

# An example of the STIE single-cell level deconvolution and clustering

Shijia Zhu ([shijia.zhu@UTSouthwestern.edu](mailto:shijia.zhu@UTSouthwestern.edu))

2023-02-26

STIE aligns the spatial transcriptome data to the matched histology image-based nucleus segmentation, thereby enabling the real single-cell level and whole-slide scale deconvolution, convolution and clustering for both low- and high-resolution spots. Here, we used the 10X CytAssist mouse brain hippocampus as an example to demonstrate the STIE deconvolution and clustering at the single-cell level with and without cell-type signature. The raw data can be found from the 10X public database: [section 1](#) and [section 2](#).

Please refer to the wiki<sup>1</sup> and the nucleus segmentation<sup>2</sup> for the full tutorial.

Contents:

- STIE dependent packages
- STIE input
- Single-cell level deconvolution in Spatial transcriptomics
- Single-cell level clustering in Spatial transcriptomics
- Grid search for  $\lambda$  and nuclear morphological features

## 1. Load STIE dependent tools and R packages

Here, we only import the dependent packages for STIE deconvolution and clustering, but did not import the other packages for reading raw data, plotting image, and examing cell-cell interaction.

```
#### for the STIE package
library("STIE")

#### for the quadratic programming
library("quadprog")
#> Warning: package 'quadprog' was built under R version 3.5.2

#### for manipulating ST gene expression
# library("Seurat")

#### for manipulating images
# library("magick", "EBImage")

#### for spatially resolved cell-cell interaction
# library("CellChat", "NMF", "ggalluvial")
```

## 2. load the ST data for the 10X CytAssist mouse brain hippocampus FFPE

STIE takes the follows as input:

- nuclear coordinates and nuclear morphology the spatial coordinates and morphological features of the nuclei
- spot-level gene expression the gene expression on spots
- cell-type transcriptomic signature the cell-type transcriptomic signature derived from scRNA-seq data

```
STIE.dir = system.file(package = "STIE")
nn = load(
  paste0(STIE.dir, "/data/MouseBrainHippocampus_10xV2ChemistryCytAssistFFPE_section1n2.Rdata") )
nn
#> [1] "ST_expr" "spot_coordinates" "cell_contour"
#> [4] "morphology_fts" "cells_on_spot" "ST_expr_s2"
#> [7] "spot_coordinates_s2" "cell_contour_s2" "morphology_fts_s2"
#> [10] "cells_on_spot_s2" "Signature"

# spot coordiantes
head(spot_coordinates,3)
#> barcode tissue row col imagerow imagecol pixel_x pixel_y
#> 1492 CTGGTGATCGCCGTAG-1 1 23 39 12468 21839 21839 12468
#> 1493 CACTCAATAAGCCGAA-1 1 23 41 12469 21474 21474 12469
#> 1494 GCTATGCGTGAACGGT-1 1 23 43 12469 21109 21109 12469

# spot-level gene expression
head(ST_expr[,1:10],3)
#> Xkr4 Rp1 Sox17 Lyplal Tcea1 Rgs20 Atp6v1h Oprk1 Npbwr1
#> CTGGTGATCGCCGTAG-1 5 0 0 3 12 0 11 0 0
#> CACTCAATAAGCCGAA-1 3 0 0 5 9 0 8 0 0
#> GCTATGCGTGAACGGT-1 0 0 0 0 5 1 12 0 0
#> Rbicc1
#> CTGGTGATCGCCGTAG-1 8
#> CACTCAATAAGCCGAA-1 3
#> GCTATGCGTGAACGGT-1 6

# nuclear coordinates and nuclear morphology
head(morphology_fts,3)
#> cell_id X.1 Area Mean StdDev Mode Min Max X Y XM YM
#> 1 1 1 0.210 192.070 24.514 212 49 232 13923.68 13680.93 20.031 17.501
#> 2 2 2 0.151 205.810 20.052 215 96 235 13637.18 14721.14 17.048 28.339
#> 3 3 3 0.107 193.578 33.204 202 44 246 14836.67 13422.50 29.543 14.809
#> Perim. BX BY Width Height Major Minor Angle Circ. Feret IntDen
#> 1 1.637 19.760 17.250 0.542 0.510 0.535 0.499 153.749 0.984 0.548 40.286
#> 2 1.395 16.812 28.125 0.469 0.427 0.466 0.412 156.331 0.975 0.473 31.064
#> 3 1.167 29.354 14.625 0.375 0.365 0.383 0.355 23.470 0.985 0.388 20.647
#> Median Skew Kurt X.Area RawIntDen Ch FeretX FeretY FeretAngle MinFeret
#> 1 199 -1.878 5.794 100 371272 1 1901.50 1665.13 140.889 0.493
#> 2 211 -2.126 5.999 100 286282 1 1618.11 2707.54 146.240 0.410
#> 3 202 -2.176 5.641 100 190287 1 2820.28 1432.34 39.136 0.353
#> AR Round Solidity Eccentricity pixel_x pixel_y size shape
#> 1 1.073 0.932 1.003 0.3606266 13923.68 13680.93 -7.229743 -1.430093
#> 2 1.132 0.883 1.001 0.4672596 13637.18 14721.14 -4.870871 -1.125815
#> 3 1.077 0.928 1.005 0.3753258 14836.67 13422.50 -2.706045 -1.419259
#> angle
#> 1 -1.568037
#> 2 -1.677228
#> 3 1.602321

# cell-type transcriptomic signature
head(Signature,3)
#> CA1 CA2 CA3 DG GABAergic Glia
#> Grm2 1.96 0.18 0.30 6.39 0.48 0.64
#> St3gal1 0.81 0.18 0.13 2.98 0.53 0.24
#> Clql3 2.05 0.00 0.44 6.66 0.22 0.90
```

## 3. STIE deconvolution at single-cell level

Load mouse brain hippocampus scRNA-seq-derived cell type transcriptomic signatures. We run STIE deconvolution by setting known\_signature=TRUE and known\_cell\_types=FALSE. We set lambda=0 and steps=30 for EM algorithm iteration.

```
#### Users could define the morphological features based on specific prior knowledge
#### In this example, we use "shape" as the morphological feature
features = c("shape")

#### run STIE deconvolution
system.time( result_deconv <- STIE(ST_expr, Signature, cells_on_spot, features,
  lambda=0, steps=30, known_signature=TRUE,
  known_cell_types=FALSE))
#> user system elapsed
#> 13.349 0.223 13.585

names(result_deconv)
#> [1] "lambda" "mu" "sigma" "PE_on_spot"
#> [5] "PM_on_cell" "PME_uni_cell" "cell_types" "uni_cell_types"
#> [9] "Signature" "cells_on_spot"
```

We use the plot\_sub\_image() function to overlay the single cells along with their cell types onto the image. Please run ?plot\_sub\_image to check the useful visualization parameters.

The high-res images are downloaded from the 10X public database: [CytAssist\\_FFPE\\_Mouse\\_Brain\\_Rep1\\_tissue\\_image.tif](#) for section 1 and [CytAssist\\_FFPE\\_Mouse\\_Brain\\_Rep2\\_tissue\\_image.tif](#) for section 2.

```
# library(EBImage)
# library(magick)

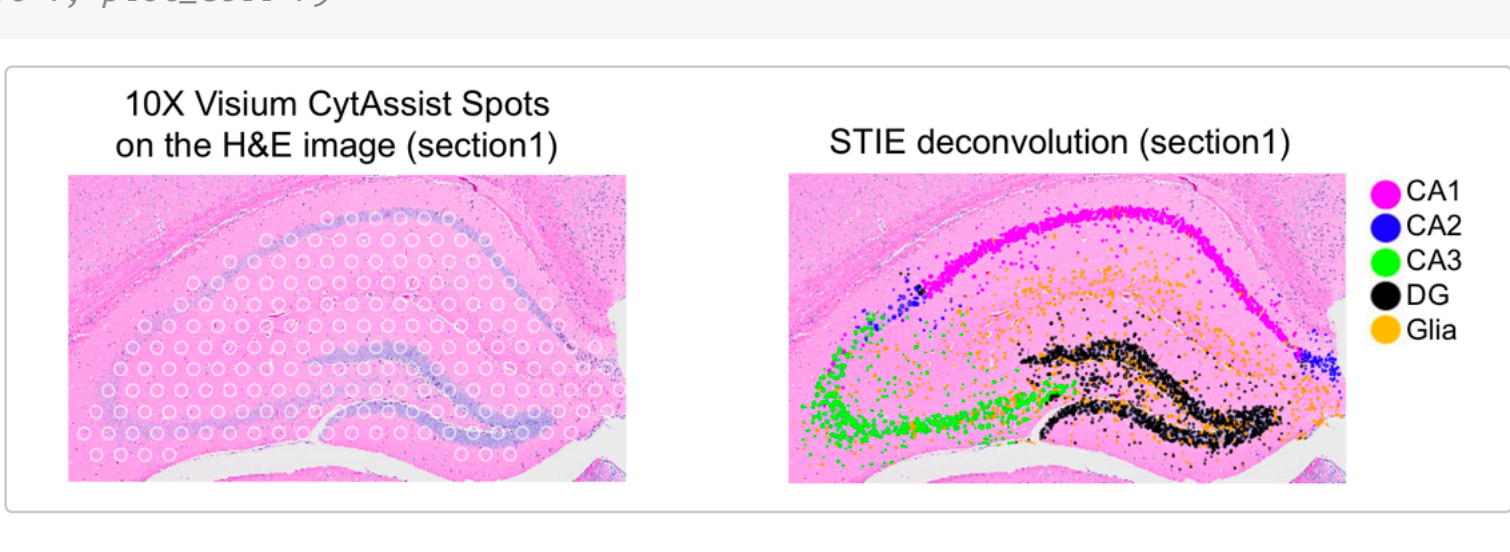
#### read image
# image_path = 'CytAssist_FFPE_Mouse_Brain_Rep1_tissue_image.tif'
# im <- image_read(image_path)[1]

#### the STIE-obtained cell types
# cell_types = result_deconv$cell_types

#### subset the cell contour
# contour = cell_contour[ match(names(cell_types), names(cell_contour)) ]

#### plot the single cells along with their cell types onto the image
# colors = c("magenta", "blue", "green", "black", "orange")

# plot spot coordinates
# plot_sub_image(im=im, w=9000, h=5000, xoff=15500, yoff=11500, x_scale=0.2,
# spot_coordinates=spot_coordinates, contour=contour, cell_types=cell_types, color_use=colors,
# plot_spot=T, plot_cell=F)
```



STIE deconvolution on the mouse brain hippocampus

## 4. STIE clustering at single-cell level

Given no cell type transcriptomic signature, STIE can perform cell type clustering at the single-cell level, and meanwhile, estimate the gene expression signature for clusters. The initial values of clusters are first given using the spot-level clustering, e.g., K-means, Louvain clustering, or SpaGCN, the cells within the spot are assigned the same initial cluster, and the initial value of cluster signature was set to be the average gene expression of spots belonging to the cluster. In each iteration, the cluster signature was re-estimated in the M-step, and the cluster of each single cell was re-assigned in the E-step. In the following example, we take the spot-level cluster at k=5 as the initial value and run STIE with by setting known\_signature and known\_cell\_types as FALSE.

```
#### choose Kmeans (k=5) on spot-level gene expression
pc = prcomp(ST_expr)$X[,1:10]
set.seed(1234)
cluster = kmeans(pc,5)$cluster
cluster = data.frame( Barcode=names(cluster), Cluster=cluster )
cluster = cluster[ match( as.character(spot_coordinates$barcode),
  as.character(cluster$Barcode)), ]
head(cluster)
#> barcode Cluster
#> CTGGTGATCGCCGTAG-1 CTGGTGATCGCCGTAG-1 4
#> CACTCAATAAGCCGAA-1 CACTCAATAAGCCGAA-1 4
#> GCTATGCGTGAACGGT-1 GCTATGCGTGAACGGT-1 4
#> CAGTTGCTCAGGTGTC-1 CAGTTGCTCAGGTGTC-1 4
#> TCAGTATGTAGGACAA-1 TCAGTATGTAGGACAA-1 4
#> ACCAAGTGATGTGAG-1 ACCAAGTGATGTGAG-1 4

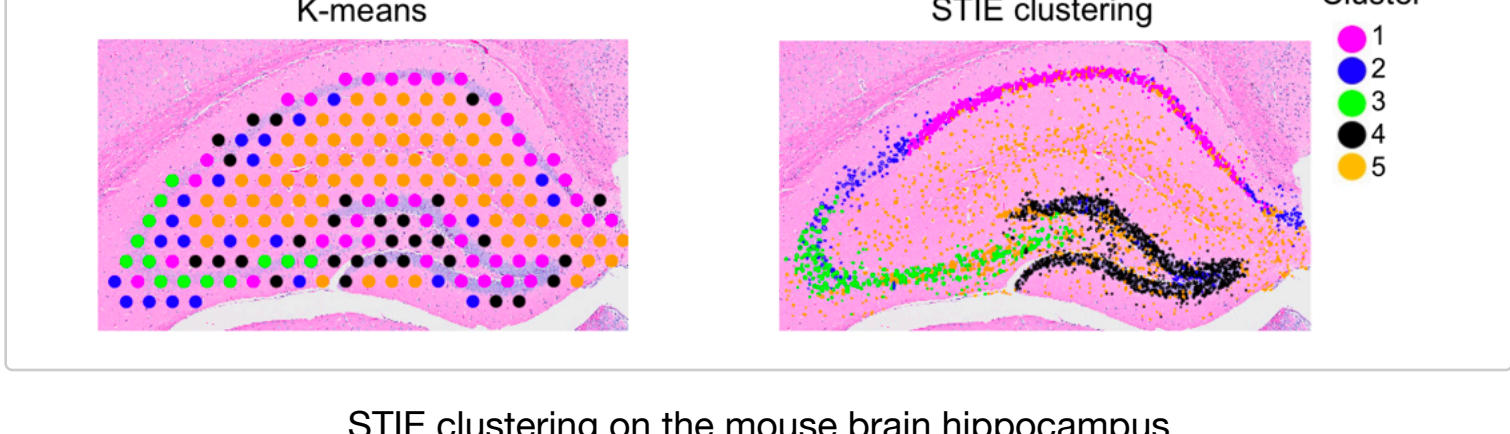
#### take the cluster average gene expression as the initial value of the cluster signature
ST_expr_ini = ST_expr[ match(as.character(cluster[,1]),rownames(ST_expr)), ]
Signature_ini = t(apply(ST_expr_ini, 2, function(x) tapply(x,cluster[,2],mean) ))

#### run STIE using "Signature_ini" as an initial value and iteratively refine "Signature"
#### by setting both "known_signature=FALSE" and "known_cell_types=FALSE"
system.time( result_cluster <- STIE(ST_expr, Signature_ini, cells_on_spot, features,
  lambda=0, steps=30,
  known_signature=FALSE, known_cell_types=FALSE))
#> user system elapsed
#> -69.035 5.611 74.712
```

We use the following codes to visualize the results of K-means and STIE clustering.

```
#### plot the Kmeans clustering at the spot level
# colors2 = c("green", "black", "magenta", "orange", "blue")
# spot_cols = colors2[ cluster$Cluster ]
# plot_sub_image(im=im, w=9000, h=5000, xoff=15500, yoff=11500, x_scale=0.2,
# spot_coordinates=spot_coordinates, plot_spot=T, plot_cell=F, spot_cols=spot_cols, fill_spot=T
# )

#### plot the STIE clustering at the single-cell level
#### the STIE-obtained cell types
# cell_types = result_cluster$cell_types
#### subset the cell contour
# contour2 = cell_contour[ match(names(cell_types), names(cell_contour)) ]
#### plot the single cells along with their cell types onto the image
# plot_sub_image(im=im, w=9000, h=5000, xoff=15500, yoff=11500, x_scale=0.2,
# spot_coordinates=spot_coordinates, contour=contour2, cell_types=cell_types, color_use=colors2,
# plot_spot=F, plot_cell=T)
```



STIE clustering on the mouse brain hippocampus

## 5. Grid search for $\lambda$ and nuclear morphological features

The hyperparameter  $\lambda$  balances the information from gene expression and morphological features.

- Given the predefined morphological features, we select  $\lambda$  by evaluating the balance between two criteria: RMSE of the spatial gene expression fitting and log-likelihood of the morphological feature fitting.
- To select the morphological features, under a predefined  $\lambda$ , we rank the nuclear morphological features based on the RMSE by running STIE on each feature individually. Next, based on their ranking, we used a greedy strategy to gradually add more features to the model. Like the selection of  $\lambda$ , the best morphological features are selected by evaluating the gene expression and morphological fittings simultaneously.
- To select the best combination of morphological features and  $\lambda$ , we investigated the morphological features over different  $\lambda$ .

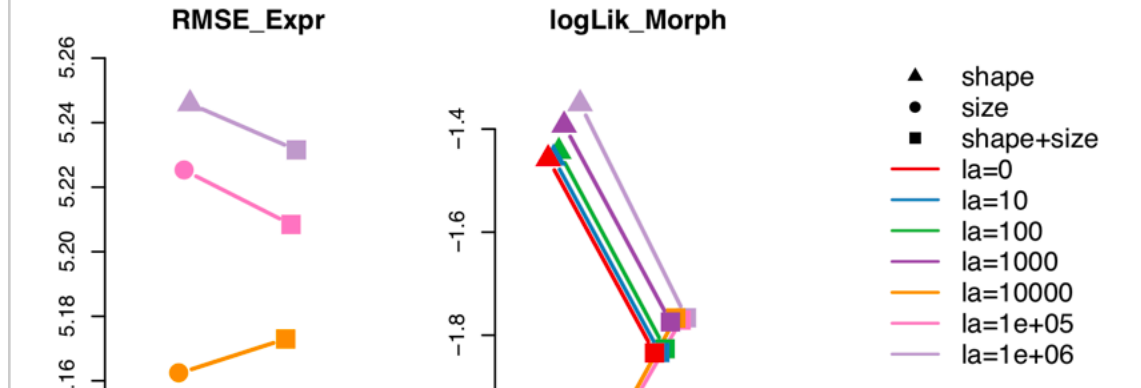
In the real datasets, we extracted multiple morphological features for the nuclei. These features are found to be highly correlated, which is consistent with their definition and mathematical calculation. We focused on two large categories: size (Area, Major, Minor, Width, Height, Feret, and Perimeter) and shape (Round and Circular). To reduce the redundancy and improve efficiency, we performed PCA for each category and took the 1st PC as the surrogate of each category.

We select the best combination of the morphological feature and  $\lambda$  simultaneously via the grid search, which is implemented using the R function STIE\_search(). In the mouse brain hippocampus deconvolution and clustering, the feature 'shape' was ranked, and 'shape' gives the lower RMSE and higher morphological likelihood at small  $\lambda$ , so we used 'shape' as the morphological feature and  $\lambda=0$  (red triangle).

```
# searching range
lambdas <- c(0,1e1,1e2,1e3,1e4,1e5,1e6)

# set RMSE as the criterion to rank the morphological features
# grid search for STIE deconvolution
# paths <- STIE_search(ST_expr, Signature, cells_on_spot, steps=30, known_signature=TRUE,
# known_cell_types=FALSE, lambdas=lambdas, criterion = "rmse" )

# names(paths)
# names(paths)[["0"]]
# paths[["0"]]$ordered_features
```



STIE search for the deconvolution

1. <https://github.com/zhushijia/STIE/wiki>
2. <https://github.com/zhushijia/STIE/wiki/Nucleus-segmentation>