

Takara Bio USA

Trekker™ Primary Analysis Pipeline User Manual

(091525)

Takara Bio USA, Inc.

2560 Orchard Parkway, San Jose, CA 95131, USA

U.S. Technical Support: technical_support@takarabio.com

United States/Canada
800.662.2566

Asia Pacific
+1.650.919.7300

Europe
+33.(0)1.3904.6880

Japan
+81.(0)77.565.6999

Page 1 of 66

Table of Contents

I.	Introduction	5
A.	Tools and Steps in the Pipeline	6
B.	Ways to Run the Pipeline.....	6
C.	Technical Support	6
II.	Trekker Pipeline Local Installation.....	6
A.	System Requirements	6
B.	Download trekker-v1.3.0.	6
C.	Set Up Paths.....	7
D.	Set Up Dependencies.....	7
E.	Check Input Format.....	8
F.	Input.....	11
G.	Trigger the Pipeline	14
H.	Monitor the Run	14
I.	Output.....	16
J.	Testing and Example Datasets	17
III.	Interpreting Trekker Pipeline Output	18
A.	Definition of Key Output Files	18
B.	Definition and Interpretations of Quality Metrics	21
C.	How Positioning Works.....	26
IV.	References.....	29
Appendix A.	Preprocessing.....	29
A.	Preprocessing: Illumina Single Cell 3' RNA Prep Kit (TrekkerU_PIP).....	29
B.	Preprocessing: Parse Evercode WT v3 Kit (TrekkerQ_P).....	34
Appendix B:	Trekker Output Merger.....	41
A.	Download trekker_merger-v1.3.0	41
B.	Set Up Paths.....	41
C.	Set Up Dependencies.....	42
D.	Check Input File Completeness.....	43
E.	Input.....	43
F.	Trigger the Merger	44
G.	Monitor the Run	45
H.	Output.....	47

I. Testing and Example Datasets	47
Appendix C. How to	48
A. How to Concatenate Multiple FASTQ Files into One FASTQ.....	48
B. How to Compress a FASTQ File into .gz Format.....	48
Appendix D. Alternative Trekker Pipeline Input Preparation	48
A. Input Preparation: BD Rhapsody Single-Cell TCR/BCR Next + WTA Kit (TrekkerU_RVDJ).....	48
B. Input Preparation: BD Rhapsody Single-Cell ATAC-Seq + mRNA WTA Kit (TrekkerU_RATAC).....	51
Appendix E. Troubleshooting.....	54
A. Troubleshooting: Local Installation	54
B. Troubleshooting: Merging Pipeline Output.....	58
C. Troubleshooting: Preprocessing for Illumina Single Cell 3' RNA Prep Kit (TrekkerU_PIP).....	60
D. Troubleshooting: Input Preparation for BD Rhapsody Single-Cell TCR/BCR Next + WTA Kit (TrekkerU_RVDJ)	
62	
E. Troubleshooting: Input Preparation for BD Rhapsody Single-Cell ATAC-Seq + mRNA Whole Transcriptome	
Analysis Kit (TrekkerU_RATAC).....	64

Table of Figures

Figure 1. Workflow of the Trekker Primary Analysis Pipeline (Trekker pipeline).....	5
Figure 2. Example read structure of the Trekker spatial barcode library FASTQ pairs.....	6
Figure 3. Examples of nuclei positioned with 1, 2, or no spatial locations.....	27
Figure 4. Assessment of the quality of spatial positioning.....	28
Figure 5. Workflow for format conversion using <code>trekkerU_PIP_converter.sh</code>	30
Figure 6. Illustration of how Trekker preprocessing step demultiplexes Trekker FASTQs to sample-specific Trekker FASTQs	35
Figure 7. Illustration of how to pair demultiplexed Trekker FASTQs with Parse' pipeline output.....	40

Table of Tables

Table 1. Trekker input preprocessing requirements and instructions	9
Table 2. Which Trekker FASTQ file to use for the Trekker pipeline, for single-cell platforms requiring preprocessing	10
Table 3. Typical location and format for required single-cell pipeline outputs.....	11
Table 4. <code>samplesheet.csv</code> column names and descriptions	13
Table 5. "sc_platform" values based on the single-cell assay platform being used.....	14
Table 6. Files in folder <code>output</code>	19
Table 7. Files in folder <code>cell_bc_plots</code>	19
Table 8. Files in folder <code>intermediates</code>	20
Table 9. Metrics in <code>\${Sample_ID}_Trekker_Report.html</code>	21
Table 10. Sample metadata and analysis parameters	23
Table 11. Nuclei positioning metrics	24

Table 12. Spatial barcode library quality24

Table 13. Spatial barcode matching quality25

Table 14. Spatial barcode library depth.....25

Table 15. Signal to noise.....26

Table 16. Nuclei salvaged26

I. Introduction

The Trekker Primary Analysis Pipeline (Figure 1) processes sequenced data from the Trekker Single-Cell Spatial Mapping Kits.

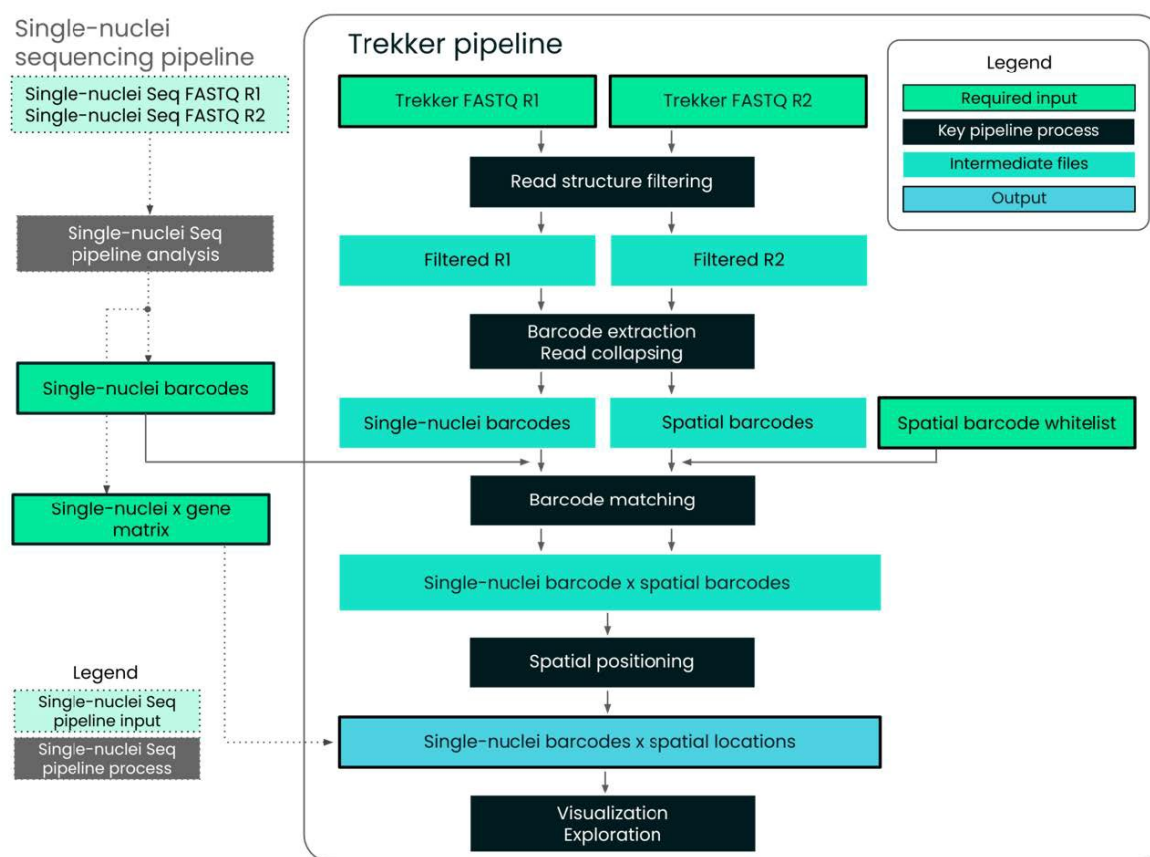


Figure 1. Workflow of the Trekker Primary Analysis Pipeline (Trekker pipeline).

The Trekker pipeline requires the following inputs:

- FASTQ pairs from the Trekker spatial barcode library
- Selected outputs from the bioinformatics pipeline of the single-nuclei sequencing library
- The bead barcode file for the Trekker tile used
- A sample sheet associating sample metadata with aforementioned inputs

The Trekker pipeline extracts single-nuclei barcodes from FASTQ Read 1 and spatial barcodes from FASTQ Read 2. It also retrieves UMIs (unique molecular identifiers) linked to each spatial barcode from Read 1 (Figure 2). After barcode matching, spatial positioning is done using the spatial barcodes associated with each single nucleus, resulting in a spatial location assigned to each single nucleus. The pipeline additionally conducts quality control to help determine the next steps.

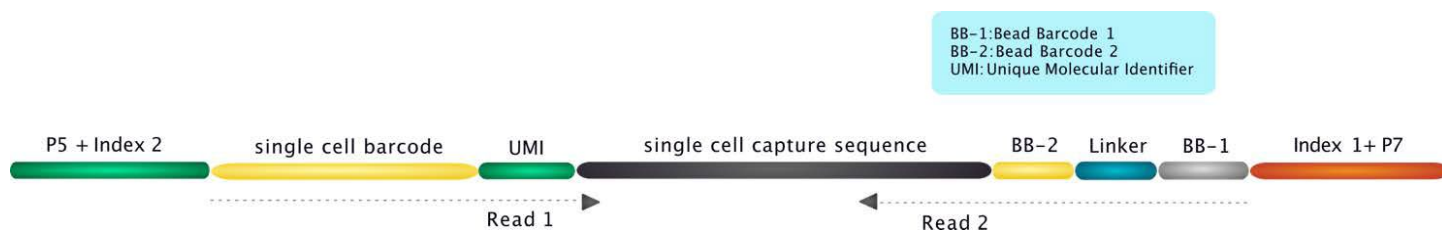


Figure 2. Example read structure of the Trekker spatial barcode library FASTQ pairs.

A. Tools and Steps in the Pipeline

The Trekker pipeline's backbone is written in Bash. Its steps are written in Python and R, and its dependencies can be executed through Singularity (now Apptainer), Docker, and Conda.

The Trekker pipeline consists of the following major steps. It utilizes both publicly available and custom-written bioinformatic software.

- Read filtering and barcode matching: custom Python scripts
- Spatial positioning: custom R scripts and DBSCAN (<https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html>)
- Analysis and visualization: custom R scripts and Seurat (<https://satijalab.org/seurat/>)

B. Ways to Run the Pipeline

The Trekker pipeline can be run in the following ways:

- Via installation on local workstations or clusters (Section II, "[Trekker Pipeline Local Installation](#)")
- Via [cloud analysis](#)

This document covers running the pipeline via local installation.

C. Technical Support

If you have any questions related to the Trekker or Seeker kits, protocols, or data interpretation downstream of bioinformatics workflows, please email technical_support@takarabio.com.

II. Trekker Pipeline Local Installation

A. System Requirements

- Linux platforms with ≥ 256 GB RAM and 12 cores.

NOTE: The pipeline does NOT support Mac or Windows machines.

B. Download trekker-v1.3.0.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

Use the following command to download trekker-v1.3.0.

```
wget https://www.takarabio.com/resourcedocument/x353749 -O - | tar -xzf-
```

Use the following command to navigate into the script folder.

```
cd trekker-*
```

C. Set Up Paths

Modify the bolded values shown in `nuclei_locator_toplevel.sh` (in the `trekker-v1.3.0` script folder) to match the paths on your platform.

```
#===== Define Directory Paths =====  
  
SCRIPT_DIR=/home/trekker-v1.3.0/  
OUT_DIR=/home/trekker_out/
```

NOTES: See below for further details on the values you are modifying.

- `SCRIPT_DIR`: Absolute path to the script folder `trekker-v1.3.0`.
- `OUT_DIR`: Absolute path to a folder where the Trekker pipeline output should be directed to. (This is the top-level output folder. The pipeline will create sample-specific folders inside.)

D. Set Up Dependencies

The pipeline can be executed through Docker, Singularity, or Conda.

1. Docker

1. Install Docker if it is not already installed on the server.
Visit the [Docker page](#) for installation instructions.
2. Pull the Trekker Docker image, using the command below.

```
docker pull tbusaspatial/trekker:v1.3.0
```

2. Singularity

Install Singularity if not already installed.

Visit the [Singularity page](#) for installation instructions.

NOTE: You DO NOT need to download the Singularity image `trekker-v1.3.0.sif`. It is in folder `trekker-v1.3.0`.

3. Conda

NOTE: Steps 1 & 2 below are only required if you are using Conda to execute dependencies.

1. Create and activate a trekker conda environment using the `environment.yml` (in the `trekker-v1.3.0` script folder) we created for the pipeline.

```
cd trekker-*  
conda env create -f environment.yml  
conda activate trekker
```

2. Modify the bolded values in the two lines shown below in `nuclei_locator_conda.sh` (in the `trekker-v1.3.0` script folder) to match the paths on your platform.

```
##### Define Conda Paths #####
PROFILE_CONDA_PATH=/home/tools/miniconda3/envs/trekker/
source /home/tools/miniconda3/etc/profile.d/conda.sh
```

NOTES: See below for further details on the values you are modifying.

- **/home/tools/miniconda3/envs/trekker/**: Replace this value with the absolute path to the trekker conda environment you created in Step 1. To find the path, type:

```
conda info --envs
```

Copy and paste the path displayed next to `trekker`, as bolded below.

```
trekker          /home/tools/miniconda3/envs/trekker
```

- **/home/tools/miniconda3/**: Replace this value with the path to your miniconda installation. To find the path, type:

```
conda info | grep -i 'base environment'
```

E. Check Input Format

1. Check if your Trekker input requires preprocessing

IMPORTANT:

- If your data comes from the following single-cell platforms, an additional preprocessing step is required to ensure your Trekker inputs are compatible with the Trekker pipeline.

- Illumina®, Single Cell 3' RNA Prep
- Parse, Evercode WT v3



Use the links in Table 1 to access preprocessing instructions for your single-cell platform.

Once preprocessing is complete, verify your *converted* Trekker input format, following the guidelines in Step 2.

- For data from all other single-cell platforms, no preprocessing is needed. Check your Trekker input format, following the guidelines in Step 2.

Table 1. Trekker input preprocessing requirements and instructions.

Single-cell assay platform	Is preprocessing required?	Preprocessing instructions
All 10x Chromium platforms	No	—
All BD Rhapsody platforms	No	—
Illumina, Single Cell 3' RNA Prep	Yes	Appendix A, Section A, " Preprocessing: Illumina Single Cell "
Parse, Evercode WT v3	Yes	Appendix A, Section B, " Preprocessing: Parse Evercode "

2. Check if your Trekker input formats conform to the following requirements

a) *FASTQ Pairs from the Trekker Spatial Barcode Library*

IMPORTANT:

- If your tissue section was processed on a single Trekker tile but split into multiple single-nuclei reactions, i.e. processed in multiple:
 - 10x Chromium wells
 - BD Rhapsody lanes
 - Illumina PIPseq reactions
 - Parse sub-libraries



These reactions will have different sample indices during sequencing. **DO NOT** combine FASTQ files from these reactions. Process Trekker FASTQ pairs for each reaction and combine their pipeline outputs using the merging script from Appendix B, "[Trekker Output Merger](#)".

- The scenario above is different from sequencing a reaction in multiple sequencing lanes, for which you must concatenate FASTQ files from multiple lanes into a single Read 1 and a single Read 2.

- For each Trekker reaction, prepare *a single* Read 1 and *a single* Read 2 FASTQ in .gz format.

NOTE: Refer to Appendix C, Section A, "[How to Concatenate Multiple FASTQ Files](#)" for instructions on concatenating multiple FASTQ files.

- The FASTQ pairs must be in the compressed .gz format.

NOTE: Refer to Appendix C, Section B, "[How to Compress a FASTQ File](#)" for instructions on compressing a FASTQ file into .gz format.

- All reads in Trekker Read 1 FASTQ should have the same length and have the minimal length required by the single-nuclei bioinformatics pipeline.

- All reads in Trekker Read 2 FASTQ should have the same length and be at least 32 bp.
- (Parse and Illumina single-cell platform only) Make sure your FASTQ pairs have gone through preprocessing. Follow Table 2 to locate the preprocessed FASTQs to be fed into the Trekker pipeline.

Table 2. Which Trekker FASTQ file to use for the Trekker pipeline, for single-cell platforms requiring preprocessing.

Single-cell assay platform	Trekker Read 1 FASTQ	Trekker Read 2 FASTQ
Illumina, Single Cell 3' RNA Prep	converted/{prefix}_converted_R1.fastq.gz (generated by the preprocessing step)	Original Trekker Read 2 without preprocessing
Parse, Evercode WT v3	In the folder defined by --opath in the preprocessing step	In the folder defined by --opath in the preprocessing step

b) *Selected Outputs from the Bioinformatics Pipeline of the Single-Nuclei Sequencing Library*

For each snRNA-seq reaction, generate the following three output files using the associated single-cell bioinformatics pipeline (and preprocessing script, in the case of Illumina Single Cell 3' RNA Prep). Place them in a single folder.

- For all 10x Chromium platforms, BD Rhapsody platforms, and Illumina platforms, the three files have the following names:
barcodes.tsv.gz, features.tsv.gz, matrix.mtx.gz
- For Parse Evercode WT v3, the three files have the following names:
cell_metadata.csv.gz, all_genes.csv.gz, count_matrix.mtx.gz

IMPORTANT:



- Most single-cell pipelines generate these three files in unfiltered and "quality-filtered" versions. Use the "quality-filtered" version where high-quality nuclei were retained.
- Use Table 3 (next page) to locate these files, and check if they have the expected format.
- See [example input](#) to check if your files have the proper format.

Table 3. Typical location and format for required single-cell pipeline outputs.

Single-cell assay platform	Which folder contains the three required files?	Typical format in barcodes.tsv.gz
All 10x Chromium platforms	filtered_feature_bc_matrix	AAACCCAAGCAAATGT-1 AAACCCAAGGTAAGAG-1 AAACCCAAGTTTGTCTG-1
All BD Rhapsody platforms	\${Sample}_RSEC_MolsPerCell_MEX	10632 14108 14878
Illumina, Single Cell 3' RNA Prep	converted (generated after preprocessing)	AAAAAAAAACCATGAAT AAAAAAAAACTGACGAC AAAAAAAAAGGTCCGAT
Parse, Evercode WT v3	DGE_filtered NOTE: DO NOT use the files from output_combined and all-samples, as this reduces positioning rate. See instructions in Appendix A, Section B.8, " Pairing Demuxed Trekker FASTQs " for which version to use.	See cell_metadata.csv.gz in Section II.J.1, " Download Example Inputs " (TrekkerQ_P)

F. Input

IMPORTANT:



- If you are processing data from BD Rhapsody Single-Cell TCR/BCR Next + mRNA Whole Transcriptome Analysis (WTA) Kit (TrekkerU_RVDJ), skip this entire “Input” section, and prepare your input following the instructions in Appendix D, Section A, "[Input Preparation: BD Rhapsody Single-Cell TCR/BCR Next](#)".
- If you are processing data from BD Rhapsody Single-Cell ATAC-Seq + mRNA WTA Kit (TrekkerU_RATAC), skip this entire “Input” section, and prepare your input following the instructions in Appendix D, Section B, "[Input Preparation: BD Rhapsody Single-Cell ATAC-Seq](#)".
- For data from all other single-cell platforms, prepare your input following the instructions below.

Prepare the following inputs before triggering the pipeline.

NOTE: Before processing your own data, we recommend using [example inputs](#) to test if the pipeline is properly set up and if your files have the proper format.

1. Bead Barcode File

```
${Tile_ID}_BeadBarcodes.txt
```

Download the bead barcode file (e.g., U0001_001_BeadBarcodes.txt) using our [Tile bead barcode file retrieval tool](#).

NOTE: Don't forget to unzip the downloaded file before use.

2. Trekker Paired FASTQ Files

Example file names:

```
MouseBrain_Trekker_R1.fastq.gz
```

```
MouseBrain_Trekker_R2.fastq.gz
```

NOTE: Don't forget to check the format conformity of these files (Section II.E, "[Check Input Format](#)"). Improper format will lead to pipeline failures.

3. Selected Outputs from the Bioinformatics Pipeline of the Single-Nuclei Sequencing Library

- For all 10x Chromium platforms, BD Rhapsody platforms, and Illumina platforms, the three files have the following names:
barcodes.tsv.gz, features.tsv.gz, matrix.mtx.gz
- For Parse Evercode WT v3, the three files have the following names:
cell_metadata.csv.gz, all_genes.csv.gz, count_matrix.mtx.gz

NOTE: Don't forget to check the format conformity of these files (Section II.E, "[Check Input Format](#)"). Improper format will lead to pipeline failures.

4. Sample Sheet

```
samplesheet.csv
```

Download this example [samplesheet.csv](#) to use as a starting point to enter information on your sample to be analyzed.

NOTES:

- Prepare a separate `samplesheet.csv` for each reaction (i.e., a pair of Trekker FASTQs). The pipeline does NOT support multiple rows (reactions) in a sample sheet.
- DO NOT swap the column orders in the sample sheet. Re-ordered columns will lead to pipeline failure.
- The box below (next page) shows an example of what the sample sheet file contents might look like. See Table 4 for a description of each column type.

```
sample,sc_sample,experiment_date,barcode_file,fastq_1,fastq_2,sc_outdir,sc_platform,profile,subsample,cores
```

```
TrekkerU_C_MouseBrain,TrekkerU_C_MouseBrain_scRNAseq,20240916,/path/to/LTTag0053_003_BeadBarcodes.txt,/path/to/TrekkerU_C_Mouse_brain_R1_001.fastq.gz,/path/to/TrekkerU_C_Mouse_brain_R2_001.fastq.gz,/path/to/TrekkerU_C_MouseBrain_scRNAseqOut/filtered_feature_bc_matrix/,TrekkerU_C,singularity,no,8
```

Table 4. `samplesheet.csv` column names and descriptions. Values are required for all columns.

Column name	Description
sample	Prefix for naming the Trekker pipeline output. Avoid using '.' or space.
sc_sample	Prefix for book-keeping single-nuclei pipeline analysis output. (For book-keeping purposes only. Use a desired string with no space or special characters.)
experiment_date	Date of analysis in format YYYYMMDD. (This will be used as part of the output folder name.)
barcode_file	Absolute path to the Trekker bead barcode file.
fastq_1	Absolute path to the Trekker FASTQ Read 1.
fastq_2	Absolute path to the Trekker FASTQ Read 2.
sc_outdir	<p>Absolute path to the folder containing the following three required single-nuclei pipeline analysis output.</p> <ul style="list-style-type: none"> For all 10x Chromium platforms, BD Rhapsody platforms, and Illumina platforms, the three files have the following names: <code>barcodes.tsv.gz</code>, <code>features.tsv.gz</code>, <code>matrix.mtx.gz</code> For Parse Evercode WT v3, the three files have the following names: <code>cell_metadata.csv.gz</code>, <code>all_genes.csv.gz</code>, <code>count_matrix.mtx.gz</code> <p>IMPORTANT: DO NOT use the files from <code>output_combined</code> and <code>all-samples</code>, as this reduces positioning rate. See instructions in Appendix A, Section B.8, "Pairing Demuxed Trekker FASTQs" for which version to use.</p>
sc_platform	<p>Select a value from the 2nd column of Table 5 (below) (e.g., <code>TrekkerU_C</code>), based on the single-cell assay platform you are using from the 1st column.</p> <p>IMPORTANT: Make sure to choose the correct value. An incorrect value will lead to pipeline failure.</p>
profile	How to execute dependencies. Choose among Docker, Singularity, or Conda based on what you set up in Section II.D, " Set Up Dependencies ". Do not use quotes; case does not matter.
subsample	If subsampling should be performed for spatial positioning parameterization. Use 'no' unless you have $>5 \times 10^6$ nuclei to be positioned. Do not use quotes; case does not matter.
cores	Number of cores to use for the spatial positioning step. We recommend 8, which is sufficient for most cases.

Table 5. "sc_platform" values based on the single-cell assay platform being used.

Single-cell assay platform	"sc_platform" value
10x Chromium, Next GEM 3' v3.1, polyT capture	TrekkerU_C
10x Chromium, GEM-X 3' v4, polyT capture	TrekkerU_CX
BD Rhapsody, WTA, polyT capture	TrekkerU_R
10x Chromium, Next GEM Multiome ATAC + Gene Expression	TrekkerU_M
Illumina, Single Cell 3' RNA Prep	TrekkerU_PIP
Parse, Evercode WT v3	TrekkerQ_P

G. Trigger the Pipeline

Follow the example below to trigger the pipeline.

NOTE: Before processing your own data, we recommend using our [example inputs](#) to test if the pipeline is properly set up.

```
$ cd trekker-v1.3.0
bash nuclei_locator_toplevel.sh /path/to/samplesheet.csv
```

H. Monitor the Run

The STDOUT (standard output) of a successful pipeline run will look similar to the text box below (next page):

```
Checking samplesheet...
'/home/TrekkerU_C_ExampleInput_MouseBrain/samplesheet.csv' exists.
Checking Trekker FASTQ R1...
'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R1_001.fastq.gz'
exists.
Checking Trekker FASTQ R2...
'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R2_001.fastq.gz'
exists.
Checking Trekker tile spatial barcode whitelist...
'/home/TrekkerU_C_ExampleInput_MouseBrain/LTTag0053_003_BeadBarcodes.txt' exists.
Checking nuclei subsampling option...
Nuclei subsampling option is set to: no
Checking the number of cores requested...
Number of cores requested is set to: 8
Checking snRNAseq outputs...
snRNAseq outputs located.
Starting Trekker Analysis (TrekkerU_C) using singularity
Start fastq_parser
fastq_parser Done
Start splitspatialbarcodes
splitspatialbarcodes Done
Start bead_matching
bead_matching Done
Start spatial
spatial Done
Start analysis
analysis Done
Start mismatch_analysis
mismatch_analysis Done
Start genmetrics
genmetrics Done
Start genreport
genreport Done
```

A log folder containing stepwise logs can be found in the folder where you stored your samplesheet.csv. Use the command below to access detailed logs.

```
$ cd /path/to/where-samplesheet.csv-is-stored
cd log/${sample}
ls
```

I. Output

Trekker outputs are in

```
${OUT_DIR}/${experiment_date}_${sample_ID}/.
```

The folder output contains key output files for downstream analysis, as shown below. A detailed description of each key output file can be found in Section III, "[Interpreting Trekker Pipeline Output](#)". The other folders contain intermediate files and run summaries.

```
output
├─ ${Sample_ID}_ConfPositioned_seurat_spatial.rds
├─ ${Sample_ID}_ConfPositioned_anndata_matched.h5ad
├─ ${Sample_ID}_MoleculesPer_ConfPositionedNuclei.mtx
├─ ${Sample_ID}_barcodes_ConfPositionedNuclei.tsv
├─ ${Sample_ID}_genes_ConfPositionedNuclei.tsv
├─ ${Sample_ID}_Location_ConfPositionedNuclei.csv
├─ ${Sample_ID}_Trekker_Report.html
├─ ${Sample_ID}_summary_metrics.csv
├─ ${Sample_ID}_variable_features_clusters.csv
├─ ${Sample_ID}_variable_features_spatial_moransi.txt
├─ coords_${Sample_ID}.txt
├─ cell_bc_plots
├─ intermediates
└─ plots
```


J. Testing and Example Datasets

To test pipeline configuration, use our example inputs to trigger the pipeline and compare the results to our example output.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

1. Download Example Inputs

Single-nuclei assay platform: TrekkerU_C

```
wget https://www.takarabio.com/resourcedocument/x353756 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_M

```
wget https://www.takarabio.com/resourcedocument/x353757 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_PIP

```
wget https://www.takarabio.com/resourcedocument/x353758 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerQ_P

```
wget https://www.takarabio.com/resourcedocument/x353759 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_RATAC

```
wget https://www.takarabio.com/resourcedocument/x353760 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_RVDJ

```
wget https://www.takarabio.com/resourcedocument/x353761 -O - |  
tar -xzf -
```

2. Download Example Outputs

Single-nuclei assay platform: TrekkerU_C

```
wget https://www.takarabio.com/resourcedocument/x353762 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_M

```
wget https://www.takarabio.com/resourcedocument/x353763 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_PIP

```
wget https://www.takarabio.com/resourcedocument/x353764 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerQ_P

```
wget https://www.takarabio.com/resourcedocument/x353765 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_RATAC

```
wget https://www.takarabio.com/resourcedocument/x353766 -O - |  
tar -xzf -
```

Single-nuclei assay platform: TrekkerU_RVDJ

```
wget https://www.takarabio.com/resourcedocument/x353767 -O - |  
tar -xzf -
```

III. Interpreting Trekker Pipeline Output

Trekker outputs can be found at `${OUT_DIR}/${experiment_date}_${sample_ID}/`.

In this section, you can find descriptions for:

- [Key output files](#)
- [Quality metrics](#)

Check out Section III.C, "[How Positioning Works](#)" to understand how the Trekker pipeline positioned the nuclei bioinformatically.

A. Definition of Key Output Files

Key output files are in folder `output`.

Table 6. Files in folder output.

File	Definition
<code>\${Sample_ID}_summary_metrics.csv</code>	Run quality metrics
<code>\${Sample_ID}_Report.html</code>	Run quality report
<code>\${Sample_ID}_Report_files</code>	Individual plots in <code>\${Sample_ID}_ST_Report.html</code>
<code>\${Sample_ID}_ConfPositioned_seurat_spatial.rds</code>	SEURAT object with expression table, spatial coordinates, and clustering results for nuclei positioned with 1 spatial location
<code>\${Sample_ID}_ConfPositioned_anndata_matched.h5ad</code>	H5AD object with expression table, spatial coordinates, and clustering results for nuclei positioned with 1 spatial location
<code>\${Sample_ID}_MoleculesPerConfPositionedNuclei.mtx</code>	Expression table in sparse matrix format for nuclei positioned with 1 spatial location (column=cell barcodes, row=genes)
<code>\${Sample_ID}_barcodesConfPositionedNuclei.tsv</code>	Cell barcodes for nuclei positioned with 1 spatial location
<code>\${Sample_ID}_genesConfPositionedNuclei.tsv</code>	Genes for nuclei positioned with 1 spatial location
<code>\${Sample_ID}_ConfPositionedNucleiLocation.csv</code>	Spatial coordinates for nuclei positioned with 1 spatial location
<code>\${Sample_ID}_variable_features_clusters.txt</code>	Top cluster defining genes
<code>\${Sample_ID}_variable_features_spatial_moransi.txt</code>	Top spatially variable genes
<code>coordinates_df_\${Sample_ID}_ST.txt</code>	Spatial coordinates of all positioned nuclei regardless of positioning status
<code>cell_bc_plots</code>	A folder containing spatial barcode plots for randomly selected nuclei
<code>plots</code>	A folder containing additional quality control plots
<code>intermediates</code>	A folder containing expression table, spatial coordinates for all nuclei (regardless of positioning status) and all positioned nuclei (regardless of positioning confidence) in Seurat and sparse matrix format.

Table 7. Files in folder cell_bc_plots.

Subfolder	Definition
<code>cells_0_coordinates_assigned</code>	Spatial plots for 50 randomly selected nuclei with no spatial location assigned
<code>cells_1_coordinates_assigned</code>	Spatial plots for 50 randomly selected nuclei positioned with 1 spatial location
<code>cells_2_coordinates_assigned</code>	Spatial plots for 50 randomly selected nuclei positioned with 2 spatial locations
<code>cells_3_coordinates_assigned</code>	Spatial plots for 50 randomly selected nuclei positioned with 3 spatial locations

Table 8. Files in folder *intermediates*.

File	Definition
<code>\${Sample_ID}_Positioned_seurat_spatial.rds</code>	Seurat object with expression table, spatial coordinates, and clustering results for all positioned nuclei (including those with 2+ spatial locations)
<code>\${Sample_ID}_Positioned_anndata_matched.h5ad</code>	H5AD object with expression table, spatial coordinates, and clustering results for all positioned nuclei (including those with 2+ more spatial locations)
<code>\${Sample_ID}_MoleculesPerPositionedNuclei.mtx</code>	Expression table in sparse matrix format for all positioned nuclei (including those with 2+ spatial locations, column=cell barcodes, row=genes)
<code>\${Sample_ID}_barcodes_PositionedNuclei.tsv</code>	Cell barcodes for all positioned nuclei (including those with 2+ spatial locations)
<code>\${Sample_ID}_genes_PositionedNuclei.tsv</code>	Genes for all positioned nuclei (including those with 2+ spatial locations)
<code>\${Sample_ID}_PositionedNucleiLocation.csv</code>	Spatial coordinates for all positioned nuclei (including those with 2+ spatial locations)
<code>\${Sample_ID}_seurat_spatial.rds</code>	Seurat object with expression table, spatial coordinates, and clustering results for all nuclei (including those without spatial locations)
<code>\${Sample_ID}_anndata_matched.h5ad</code>	H5AD object with expression table, spatial coordinates, and clustering results for all nuclei (including those without spatial locations)
<code>\${Sample_ID}_MoleculesPer_Nuclei.mtx</code>	Expression table in sparse matrix format for all nuclei (including those without spatial locations; column=cell barcodes, row=genes)
<code>\${Sample_ID}_barcodes_Nuclei.tsv</code>	Cell barcodes for all nuclei (including those without spatial locations)
<code>\${Sample_ID}_genes_Nuclei.tsv</code>	Genes for all nuclei (including those without spatial locations)
<code>\${Sample_ID}_Location_Nuclei.csv</code>	Spatial coordinates for all nuclei (including those without spatial locations)

B. Definition and Interpretations of Quality Metrics

Below are definitions and interpretations for metrics in:

1. [\\${Sample_ID}_Trekker_Report.html](#)
2. [\\${Sample_ID}_summary_metrics.csv](#)

1. Metrics in `${Sample_ID}_Trekker_Report.html`

Table 9. Metrics in `${Sample_ID}_Trekker_Report.html`.

Metric name	Definition	Interpretation and suggested actions
Min spatial barcodes used to locate a nucleus centroid	Minimum number of spatial barcodes used to assign 1 spatial location. This is the parameter minPts used for DBSCAN. For each sample, it is dynamically chosen within an expected range (10x Chromium 3' v3.1, 3' v4 [4,15]; 10x Chromium Multiome [4,26]; BD Rhapsody WTA, ATAC+WTA, VDJ+WTA [9,31]; Parse Evercode WT v3, Illumina 3' RNA Prep [4,60]). The chosen number maximizes the metric "Pct nuclei positioned with 1 spatial location."	—
Total nuclei from single-nuclei sequencing library	Total high-quality nuclei retained from single-nuclei pipeline analysis	If significantly lower or higher than expected, check if the threshold set by the single-nuclei analysis pipeline to retain high-quality nuclei is appropriate. Alternatively, check if nuclei counting was performed appropriately. If lower than expected, check if the assayed tissue had compromised quality.
Pct nuclei in Trekker library	% nuclei from single-nuclei pipeline analysis found in the Trekker library	If <95%, check your samplesheet.csv to see if a wrong single-nuclei assay platform was selected for the Trekker pipeline analysis, and if your Trekker FASTQ inputs and single-nuclei output are incorrectly paired and are not from the same group of nuclei.
Pct nuclei in Trekker library with valid spatial barcodes	% nuclei from single-nuclei pipeline analysis found in the Trekker library with R2 matched to the Trekker bead barcode file	If <50%, check your samplesheet.csv to see if a wrong tile ID was used for the Trekker pipeline analysis.
Pct nuclei positioned	% nuclei from single-nuclei pipeline analysis with at least 1 spatial location assigned	If low, check for noticeable bead loss during nuclei counting. Alternatively, check if your Trekker library has been sufficiently sequenced to reach saturation, using metric 'Median useful reads per nuclei' (10x Chromium 3' v3.1, 3' v4, Multiome, Parse Evercode WT v3, Illumina 3' RNA Prep ≥ 400 , BD Rhapsody WTA, ATAC+WTA, VDJ+WTA ≥ 250).

Trekker™ Primary Analysis Pipeline User Manual

Metric name	Definition	Interpretation and suggested actions
Pct nuclei positioned with 1 spatial location	% nuclei from single-nuclei pipeline analysis with 1 spatial location assigned (singlets)	If low, check for noticeable bead loss or multiplets during nuclei counting. Alternatively, check if your Trekker library has been sufficiently sequenced to reach saturation, using metric 'Median useful reads per nuclei' (10x Chromium 3' v3.1, 3' v4, Multiome, Parse Evercode WT v3, Illumina 3' RNA Prep ≥ 400, BD Rhapsody WTA, ATAC+WTA, VDJ+WTA ≥ 250).
Pct nuclei positioned with 2+ spatial locations	% nuclei from single-nuclei pipeline analysis with 2 or more spatial locations assigned (likely multiplets, or nucleus with bead contamination, or occasional bioinformatic artifact)	If high, check for noticeable multiplets or bead loss during nuclei counting.
Total nuclei positioned with 1 spatial location	Nuclei from single-nuclei pipeline analysis with 1 spatial location assigned (singlets)	If low, check for noticeable bead loss or multiplets during nuclei counting. Alternatively, check if your Trekker library has reached saturation, using metric 'Median useful reads per nuclei' (10x Chromium 3' v3.1, 3' v4, Multiome, Parse Evercode WT v3, Illumina 3' RNA Prep ≥ 400, BD Rhapsody WTA, ATAC+WTA, VDJ+WTA ≥ 250).
Total readpairs in Trekker library	Total raw FASTQ read pairs in the Trekker library	If notably lower than expected, check sequencing run quality and if demultiplexing was performed appropriately.
Pct readpairs with proper structure	% Trekker spatial barcode reads with expected linker sequence (hamming dist ≤ 1)	If <80%, check your samplesheet.csv for a potential swap between Trekker R1 and R2 FASTQs or if a wrong primer was used during Trekker library preparation.
Pct readpairs matched to single nuclei barcodes	% Trekker read pairs properly structured and matched to nuclei barcodes from single-nuclei pipeline analysis	If <5%, check your samplesheet to see if your Trekker FASTQ inputs and single-nuclei pipeline output are incorrectly paired and are not from the same group of nuclei. Check also your nuclei prep quality. Excess debris or bead loss may decrease this metric.
Pct readpairs matched to single nuclei barcodes with valid spatial barcodes	% properly structured, single-nuclei matched read pairs that matched to the Trekker bead barcode file	If <50%, check your samplesheet to see if a wrong tile ID was used for the Trekker pipeline analysis.
Median spatial barcodes per nuclei	Median valid spatial barcodes per nuclei	—
Median spatial barcodes UMI per nuclei	Median number of molecules of valid spatial barcodes per nuclei	—
Median reads per nuclei	Median reads per nuclei that matched to nuclei barcodes and are properly structured	—

Metric name	Definition	Interpretation and suggested actions
Median useful reads per nuclei	Median reads per nuclei that matched to nuclei barcodes, properly structured, and matched to the Trekker bead barcode file	If <400 for 10x Chromium 3' v3.1, 3' v4, Multiome, Parse Evercode WT v3, Illumina 3' RNA Prep or <250 for BD Rhapsody WTA, ATAC+WTA, VDJ+WTA and no severe bead loss was observed, consider performing additional sequencing to obtain more reads and reach saturation.
Pct nuclei salvaged	% nuclei from single-nuclei pipeline analysis converted from 2+ to 1 spatial location assigned after bead removal	–

2. Metrics in \${Sample_ID}_summary_metrics.csv

Table 10. Sample metadata and analysis parameters.

Metric	Definition	Expected range or values
Sample ID	Sample ID used for naming pipeline output files	User-defined
Single cell assay	Single-nuclei assay platform selected for pipeline analysis	TrekkerU_C, TrekkerU_CX, TrekkerU_R, TrekkerU_M, TrekkerU_PIP, TrekkerQ_P TrekkerU_RVDJ, TrekkerU_RATAC
Tile ID	Tile ID used for pipeline analysis	CXXXXX_XXX
eps	Max radius allowed in a nucleus-associated cluster (defined by the pipeline)	50
Min spatial barcodes required to locate a nucleus centroid	Minimum number of spatial barcodes required to assign 1 spatial location. This is the parameter minPts used for DBSCAN. For each sample, it is dynamically chosen within an expected range (10x Chromium 3' v3.1, 3' v4 [4,15]; 10x Chromium Multiome [4,26]; BD Rhapsody WTA, ATAC+WTA, VDJ+WTA [9,31]; Parse Evercode WT v3, Illumina 3' RNA Prep [4,60]). The chosen number maximizes the metric 'Pct nuclei positioned with 1 spatial location'	TrekkerU_C, TrekkerU_CX: [4, 15] TrekkerU_R, TrekkerU_RVDJ, TrekkerU_RATAC: every other number in [9, 31] TrekkerU_PIP: [4, 60]
Maximum UMI cutoff	Max number of transcripts allowed for a valid spatial barcode. Spatial barcodes with more transcripts than the cutoff will be removed from spatial positioning as they likely are from dislodged Trekker beads.	256

Table 11. Nuclei positioning metrics.

Metric	Definition	Expected range or values
Total nuclei from single-nuclei sequencing library	Total high-quality nuclei retained from single-nuclei pipeline analysis	Dependent on single-nuclei platform and experimental design
Nuclei from single-nuclei sequencing library found in Trekker library	Number of nuclei from single-nuclei pipeline analysis found in Trekker library	Dependent on total nuclei identified from single-nuclei pipeline analysis
Pct nuclei in Trekker library	% nuclei from single-nuclei pipeline analysis found in the Trekker library	>95%
Nuclei from single-nuclei sequencing library found in Trekker library with valid spatial barcodes	Number of nuclei from single-nuclei pipeline analysis both found in Trekker library and matched to the Trekker bead barcode file	Dependent on total nuclei identified from single-nuclei pipeline analysis
Pct nuclei in Trekker library with valid spatial barcodes	% nuclei from single-nuclei pipeline analysis found in the Trekker library with R2 matched to the Trekker bead barcode file	>95%
Total nuclei positioned	Number of nuclei from single-nuclei pipeline analysis with at least one spatial location assigned	Dependent on total nuclei identified from single-nuclei pipeline analysis
Pct nuclei positioned	% nuclei from single-nuclei pipeline analysis with at least 1 spatial location assigned	>60%
Total nuclei positioned with 1 spatial location	Nuclei from single-nuclei pipeline analysis with 1 spatial location assigned (singlets)	Dependent on total nuclei identified from single-nuclei pipeline analysis
Pct nuclei positioned with 1 spatial location	% nuclei from single-nuclei pipeline analysis with 1 spatial location assigned (singlets)	>40%
Pct nuclei positioned with 2+ spatial locations	% nuclei from single-nuclei pipeline analysis with 2 or more spatial locations assigned (likely multiplets, or nucleus with bead contamination, or occasional bioinformatics artifact)	<20%

Table 12. Spatial barcode library quality.

Metric	Definition	Expected range or values
Total readpairs in Trekker library	Total raw FASTQ read pairs in the Trekker library	Dependent on single-nuclei platform and experimental design
Readpairs with proper structure	Trekker spatial barcode reads with expected linker sequence (hamming dist ≤ 1)	Dependent on total input reads
Pct readpairs with proper structure	% Trekker spatial barcode reads with expected linker sequence (hamming dist ≤ 1)	>80%
Readpairs used for matching to single nuclei barcodes	Properly structured read pairs used for barcode matching with single nuclei barcodes	Dependent on total input reads
Readpairs matched single nuclei barcodes	Properly structured read pairs containing barcodes matched to single nuclei barcodes (hamming dist = 0)	Dependent on total input reads

Metric	Definition	Expected range or values
Pct readpairs matched to single nuclei barcodes	% Trekker read pairs properly structured and matched to nuclei barcodes from single-nuclei pipeline analysis	>7%
Readpairs matched to single nuclei barcodes with valid spatial barcodes	Properly structured, single-nuclei matched read pairs that matched to the Trekker bead barcode file	Dependent on total input reads
Pct readpairs matched to single nuclei barcodes with valid spatial barcodes	% properly structured, single-nuclei matched read pairs that matched to the Trekker bead barcode file	>75%
Pct useful reads	% raw read pairs that are properly structured, matched to both nuclei barcodes from single-nuclei pipeline analysis and the Trekker bead barcode file	>6%

Table 13. Spatial barcode matching quality.

Metric	Definition	Expected range or values
Total sequenced spatial barcodes	Number of unique spatial barcodes identified	Dependent on total input reads
Sequenced spatial barcodes perfectly matched to whitelist	Number of spatial barcodes with exact match in the bead barcode file (hamming dist = 0)	Dependent on total input reads
Sequenced spatial barcodes approximately matched to whitelist	Number of spatial barcodes matched to the bead barcode file (hamming dist = 1)	Dependent on total input reads
Pct valid spatial barcodes	% unique spatial barcodes matched to the bead barcode file (hamming dist ≤1)	>60%

Table 14. Spatial barcode library depth. All expected range or values in this table are dependent on total input reads, tissue type, and single-nuclei platform.

Metric	Definition
Median spatial barcodes per nuclei	Median valid spatial barcodes per nuclei
Mean spatial barcodes per nuclei	Mean valid spatial barcodes per nuclei
Median spatial barcodes UMI per nuclei	Median number of molecules of valid spatial barcodes per nuclei
Mean spatial barcodes UMI per nuclei	Mean number of molecules of valid spatial barcodes per nuclei
Median reads per nuclei	Median reads per nuclei that matched to nuclei barcodes and are properly structured
Mean reads per nuclei	Mean reads per nuclei that matched to nuclei barcodes and are properly structured
Median useful reads per nuclei	Median reads per nuclei that matched to nuclei barcodes, properly structured, and matched to the Trekker bead barcode file
Mean useful reads per nuclei	Mean reads per nuclei that matched to nuclei barcodes, properly structured, and matched to the Trekker bead barcode file

Table 15. Signal to noise.

Metric	Definition	Expected range or values
Median proportion unique SB top cluster total	Median proportion of total spatial barcodes that contributed to spatial coordinate assignment per nucleus for all positioned nuclei	>0.02
Mean proportion unique SB top cluster total	Mean proportion of total spatial barcodes that contributed to spatial coordinate assignment per nucleus for all positioned nuclei	>0.04
Median proportion SB UMI top cluster total	Median proportion of total spatial barcode transcripts that contributed to spatial coordinate assignment per nucleus for all positioned nuclei	>0.04
Mean proportion SB UMI top cluster total	Mean proportion of total spatial barcode transcripts that contributed to spatial coordinate assignment per nucleus for all positioned nuclei	>0.07
Median mapped cells proportion unique SB top cluster	Median proportion of total spatial barcodes that contributed to spatial coordinate assignment per nucleus for confidently positioned nuclei	>0.05
Mean mapped cells proportion unique SB top cluster	Mean proportion of total spatial barcodes that contributed to spatial coordinate assignment per nucleus for confidently positioned nuclei	>0.06
Median mapped cells proportion SB UMI top cluster	Median proportion of total spatial barcode transcripts that contributed to spatial coordinate assignment per nucleus for confidently positioned nuclei	>0.07
Mean mapped cells proportion SB UMI top cluster	Mean proportion of total spatial barcode transcripts that contributed to spatial coordinate assignment per nucleus for confidently positioned nuclei	>0.1

Table 16. Nuclei salvaged.

Metric	Definition	Expected range or values
Median spatial barcodes per nuclei	% nuclei from single-nuclei pipeline analysis converted from 2+ to 1 spatial location assigned after bead removal	—

C. How Positioning Works

1. How Are the Nuclei Positioned Bioinformatically?

After nuclei are bioinformatically associated with their spatial barcodes, DBSCAN (Ester et al. 1996) is used to filter out the noise from the spatial barcodes before the nuclei are spatially positioned.

For each nucleus, DBSCAN clusters the spatial barcodes in their spatial coordinates and assigns a Cluster ID to each spatial barcode. DBSCAN identifies clusters based on density and assigns two types of Cluster IDs:

1. Cluster ID >0

Spatial barcodes assigned have discrete spatial clustering. They appear in high-density regions in the spatial coordinates and are signal spatial barcodes denoting the nucleus' true spatial location.

2. Cluster ID = 0

Spatial barcodes assigned have no clear spatial distribution. They appear in low-density regions in the spatial coordinates and are denoted as noise.

For nuclei with non-noise DBSCAN cluster (Cluster ID >0) identified, the pipeline filters out the noise spatial barcodes (Cluster ID = 0) and computes a UMI-weighted centroid for spatial barcodes in the cluster with Cluster ID = 1. This centroid is the nucleus' spatial coordinate.

2. Categories in Nuclei Positioning

The Trekker Primary Analysis Pipeline categorizes nuclei positioning based on the number of non-noise DBSCAN clusters (Cluster ID >0) assigned.

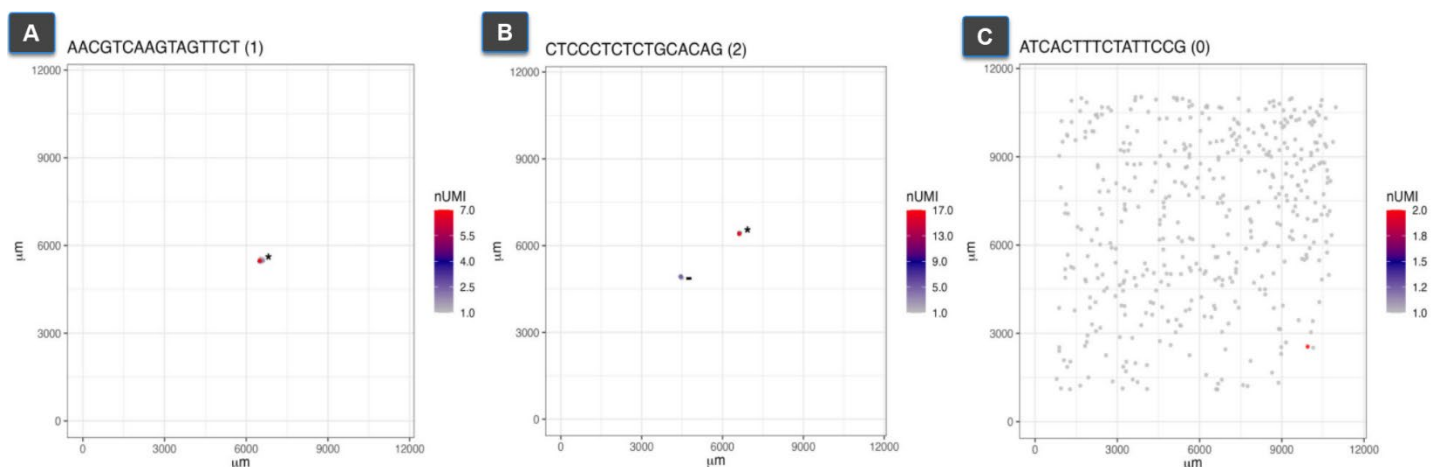


Figure 3. Examples of nuclei positioned with 1, 2, or no spatial locations. Each plot represents the spatial barcodes bioinformatically associated with a single nucleus. Each nucleus was assigned to one (1) spatial location (**Panel A**), two (2) spatial locations (**Panel B**), or no spatial location since only noise spatial barcodes were identified (**Panel C**). Each dot is a spatial barcode colored by the number of unique molecular identifiers (nUMI). Spatial barcodes are plotted in the spatial coordinates in μm. "*" points to the cluster with Cluster ID = 1, and "-" points to the cluster with Cluster ID >1.

1. "Nuclei positioned with 1 spatial location": Nuclei with 1 non-noise cluster assigned (Figure 3, Panel A)
2. "Nuclei positioned with 2+ spatial locations": Nuclei with >1 non-noise cluster assigned (Figure 3, Panel B). These nuclei are usually from one of the three scenarios. Nuclei under scenarios b. and c. below are salvageable for spatial positioning.
 - a. Two or more nuclei were captured in the same droplet or well
 - b. A nucleus and a Trekker bead were captured in the same droplet or well
 - c. A DBSCAN cluster was called as two clusters due to parameterization artifacts
3. "Nuclei positioned" = Nuclei in category 1 + Nuclei in category 2

4. We do not assign spatial coordinates to nuclei with only noise cluster (Cluster ID = 0) identified (Figure 3, Panel C)

3. Evaluate Your Trekker Library Quality Based on Nuclei Positioning

In the pie chart below (Figure 4, below), nuclei are categorized based on the number of non-noise clusters identified. In general, the larger the slice labeled with "1", the higher the quality of nuclei positioning.

See below for a detailed description of different slices of the pie.

Nuclei vs. the number of spatial locations assigned

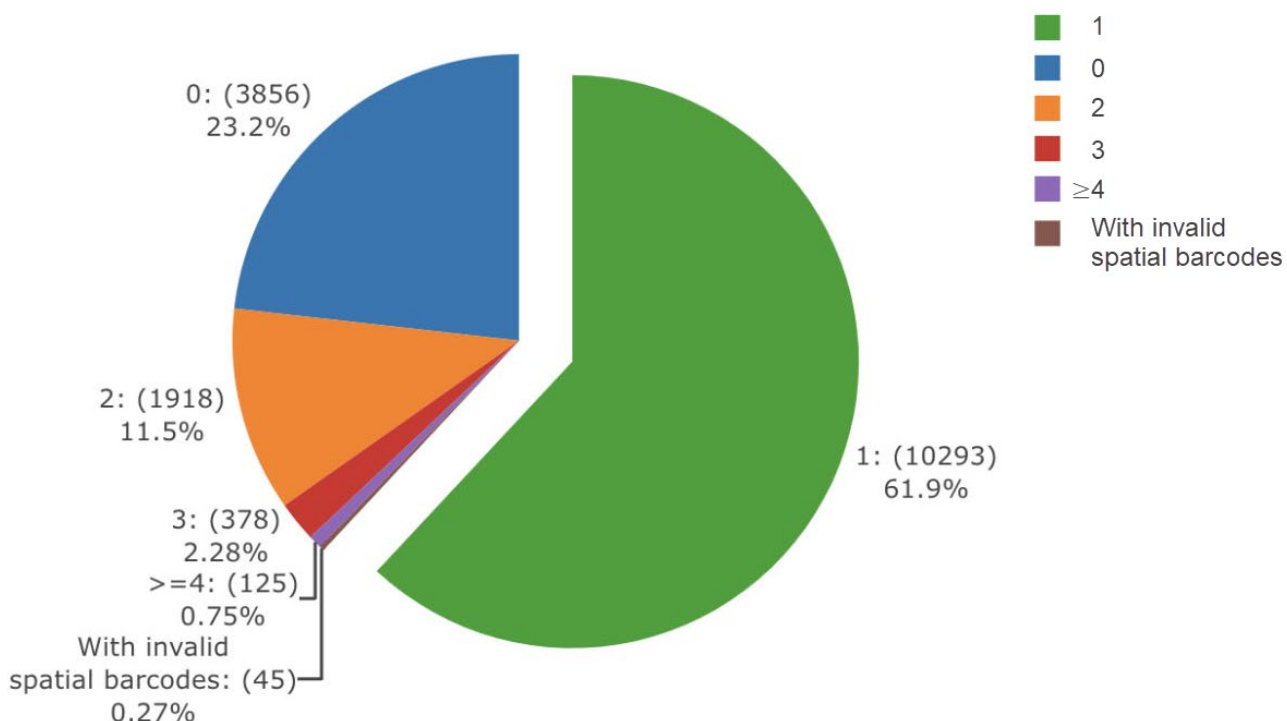


Figure 4. Assessment of the quality of spatial positioning.

- "1": (10293) 61.9%: Nuclei have 1 non-noise DBSCAN cluster assigned, consisting of 61.9% of all nuclei.
- "2": (1918) 11.5%: Nuclei have 2 non-noise DBSCAN clusters assigned, consisting of 11.5% of all analyzed nuclei.
- "3": (378) 2.28%: Nuclei have 3 non-noise DBSCAN clusters assigned, consisting of 2.28% of all analyzed nuclei.
- "≥4": (125) 0.75%: have 4 or more spatially discrete clusters assigned, consisting of 0.75% of all analyzed nuclei.

- "0": (3856) 23.2%: Nuclei have 0 non-noise DBSCAN clusters assigned. All spatial barcodes were assigned as Cluster ID = 0. There are 1702 nuclei, consisting of 23.2% of all nuclei.
- "With invalid spatial barcodes": (45) 0.27% have NO valid spatial barcodes identified and were not processed by DBSCAN.

DBSCAN requires two parameters as input: minPts and eps. To determine the optimal parameter set for each Trekker run, the pipeline iterates through different minPts within a range (e.g., 4–15) under a constant eps (50) and chooses the minPts that leads to the highest proportion of nuclei that are positioned with one spatial location (with one non-noise DBSCAN cluster assigned).

IV. References

Ester, M., Kriegel, H., Sander, J., et al. A density-based algorithm for discovering clusters in large spatial databases with noise. in *KDD* **96**, 226–231 (1996).

Appendix A. Preprocessing

A. Preprocessing: Illumina Single Cell 3' RNA Prep Kit (TrekkerU_PIP)

1. Overview

If you are processing data generated from the Illumina Single Cell 3' RNA Prep Kit (Fluent PIPseq V), additional format conversion is needed for:

- Trekker FASTQ pairs
- Single-nuclei bioinformatics pipeline (PIPseeker) output

This script converts the Trekker FASTQ pairs and single-nuclei pipeline output into formats that are compatible with the Trekker Primary Analysis Pipeline.

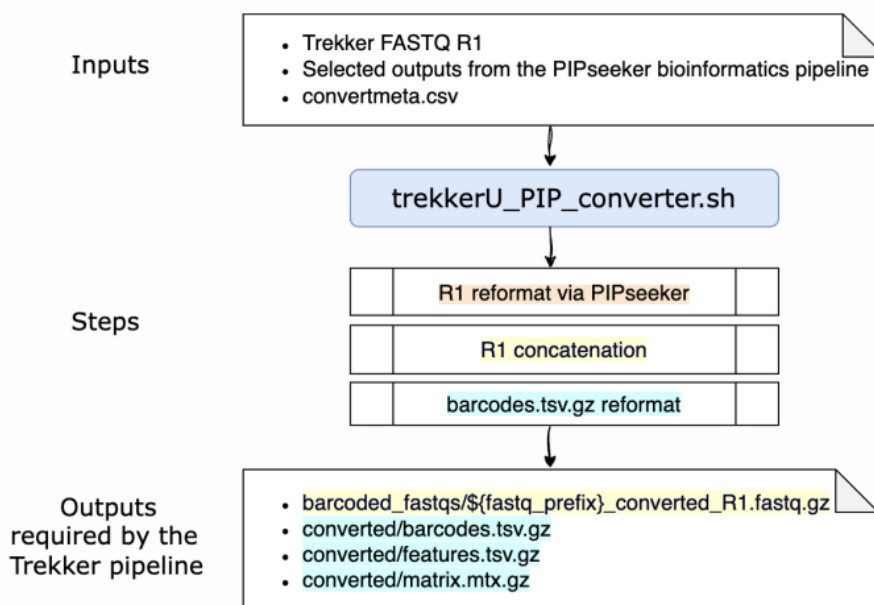


Figure 5. Workflow for format conversion using `trekkerU_PIP_converter.sh`.

After successful format conversion, return to Section II.E, "[#Check Input Format](#)" and proceed with running the Trekker Primary Analysis Pipeline.

Follow the instructions below to convert Trekker input format using `trekkerU_PIP_converter.sh`.

2. Download

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

```
wget https://www.takarabio.com/resourcedocument/x353768 -O - |
tar -xzf -
cd trekkerU_PIP_converter_v1.3.0
```

3. Set Up Dependencies

PIPseeker

Download PIPseeker by following the instructions on the [Fluent BioSciences PIPseeker](#) page*.

NOTE: You will need to complete a form to access the user guide and downloads. Once submitted, you'll gain access to the PIPseeker User Guide as well as the PIPseeker software package that can be downloaded for Linux.

For proper setup, refer to the "Setting up and Running the PIPseeker Executable" section in the PIPseeker User Guide.

*If you are unable to download PIPseeker, please email technical_support@takarabio.com.

4. Set Up Paths

Modify the bolded values below in `trekkerU_PIP_converter.sh` to match the paths on your platform.

```
#===== Define Directory Paths=====
PIPSEEKER_PATH=/home/tools/pipseeker-v3.3.0-linux/pipseeker
```

NOTES: See below for further details on the values you are modifying.

- `PIPSEEKER_PATH`: Absolute path to the pipseeker executable.

5. Input

Prepare the following inputs before triggering the converter

1. Trekker FASTQ R1 and R2

`TrekkerU_PIP_MouseBrain_R1_001.fastq.gz`

`TrekkerU_PIP_MouseBrain_R2_001.fastq.gz`

2. Selected outputs from the PIPseeker bioinformatics pipeline

`barcodes.tsv.gz`, `features.tsv.gz`, `matrix.mtx.gz`

NOTE: For PIPseeker, these files are usually in a folder and are provided in 5 sensitivity levels. Use the files in your preferred sensitivity level, e.g. 'sensitivity_5'.

3. Sample sheet

`convertmeta.csv`

Use this example [convertmeta.csv](#) as a starting point.

Columns in a sample sheet include:

`fastq_R1`, `fastq_prefix`, `sc_outdir`, `chem`

`/path/to/TrekkerU_PIP_MouseBrain_R1_001.fastq.gz,TrekkerU_PIP_MouseBrain,/path/to/TrekkerU_PIP_MouseBrain_scRNAseqOut/filtered_matrix/sensitivity_5/,V`

NOTES: See below for further details on columns in a sample sheet.

- `R1_fastq`: Absolute path to the Trekker FASTQ Read 1.
- `fastq_prefix`: Used for the Trekker FASTQ files.
- `sc_outdir`: Absolute path to the folder containing the following 3 required single-nuclei pipeline analysis output.
`barcodes.tsv.gz`, `features.tsv.gz`, `matrix.mtx.gz`
- `chem`: Version of the PIPseq assay used. Choose from v3, v4 or V PIPseeker chemistry.
- DO NOT swap the column orders in the sample sheet. Re-ordered columns will lead to converter failure.

6. Trigger the Converter

Follow the example below to trigger the converter.

NOTE: Before processing your own data, we recommend using our [example inputs](#) to test if the converter is properly set up.

```
bash trekkerU_PIP_converter.sh /path/to/convertmeta.csv
```

7. Monitor the Run

The STDOUT of a successful run will look similar to below:

```
Checking convertmeta.csv...
'/home/TrekkerU_PIP_converter_ExampleInput_MouseBrain/convertmeta.csv' exists.
Checking convertmeta.csv format...
convertmeta.csv is in Linux format.
Checking FASTQ R1...
'/home/TrekkerU_PIP_converter_ExampleInput_MouseBrain/TrekkerU_PIP_MouseBrain_R1_001.fastq.gz' exists.
Checking sc_outdir...
'/home/TrekkerU_PIP_converter_ExampleInput_MouseBrain/TrekkerU_PIP_MouseBrain_scRNAseqOutput/filtered_matrix/sensitivity_5/' exists
Checking snRNAseq output files in sc_outdir...
snRNAseq output files located.
Checking PIPseq chemistry...
Chemistry used: V
Starting the trekkerU_PIP_converter...
Start convertBarcode
convertBarcode Done
Start concatR1
concatR1 Done
Validating FASTQ R1...
FASTQ R1 is validated. Total reads in
'/home/TrekkerU_PIP_converter_ExampleOutput_MouseBrain_v1.3.0/barcoded_fastqs/TrekkerU_PIP_MouseBrain_converted_R1.fastq.gz' match the expected 979647 reads in the PIPseeker output file
'/home/TrekkerU_PIP_converter_ExampleOutput_MouseBrain_v1.3.0/metrics/barcode_stats.csv'
.
Start updateBarcodes
updateBarcodes Done
```


A log folder containing stepwise logs can be found in the folder where you stored your `convertmeta.csv`. Use the command below to access detailed logs.

```
$ cd /path/to/where-convertmeta.csv-is-stored
cd log/${fastq_prefix}
ls
```

8. Output

The TrekkerU_PIP converter outputs are deposited both in Trekker FASTQs and single-nuclei input directories specified in the `convertmeta.csv`

It contains the converted Trekker FASTQ pairs and the updated single-nuclei analysis output files.

IMPORTANT: Use the following paths for executing the main Trekker Primary Analysis Pipeline.



`fastq_1`: Absolute path to file `${fastq_prefix}_converted_R1.fastq.gz`
`fastq_2`: Absolute path to the `TrekkerU_PIP_MouseBrain_R2_001.fastq.gz` (original `R2.fastq.gz`)
`sc_outdir`: Absolute path to folder `converted`

```
.
├── barcoded_fastqs
│   ├── ${fastq_prefix}_converted_R1.fastq.gz
│   ├── barcoded_1_R1.fastq.gz
│   └── barcoded_1_R2.fastq.gz
├── barcodes
│   └── generated_barcode_read_info_table.csv
├── logs
│   ├── pipseeker_${Date_Time}.log
│   └── progress.log
├── metrics
│   └── barcode_stats.csv
└── run_config.csv
```

```

.
├── filtered_matrix
│   ├── sensitivity_5
│   │   ├── converted
│   │   │   ├── barcodes.tsv.gz
│   │   │   ├── features.tsv.gz
│   │   │   └── matrix.mtx.gz

```

9. Testing and Example Datasets

To test converter configuration, use our example inputs to trigger the converter and compare the results to our example output.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

1. Download example inputs

```
wget https://www.takarabio.com/resourcedocument/x353769 -O - |
tar -xzf -
```

2. Download example outputs

```
wget https://www.takarabio.com/resourcedocument/x353770 -O - |
tar -xzf -
```

For troubleshooting instructions, refer to Appendix E, Section C, "[Troubleshooting: Preprocessing for Illumina](#)".

B. Preprocessing: Parse Evercode WT v3 Kit (TrekkerQ_P)

1. Overview

If you are processing data generated from the Parse Evercode WT v3 Kit (TrekkerQ_P), you will need to demultiplex your Trekker FASTQ pairs into sample-specific FASTQ files (Figure 6). These sample-specific FASTQ pairs are the required input for the Trekker Primary Analysis Pipeline.

This script demultiplexes paired Read 1 and Read 2 from a Trekker library, using a “sample” barcode defined by a sample’s well coordinates in the first plate of Parse assay. The script outputs sample-specific Trekker FASTQ files.

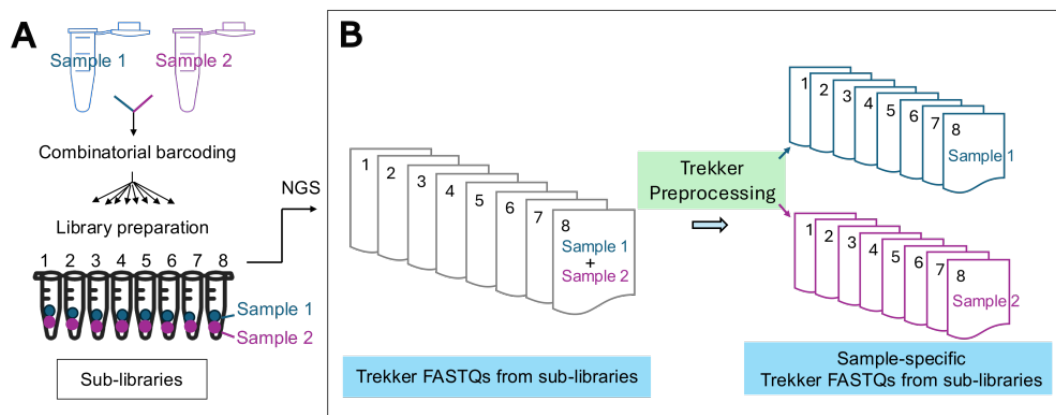


Figure 6. Illustration of how Trekker preprocessing step demultiplexes Trekker FASTQs to sample-specific Trekker FASTQs. Panel A. Single-cell workflow. **Panel B.** Trekker FASTQ preprocessing schematics.

After successful demultiplexing, use sample-specific FASTQ files as input for running the Trekker Primary Analysis Pipeline, as outlined in Section II, "[Trekker Pipeline Local Installation](#)".

Follow the instructions below to demultiplex your Trekker FASTQ pairs using the `fastq_sep_groups_v0.5.py` script.

2. System Requirements

- A Linux workstation with Python ≥ 3.10 or greater installed.
- Python module “pandas”. If you don’t have this module installed, you can run `pip install pandas`.

Read more on the [Parse support](#) page.

3. Download

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

```
wget https://www.takarabio.com/resourcedocument/x353771
```

4. Input

Prepare the following inputs before triggering the demultiplexer.

1. Multiplexed Trekker paired FASTQ files, for each sub-library.



IMPORTANT: For each Trekker+Parse run, you will likely have multiple Trekker sub-libraries, leading to multiple Trekker FASTQ pairs (defined by different Illumina sample indices). If so, you must demultiplex each pair individually.

Example file names:

```

Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_R1_001.fastq.gz
Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_R2_001.fastq.gz
Multiplexed_TrekkerQ_P_HumanBreastCancer_SL2_R1_001.fastq.gz
Multiplexed_TrekkerQ_P_HumanBreastCancer_SL2_R2_001.fastq.gz

```

5. Trigger the Demultiplexer

Follow the example below to trigger the demultiplexer.

```

python fastq_sep_groups_v0.5.py \
    --chemistry v3 \
    --fq1
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBrea
stCancer_SL1_R1_001.fastq.gz \
    --fq2
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBrea
stCancer_SL1_R2_001.fastq.gz \
    --opath /home/TrekkerQ_P_DEMUX_example_output/ \
    --group HBC1 A1-A12,B1-B12 \
    --group HBC2 C1-C12,D1-D12

python fastq_sep_groups_v0.5.py \
    --chemistry v3 \
    --fq1
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBrea
stCancer_SL2_R1_001.fastq.gz \
    --fq2
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBrea
stCancer_SL2_R2_001.fastq.gz \
    --opath /home/TrekkerQ_P_DEMUX_example_output/ \
    --group HBC1 A1-A12,B1-B12 \
    --group HBC2 C1-C12,D1-D12

```

NOTES: See below for further details on command arguments:

Read more about each input argument on the [Parse support](#) page.

- chemistry: Chemistry used for the Parse assay.
- fq1: Absolute path to Trekker FASTQ Read 1 without sample demultiplexing.
- fq2: Absolute path to Trekker FASTQ Read 2 without sample demultiplexing.
- opath: Absolute path to the desired output directory.

- group: Each group defines a sample by a desired name (in this example, we have used HBC1 and HBC2) and the sample's well coordinates in the first plate of the Parse assay. Well coordinates can be separated by a comma (,).

6. Monitor the Run

The STDOUT of a successful run will look similar to the example output below (next page):

```

# Set amplicon sequence for chemistry {args.chemistry}
# Start time 2025-07-11 13:33:18.878248
# Initializing...
# Loaded barcode data (9 bc sets)
# Sampling 500000 reads from
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_R2_001
.fastq.gz
# Collected 500000 data from 600000 fastq records
# Loaded fastq R2 sample
# Scoring fastq data against 3 kits, chemistry v3
#   WT_mini (n26_R1_v3_3) = 0.134
#   WT (n107_R1_v3_3) = 0.694
#   WT_mega (n222_R1_v3_3) = 0.431
# Best scoring kit = WT, 0.694
# Fasta R1
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_R1_001
.fastq.gz
#       R2
/home/TrekkerQ_P_DEMUX_example_input/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_R2_001
.fastq.gz
# Read range      No restrictions
# Kit              WT
# Barcodes         ['n107_R1_v3_3', 'v1', 'R3_v3']
# Amplicon         NNNNNNNNNN3333333ATGAGGGGTCAG2222222TCCAACCACCTC11111111
# Max edit dist 2
# Matching         Permissive first barcode matches
# Out path         ./Demux_out/
# Out base         Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_group
#
# Subsets          2 groups (by wells)
#   HBC1   A1-A12,B1-B12      (24): [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
16, 17, 18, 19, 20, 21, 22, 23, 24]
#   HBC2   C1-C12,D1-D12      (24): [25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37,
38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48]
#

```

```

# Output files
#   HBC1
#     /home/TrekkerQ_P_DEMUX_example_output/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_
group_HBC1_R1.fastq.gz
#   HBC2
#     /home/TrekkerQ_P_DEMUX_example_output/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_
group_HBC2_R1.fastq.gz
# Processing reads...
# record 100000
# record 1000000
# record 2000000
# record 3000000
# record 4000000
# record 5000000
# record 6000000
# record 7000000
# record 8000000
# record 9000000
# record 10000000
=====
number_of_reads  10000000
reads_too_short  0
reads_valid_bc   9666853
reads_ambig_bc1  452
bc1_Q30          0.935
bc2_Q30          0.964
bc3_Q30          0.957
bc_edit_dist_NA  333147
bc_edit_dist_0   8777028
bc_edit_dist_1   283751
bc_edit_dist_2   606074
index_with_outs  9666853
index_mult_outs  271160
index_no_outs    0
total_outputs    9938013

```

New files:

```
/home/TrekkerQ_P_DEMUX_example_output/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_group_HBC1_R1.fastq.gz (and R2) 4347111
```

New files:

```
/home/TrekkerQ_P_DEMUX_example_output/Multiplexed_TrekkerQ_P_HumanBreastCancer_SL1_group_HBC2_R1.fastq.gz (and R2) 5590902
```

Total time 0:02:15.99

7. Output

The demultiplexed Trekker FASTQ pairs are written to directory specified by the `--opath` parameter when executing the demux command.

```
/home/TrekkerQ_P_DEMUX_example_output/
```

8. Pairing Demuxed Trekker FASTQ with Parse Output for Trekker Pipeline Run

In Parse's Trailmaker output, each sample has multiple versions of three files below with varying multiplex levels. For Trekker, use the versions demultiplexed by both sequencing index and sample (Figure 7).

```
cell_metadata.csv.gz, all_genes.csv.gz, count_matrix.mtx.gz
```

NOTE: DO NOT use the files from `output_combined` and `all-samples`, as this reduces positioning rate.

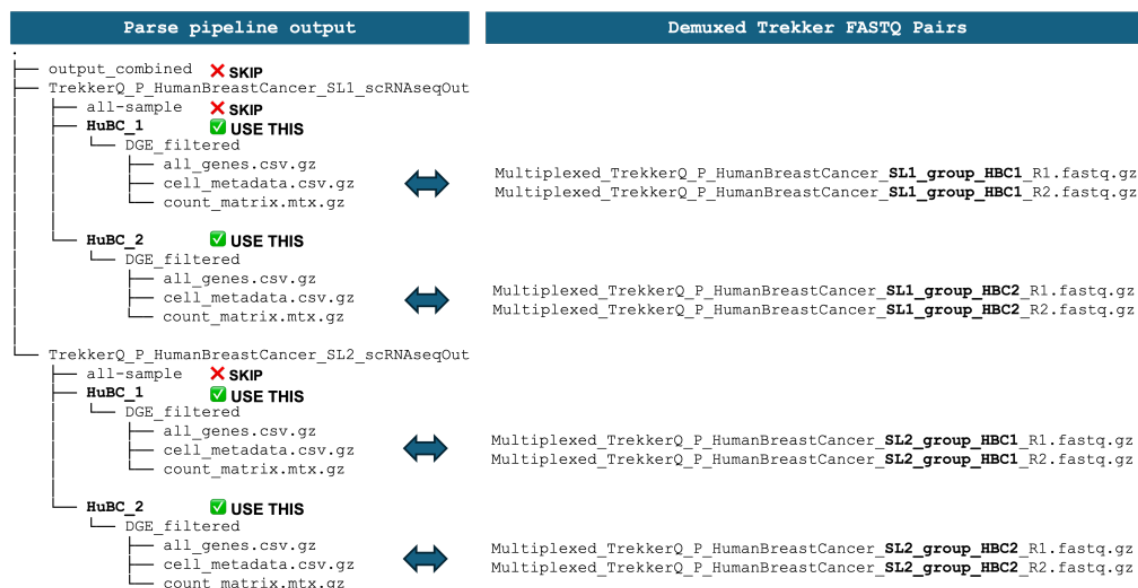


Figure 7. Illustration of how to pair demultiplexed Trekker FASTQs with Parse' pipeline output.

After running Trekker pipeline on the demuxed data, use the merging script from Appendix B, "[Trekker Output Merger](#)" to combine outputs for samples tagged with the same Trekker tile.

9. Testing and Example Datasets

To test the demultiplexer configuration, use our example inputs to trigger the demultiplexer and compare the results to our example output.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

1. Download example inputs.

```
wget https://www.takarabio.com/resourcedocument/x353772 -O - |
tar -xzf -
```

2. Download example outputs.

```
wget https://www.takarabio.com/resourcedocument/x353773 -O - |
tar -xzf -
```

Appendix B: Trekker Output Merger

If your tissue section was processed on a single Trekker tile but split into multiple single-nuclei reactions (e.g. processed in multiple 10x Chromium wells or BD Rhapsody lanes with different sample indices during sequencing), you can process the Trekker FASTQ pairs for each reaction and combine their pipeline outputs.

NOTE: The scenario above is different from sequencing a reaction in multiple sequencing lanes. In this case, concatenate FASTQ files from multiple lanes into a single Read 1 and a single Read 2, and process the concatenated FASTQ pair through the Trekker pipeline. Refer to Appendix C, "[How to](#)" for instructions.

A. Download trekker_merger-v1.3.0

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

Use the following command to download trekker_merger-v1.3.0.

```
wget https://www.takarabio.com/resourcedocument/x353750 -O - | tar -xzf -
```

Use the following command to navigate into the appropriate folder.

```
cd trekker_merger*
```

B. Set Up Paths

Modify the bolded values below in `trekker_merger.sh` (in the `trekker_merger-v1.3.0` script folder) to match the paths on your platform.

```
#===== Define Directory Paths =====
SCRIPT_DIR=/home/trekker_merger-v1.3.0/trekker
```

NOTES: See below for further details on the values you are modifying.

– `SCRIPT_DIR`: Absolute path to the script folder `trekker_merger-v1.3.0`.

C. Set Up Dependencies

The merger can be executed through Docker, Singularity, or Conda.

1. Docker

- If you have already pulled the trekker docker image (`tbusaspatial/trekker:v1.3.0`) for your Trekker pipeline, you don't need to do anything. (You pulled it [here](#).)
- If you have not pulled the docker image, follow steps below to pull it.
 1. Install docker if not already. Visit the [docker](#) page for installation instructions.
 2. Pull the trekker docker image, using the command below.

```
docker pull tbusaspatial/trekker:v1.3.0
```

2. Singularity

- If you used Singularity to execute dependencies for your Trekker pipeline, you don't need to do anything.
- If you did not use Singularity for your Trekker pipeline dependencies, make sure you have the singularity software installed.

Install singularity if you have not already. Visit the [singularity](#) page for installation instructions.

NOTE: You DO NOT need to download the singularity image `trekker-v1.3.0.sif`. It is in the folder `trekker_merger-v1.3.0`.

3. Conda

NOTE: Step 2 is required, regardless of whether you skipped Step 1.

- If you used Conda to execute dependencies for your Trekker pipeline, you can use the same environment for the merger and skip Step 1. You created the environment [here](#).
- If you did not use Conda for your Trekker pipeline dependencies, follow Step 1 to create the environment.
 1. You can create and activate a Conda environment `trekker` using the `environment.yml` (in the `trekker-v1.3.0` script folder) we created for the pipeline.

```
cd trekker-*
conda env create -f environment.yml
conda activate trekker
```

2. Modify the bolded values in the two lines shown below (next page) in `trekker_merger.sh` (in the `trekker_merger-v1.3.0` script folder) to match the paths on your platform.

```
#===== Define Conda Paths =====
CONDA_SH_PATH=/home/tools/miniconda3/etc/profile.d/conda.sh
PROFILE_CONDA_PATH=/home/tools/miniconda3/envs/trekker/
```

NOTES: See below for further details on the values you are modifying.

- /home/tools/miniconda3/: Replace this value with the path to your miniconda installation. To find the path, type:

```
conda info | grep -i 'base environment'
```
- /home/tools/miniconda3/envs/trekker/: Replace this value with absolute path to the trekker conda environment you created in Step 1. To find the path, type:

```
conda info --envs
```

3. Copy and paste the path displayed next to trekker, as bolded below.

```
trekker          /home/tools/miniconda3/envs/trekker
```

D. Check Input File Completeness

Selected trekker-v1.3.0 outputs for all samples to be merged are required and must conform to the following requirement:

For each output, the following two files should be present and in the following folder structure.

```
/path_to/output/intermediates/${SAMPLE_ID}_seurat_spatial.rds
/path_to/output/${SAMPLE_ID}_summary_metrics.csv
```

E. Input

Prepare the following inputs before triggering the merger.

NOTE: Before processing your own data, we recommend using [example inputs](#) to test if the pipeline is properly set up and if your files have the proper format.

1. Selected trekker-v1.3.0 pipeline outputs for all outputs to be merged

NOTE: Don't forget to check the [completeness of the outputs](#). Missing files will lead to merger failures.

2. Sample sheet

```
samplesheet.csv
```

Use this example [samplesheet.csv](#) as a starting point.



IMPORTANT: DO NOT use Excel to edit the sample sheet. Doing so will lead to unwanted modification to the 'new line' token and merger failure. Use a text editor (e.g., [Vim](#) or equivalent).

Columns in a sample sheet include:

```
sample,out_dir
TrekkerR_MouseEmbryo_LaneA,/path/to/20241020_TrekkerR_MouseEmbryo_LaneA/trekker_Trekker
TrekkerR_MouseEmbryo_LaneB,/path/to/20241020_TrekkerR_MouseEmbryo_LaneB/trekker_Trekker
TrekkerR_MouseEmbryo_LaneC,/path/to/20241020_TrekkerR_MouseEmbryo_LaneC/trekker_Trekker
```

NOTES: See below for further details on columns in a sample sheet.

- **sample:** Prefixes used for naming the Trekker pipeline output for each sample.
This value is used to locate the two required input files (above). If you named the file like
`/output/${SAMPLE_ID}_summary_metrics.csv`
you should use
`${SAMPLE_ID}`
for the value of **sample**.
This is also the value for **sample** you used in Section II.F.4, "[Sample Sheet](#)".
- **out_dir:** Absolute path to the trekker-v1.3.0 pipeline folder output for each sample.
- **DO NOT** swap the column orders in the sample sheet. Re-ordered columns will lead to merger failure.

F. Trigger the Merger

Follow the example below to trigger the merger. Modify the bolded values below to match the paths and files on your platform. Before processing your own data, we recommend using our [example inputs](#) to test if the merger is properly set up.

```
$ cd trekker_merger-v1.3.0
bash trekker_merger.sh \
/path/to/samplesheet.csv \
${output_dir} \
${sample_ID} \
${profile}
```



IMPORTANT: Please replace `/path/to/samplesheet.csv` with the absolute PATH to `samplesheet.csv` on your platform.

NOTES: See below for further details on input arguments.

- `/path/to/samplesheet.csv`: Absolute path to the `samplesheet.csv`.
- `${output_dir}`: Absolute path to a folder where the merger output should be directed to. If you used a value like this:
`/home/merger_out/MouseEmbryo`
the base directory `/home/merger_out/` should exist. The merger will create folder `MouseEmbryo`.
- `${sample_ID}`: Prefix for naming the final Trekker merged output. Avoid using `'` or space.

- `${profile}`: How to execute dependencies. Choose among Docker, Singularity, or Conda based on what you set up in Section II.D, "[Set Up Dependencies](#)". (Do not use quotes. Case does not matter.)

Example command:

```
$ cd trekker_merger-v1.3.0
bash trekker_merger.sh \
    /home/trekker/samplesheet.csv \
    /home/trekker/merger_out/MouseEmbryo/ \
    MouseEmbryo \
    singularity
```

G. Monitor the Run

The STDOUT of a successful pipeline run will look similar to the example below (next page):

```
Checking samplesheet... '/home/trekker_merger_example_input/samplesheet.csv' exists
Checking samplesheet format
Samplesheet is in Linux format.
Checking Output directory path...
'/home/merger_out exists'
Checking Output Sample ID prefix...
Using SAMPLE_ID: MouseEmbryo
TrekkerR_MouseEmbryo_LaneA
seurat_file and metrics_file for TrekkerR_MouseEmbryo_LaneA are copied to a temporary
input directory
TrekkerR_MouseEmbryo_LaneB
seurat_file and metrics_file for TrekkerR_MouseEmbryo_LaneB are copied to a temporary
input directory
TrekkerR_MouseEmbryo_LaneC
seurat_file and metrics_file for TrekkerR_MouseEmbryo_LaneC are copied to a temporary
input directory
All samples in the samplesheet have the same Trekker Tile ID: LTTAG0050_015 and are from
the same Single_cell_assay platform: TrekkerR
Starting trekker_merger
Running the trekker_merger using 'docker'
Start mergeSeurats
mergeSeurats Done
Start mergeMetrics
mergeMetrics Done
Start genreport
genreport Done
Your trekker output has been merged. Please find the results in
/home/merger_out/MouseEmbryo
```

A log folder containing stepwise logs can be found in the folder where you stored your `samplesheet.csv`. Use the command below to access detailed logs, modifying the bolded values to match the paths and files on your platform.

```
$ cd /path/to/where-samplesheet.csv-is-stored
cd log/${sample_ID}
ls
```

H. Output

The merger outputs are in `${output_dir}`. It contains key output files for downstream analysis, as shown below. Detailed descriptions of each key output file can be found in Section III, "[Interpreting Trekker Pipeline Output](#)".

```
├── intermediates
├── ${sample_ID}_Report.html
├── ${sample_ID}_summary_metrics_merged.csv
├── ${sample_ID}_ConfPositioned_anndata_merged.h5ad
├── ${sample_ID}_ConfPositioned_seurat_spatial_merged.rds
├── ${sample_ID}_MoleculesPer_ConfPositionedNuclei_merged.mtx
├── ${sample_ID}_barcodes_ConfPositionedNuclei_merged.tsv
├── ${sample_ID}_genes_ConfPositionedNuclei_merged.tsv
├── ${sample_ID}_Location_ConfPositionedNuclei_merged.csv
├── ${sample_ID}_variable_features_clusters_merged.csv
└── ${sample_ID}_variable_features_spatial_moransi_merged.txt
```

I. Testing and Example Datasets

To test pipeline configuration, use our example inputs to trigger the merger and compare the results to our example outputs.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

1. Download example inputs

```
wget https://www.takarabio.com/resourcedocument/x353774 -O - |
tar -xzf -
```

2. Download example outputs

```
wget https://www.takarabio.com/resourcedocument/x353748 -O - |
tar -xzf -
```

Appendix C. How to

A. How to Concatenate Multiple FASTQ Files into One FASTQ

In a terminal, type the following command:

```
cat ${s1_1}_R1.fastq.gz ${s1_2}_R1.fastq.gz > ${sample1}_R1.fastq.gz
```

NOTE: Don't forget to replace `${s1_*}` and `${sample1}` with the prefixes of your own FASTQs

In the command above, three `R1.fastq.gz` files are concatenated into one FASTQ `${sample1}_R1.fastq.gz`. You can concatenate as many FASTQs as needed by appending file names to the `cat` command before `>`.

B. How to Compress a FASTQ File into .gz Format

In a terminal, type the following command:

```
gzip ${sample1}.fastq
```

Replace `${sample1}` with the prefix of your own FASTQ.

Appendix D. Alternative Trekker Pipeline Input Preparation

A. Input Preparation: BD Rhapsody Single-Cell TCR/BCR Next + WTA Kit (TrekkerU_RVDJ)

1. Overview

If you are processing data generated from the BD Rhapsody Single-Cell TCR/BCR Next + WTA Kit, additional input and a slightly different `samplesheet.csv` are required by the trekker pipeline.

After you have all the required input, return to Section II.G, "[Trigger the Pipeline](#)" and proceed with running the Trekker Primary Analysis Pipeline.

Follow the instructions below to prepare your input.

2. Input

Prepare the following inputs before triggering the pipeline.

NOTE: Before processing your own data, we recommend using [example inputs for TrekkerU_RVDJ](#) to test if the pipeline is properly set up and if your files have the proper format.

1. Bead barcode file

```
${Tile_ID}_BeadBarcodes.txt
```

Download the bead barcode file (e.g., `U0001_001_BeadBarcodes.txt`) using our [Tile bead barcode file retrieval tool](#).

NOTE: Don't forget to unzip the downloaded file before use.

2. Trekker paired FASTQ files

Example file names:

```
TrekkerU_RVDJ_HumanBreastCancer_R1_001.fastq.gz
```

```
TrekkerU_RVDJ_HumanBreastCancer_R2_001.fastq.gz
```

NOTE: Don't forget to check the format conformity of these files (Section II.E, "[Check Input Format](#)"). Improper format will lead to pipeline failures.

3. Selected outputs from the bioinformatics pipeline of the single-nuclei sequencing library

```
barcodes.tsv.gz, features.tsv.gz, matrix.mtx.gz
```

Follow the steps below to ensure the Trekker pipeline locate these files.

- For each snRNAseq reaction, generate the three output files using the associated bioinformatics pipeline.
- Identify the 'quality-filtered' version of these three files and place them in a single folder.
- Update the column `sc_outdir` in `samplesheet.csv` such that these three files can be found within the folder defined by the `sc_outdir`.

NOTE: Don't forget to check the format conformity of these files (Section II.E, "[Check Input Format](#)"). Improper format will lead to pipeline failures.

4. VDJ-associated output from the bioinformatics pipeline of the single-nuclei sequencing library.

```
TrekkerU_RVDJ_HumanBreastCancer_VDJ_Seurat.rds
```

5. Sample sheet

```
samplesheet.csv
```

Use this example [TrekkerU_RVDJ_samplesheet.csv](#) as a starting point to enter information on your sample to be analyzed.

NOTES:

- Prepare a separate `samplesheet.csv` for each reaction (i.e., a pair of Trekker FASTQs). The Trekker Primary Analysis Pipeline does NOT support multiple rows (reactions) in a sample sheet.
- DO NOT swap the column orders in the sample sheet. Re-ordered columns will lead to pipeline failure.

Columns in a sample sheet include:

```
sample,sc_sample,experiment_date,barcode_file,fastq_1,fastq_2,sc_outdir,sc_platform,profile,subsample,cores,scmulti_outdir
```

```
TrekkerU_RVDJ_HumanBreastCancer,TrekkerU_RVDJ_HumanBreastCancer_VDJ,20250331,/path/to/U0043_014_BeadBarcodes.txt,/path/to/TrekkerU_RVDJ_HumanBreastCancer_R1_001.fastq.gz,/path/to/TrekkerU_RVDJ_HumanBreastCancer_R2_001.fastq.gz,/path/to/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_HumanBreastCancer_scRNAseqOut/TrekkerU_RVDJ_HumanBreastCancer_RSEC_MolsPerCell_MEX/,TrekkerU_RVDJ,singularity,No,8,/path/to/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_HumanBreastCancer_scRNAseqOut
```

NOTES: See below for further details on columns in a sample sheet.

- **sample:** Prefix for naming the Trekker pipeline output. Avoid using ‘.’ or space.
- **sc_sample:** Prefix for the following VDJ-associated single-nuclei pipeline analysis output. (Use a desired string with no space or special characters.)
`${sc_sample}_Seurat.rds`
- **experiment_date:** Date of analysis in format YYYYMMDD. This will be used as part of the output folder name.
- **barcode_file:** Absolute path to the Trekker bead barcode file.
- **fastq_1:** Absolute path to the Trekker FASTQ Read 1.
- **fastq_2:** Absolute path to the Trekker FASTQ Read 2.
- **sc_outdir:** Absolute path to the folder containing the three following required single-nuclei pipeline analysis output.
`barcodes.tsv.gz, features.tsv.gz, matrix.mtx.gz`
- **sc_platform:** Select a value from the second column of the table below (e.g., TrekkerU_RVDJ), based on the single-cell assay platform you are using from the first column:



IMPORTANT: Make sure to choose the correct value. An incorrect value will lead to poor positioning results.

Single-cell assay platform	sc_platform value
BD Rhapsody Single-Cell TCR/BCR Next + WTA, polyT capture	TrekkerU_RVDJ

- **profile:** How to execute dependencies. Choose among Docker, Singularity, or Conda based on what you set up in Section II.D, "[Set Up Dependencies](#)". (Do not use quotes. Case does not matter.)
- **subsample:** If subsampling should be performed for spatial positioning parameterization. Use “no” unless you have $>5 \times 10^6$ nuclei to be positioned. (Do not use quotes. Case does not matter.)

- `cores`: Number of cores to use for the spatial positioning step. We recommend 8, which is sufficient for most cases.
- `scmulti_outdir`: Absolute path to the folder containing the following VDJ-associated single-nuclei pipeline analysis output.
`${sc_sample}_Seurat.rds`

3. Testing and example datasets

To test pipeline configuration, use our example inputs to trigger the pipeline and compare the results to our example output.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

1. Download example inputs

Single-nuclei assay platform: TrekkerU_RVDJ

```
wget https://www.takarabio.com/resourcedocument/x353761 -O - |
tar -xzf -
```

2. Download example outputs

Single-nuclei assay platform: TrekkerU_RVDJ

```
wget https://www.takarabio.com/resourcedocument/x353767 -O - |
tar -xzf -
```

For troubleshooting instructions, refer to Appendix E, Section D, "[Troubleshooting: Input Preparation for BD Rhapsody Single-Cell TCR/BCR Next](#)".

B. Input Preparation: BD Rhapsody Single-Cell ATAC-Seq + mRNA WTA Kit (TrekkerU_RATAC)

1. Overview

If you are processing data generated from the BD Rhapsody Single-Cell ATAC-Seq + mRNA WTA Kit (TrekkerU_RATAC), additional input and a slightly different `samplesheet.csv` are required by the Trekker Primary Analysis Pipeline.

After you have all the required input, return to Section II.G, "[Trigger the Pipeline](#)" and proceed with running the Trekker Primary Analysis Pipeline.

Follow the instructions below to prepare your input.

2. Input

Prepare the following inputs before triggering the pipeline.

NOTE: Before processing your own data, we recommend using [example inputs for TrekkerU_RATAC](#) to test if the pipeline is properly set up and if your files have the proper format.

1. Bead barcode file

```
${Tile_ID}_BeadBarcodes.txt
```

Download the bead barcode file (e.g., U0001_001_BeadBarcodes.txt) using our [Tile bead barcode file retrieval tool](#).

NOTE: Don't forget to unzip the downloaded file before use.

2. Trekker paired FASTQ files

Example file names:

```
TrekkerU_RATAC_MouseKidney_R1_001.fastq.gz
```

```
TrekkerU_RATAC_MouseKidney_R2_001.fastq.gz
```

NOTE: Don't forget to check the format conformity of these files (Section II.E, "[Check Input Format](#)"). Improper format will lead to pipeline failures.

3. Selected outputs from the bioinformatics pipeline of the single-nuclei sequencing library

```
barcodes.tsv.gz, features.tsv.gz, matrix.mtx.gz
```

Follow the steps below to ensure the Trekker pipeline locate these files.

- For each snRNAseq reaction, generate the three output files using the associated bioinformatics pipeline.
- Identify the 'quality-filtered' version of these three files and place them in a single folder.
- Update the column `sc_outdir` in `samplesheet.csv` such that these three files can be found within the folder defined by the `sc_outdir`.

NOTE: Don't forget to check the format conformity of these files (Section II.E, "[Check Input Format](#)"). Improper format will lead to pipeline failures.

4. ATAC-associated output from the bioinformatics pipeline of the single-nuclei sequencing library.

```
atac-barcodes.tsv.gz, atac-features.tsv.gz, atac-matrix.mtx.gz
```

5. Sample sheet

```
samplesheet.csv
```

Use this example [TrekkerU_RATAC_samplesheet.csv](#) as a starting point to enter information on your sample to be analyzed.

NOTES:

- Prepare a separate `samplesheet.csv` for each reaction (i.e., a pair of Trekker FASTQs). The Trekker Primary Analysis Pipeline does NOT support multiple rows (reactions) in a sample sheet.
- DO NOT swap the column orders in the sample sheet. Re-ordered columns will lead to pipeline failure.

Columns in a sample sheet include:

```
sample,sc_sample,experiment_date,barcode_file,fastq_1,fastq_2,sc_outdir,sc_platform,profile,subsample,cores,scmulti_outdir
TrekkerU_RATAC_MouseKidney,TrekkerU_RATAC_MouseKidney_scRNAseq,20250624,/path/to/U0027_016_BeadBarcodes.txt,/path/to/TrekkerU_RATAC_MouseKidney_R1_001.fastq.gz,/path/to/TrekkerU_RATAC_MouseKidney_R2_001.fastq.gz,/path/to/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_RSEC_MolsPerCell_MEX/,TrekkerU_RATAC,singularity,no,8,/path/to/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/
```

NOTES: See below for further details on columns in a sample sheet.

- `sample`: Prefix for naming the Trekker pipeline output. Avoid using ‘.’ or space.
- `sc_sample`: Prefix for single-nuclei pipeline analysis output. (For book-keeping purposes only. Use a desired string with no space or special characters.)
- `experiment_date`: Date of analysis in format YYYYMMDD. This will be used as part of the output folder name.
- `barcode_file`: Absolute path to the Trekker bead barcode file.
- `fastq_1`: Absolute path to the Trekker FASTQ Read 1.
- `fastq_2`: Absolute path to the Trekker FASTQ Read 2.
- `sc_outdir`: Absolute path to the folder containing the three following required single-nuclei pipeline analysis output.
`barcodes.tsv.gz, features.tsv.gz, matrix.mtx.gz`
- `sc_platform`: Select a value from the second column of the table below (e.g., TrekkerU_RVDJ), based on the single-cell assay platform you are using from the first column:



IMPORTANT: Make sure to choose the correct value. An incorrect value will lead to poor positioning results.

Single-cell assay platform	sc_platform value
BD Rhapsody™ Single-Cell ATAC-Seq + mRNA Whole Transcriptome Analysis Kit	TrekkerU_RATAC

- `profile`: How to execute dependencies. Choose among Docker, Singularity, or Conda based on what you set up in Section II.D, "[Set Up Dependencies](#)". (Do not use quotes. Case does not matter.)
- `subsample`: If subsampling should be performed for spatial positioning parameterization. Use "no" unless you have $>5 \times 10^6$ nuclei to be positioned. (Do not use quotes. Case does not matter.)
- `cores`: Number of cores to use for the spatial positioning step. We recommend 8, which is sufficient for most cases.
- `scmulti_outdir`: Absolute path to the folder containing the following ATAC-associated single-nuclei pipeline analysis output.
`atac-barcodes.tsv.gz, atac-features.tsv.gz, atac-matrix.mtx.gz`

3. Testing and example datasets

To test pipeline configuration, use our example inputs to trigger the pipeline and compare the results to our example output.

NOTE: For best results, use Adobe Acrobat Reader to view this user manual. If you are copying and pasting these commands, ensure that the commands include the correct punctuation (e.g., hyphens) before execution.

1. Download example inputs

Single-nuclei assay platform: TrekkerU_RATAC

```
wget https://www.takarabio.com/resourcedocument/x353760 -O - |
tar -xzf -
```

2. Download example outputs

Single-nuclei assay platform: TrekkerU_RATAC

```
wget https://www.takarabio.com/resourcedocument/x353766 -O - |
tar -xzf -
```

For troubleshooting instructions, refer to Appendix E, Section E, "[Troubleshooting: Input Preparation for BD Rhapsody Single-Cell ATAC-Seq](#)".

Appendix E. Troubleshooting

A. Troubleshooting: Local Installation

- **A required input is missing.**

You will see an error message like the one below. If this happens, check your `samplesheet.csv` to ensure the path for the missing input is correct.

```

Checking samplesheet...
'/home/TrekkerU_C_ExampleInput_MouseBrain/samplesheet.csv' exists.

Checking Trekker FASTQ R1...
'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R1_001.fastq.gz' exists.

Checking Trekker FASTQ R2...
'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R2_001.fastq.gz' exists.

Checking Trekker tile spatial barcode whitelist...
'/home/TrekkerU_C_ExampleInput_MouseBrain/LTTag0053_003_BeadBarcodes.txt' exists.

Checking snRNAseq outputs...
Abort the Pipeline. Error: Unable to locate single-nuclei RNAseq pipeline output folder. Please ensure the path specified by 'sc_outdir' in your samplesheet exists.

```

- **A step failed.**

The error message will direct you to the log of the failed step. If this happens, review the step-log for details or share it with technical_support@takarabio.com.

```

...

Checking snRNAseq outputs...
snRNAseq outputs located.

Starting Trekker Analysis (TrekkerU_C) using singularity

Start fastq_parser

Abort the Pipeline. Error: fastq_parser failed, Please refer to
'/home/TrekkerU_C_ExampleInput_MouseBrain/log/TrekkerU_C_MouseBrain/fastq_parser.log' for details.

```

- (TrekkerU_RVDJ only) **The required VDJ-associated input is missing.**

You will see error messages like below. If this happens, check your `samplesheet.csv` to ensure `sc_sample` and `scmulti_outdir` values are correct such that the path to the VDJ-associated file follows this pattern:

```
${scmulti_outdir}/${sc_sample}_Seurat.rds
```

- Error due to missing `scmulti_outdir` column in the `samplesheet.csv`

```
...
Checking scmulti_outdir...
Abort the Pipeline. Error: The scmulti_outdir value in the samplesheet
is empty. Please check your samplesheet and provide a valid value.
```

- o **Error due to incorrect scmulti_outdir value in the samplesheet.csv**

```
...
Checking scmulti_outdir...
Error message:
Abort the Pipeline. Error: Unable to locate folder
'/home/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_Human
BreastCancer_scRNAseqOut/'.
Please ensure the path specified by 'scmulti_outdir' in your
samplesheet exists.
```

- o **Error due to incorrect sc_sample value in the samplesheet.csv**

```
... Checking scmulti_outdir...
'/home/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_Hum
anBreastCancer_scRNAseqOut/' exists.
Checking scmulti_outdir/vdj_seurat file...
Error message:
Abort the Pipeline. Error: File
'/home/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_Hum
anBreastCancer_scRNAseqOut/TrekkerU_RVDJ_HumanBreastCancer_VDJ_Seurat
.rds' does not exist.
Please check your samplesheet and provide a valid value to
'sc_sample' (prefix for vdj_seurat file) or ensure that file
'TrekkerU_RVDJ_HumanBreastCancer_VDJ_Seurat.rds' exists in the path
specified by 'scmulti_outdir' in your samplesheet.
```

- **(TrekkerU_RATAC only) The required ATAC-associated input is missing.**

You will see error messages like below. If this happens, check your `samplesheet.csv` to ensure the `scmulti_outdir` value is correct and it contains these three ATAC-associated files:

```
${scmulti_outdir}/atac-barcodes.tsv.gz
${scmulti_outdir}/atac-features.tsv.gz
${scmulti_outdir}/atac-matrix.mtx.gz
```


- o Error due to missing `scmulti_outdir` column in the `samplesheet.csv`

```
...
Checking scmulti_outdir...

Abort the Pipeline. Error: The scmulti_outdir value in the
samplesheet is empty. Please check your samplesheet and provide a
valid value.
```

- o Error due to incorrect `scmulti_outdir` value in the `samplesheet.csv`

```
...
Checking scmulti_outdir...

Abort the Pipeline. Error: Unable to locate the multiomic output folder
'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseKidn
ey_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/' from the
single-nuclei pipeline. Please ensure the path specified by
'scmulti_outdir' in your samplesheet exists.
```

- o Error due to missing files in folder defined by `scmulti_outdir`

```
...
Checking scmulti_outdir...

'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseK
idney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/'
exists.

Checking scmulti_outdir/atac files...

'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseK
idney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/atac
-barcodes.tsv.gz' exists.

'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseK
idney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/atac
-features.tsv.gz' exists.

Abort the Pipeline. Error: File
'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseK
idney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/atac
-matrix.mtx.gz' does not exist. Please check your samplesheet and
provide a valid path to the 'scmulti_outdir' value or ensure that
the 'atac-matrix' can be located in the path specified by
'scmulti_outdir' of your samplesheet.
```

- (TrekkerU_RATAC only) **The numbers of nuclei in RNA-Seq and ATAC-Seq matrices do not match.**

You will see error messages like below. If this happens, check your `samplesheet.csv` to ensure `sc_outdir` and `scmulti_outdir` point to the RNA-seq and ATAC-seq matrices that came from the same run of the single-cell bioinformatics pipeline.

```
...

Checking the number of nuclei in snRNAseq and snATACseq count matrices
from the single-nuclei pipeline...

Abort the Pipeline. Error: Different number of nuclei found. Please
ensure snRNAseq and snATACseq count matrices have the same number of
nuclei.
```

B. Troubleshooting: Merging Pipeline Output

- **The path to the script folder is not defined.**

You will see an error message like the one below. If this happens, define path following instructions in Appendix B, Section B, "[Set Up Paths](#)".

STDOUT:

```
...

Abort the trekker_merger. Error: mergeSeurats failed, Please refer to
/trekker_merger_example_input/log/MouseEmbryo/mergeSeurats.log for
details.
```

mergeSeurats.log:

```
...

Fatal error: cannot open file '/home/trekker_merger-
v1.3.0//mergeSeurats.R': No such file or directoryERROR
conda.cli.main_run:execute(125): `...` failed. (See above for error)
```

- **A required input is missing.**

You will see an error message like the one below. If this happens, check your `samplesheet.csv` to ensure the path for the missing input is correct.

```
...

Seurat file not found for TrekkerR_MouseEmbryo_LaneD. Please check your
samplesheet and provide a valid path to the output folder for sample
TrekkerR_MouseEmbryo_LaneD.

Metrics file not found for TrekkerR_MouseEmbryo_LaneD. Please check
your samplesheet and provide a valid path to the output folder for
sample TrekkerR_MouseEmbryo_LaneD.
```

- **Intermediate folder input exists.**

If this happens, remove the input folder and re-trigger the run.

...

/home/TrekkerR_ExampleInput_MouseEmbryo/input/ exists. Please delete the input folder or update its name and start the trekker_merger again.

- **Outputs being merged are not from the same Trekker tile or single-nuclei platform.**

If this happens, modify your samplesheet.csv so only outputs from the same tile and single-nuclei platform are included.

...

Error: Trekker Tile ID must be the same for all samples. Please check your samplesheet and provide the output folders for samples from the same Trekker tile.

Error: The Single_cell_assay must be the same for all samples. Please check your samplesheet and provide the output folders for samples processed by the same single cell assay platform.

- **A step failed.**

The error message will direct you to the log of the failed step. If this happens, review the step-log for details or share it with technical_support@takarabio.com.

...

Running the trekker_merger using 'conda'

Start mergeSeurats

Abort the trekker_merger. Error: mergeSeurats failed. Please refer to /home/TrekkerR_ExampleInput_MouseEmbryo/log/TrekkerR_MouseEmbryo/mergeSeurats.log for details

C. Troubleshooting: Preprocessing for Illumina Single Cell 3' RNA Prep Kit (TrekkerU_PIP)

- **A required input is missing.**

You will see an error message like the one below. If this happens, check your `convertmeta.csv` to ensure the path for the missing input is correct.

```
Checking samplesheet...  
  
'/home/trekkerU_PIP_converter_example_input/convertmeta.csv' exists  
  
Checking samplesheet format...  
  
samplesheet is in Linux format.  
  
Checking FASTQ R1...  
  
Abort the trekkerU_PIP_converter. Error: File  
'/home/trekkerU_PIP_converter_example_input/TrekkerU_PIP_MouseBrain_R1_0  
01.fastq.gz' does not exist. Please provide a valid path to the  
'fastq_R1' in  
'/home/trekkerU_PIP_converter_example_input/convertmeta.csv'.
```

- **Incorrect chemistry argument used in the sample sheet.**

If this happens, modify your `convertmeta.csv` to ensure that the value for `chem` is one of the following PIPseeker chemistry: v3, v4, or V.

```
...  
  
Checking snRNAseq output files in sc_outdir...  
  
snRNAseq output files located.  
  
Checking PIPseq chemistry...  
  
Chemistry used: v5  
  
Abort the trekkerU_PIP_converter. Error: The PIPseq chemistry must be  
v3, v4 or V. Please provide a valid value to the 'chem' in  
'/home/trekkerU_PIP_converter_example_input/convertmeta.csv'
```

- **Validation failed for updated FASTQ R1**

FASTQ R1 retained by pipseeker and updated FASTQ R1 generated does not have same number of reads. If this happens, please delete all the intermediate files and retrigger a clean trekkerU_PIP_converter run.

```
...
Starting the trekkerU_PIP_converter...
Start convertBarcode
convertBarcode Done
Start concatR1
concatR1 Done
Validating FASTQ R1...

Abort the trekkerU_PIP_converter. Error: Total reads mismatch after
concatenation. Found 900000 reads in
'/home/trekkerU_PIP_converter_example_input/barcoded_fastqs/TrekkerU_PIP
_MouseBrain_converted_R1.fastq.gz', but expected 979647 reads based on
PIPseeker output file
'/home/trekkerU_PIP_converter_example_input/metrics/barcode_stats.csv'.
Please delete all the intermediate files and retrigger
trekkerU_PIP_converter.
```

- **A step failed.**

The error message will direct you to the log of the failed step. If this happens, review the step-log for details or share it with technical_support@takarabio.com.

```
...
Starting the trekkerU_PIP_converter...
Start convertBarcode

Abort the TrekkerU_PIP_converter. Error: convertBarcode failed. Please
refer to
'/home/trekkerU_PIP_converter_example_input/log/TrekkerU_PIP_MouseBrain/c
onvertBarcode.log' for details.
```

D. Troubleshooting: Input Preparation for BD Rhapsody Single-Cell TCR/BCR Next + WTA Kit (TrekkerU_RVDJ)

- **A required input is missing.**

You will see an error message like the one below. If this happens, check your `samplesheet.csv` to ensure the path for the missing input is correct.

```
Checking samplesheet...
'/home/TrekkerU_C_ExampleInput_MouseBrain/samplesheet.csv' exists.
Checking Trekker FASTQ R1...

'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R1_001.fastq.gz' exists.
Checking Trekker FASTQ R2...

'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R2_001.fastq.gz' exists.
Checking Trekker tile spatial barcode whitelist...

'/home/TrekkerU_C_ExampleInput_MouseBrain/LTTag0053_003_BeadBarcodes.txt' exists.
Checking snRNAseq outputs...
Abort the Pipeline. Error: Unable to locate single-nuclei RNAseq pipeline output folder. Please ensure the path specified by 'sc_outdir' in your samplesheet exists.
```

- **A step failed.**

The error message will direct you to the log of the failed step. If this happens, review the step-log for details or share it with technical_support@takarabio.com.

```
...
Checking snRNAseq outputs...
snRNAseq outputs located.
Starting Trekker Analysis (TrekkerU_C) using singularity
Start fastq_parser
Abort the Pipeline. Error: fastq_parser failed, Please refer to
'/home/TrekkerU_C_ExampleInput_MouseBrain/log/TrekkerU_C_MouseBrain/fastq_parser.log' for details.
```

- **The required VDJ-associated input is missing.**

You will see error messages like below. If this happens, check your `samplesheet.csv` to ensure `sc_sample` and `scmulti_outdir` values are correct such that the path to the VDJ-associated file follows this pattern:

```
${scmulti_outdir}/${sc_sample}_Seurat.rds
```

- Due to missing `scmulti_outdir` column in the `samplesheet.csv`

```
...
Checking scmulti_outdir...

Abort the Pipeline. Error: The scmulti_outdir value in the
samplesheet is empty. Please check your samplesheet and provide a
valid value.
```

- Due to incorrect `scmulti_outdir` value in the `samplesheet.csv`

```
...
Checking scmulti_outdir...

Abort the Pipeline. Error: Unable to locate the multiomic output
folder
'/home/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_Hum
anBreastCancer_scRNAseqOut/' from the single-nuclei pipeline. Please
ensure the path specified by 'scmulti_outdir' in your samplesheet
exists.
```

- Due to incorrect `sc_sample` value in the `samplesheet.csv`

```
...
Checking scmulti_outdir...

'/home/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_Hum
anBreastCancer_scRNAseqOut/' exists.

Checking scmulti_outdir/vdj_seurat file...

Abort the Pipeline. Error: File

'/home/TrekkerU_RVDJ_ExampleInput_HumanBreastCancer/TrekkerU_RVDJ_Hum
anBreastCancer_scRNAseqOut/TrekkerU_RVDJ_HumanBreastCancer_VDJ_Seurat
.rds' does not exist.

Please check your samplesheet and provide a valid value to
'sc_sample' (prefix for vdj_seurat file) or ensure that file
'TrekkerU_RVDJ_HumanBreastCancer_VDJ_Seurat.rds' exists in the path
specified by 'scmulti_outdir' in your samplesheet.
```

E. Troubleshooting: Input Preparation for BD Rhapsody Single-Cell ATAC-Seq + mRNA Whole Transcriptome Analysis Kit (TrekkerU_RATAC)

- **A required input is missing.**

You will see an error message like the one below. If this happens, check your `samplesheet.csv` to ensure the path for the missing input is correct.

```
Checking samplesheet...
'/home/TrekkerU_C_ExampleInput_MouseBrain/samplesheet.csv' exists.
Checking Trekker FASTQ R1...

'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R1_001.
fastq.gz' exists.
Checking Trekker FASTQ R2...

'/home/TrekkerU_C_ExampleInput_MouseBrain/TrekkerU_C_Mouse_brain_R2_001.
fastq.gz' exists.
Checking Trekker tile spatial barcode whitelist...

'/home/TrekkerU_C_ExampleInput_MouseBrain/LTTag0053_003_BeadBarcodes.txt
' exists.
Checking snRNAseq outputs...
Abort the Pipeline. Error: Unable to locate single-nuclei RNAseq
pipeline output folder. Please ensure the path specified by 'sc_outdir'
in your samplesheet exists.
```

- **A step failed.**

The error message will direct you to the log of the failed step. If this happens, review the step-log for details or share it with technical_support@takarabio.com.

```
...
Checking snRNAseq outputs...
snRNAseq outputs located.
Starting Trekker Analysis (TrekkerU_C) using singularity
Start fastq_parser
Abort the Pipeline. Error: fastq_parser failed, Please refer to
'/home/TrekkerU_C_ExampleInput_MouseBrain/log/TrekkerU_C_MouseBrain/fast
q_parser.log' for details.
```


- **The required ATAC-associated input is missing.**

You will see error messages like below. If this happens, check your samplesheet.csv to ensure 'scmulti_outdir' value is correct and it contains these three ATAC-associated files.

```
${scmulti_outdir}/atac-barcodes.tsv.gz
```

```
${scmulti_outdir}/atac-features.tsv.gz
```

```
${scmulti_outdir}/atac-matrix.mtx.gz
```

- **Due to missing scmulti_outdir column in the samplesheet.csv**

```
...
```

```
Checking scmulti_outdir...
```

```
Abort the Pipeline. Error: The scmulti_outdir value in the samplesheet is empty. Please check your samplesheet and provide a valid value.
```

- **Due to incorrect scmulti_outdir value in the samplesheet.csv**

```
...
```

```
Checking scmulti_outdir...
```

```
Abort the Pipeline. Error: Unable to locate the multiomic output folder '/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/' from the single-nuclei pipeline. Please ensure the path specified by 'scmulti_outdir' in your samplesheet exists.
```

- **Due to missing files in folder defined by scmulti_outdir**

```
...
```

```
Checking scmulti_outdir...
```

```
'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/' exists.
```

```
Checking scmulti_outdir/atac files...
```

```
'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/atac-barcodes.tsv.gz' exists.
```

```
'/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/atac-features.tsv.gz' exists.
```

```
Abort the Pipeline. Error: File '/home/TrekkerU_RATAC_ExampleInput_MouseKidney/TrekkerU_RATAC_MouseKidney_scseqOut/TrekkerU_RATAC_MouseKidney_ATAC_Cell_by_Peak_MEX/atac-matrix.mtx.gz' does not exist. Please check your samplesheet and provide a valid path to the 'scmulti_outdir' value or ensure that the 'atac-matrix' can be located in the path specified by 'scmulti_outdir' of your samplesheet.
```

- **The numbers of nuclei in RNA-Seq and ATAC-Seq matrices do not match.**

You will see error messages like below. If this happens, check your `samplesheet.csv` to ensure `sc_outdir` and `scmulti_outdir` point to the RNA-Seq and ATAC-Seq matrices that came from the same run of the single-cell bioinformatics pipeline.

...

Checking the number of nuclei in snRNAseq and snATACseq count matrices from the single-nuclei pipeline...

Abort the Pipeline. Error: Different number of nuclei found. Please ensure snRNAseq and snATACseq count matrices have the same number of nuclei.

Contact Us	
Customer Service/Ordering	Technical Support
tel: 800.662.2566 (toll-free)	tel: 800.662.2566 (toll-free)
fax: 800.424.1350 (toll-free)	fax: 800.424.1350 (toll-free)
web: takarabio.com/service	web: takarabio.com/support
e-mail: ordersUS@takarabio.com	e-mail: technical_support@takarabio.com

Notice to Purchaser

Our products are to be used for **Research Use Only**. They may not be used for any other purpose, including, but not limited to, use in humans, therapeutic or diagnostic use, or commercial use of any kind. Our products may not be transferred to third parties, resold, modified for resale, or used to manufacture commercial products or to provide a service to third parties without our prior written approval.

Your use of this product is also subject to compliance with any applicable licensing requirements described on the product's web page at takarabio.com. It is your responsibility to review, understand and adhere to any restrictions imposed by such statements.

© 2025 Takara Bio Inc. All Rights Reserved.

All trademarks are the property of Takara Bio Inc. or its affiliate(s) in the U.S. and/or other countries or their respective owners. Certain trademarks may not be registered in all jurisdictions. Additional product, intellectual property, and restricted use information is available at takarabio.com.

This document has been reviewed and approved by the Quality Department.