Package: Spaniel (via r-universe)

November 9, 2026

Type Package

Title Spatial Transcriptomics Analysis

Version 1.26.0

Description Spaniel includes a series of tools to aid the quality control and analysis of Spatial Transcriptomics data. Spaniel can import data from either the original Spatial Transcriptomics system or 10X Visium technology. The package contains functions to create a SingleCellExperiment Seurat object and provides a method of loading a histologial image into R. The spanielPlot function allows visualisation of metrics contained within the S4 object overlaid onto the image of the tissue.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 4.0)

Imports Seurat, SingleCellExperiment, SummarizedExperiment, dplyr, methods, ggplot2, scater (>= 1.13), scran, igraph, shiny, jpeg, magrittr, utils, S4Vectors, DropletUtils, jsonlite, png

Suggests knitr, rmarkdown, testthat, devtools

VignetteBuilder knitr

biocViews SingleCell, RNASeq, QualityControl, Preprocessing, Normalization, Visualization, Transcriptomics, GeneExpression, Sequencing, Software, DataImport, DataRepresentation, Infrastructure, Coverage, Clustering

Collate 'utilities.R' 'addClusterCols.R' 'addCoordinates.R' 'createObjects.R' 'parseImage.R' 'removeSpots.R' 'spanielPlotInternals.R' 'spanielPlot.R' 'shinySpaniel.R' 'tenX.R'

RoxygenNote 7.1.1

Config/pak/sysreqs cmake libglpk-dev make libicu-dev libjpeg-dev libpng-dev libuv1-dev libxml2-dev libssl-dev zlib1g-dev

Repository Bioconductor 3.23

Date/Publication 2026-04-28 12:50:59 UTC

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

RemoteRef RELEASE_3_23

RemoteSha fc60d1f8be99b3fe9211dcc0d96732aa80655ee1

Contents

addCoordinates 2
createSCE 3
createSeurat 4
createVisiumSCE 5
markClusterCol 5
parseImage 6
removeSpots 7
runShinySpaniel 7
selectSpots 8
spanielPlot 9

Index **11**

`addCoordinates` *Add coordinates to Object Adds output of Spot Detector coordinates to Seurat object or SCE object created by createSeurat/createSCE. Details about how to use Spot Detector can be found: https://github.com/SpatialTranscriptomicsResearch/st_spot_detector*

Description

Add coordinates to Object Adds output of Spot Detector coordinates to Seurat object or SCE object created by createSeurat/createSCE. Details about how to use Spot Detector can be found: https://github.com/SpatialTranscriptomicsResearch/st_spot_detector

Usage

```
addCoordinates(object, coordinatesFile, scaleFactor = NULL)
```

Arguments

`object`	either a Seurat object or SCE
`coordinatesFile`	path to coordinates file exported from Spot Detector
`scaleFactor`	a scaling factor which can be used if the image file has been reduced in size after the coordinates were generated. For example if the image to be used is 10 percent the size of the original factor scaleFactor = 10

Value

object

Examples

```
### load a SingleCellExperiment Object
sceObj <- readRDS(file.path(system.file(package = "Spaniel"),
                        "extdata/sceData.rds"))
### path to coordinates file exported from spot detector
coordinatesFile <-  file.path(system.file(package = "Spaniel"),
                        "spot_positions.tsv")
sceObj <- addCoordinates(sceObj, coordinatesFile)
```

`createSCE` *Create a SingleCellExperiment Object From Spatial Transcriptomics Data*

Description

This function converts a count matrix into a SingleCellExperiment object. The barcodes for each spot are added to the coldata of the SingleCellExperiment object and are used in plotting the data.

Usage

```
createSCE(counts, barcodeFile, projectName=projectName,
                sectionNumber=sectionNo)
```

Arguments

`counts`	Raw count matrix or data frame where each row represents a gene and each column represents barcoded location on a spatial transcriptomics slide. The columns should be named using the spot barcode (eg "GTCCGATATGATTGCCGC")
`barcodeFile`	a tab seperated barcode file supplied by Spatial Trancscriptomics. The file should contains three column: The first column contains the Spatial Transcriptomics barcode, the second and third column equate to the x and y location
`projectName`	The name of the project which is stored in the Seurat Object.
`sectionNumber`	The location of the sample on the slide

Value

A SingleCellExeriment Object

Examples

```
## Data is taken from DOI: 10.1126/science.aaf2403
examplecounts <- readRDS(file.path(system.file(package = "Spaniel"),
                           "extdata/counts.rds"))
exampleBarcodes <- file.path(system.file(package = "Spaniel"),
                           "1000L2_barcodes.txt")
seuratOb <- createSCE(examplecounts,
                      exampleBarcodes,
                      projectName = "TestProj",
                      sectionNumber = 1)
```

---

createSeurat *Create a Seurat Object From Spatial Transcriptomics Data*

Description

This function converts a count matrix into a Seurat object. The barcodes for each spot are added to the metadata of the Seurat object and are used in plotting the data.

Usage

```
createSeurat(counts, barcodeFile, projectName = projectName,
                         sectionNumber = sectionNo)
```

Arguments

counts	Raw count matrix or data frame where each row represents a gene and each column represents barcoded location on a spatial transcriptomics slide. The columns should be named using the spot barcode (eg "GTCCGATATGATTGCCGC")
barcodeFile	a tab seperated barcode file supplied by Spatial Trancscriptomics. The file should contains three column: The first column contains the Spatial Transcriptomics barcode, the second and third column equate to the x and y location
projectName	The name of the project which is stored in the Seurat Object.
sectionNumber	The location of the sample on the slide

Value

A Seurat Object

Examples

```
## Data is taken from DOI: 10.1126/science.aaf2403
examplecounts <- readRDS(file.path(system.file(package = "Spaniel"),
                           "extdata/counts.rds"))
exampleBarcodes <- file.path(system.file(package = "Spaniel"),
                           "1000L2_barcodes.txt")
```

```
SeuratObj <- createSeurat(examplecounts,
                          exampleBarcodes,
                          projectName = "TestProj",
                          sectionNumber = 1
                          )
```

---

createVisiumSCE *createVisiumSCE*

Description

A function to select to import 10X data into an SCE object

Usage

```
createVisiumSCE(tenXDir = "../outs", resolution = "Low")
```

Arguments

tenXDir	The path to Space Ranger outs directory containing spatial directory and filtered_feature_bc_matric
resolution	Resolution of the tissue image to be used for plotting. Can be either "High", or "Low". Default is "Low".

Value

SingleCellExperimentObject

Examples

```
tenXDir <- file.path(system.file(package = "Spaniel"), "extdata/outs")
sce <- createVisiumSCE(tenXDir, resolution = "Low")
```

---

markClusterCol *markClusterCol*

Description

A function to mark the columns containing cluster information in the metadata or colData of a Seurat or SCE object. Columns are marked with "cluster_" prefix.

Usage

```
markClusterCol(object, pattern)
```

Arguments

object	Either a Seurat or SCE object containing clustering information
pattern	pattern indicating which columns contain cluster information

Value

A Seurat or SCE object

Examples

```
sceObj <- readRDS(file.path(system.file(package = "Spaniel"),
                        "extdata/sceData.rds"))
sceObj <- markClusterCol(sceObj, "res")
```

---

parseImage *This function parses a HE image to use as the background for plots*

Description

This function parses a HE image to use as the background for plots

Usage

```
parseImage(imgFile, imgType = "jpg")
```

Arguments

imgFile	Path to the image file
imgType	Type of image options jpg (default), png

Value

A rasterized grob

Examples

```
imgFile <- file.path(system.file(package = "Spaniel"),
                     "extdata/outs/spatial/tissue_lowres_image.png")
img <- parseImage(imgFile, imgType = "png")
```

removeSpots *removeSpots*

Description

A function to filter spots from analysis. It requires selectSpots to be run first.

Usage

```
removeSpots(sObj, pointsToRemove = "points_to_remove.txt")
```

Arguments

sObj	Either a Seurat object (version 3) or a SingleCellExperiment object containing barcode coordinates in the metadata (Seurat) or colData (SingleCellExperiment).
pointsToRemove	path to points to remove file. Default is "points_to_remove.txt"

Value

A filtered Seurat or SingleCellExperiment Object

Examples

```
sceObj <- readRDS(file.path(system.file(package = "Spaniel"),
                        "extdata/sceData.rds"))
toRemove <- file.path(system.file(package = "Spaniel"),
                        "points_to_remove.txt")
sceObj_filtered <- removeSpots(sObj = sceObj, pointsToRemove = toRemove)
```

runShinySpaniel *RunShinySpaniel*

Description

A function to visualise Spatial Transcriptomics. It requires a prepocessed Seurat Object or a SingleCellExperiment object as well as a rasterised image saved as an .rds object. There are 4 plots available in the app showing: a) the number of genes detected per spot, b) the number of reads detected per spot, c) clustering results, d) the gene expression of a selected gene." To view the clustering results the columns of the meta.data or colData containing clustering results must be prefixed with cluster_ . This can be done by using the markClusterCol() function included in Spaniel.

Usage

```
runShinySpaniel()
```

Value

Runs a Shiny App

Examples

```
## mark the columns of metadata/colData that contain clustering
## information see ?markClusterCol for more details#'
sObj <-  readRDS(file.path(system.file(package = "Spaniel"),
                    "extdata/sceData.rds"))

img <- readRDS(file.path(system.file(package = "Spaniel"),
                    "extdata/image.rds"))

## run shinySpaniel (upload data.rds and image.rds in the shiny app)
## Not Run:
# runShinySpaniel()
```

---

selectSpots *selectSpots*

Description

A function to select spots to remove from analysis

Usage

```
selectSpots(sObj, imgObj)
```

Arguments

sObj	Either a Seurat object (version 3) or a SingleCellExperiment object containing barcode coordinates in the metadata (Seurat) or colData (SingleCellExperiment).
imgObj	a ggplot grob (see parseImage function)

Value

Runs a shiny application

Examples

```
## Run the shiny app (Not run):
# selectSpots(sObj, imgObj)

# Click on the spots to remove from downstream analysis. Once all the spots
# have been selected close the shiny app window. A list of spots is
# stored in a text file called points_to_remove.txt in the working directory.
```

```
# Once this step has been run a filtered Seurat or SCE object can be
# created using removeSpots (see removeSpots for more details)
```

---

spanielPlot *Spatial Transcriptomics Plot*

Description

This function overlays information from a Seurat object or SingleCellExperiment object containing barcodes onto a H & E image. There are 4 plots available showing a) the number of genes detected per spot, b) the number of reads detected per spot, c) clustering results, d) the gene expression of a selected gene.

Usage

```
spanielPlot(object,
grob = NULL,
techType =
"Original",
byCoord = FALSE,
imgDims = NULL,
plotType = c("NoGenes", "CountsPerSpot", "Cluster", "Gene"),
gene= NULL,
clusterRes = NULL,
customTitle = NULL,
scaleData = TRUE,
showFilter = NULL,
ptSize = 2,
ptSizeMin = 0,
ptSizeMax = 5)
```

Arguments

object	Either a Seurat object (version 3) or a SingleCellExperiment object containing barcode coordinates in the metadata (Seurat) or colData (SingleCellExperiment).
grob	an grob to be used as the backgound image see(parseImage). This is used for original Spatial Transcriptomics objects but not Visium
techType	Either 1) "Original" (default) for the original Spatial Transcriptomics slides where the image has been cropped to the edge of the spots 2) "Visium" for 10X slides.
byCoord	TRUE/FALSE option to plot original Spatial Transcriptomics data using pixel coordinates instead of by spot coordinates. Not required if techType = "Visium". Default is FALSE.

imgDims	pixel dimensions of histological image. Required when byCoord parameter is set to TRUE, Not required if techType = "Visium".
plotType	There are 5 types of plots avaiable: 1) NoGenes - This shows the number of genes per spot and uses information from "nFeature_RNA" column of Seurat object or "detected" from a SingleCellExperiment object. 2) CountsPerSpot - This shows the number of counts per spot. It uses information from "nCount_RNA" column of Seurat object or "sum" from a singleCellExperiment object. 3) Cluster - This plot is designed to show clustering results stored in the meta.data or colData of an object 4) Gene- This plot shows the expression of a single gene. This plot uses scaled/normalised expressin data from the scale.data slot of Seurat object or logcounts of a SingleCellExperiment object. 5) Other - A generic plot to plot any column from the meta.data or colData of an object.
gene	Gene to plot
clusterRes	which cluster resolution to plot
customTitle	Specify plot title (optional)
scaleData	Show scaled data on plot (default is TRUE)
showFilter	Logical filter showing pass/fail for spots
ptSize	Point size used for cluster plot default is 2
ptSizeMin	Minimum point size used for QC and Gene Expression plots default is 0
ptSizeMax	Maximum point size used for QC and Gene Expression plots default is 5

Value

A ggplot spatial transcriptomics plot

Examples

```
pathToTenXOuts <- file.path(system.file(package = "Spaniel"), "extdata/outs")
sceObj <- createVisiumSCE(tenXDir=pathToTenXOuts,
                      resolution="Low")
filter <- sceObj$detected > 0
spanielPlot(object = sceObj,
plotType = "NoGenes",
showFilter = filter,
techType = "Visium",
ptSizeMax = 3)
```

Index

addCoordinates, 2

createSCE, 3
createSeurat, 4
createVisiumSCE, 5

markClusterCol, 5

parseImage, 6

removeSpots, 7
runShinySpaniel, 7

selectSpots, 8
spanielPlot, 9