

Stereo-seq

ANALYSIS WORKFLOW

Q&A



TABLE OF CONTENTS

CHAPTER 1: COMMON QUESTIONS RELATED TO Stereo-seq BIOINFORMATICS ANALYSIS	1
CHAPTER 2: IMAGE PROCESSING	9
CHAPTER 3: SAW (Stereo-seq Analysis Workflow)	18
CHAPTER 4: SAW RESULT INTERPRETATION	21
CHAPTER 5: VISUALIZATION	23

CHAPTER 1

COMMON QUESTIONS

1.1. What is the relationship between the sizes of a single biological cell and a square bin spot?

Given that the diameter of a typical mammalian cell is approximately 10µm, it is analogous to a bin20 spot that is 10µm x 10µm in area or a bin14 spot with a diagonal of 10µm.

1.2. How to choose appropriate bin sizes when analyzing the data?

Some information, such as cell sizes of specific tissue types, can be used. It is recommended to vary the bin level repeatedly based on the results of downstream analyses, with a spectrum of bin20, 50, 100, and 200. Bin20 is about the size of a regular mammalian cell, while bin50 and bin100 are both frequently adopted in the analysis. And bin200 is generally used for immediate visualization of SAW outputs.

1.3. How to deal with the situation of abnormal valid CID ratio?

There are three directions in which investigation can be carried out.

1. Sequencing quality. Low sequencing quality can affect alignment results. In addition to Q30, the presence of unknown base calls needs to be considered as well, which can be examined by reviewing base distribution in the sequencing report. If the proportion of N bases is high, it needs to be considered that sequencing problems have affected the valid CID ratio. It is recommended to prioritize such inspection.

2. The chip mask h5 file does not correspond to the FASTQ datasets. Because the CID recorded in the mask does not match the CID obtained by sequencing the sample, the valid CID ratio is low. If this situation occurs alone, the proportion is usually extremely low. If the next situation is also involved, the variation would be of significance, requiring a case by case analysis.

3. (Cross) Contamination. It occurred when other samples got mixed in during the experiment, library preparation, or sequencing, which affected the valid CID ratio because of being contaminated. Here comes a likelihood that two chips can be both mapped to the sequencing data of the same library. If there is a lot of mixing, a distinct tissue pattern should be visible. If the proportion is extremely small, in some cases there will be some local bright spots.

1.4. What is the version compatibility between ImageQC/ImageStudio, SAW and StereoMap?

ImageStudio	ImageStudio description	SAW	SAW description	StereoMap	StereoMap description
≤ 1.0.8	File format: .json + .tar.gz Features: ssDNA image QC	≤ v4.1.0	Support ssDNA image registration and tissue segmentation		
≥ 1.1.0	File format: .ipr + .tar.gz Features: ssDNA image QC	≥ v5.1.3	Support cell segmentation on ssDNA image; enable analysis of FASTQ data in Q4 format		

ImageStudio	ImageStudio description	SAW	SAW description	StereoMap	StereoMap description
1.0.0	File format: .ipr + .tar.gz Features: ssDNA image QC and manual processing	≥ v5.5.0	Support cell segmentation on ssDNA image; enable analysis of FASTQ data in Q4 format	1.0.0	Support displaying spatial expression heatmap, co-visualization of gene distribution, and ssDNA image. Manual registration enabled
2.0.0	File format: .ipr + .tar.gz Features: Image QC for ssDNA, DAPI, mIF stains and manual processing	≥ v6.0.0	Support mIF image registration; allow for rRNA filtering	2.0.0	Display of individual mIF images and the ones stacked with different image layers
2.1	File format: .ipr + .tar.gz Features: Image QC for ssDNA, DAPI, mIF stains and their manual image processing; Fully manual procedure for QC-failed images	< 7.0	Support analysis with the results of the manual register procedure done by a combination of ImageStudio and StereoMap	< 3.0	Support reading multiple gef files at a time, which will be displayed by individual tabs
2.2	File format: .ipr + .tar.gz Features: Image QC for ssDNA, DAPI, mIF stains and their manual image processing; fully manual procedure for QC failed images	≥6.1 <7.0	Support analysis with the results of fully manual procedure done by ImageStudio	2.1 < 3.0	Support reading multiple gef files at a time, which will be displayed by individual tabs
3.0	File format: .ipr + .tar.gz File format: .ipr + .tar.gz Features: Image QC for ssDNA, DAPI, H&E, mIF stains and their manual image processing; fully manual procedure for QC failed images	7.0	Reconstructed 'count' go online; 'register' reconstructed with new tissue segmentation algorithm and new 'V03' cell segmentation algorithm; Support H&E whole process; Support cell correction using EDM algorithm based on mask file of cell segmentation result	≤ 3.0	Support reading h5ad files with different binsize/ resolution; /codedCellBlock information is written into cgef file after the SAW cellChunk module; Render cellbin heatmap while loading cgef files

Reference & GTF/GFF Annotation Files

1.5. What are requirements for genomic annotation files including GTF/GFF formats?

1. File format:

GFF files or GTF files, supporting `gtf/gtf.gz`, `gff/gff.gz`, `gff3/gff3.gz` as file suffix names.

2. GTF file format:

- Comment lines begin with #
- The main body has 9 columns, separated by 'tab': `seqname source feature start end score strand frame attributes`
 1. `type`: types of annotation information **must** contain `gene`, `transcript` **and** `exon`
 2. `start/end`: need to be less than 2^{31}
 3. `strand`: forward and reverse of strands, represented as + and -, respectively
 4. `attributes` as the 9th column, whose format is `tag "value"`, with different attributes separated by **space**; of which **the following four are required**.
 - a. `gene_name value`
 - b. `gene_id value`: represents the unique ID of a transcript for the given gene loci of the genome. 'gene_id' and 'value' are separated by space. If the value is empty, it means that there is no corresponding gene.
 - c. `transcript_name value`
 - d. `transcript_id value`: a unique ID to identify a transcript. Empty value means no transcript.
- At present, the maximum valid gene number must be less than 2^{20} , that is 1048576
- **Do not disrupt order**. The same gene's transcript/exons need to be arranged in order

3. GTF file format:

- Comment lines begin with #
- The main body has 9 columns, separated by 'tab': `seqid source type start end score strand phase attributes`
 1. `type`: types of annotation information **must** contain `gene`, `mRNA` **and** `exon`
 2. `start/end`: max of them need to be less than 2^{31}
 3. `strand`: "+" stands for forward strands, "-" stands for reverse strands, "." indicates there is no need to specify positive or negative strands, "?" means unknown
 4. `attributes` as the 9th column, whose format is `tag=value`, with different attributes separated by **semicolon**
 - a. `ID Name Parent` must provide (Parent is not required for each gene)
 - b. **For naming rules of the 3rd column, please carefully check on-->** "dendrarchy" (tree-shaped hierarchy) (do not list 'child' rows without '**parent**' rows!) An example is shown as follows:

```

1 ##sequence-region ctg123 1 1497228
2 ctg123 . gene 1000 9000 . + . ID=gene00001;Name=EDEN
3 ctg123 . TF_binding_site 1000 1012 . + . ID=tfbs00001;Parent=gene00001
4 ctg123 . mRNA 1050 9000 . + . ID=mRNA00001;Parent=gene00001;Name=EDEN.1
5 ctg123 . mRNA 1050 9000 . + . ID=mRNA00002;Parent=gene00001;Name=EDEN.2
6 ctg123 . mRNA 1300 9000 . + . ID=mRNA00003;Parent=gene00001;Name=EDEN.3
7 ctg123 . exon 1300 1500 . + . ID=exon00001;Parent=mRNA00003
8 ctg123 . exon 1050 1500 . + . ID=exon00002;Parent=mRNA00001,mRNA00002
9 ctg123 . exon 3000 3902 . + . ID=exon00003;Parent=mRNA00001,mRNA00003
10 ctg123 . exon 5000 5500 . + . ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11 ctg123 . exon 7000 9000 . + . ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12 ctg123 . CDS 1201 1500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13 ctg123 . CDS 3000 3902 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14 ctg123 . CDS 5000 5500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15 ctg123 . CDS 7000 7600 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16 ctg123 . CDS 1201 1500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17 ctg123 . CDS 5000 5500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18 ctg123 . CDS 7000 7600 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19 ctg123 . CDS 3301 3902 . + 0 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20 ctg123 . CDS 5000 5500 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21 ctg123 . CDS 7000 7600 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22 ctg123 . CDS 3391 3902 . + 0 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23 ctg123 . CDS 5000 5500 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24 ctg123 . CDS 7000 7600 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4

```

- At present, the maximum valid gene number must be less than 2^{20} , that is 1048576
- Although ordering is not required, the rules that **'gene' must appear ahead of corresponding mRNA, and mRNA must appear ahead of corresponding exon** still need to be met.

4. Others to note:

- gene/gene_name **should not contain any special symbols** (space, all types of brackets, quotation marks, <>, %, etc.) **other than** common symbols such as “_” and “.”
- gene/gene_name **shorter than 64 characters**
- Although the mainly used GFF files are version 3 (GFF3), please name them as .gff; likewise, please name GTF files as .gtf

1.6. What are the situations where corresponding genes are omitted while reading annotation files?

- There are no attributes of gene_name gene_id transcript_name transcript_id in gtf format (only gene_name and gene_id are needed for each gene)
- There are no attributes of ID Name Parent in gff format (Parent is not needed for gene entities)
- Multiple gene IDs are assigned to the same gene, as printed by log **"Multiple gene IDs for gene xxx: id1, id2..."**
- Both forward and reverse strands are assigned to the same gene, as printed by log **"Strand disagreement for gene xxx - skipping"**
- No transcript_id for transcript/exon, as printed by log **"Record does not have transcriptID for gene xxx"**
- If a gene has multiple transcripts and the same transcript_id / ID, as printed by log **"Transcript appears more than once for xxx"**

- start > end for some exons, as printed by log **"Exon has 0 or negative extent for xxx"**
- There is overlap between exons of the same transcript, as printed by log **"Exons overlap for xxx"**
- A gene has no transcript present, as printed by log **"No transcript for gene xxx"**
ps: One contig with multiple genes sharing the same gene_name will merge them into one.

1.7. How to deal with the error reporting "Fatal INPUT FILE error, no valid exon lines in the GTF file" during reference genome indexing

One possibility is that GTF/GFF annotation files are not completely consistent with genome FASTA files in terms of chromosome naming. Please keep the chromosome name unified.

1.8. Why are most genes in the annotation file not annotated?

It is possible that the input annotation file does not conform to the norms. Please double check according to the file format requirements mentioned above.

Another possibility is that the forward/reverse symbols of the strand are not in the right format. Strand values in annotation files should only be either "+" (forward) or "-" (reverse), do not confuse "-" (hyphen) with "_" (underline)

If there are direction inconsistencies with genes that have same name and come from the same chromosome, the annotation file will be regarded as abnormal, and all genes of this kind will be discarded.

1.9. Is there any helpful tool for checking for errors in annotation files?

The checkGTF tool of SAW sif has been developed for such a purpose. The execution commands are shown as follows:

```
## export SINGULARITY_BIND="/path/to/input/dir,/path/to/output/dir"
singularity exec SAW.sif checkGTF \
  -i <input.gtf/gff> \  ## GTF/GFF file input to be checked
  -o <output.gtf/gff>  ## [optional]. Set to output revised GTF/GFF file. Be
aware that this may remove some genes which do not meet the requirements and can-
not be fixed.
```

Gene annotation records that can not be fixed by the program will be removed from the output. But these records will be written into the log file. Please rectify the incorrect items and run the program again.

1.10. How to remove rRNA alignments during analysis? Can rRNA sequences that need to be removed be specified manually?

It is allowed to manually add rRNA sequences to the reference genome FASTA file, followed by rebuilding reference indices. With rRNARemove switch on, SAW mapping will filter out the reads that are mapped to rRNA sequences. rRNA filtering function is recently added in SAW v6.0.

Rules to add rRNA sequence: include rRNA sequences to filter out in the FASTA file, and append '_rRNA' at the end of the usual sequence name starting with ">", for program identification. Examples are as follows:

- Using the different lengths in genome indexing can specify the range of adjacent genome sequences around the annotated splicing sites, which is the maximum length used for identification of alternative transcript splicing.
- The impact of using different lengths is limited to very few splicing sites that are susceptible to variation of several bases in the alignment of the regions around boundaries of exons/introns. The difference is marginal that there is no need to have a distinct index for each read length.

Reference material:

[Accuracy of --sjdbOverhang parameter · Issue #1486 · alexdobin/STAR](#)

[STARmanual.pdf](#)

CHAPTER 2

IMAGE PROCESSING

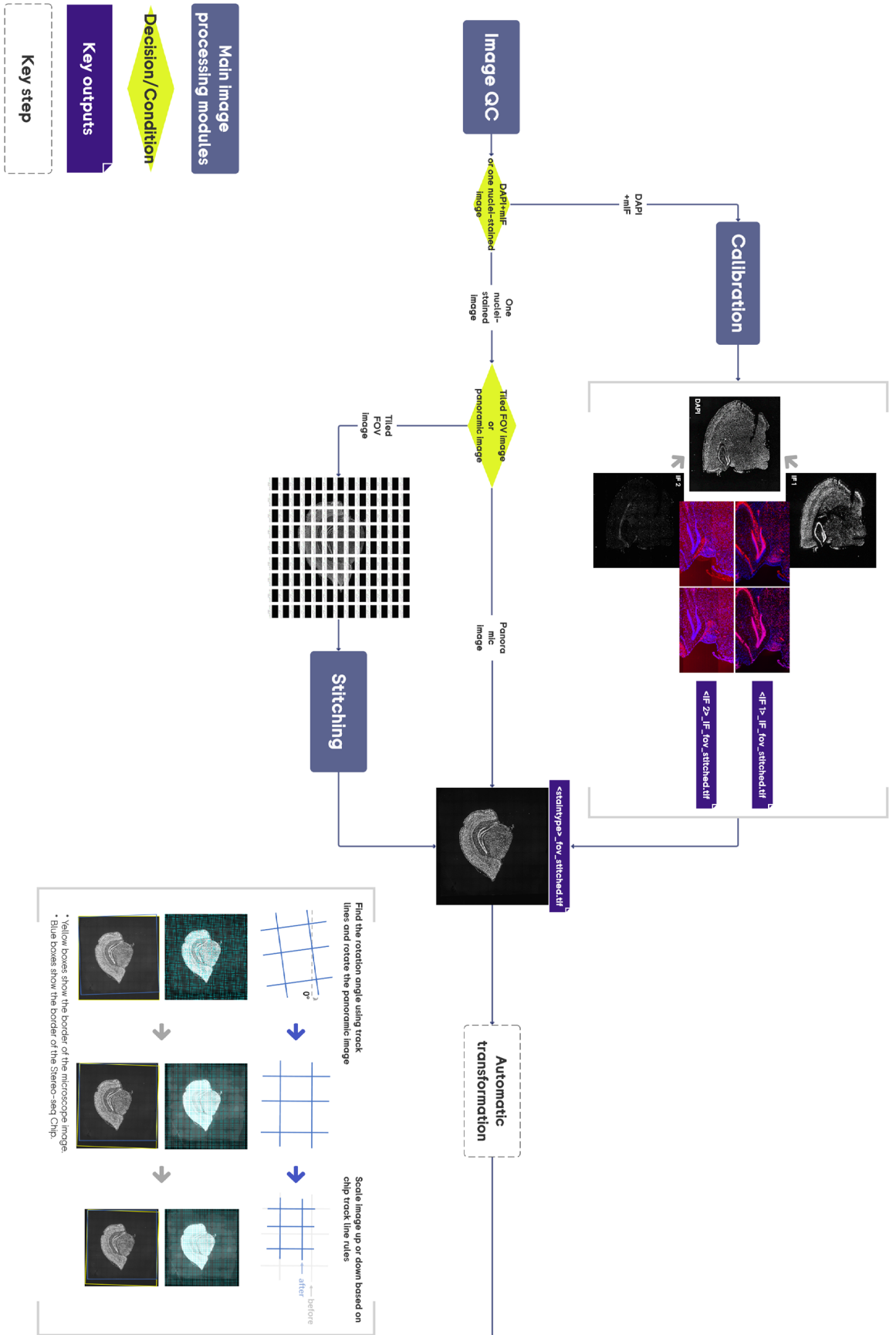
Whole procedure of image processing

2.1. What does the overall image processing procedure of Stereo-seq analysis look like?

Auto process

ImageStudio

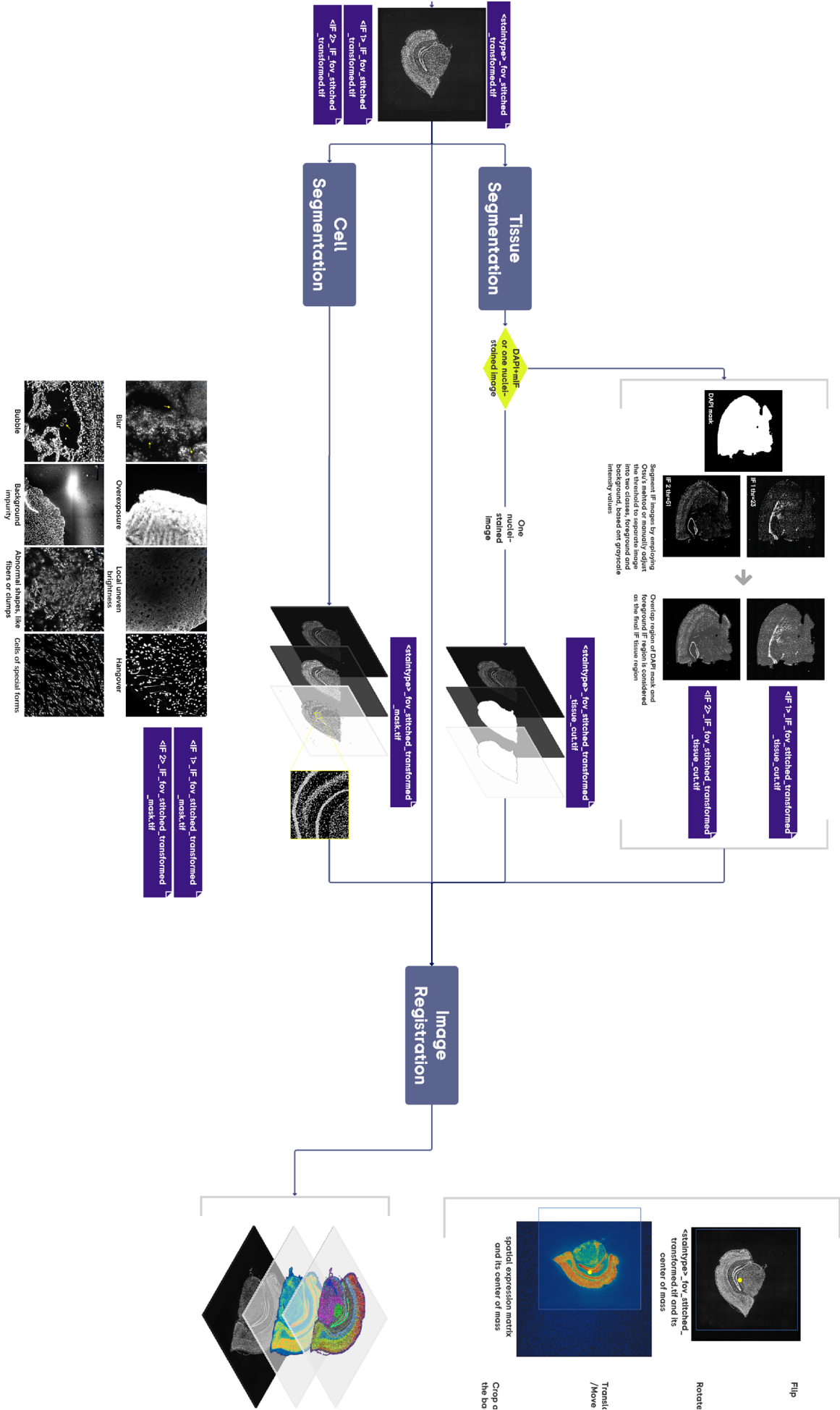
FAQ

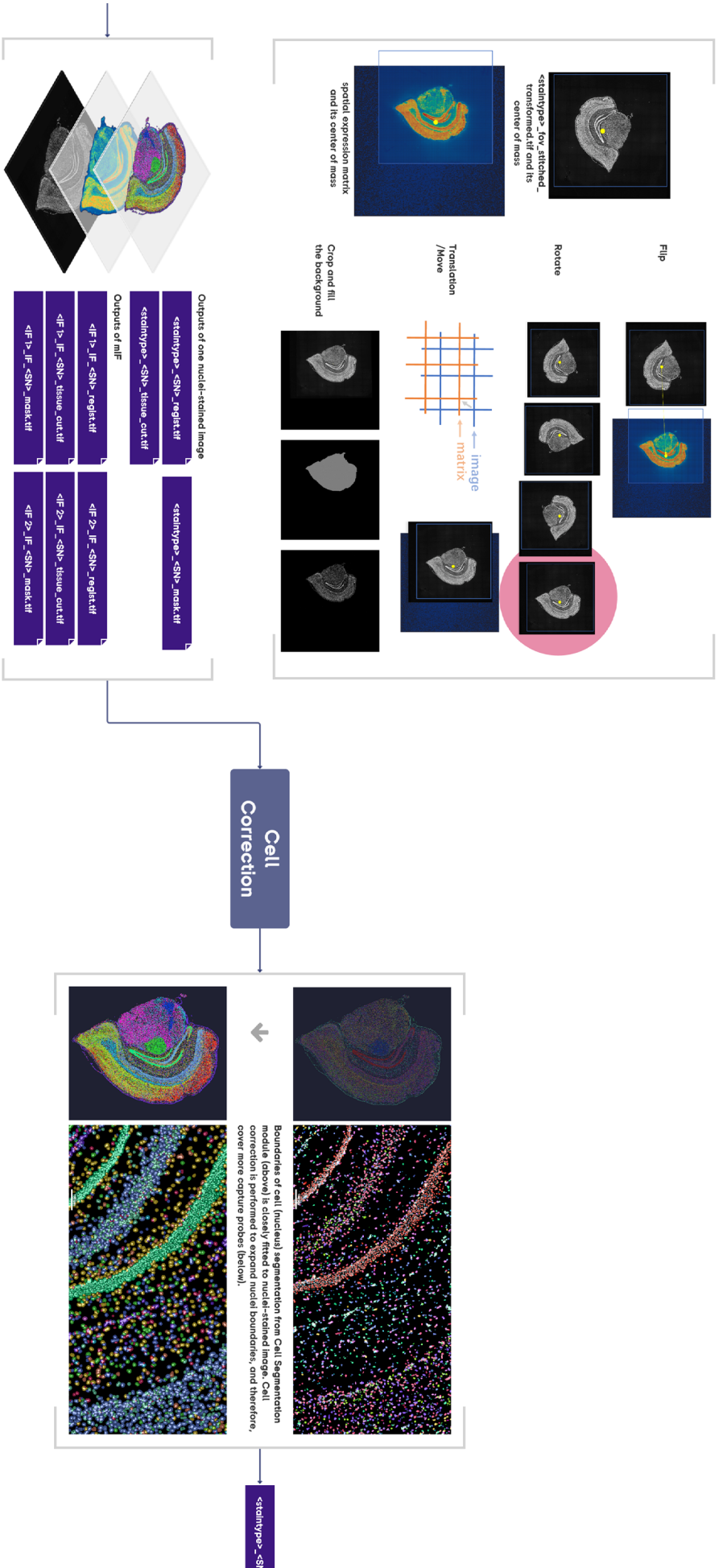


Manual process

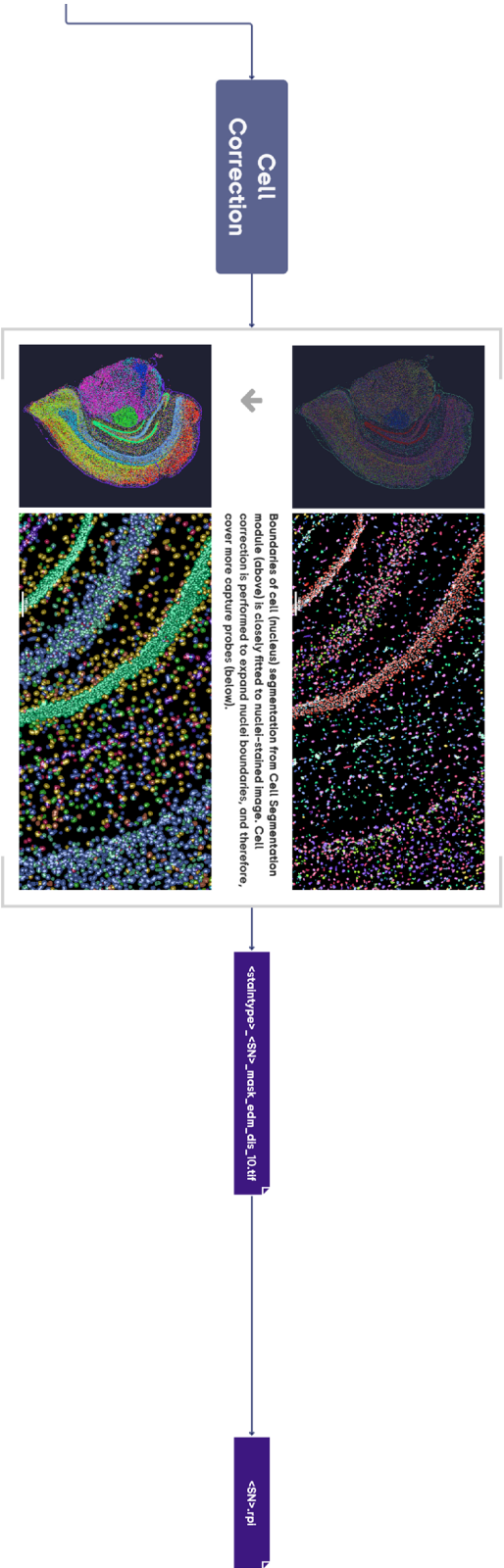
ImageStudio







STOmics®



Multiple Immunofluorescence Stains

2.1. How are immunofluorescence (IF) images mapped to the gene expression matrix?

The alignment between IF image and the spatial gene expression matrix is achieved indirectly by taking the DAPI image as a reference frame.

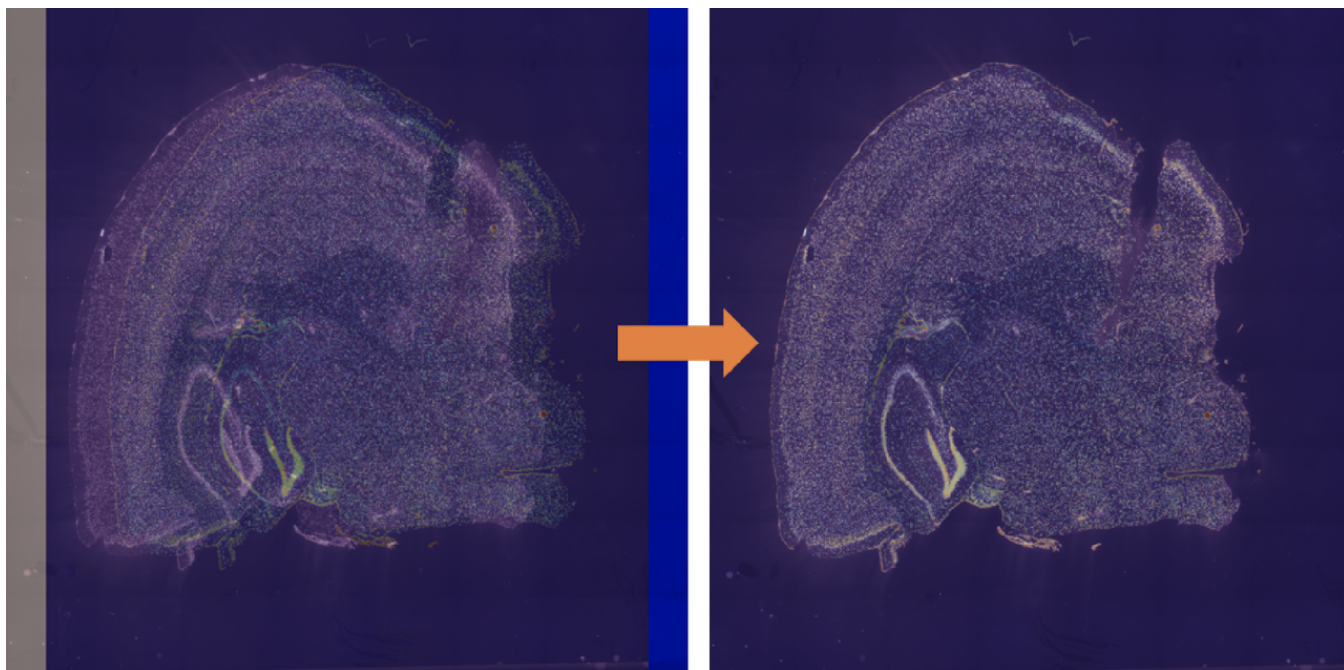
DAPI and IF images of the same tissue slice were shot back to back by switching channels. With the chip fixed during imaging, DAPI and IF images share the same stitching, scale, and angle parameters as compared to those of the spatial gene expression map. So the information used for DAPI image stitching, rotation, scaling, translation, and transformation can be applied to image processing of the IF layer as well, including alignment with the expression matrix.

2.2. What's the principle behind IF image QC?

Our current quality check strategy for IF images requires a paired DAPI image to be input together. The assessment contents include track line recognition of DAPI image, evaluation of microscope stitching for DAPI/IF images, and calibration between DAPI and IF images based on tissue morphology.

The detected **track lines** from the DAPI image during the QC step provide a fiducial reference frame for automatic image registration with the chip. **Microscope stitching evaluation** is used to determine whether there are obvious stitching errors in the microscope-stitched global image, guaranteeing the quality of subsequent tissue segmentation and alignment. **Calibration evaluation** is aimed to ensure that IF images can be processed in the same way as the DAPI image in terms of stitching, rotation, scaling, translation, and flip, and finally register the IF images with the expression matrix

However, it is possible that the IF images have dissimilar tissue morphology with DAPI, which might fail calibration QC. In such cases, **ImageStudio** can be used to make adjustments pairwise with the "Calibration" module.



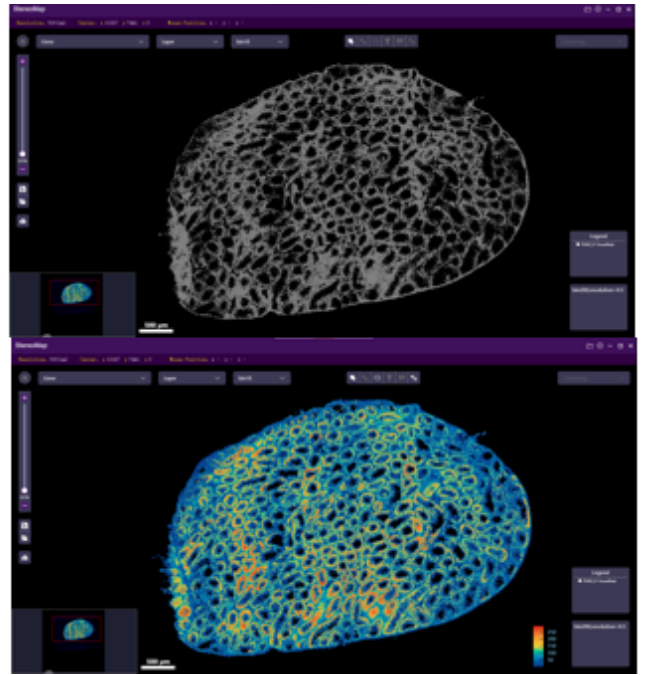
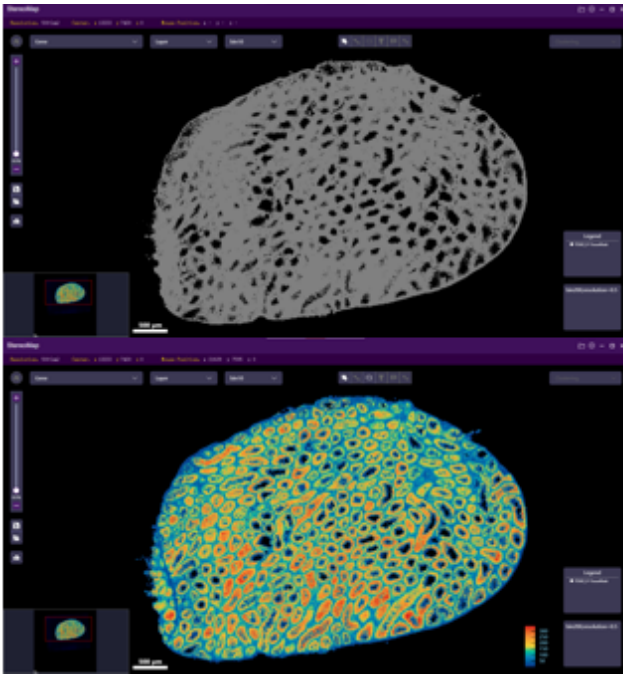
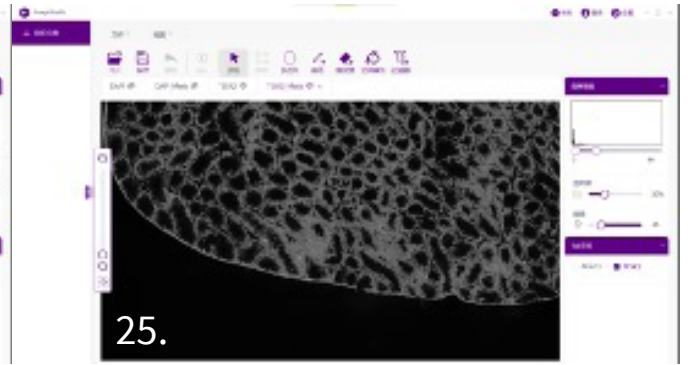
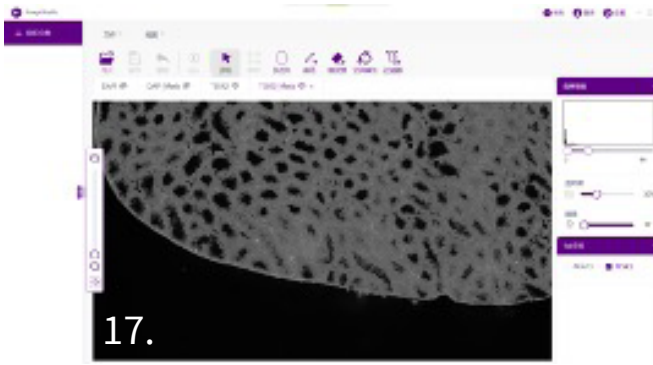
In the situation where DAPI image fails QC for track line recognition and microscope stitching, the related IF images can not be further processed automatically.

2.3. How to extract the corresponding regions from expression matrix according to IF images?

The immunofluorescence signal visualizes the location of the targeted proteins on the tissue slice. High fluorescent intensity indicates that a large number of cells in that region actively express the target proteins.

In the SAW workflow, the `register` module takes the use of an automatic global thresholding algorithm to compute the threshold value of the gray level that binarizes the IF image into the foreground and background region. The foreground region of the IF image is used as the mask file in the `tissueCut` module to acquire the gene expression matrix of the corresponding region.

If the segmentation result based on gray level calculated automatically is not satisfying, users can utilize the **ImageStudio** "Tissue Segmentation" module to manually adjust the grayscale threshold of the IF image to obtain a new tissue segmentation result.

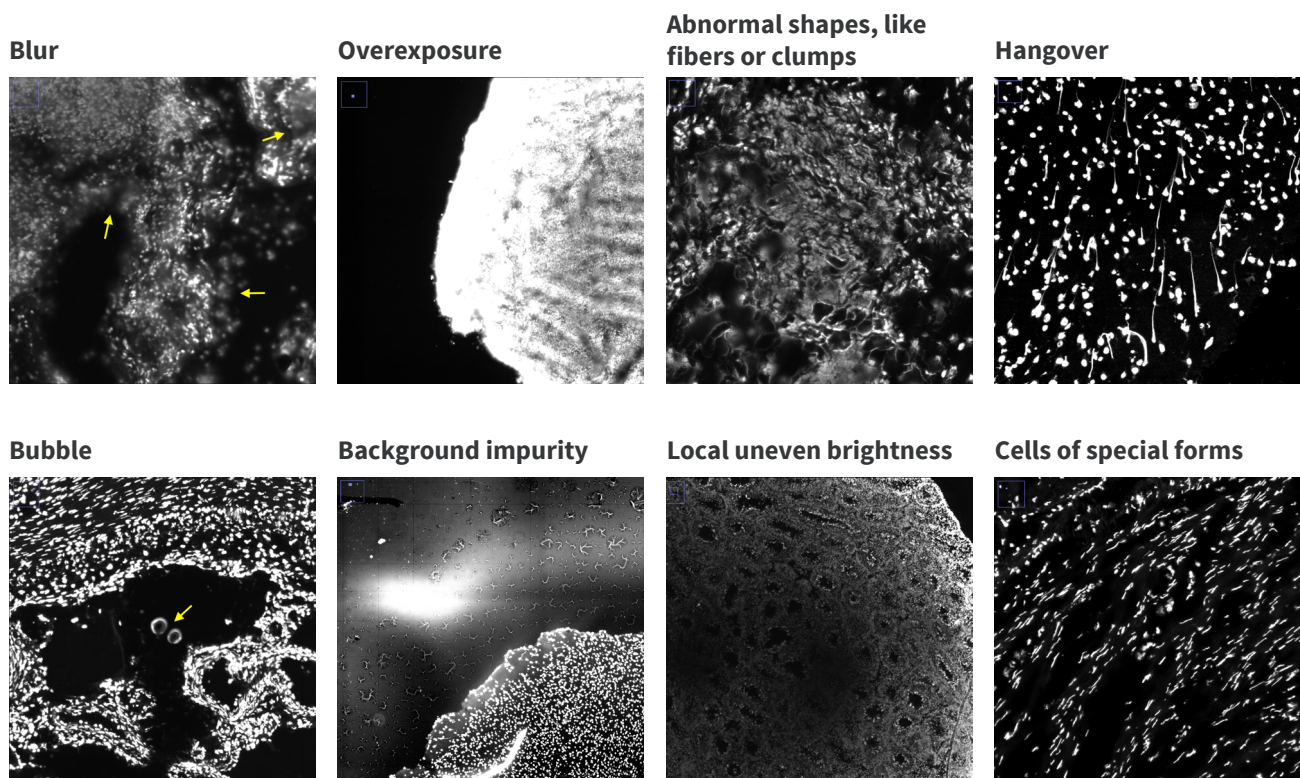


2.4. What factors affect cell segmentation results? How to get optimized segmentation?

The outcome of cell segmentation is determined by multiple factors such as the performance of microscope imaging and the segmentation algorithm used. Factors like overexposure and blurring can affect the automatic identification of cell areas which results in poor segmentation output. For some dense areas that are also blurred, and even accompanied by overlapping cells, it is especially difficult for the algorithm to do segmentation accurately. Also, segmentation mistakes will arise in cases where brightness is locally uneven over the tissue areas or background impurities and hangover of cell movement were introduced during experiments (see examples below).

From the perspective of the algorithm itself, training of automatic segmentation was done on specific datasets with manually assigned labels. Hence, the algorithm could perform poorly in identifying some particularly rare cell morphology that is not encompassed in the datasets.

If the algorithm segmentation does not work well, users can manually adjust results using **ImageStudio**, a desktop image processing software, or try to do it again with **Stereopy** or other algorithms. If there is a need to enlarge the identified cells, the cell correction algorithm in **Stereopy** can be employed to increase the cell diameters and have a larger cell coverage.



CHAPTER 3

SAW (Stereo-seq Analysis Workflow)

3.1. Format and requirements of the original input FASTQ file to SAW

STOmics SAW supports Q40 FASTQ and Q4 FASTQ files as input parameters.

Q40: A quality system that describes the quality of all sequenced bases with 41 quality values. Q40 FASTQ is the common form of PE FASTQ, with data appearing in pairs, usually named <lane>*_1.fq.gz and <lane>*_2.fq.gz. In general, the Q40 FASTQ file usually has "read" field, such as lane_read_1.fq.gz and lane_read_2.fq.gz. In a few cases, there may be no "read" in the file name, you can check whether it is in pairs, or check the file format.

```
## Q40 read 1
@V350156489L4C001R00100001484/1
TCTGCAGCCAACATGGACAGATCCTTTTAGAACTT
+
D>DC<CBEADEABCB(AADCDD2DD"DBE*$F'C'

## Q40 read 2
@V350156489L4C001R00100001484/2
CTATGAAACACACTATCCTCAATCGGCTCCTTAATTTCAATACCAGCCGT
+
ECFCCB?CAECEBDBCCFDFFDF?<FCF>B?2EC=C?C<CE(AA=;B5B
```

Q4: A quality system that describes the quality of all sequenced bases with 4 quality values. Q4 FASTQ comes as a set of 16 or 64 individual files, usually named <Lane>_<barcode>_<splitIndex>.fq.gz. In a few cases, there may be no barcode in the file prefix, you may distinguish the barcode from the directory name as below. Although it's not accurate, Q4 FASTQ is previously often called SE FASTQ. In fact, it is still paired-end sequencing, but when FASTQ is written out of the machine after sequencing, it is output as a set of single files, which looks the same as SE data. This format encodes the information in read1 and writes it into the readID of read2. At the same time, the sequence quality information is recorded in the form of Q4 to reduce storage.

Q4 FASTQ name

/path/to/data/E100026571_L01/barcode_2/E100026571_L01_2_16.fq.gz

lane

barcode

lane

barcode

split index

```
## Q4 read
@FP300000513L1C002R00400000218 CE242DF29A57 97D26
GTGTAGTGAACCCCATGGTAGTTTTCTGATTGTTGTTAAAAAAATGACTTAACATATTACATGGACACTCAATAAAAAAT-
GTTTTATTTCTGTTGAAAA
+
FFFFFFFFFFFFFFF8F8FFFFFFFFFFFFFFF8FFFFFFFFF8FF8FFF8FFFFFFFF,FFFFFFFFF8FFFFFFFF8F8F,F8FFF
FFF,FFFFFFFFF,FFF
```

3.2. What are the major filtering steps for the sequencing data in SAW pipelines?

CID filtering: to filter out reads with CID that can not be matched with any CID recorded in the Stereo-seq Chip T mask file.

MID filtering: to filter out reads with MID containing N base, reads with MID having ploy-A content, and reads with at least one base whose quality scores are lower than 10.

Reads filtering: to filter out reads containing adapters and DNB sequences.

3.3. How to deal with abnormal gene expression visualization result that does not show any tissue morphology?

Step 1: Check if the **"Valid CID Reads"** ratio in the HTML report is lower than 10%. If so, please check whether the FASTQ corresponds with chip SN.

Step 2: Two possibilities that can lead to a low **"Valid CID Reads"** ratio of around 10% - 30%:

- Reference genome does not meet the format requirement: if the ratio of multi-mapped reads is high, and the uniquely mapped reads ratio is extremely low, please run SAW checkGTF for the GFF/GTF file to verify the file format is valid for running pipelines.
- Contamination: please perform troubleshooting on the wet lab workflow.

3.4. What is the difference between the two SAW registration modules, register and rapidRegister?

SAW register pipeline includes a cell segmentation procedure, whereas rapidRegister does not.

3.5. What are the purposes of the three Saturation curves?

The first one can indicate whether the sequencing is saturated. If the fitted curve reaches or approximates a plateau, this means the sample is about to saturate. Depending on the goal of each individual project, you may need additional sequencing runs. For example, a project designed to recover very lowly expressed transcripts or involves precious samples may desire a higher sequencing saturation. A recommended saturation of 80% is an empirical threshold, it is not a rigid value.

The second and third figures are plotted with statistics computed at bin levels, and their stationary stages are lagging behind Figure 1. The first plot serves as the main indicator for the potential benefit of additional sequencing.

CHAPTER 4

SAW RESULT INTERPRETATION

4.1. How to interpret GEF-format files?

Option 1: use C++ compiled geftools;

- <https://github.com/STOmics/geftools>

Option 2: use Python package - gefpy (e.g. 0.6.1);

- <https://pypi.org/project/gefpy/>
- <https://gefpy.readthedocs.io/en/latest/index.html>
- `pip install gefpy==0.6.1`

Option 3: with installed SAW sif (e.g. v5.1.3):

- <https://hub.docker.com/repository/docker/stomics/saw>
- `singularity exec SAW_v5.1.3.sif cellCut`
- Please use Singularity version 3.8 or later

```
export HDF5_USE_FILE_LOCKING=FALSE
## gef2gem using geftools
geftools view -i <SN>.gef -o <SN>.gem -s <SN>
# -i input square bin GEF, e.g. SN.raw.gef or SN.gef
# -o output GEM
# -s SN
```

```
## gef2gem using gefpy
python
>>> from gefpy.bgef_reader_cy import BgefR
>>> bgef=BgefR(filepath='<SN>.gef',bin_size=200,n_thread=4)
>>> bgef.to_gem('<SN>.bin200.gem')
```

```
## gef2gem using SAW sif
## export SINGULARITY_BIND="/path/to/input/dir,/path/to/output/dir"
singularity exec SAW_v5.1.3.sif cellCut view -i <SN>.gef -o <SN>.gem -s <SN>

## cgef2cgem
geftools view -i <SN>.cellbin.gef -o <SN>.cellbin.gem -d <SN>.raw.gef -s <SN>
# -i input cellbin GEF
# -o output cellbin GEM
# -d input square bin GEF, e.g. SN.raw.gef or SN.gef
# -s SN

## gem2gef
geftools bgef -i <SN>.gem -o <SN>.gef -b 1,20,50 -O Transcriptomics
# -i input square bin GEM
# -o output square bin GEF
# -b bin sizes separate by comma, default: 1,10,20,50,100,200,500
# -O omics name
```


CHAPTER 5 VISUALIZATION

5.1. As SAW generates multiple GEF files, what information do they store respectively? Do they support visualization?

File Name	Common Application	Visualization Supported	Description
SN.raw.gef	transcription		Original transcription matrix of whole chip area, containing only bin1 geneExp group.
SN.tissue.gef	transcription		Transcription matrix of tissue-covered region, containing only bin1 geneExp group.
SN.gef	transcription	✔	Visual transcription matrix of whole chip area, containing multiple bin geneExp and wholeExp groups.
SN.cellbin.gef	transcription/ ssDNA/DAPI	✔	Cell-gene expression matrix of tissue-covered region.
SN.<protein_IF>.gef	transcription/ mIF	✔	Completion matrix based on matrix that is extracted by mask of IF image gray scale threshold filtering. [Recommended to name like this, it is not generated by default. Please complete it by running cellCut after tissueCut.]
SN.<protein_IF>.cellbin.gef	transcription/ mIF	✔	Cell-gene expression matrix that is extracted by cellmask of IF image gray scale threshold filtering. [Recommended to name like this, it is not generated by default but needed to switch to cellCut after tissueCut.]
SN.<label>.raw.label.gef	transcription/ H&E		Original transcription matrix of labeled area, containing only bin1 geneExp group.
SN.<label>.label.gef	transcription/ H&E	✔	Visual matrix of labeled area, containing multiple bin geneExp and wholeExp groups.
SN.<label>.label.cellbin.gef	transcription/ H&E	✔	Cell-gene expression matrix of labeled area.