2026

Title Utility functions for working with histopathology images

Version 1.0.0

Description Utility functions for working with CONCH data, listing remote files. One function assigns HoverNet nuclei to ProvGigaPath tiles with a scale factor to align coordinates. Provides internal utility functions for 'imageFeatureTCGA' and most functions are not meant for end users.

Depends R (>= 4.5.0)

Imports BiocBaseUtils, data.table, dplyr, grDevices, methods, rlang, S4Vectors, SpatialExperiment, SummarizedExperiment

Suggests anndataR, BiocStyle, imageFeatureTCGA, ggplot2, knitr, paws, rhdf5, rmarkdown, sfdep, spdep, SpatialFeatureExperiment, tinytest

biocViews Software, WorkflowStep, Preprocessing

License Artistic-2.0

BugReports https://github.com/waldronlab/imageTCGAutils/issues

URL https://github.com/waldronlab/imageTCGAutils

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

VignetteBuilder knitr

Config/pak/sysreqs libmagick++-dev gsfonts libicu-dev libssl-dev zlib1g-dev

Repository https://bioc-release.r-universe.dev

Date/Publication 2026-04-28 13:07:00 UTC

RemoteUrl https://github.com/bioc/imageTCGAutils

RemoteRef RELEASE_3_23

RemoteSha 40b6a03d18f466db0c764b7c3d448f0e9412b1e5

Contents

`importCONCH` *Import CONCH as a SpatialFeatureExperiment*

Description

This function reads CONCH output stored in an HDF5 (`.h5`) file and converts it into a `SpatialFeatureExperiment` object for downstream analysis and visualization. The function extracts spatial coordinates and feature embeddings from the file, ensuring proper formatting for the SFE class.

Usage

```
importCONCH(file_path, patch_size = 224)
```

Arguments

`file_path`	`character(1)` Path to the .h5 file containing CONCH output.
`patch_size`	`numeric(1)` The width/height of the patch in pixels (default 224). Used to set the spatial metadata.

Value

A `SpatialFeatureExperiment` class object.

Examples

```
conch_file <- system.file(
    "extdata/mini_tcga_conch.h5",
    package = "imageTCGAutils",
    mustWork = TRUE
)
importCONCH(conch_file)
```

`listFiles` *List available HoVerNet and Prov-Giga-Path data for TCGA cancers*

Description

Functions to list available HoverNet and ProvGiga data for TCGA cancers. HoverNet data is only available for TCGA-OV, while ProvGiga data is available for multiple TCGA cancer types at slide and tile levels. See the `TCGAcodesAvailable` dataset for a summary of available data. These functions return a `data.frame` with filenames and file sizes.

Usage

```
listHoverNet(format = c("geojson", "h5ad", "json", "thumb"), maxkeys = 1000L)

listProvGiga(level = c("slide_level", "tile_level"), maxkeys = 1000L)
```

Arguments

`format`	`character(1L)` One of "geojson", "h5ad", "json", or "thumb" specifying the desired HoverNet data format. Default is "geojson".
`maxkeys`	`integer(1L)` Maximum number of files to return. Default is 1000.
`level`	`character(1L)` One of "slide_level" or "tile_level" specifying the desired ProvGiga data level. Default is "slide_level".

Value

`listHoverNet`,`listProvGiga`: A `tibble` listing available HoverNet or ProvGigaPath files with `Filename`, `Modified`, and `Size` columns.

Examples

```
## List available HoverNet data for TCGA-OV
listHoverNet(format = "geojson", maxkeys = 10)


## List available ProvGiga slide-level data for TCGA-BRCA
listProvGiga(level = "slide_level", maxkeys = 10)
```

`matchHoverNetToTiles` *Match HoverNet Nuclei to ProvGigaPath Tiles*

Description

Assigns HoverNet nuclei to ProvGigaPath tiles by computing a scale factor to align coordinate systems, then performing spatial matching. Returns tile-level cell type counts and dominant cell types.

Usage

```
matchHoverNetToTiles(
  hovernet,
  tiles,
  tile_size = 224,
  cell_x = "x_centroid",
  cell_y = "y_centroid",
  tile_x = "tile_x",
  tile_y = "tile_y",
  tile_id = "tile_id",
  cell_type = "type"
)
```

Arguments

`hovernet`	A `SpatialExperiment` or `SpatialFeatureExperiment` object containing HoverNet nuclei data with spatial coordinates and cell type information.
`tiles`	A `data.frame` or `tibble` containing ProvGigaPath tile data with tile coordinates and embeddings.
`tile_size`	Numeric. Size of tiles in pixels. Default is `224`.
`cell_x`	Character. Name of the x-coordinate column in HoverNet data. Default is `"x_centroid"` for h5ad data.
`cell_y`	Character. Name of the y-coordinate column in HoverNet data. Default is `"y_centroid"` for h5ad data.
`tile_x`	Character. Name of the tile x-coordinate column in tiles data. Default is `"tile_x"`.
`tile_y`	Character. Name of the tile y-coordinate column in tiles data. Default is `"tile_y"`.
`tile_id`	Character. Name of the tile ID column in tiles data. Default is `"tile_id"`.
`cell_type`	Character. Name of the cell type column in HoverNet data. Default is `"type"`.

Details

The function performs the following steps:

1. Computes a scale factor to align nuclei coordinates with tile coordinates
2. Scales nuclei coordinates using the computed scale factor

3. Creates bounding boxes for each tile based on tile_size
4. Assigns nuclei to tiles using spatial overlap
5. Counts cell types per tile
6. Identifies the dominant cell type for each tile
7. Merges results back to the original tiles data

The scale factor is computed as the mean of x and y scale factors, where each is the ratio of coordinate ranges between nuclei and tiles.

Value

A list with three elements:

tiles_with_nuclei A `data.frame` with original tile data plus cell type counts (`N`) and labels (`cell_type_label`) for each tile.

tiles_dominant A `data.frame` with tile IDs and their dominant (most frequent) cell type.

scale_factor A list containing the computed scale factor, and separate x and y scale factors.

Examples

```
library(imageFeatureTCGA)
hov_file <- paste0(
    "https://store.cancerdatasci.org/hovernet/h5ad/",
    "TCGA-23-1021-01Z-00-DX1.F07C221B-D401-47A5-9519-10DE59CA1E9D.h5ad.gz"
)
hn_spe <- HoverNet(hov_file, outClass = "SpatialExperiment") |> import()

tile_prov_url <- paste0(
    "https://store.cancerdatasci.org/provgigapath/tile_level/",
    "TCGA-23-1021-01Z-00-DX1.F07C221B-D401-47A5-9519-10DE59CA1E9D.csv.gz"
)
pg_spe <- ProvGiga(tile_prov_url) |> import()

result         <- matchHoverNetToTiles(hn_spe, pg_spe)
tiles_matched  <- result$tiles_with_nuclei
dominant_types <- result$tiles_dominant
scale_info     <- result$scale_factor
```

Index