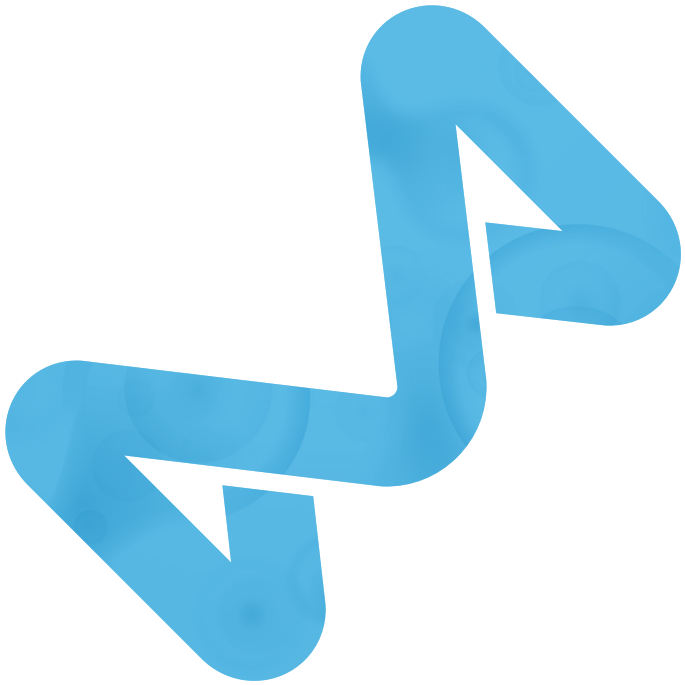
MobiDrop (Zhejiang) Co., Ltd.

MobiVision v3.0

Handbook



MOBIDROP

www.mobidrop.com

CONTENTS

MOBIVISION INTRODUCTION

MobiVision is a bioinformatic software that can be used for the analysis of single-cell transcriptome, full-length V(D)J, CITE and ChIP sequencing data from the MobiNova platform. Currently, MobiVision v3.0 is divided into four modules, with a total of 10 sub-commands:

Transcriptomic analysis, which includes the following 6 sub-commands:

- **quantify**

Process single cell transcriptomic data (.fastq format) into cell-gene expression matrix and generate a QC report.

- **fmt_gtf**

Format the annotation file (.gtf).

- **mkindex**

Generate transcriptomic reference used by "mobivision quantify".

- **rename**

Rename the sample ID of output files from "mobivision quantify".

- **mtx2csv**

Convert the cell-gene expression matrix into dataframe (.csv) file.

- **re_call_cell**

Redo cell-calling using the outputs from "mobivision quantify".

V(D)J analysis, which includes the following sub-command:

- **vdj**

Process single-cell V(D)J sequencing data (.fastq format), and generate the results such as V(D)J library quality control reports, annotation and related clonotype results.

CITE analysis, which includes the following sub-command:

- **cite**

Process CITE sequencing data (.fastq format), and generate the results such as quality control reports, gene and protein expression matrix.

ChIP analysis, which includes the following 2 sub-command:

- **chip**

Process single cell ChIP-seq data (.fastq format) into cells-peaks/fragments files and generate a QC report.

- **mk_chip_ref**

Generate index reference used by "mobivision chip".

SYSTEM REQUIREMENTS

- 8-core Intel or AMD processor, x86 architecture required (16 cores or more recommended)
- 64GB recommended
- Linux operation system, 64-bit CentOS, Ubuntu 20.04 or later version
- 1TB free disk space

INSTALLATION

First, decompress mobivision-v3.0.tar.gz. Then, execute the source command in the Shell environment and the MobiVision v3.0 is ready to run. Run the 'source' command each time when you open a new terminal or new shell window.

```
$ tar -zxvf mobivision-v3.0.tar.gz
$ source mobivision-v3.0/source.sh

#Check if MobiVision is successfully installed
$ mobivision --help

#The MobiVision software is installed successfully when the following information is shown:

mobivision-v3.0
Process MobiDrop Single Cell Sequencing data

USAGE:
mobivision <SUBCOMMAND>

FLAGS:
    -h, --help          Print help information
    -V, --version       Print version information
```

SUBCOMMANDS:

quantify	Count gene expression reads from a single sample
mkindex	Prepare a reference for mobivision software
fmt_gtf	Format and filter gene type features in annotation GTF file
rename	Rename the sample_id of output files
mtx2csv	Convert the matrix files to csv
re_call_cell	Re do cell-calling using the results from mobivision quantify analysis
vdj	Analysis of single-cell V(D)J sequencing data
mk_chip_ref	Prepare a reference for mobivision chip analysis
chip	Analyze ChIP fastq data from a single sample
cite	Analysis of data from CITE-seq

TUTOTIAL

1. Transcriptomics Analysis Module (3’RNA、5’RNA)

1.1 quantify

“mobivision quantify” is adapted to libraries and sequencing data prepared with the MobiCube® High-Throughput Single-Cell 3' Transcriptome Kit (3'RNA) and the MobiCube® High-Throughput Single-Cell 5' Transcriptome Kit (5'RNA). “mobivision quantify” can be used for the quantification of gene expression for each cell from single-cell transcriptomic sequencing data. Using paired-end fastq files as input, it generates quality control summaries as well as single cell gene expression matrices for further analysis.

1.1.1 Preparation of Fastq Data

The input fastq files should be named as follows:

[SampleName]_[ReadType].[Suffix]

For detail,

[SampleName] should be composed of Letters, Numbers or Underscore;

[ReadType] is recommended to be in one of the following formats: ‘R1’, ‘R2’, ‘R1_001’ or ‘R2_001’;

[Suffix] currently supports four formats: ‘fastq.gz’, ‘fq.gz’, ‘fastq’ or ‘fq’.

For example,

```
/Data/Sample_fastq/
├—— R22001221-20220105-Emix-3-20220105-S-F5-2-2_R1.fq.gz
└—— R22001221-20220105-Emix-3-20220105-S-F5-2-2_R2.fq.gz
```

1.1.2 Getting Start with MobiVision quantify

"mobiVision quantify" requires the path of corresponding index folder(-i), the threads used for parallel computing(-t), the path of input fastq files(-f) and the path for output files(-o). Replace the following code in red by actual parameters and run:

```
$ mobivision quantify -i /media/mz-3/db/gencode_hsa/GRCh38 \
-t 12 \
-f /media/mz-3/mzRaid10/Mobi_data/Sample_R22014039-C07 \
-o /media/mz-3/analysis/R22014039-C07
```

Argument description:

	Argument	Description
Required	-f, --fastqDir	The directory of fastq files(e.g., /media/mz-3/mzRaid10/Mobi_data/Sample_R22014039-C07).
	-i, --indexDir	The path of folder containing mobivision-compatible transcriptome reference. If the index file in tar.gz format is officially provided by MobiDrop, it is necessary to decompress the tar.gz file first, and then use the decompressed file path as the path of the -i parameter(e.g., /media/mz-3/db/gencode_hsa/GRCh38).
	-o, --outDir	The path of output files(e.g., /media/mz-3/analysis/R22014039-C07).
	-t, --threads	The number of threads used by MobiVision(e.g., 12).
Optional	--intron	<IntronInclude_Option> Include introns of genes or not for analysis. Options are "excluded" and "included". The default setting is "included".
	--cellnumber	<TOP_CELL> Force cell number for cell filtering.
	--cr2.2	Set CR2.2 algorithm for cell filtering. If not designated, the EmptyDrops algorithm will be used.
	-s, --sampleID	<SAMPLE_ID> The ID of the result file defined by the user.
	--kit	Set the version of RNA kit used for the library construction. The default setting is 'Unknown'.
Help	-h, --help	Show help messages.

The "mobivision quantify" command speicifies the path of the input fastq files by the argument “-f”. If there are only one pair of fastq files in the folder, the fastq data would be analysed by default. If there are multiple pairs of fastq files, the multiple pairs of fastq files would be merged, and the merged data would be stored in a folder named “merge/”. The subsequent analysis would be processed using the merged data. So it is recommended to put fastq files from the same sample instead of different samples in one folder.

There are three algorithms for cell calling:
1. The argument “--cellnumber int“ will allow the “mobivision quantify” to force the number of cells to be this designated number.
2. The “--cr2.2" argument will allow “mobivision quantify” to use CellRanger2.2 algorithm for cell calling.
3. EmptyDrops algorithm is used by default.

If not designated, “mobivision quantify” would quantify the genes including introns by default. Excluding introns option could be accomplished through setting “--intron excluded”.

The output path contains the following folders or files:

├── _flagdone	The flag for successful analysis
├── _log	The running log file
└── SAMPLEID_outs	The root of output results
├── filtered_cell_gene_matrix	The root of filtered matrix files
│ ├── barcodes.tsv.gz	Filtered barcodes file
│ ├── features.tsv.gz	Filtered gene list file
│ └── matrix.mtx.gz	Filtered count matrix file
├── SAMPLEID_Aligned.sort.bam	Mapping result file in bam format
├── SAMPLEID_Aligned.sort.bam.bai	Index file of bam file
├── SAMPLEID_filtered.h5ad	H5ad format of filtered matrix
├── SAMPLEID_Report.html	QC report in html format
├── SAMPLEID_Report.json	QC report in json format
├── SAMPLEID_summary.csv	Summary information in csv format
├── raw_cell_gene_matrix	The root of unfiltered matrix files
│ ├── barcodes.tsv.gz	Unfiltered barcodes file
│ ├── features.tsv.gz	Unfiltered gene list file
│ └── matrix.mtx.gz	Unfiltered count matrix file
├── result_mito_percentage.csv	Mitochondrial percentage information
└── run_analysis_cmds.txt	The command line records

1.2 fmt_gtf

"mobivision fmt_gtf" is used for editing the annotation file which will be used in “mobivision mkindex”. It requires the path of the input GTF file (-i), the path of output GTF file (-o) and the gene feature types (-t) which will be kept in output GTF, the "-t" option can be designated multiple times. Replace the following code in red by actual parameters and run:

```
$ mobivision fmt_gtf -i /media/db/gencode.v38.primary_assembly.annotation.gtf.gz \
-o /media/db_out/gencode.v38.primary_assembly.annotation.filtered.gtf \
-t protein_coding \
-t lncRNA \
-t antisense \
-t IG_LV_gene \
-t IG_V_gene \
-t IG_V_pseudogene \
-t IG_D_gene \
-t IG_J_gene \
-t IG_J_pseudogene \
-t IG_C_gene \
-t IG_C_pseudogene \
-t TR_V_gene \
-t TR_V_pseudogene \
-t TR_D_gene \
-t TR_J_gene \
-t TR_J_pseudogene \
-t TR_C_gene
```

Argument description:

	Argument	Description
Required	-i	Filename for input GTF, the format of ".gtf" file or ".gtf.gz" file is supported.
	-o	The name for output GTF filename, default value is "out_filtered.gtf".
	-t	The gene feature types which will be kept in output gtf, this can be designated multiple times.
Help	-h, --help	Show help messages.

1.3 mkindex

"mobivision mkindex" could generate the MobiVision-compatible transcriptome reference index using genome fasta and annotation file. It requires the name of genome (-n), genome fasta file path (-f), GTF file path for genome (-g), and the generated index will is stored in a folder named by the argument "-n". The folder will be created in the path where you run the command. Replace the following code in red by actual parameters and run:

```
$ mobivision mkindex -n GRCh38_gencode \
-f /media/db_out/GRCh38.primary_assembly.genome.fa.gz\
-g /media/db_out/gencode.v38.primary_assembly.annotation.filtered.gtf
```

If you want to generate a reference index using two or more genomes, please specify -n\-f\-g arguments two or more times. And the name of genome, the genome fasta file and the genome annotation file specified by -n, -f and -g must correspond one to one in order. For example,

```
$ mobivision mkindex -n GRCh38_gencode \
-f /media/db_out/GRCh38.primary_assembly.genome.fa.gz \
-g /media/db_out/gencode.v38.primary_assembly.annotation.filtered.gtf \
-n GRCm39_gencode \
-f /media/db_out/GRCm39.primary_assembly.genome.fa.gz \
-g /media/db_out/gencode.vM39.primary_assembly.annotation.filtered.gtf
```

Argument description:

	Argument	Description
Required	-n	Unique genome name containing [a-zA-Z0-9_]. If the index is created with multiple genomes, please specify the -n argument multiple times. For example, if one specifies -n <genome1> and another -n <genome2>, the output index folder will be named <genome1>_and_<genome2>.
	-f	Genome fasta file path. If the index is created with multiple FASTA files, please specify the -f argument multiple times. The supported formats of genome fasta files include '.fasta', '.fa.gz' and '.fna.gz'.
	-g	GTF file path for genome. If the index is created with multiple GTFs, please specify the -g argument multiple times. If -n and -f arguments were specified multiple times, -g argument needs to be specified an equal amount of times and the specified order must be same.
Optional	-r	The version of output genome index, e.g., "ref_v1.0", "ref-2022-06".
	-m	Maximum memory (GB) used when building index files with STAR.
Help	-h, --help	Show help messages.

In transgenic or viral infection researches, it may be necessary to add genes of interest to existing genomes for alignment. Here we show how to add a subtype HIV gene to the human reference genome:

First, use the "wget" command to download and decompress the fasta and gtf files needed to build the reference genome from ***Genecode*** official website. Replace the following code in red by actual parameters and run:

```
$wget
http://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_38/GRCh38.primary_assembly.genome.fa.gz
$wget
http://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_38/gencode.v38.primary_assembly.annotation.gtf.gz
$gunzip -c GRCh38.primary_assembly.genome.fa.gz >
GRCh38.primary_assembly.genome.fa
$gunzip -c gencode.v38.primary_assembly.annotation.gtf.gz >
gencode.v38.primary_assembly.annotation.gtf
```

Secondly, download the sequence of the gene of interest, use the "cat" command to create a new fa file, copy the gene sequence to the fa file, and enter Ctrl+D to save it. Replace the long name in the first line with a shorter name that is easy to identify, and count the number of bases in the sequence for the production of the gtf file. Replace the following code in red by actual parameters and run:

```
$cat > HIV.fa
$sed -i '1d' HIV.fa
$sed -i '1 i >HIV' HIV.fa
$cat HIV.fa | grep -v "^>" | tr -d "\n" | wc -c
```

Then, build the gtf file of the gene of interest according to the basic format of the gtf file on ***ensembl*** official website . Replace the following code in red by actual parameters and run:

```
$echo -e 'HIV\tunknown\texon\t1\t999\t.\t+\t.\tgene_id "HIV-1"; transcript_id "HIV-1"; gene_name "HIV-1"; gene_biotype "protein_coding";' > HIV.gtf
```

Then, merge the fa file and gtf file of the original reference genome and the gene of interest. Replace the following code in red by actual parameters and run:

```
$cp -rf GRCh38.primary_assembly.genome.fa GRCh38.hiv.genome.fa
$cp -rf gencode.v38.primary_assembly.annotation.gtf GRCh38.hiv.genome.gtf
$cat HIV.fa >> GRCh38.hiv.genome.fa
$cat HIV.gtf >> GRCh38.hiv.genome.gtf
```

Finally, use the “mobivision mkindex” command to build new reference genome. Replace the following code in red by actual parameters and run:

```
$ mobivision mkindex \
-n GRCh39-HIV \
-f GRCh38.hiv.genome.fa \
-g GRCh38.hiv.genome.gtf \
-r GRCh39-HIV0923
```

1.4 re_call_cell

The outputs from “mobivision quantify” could be the inputs of “mobivision re_call_cell”, which could redo cell-calling, generate a new cell-gene matrix (reanalysis_matrix), obtain a new QC report and other relevant result files. “mobivision re_call_cell” requires the outputs from “mobivision quantify” (-i) and the results directory of “mobivision re_call_cell” analysis (-o). Replace the following code in red by actual parameters and run:

```
$ mobivision re_call_cell -i 230106-WJ-E12-W04_combined_outs \
-o 230106-WJ-E12-W04_reCallCell_outs \
--cellnumber 1000
```

Argument description:

	Argument	Description
Required	-i	The result directory from mobivision quantify.
	-o	The output directory of mobivision re_call_cell .
Optional	--cellnumber	<TOP_CELL> Force cell number for cell filtering.
	--ed	Set EmptyDrops algorithm for cell filtering. If not designated, the CellRanger2.2 algorithm will be used.
Help	-h, --help	Show help messages.

There are three algorithms for cell calling:

1. The argument “--cellnumber int“ will allow the “mobivision re_call_cell” to force this number of cells.
2. The argument “--ed" option will allow “mobivision re_call_cell” to use EmptyDrops algorithm for cell calling;
3. CellRanger2.2 algorithm is used by default.

1.5 rename

"mobivision rename" could rename all output files in which sample ID is included and rename the “sampleID” shown in html report. It requires new sample_id of result files (-i) and the results directory of “mobivision quantify” (-d). Replace the following code in red by actual parameters and run:

```
$ mobivision rename -i human_PBMC_Control \
-d /media/mz/analysis/R1-S0-1_outs
```

Argument description:

	Argument	Description
Required	-i	New sample_id of result files.
	-d	The results directory of mobivision quantify.
Help	-h, --help	Show help messages.

1.6 mtx2csv

After completing “mobivision quantify”, filtered cell gene matrix and raw cell gene matrix would be generated. “mobivision mtx2csv” could convert the filtered cell gene matrix or raw cell gene matrix into dataframe (.csv) file. It requires the input directory of matrix, features and barcodes files (-i) and the filename of result csv(-o). Replace the code in red by actual parameters and run:

```
$ mobivision mtx2csv -i /media/mz/analysis/R1-S0-1_outs/filtered_cell_gene_matrix \
-o /output/R1-S0-1.csv
```

Argument description:

	Argument	Description
Required	-i	The input directory of mtx, features and barcodes files.
	-d	The filename of result csv.
Help	-h, --help	Show help messages.

2. Immunomics Analysis Module (V(D)J)

2.1 vdj

"mobivision vdj" adapted to libraries and sequencing data prepared with MobiCube® high-throughput single-cell V(D)J kits. "mobivision vdj" be used to analyze the single-cell V(D)J sequencing data from the MobiNova platform. Using paired-end sequencing fastq files as input, it can generate quality control reports, annotations and related clonotyp results.

2.1.1 Preparation of Fastq Data

The input fastq files should be named as follows:

[SampleName]_[ReadType].[Suffix]

where,

[SampleName] should be composed of letters, numbers or underscores, and it is not recommended to include 'R1' or 'R2';

[ReadType] is recommended to be in one of the following formats: R1, R2, R1_001, R2_001;

[Suffix] currently supports four formats: fastq.gz, fq.gz, fastq, fq.

For example:

```
/Data/Sample_fastq/
├── R22001221-20220105-Emix-3-20220105-S-F5-2-2_R1.fq.gz
└── R22001221-20220105-Emix-3-20220105-S-F5-2-2_R2.fq.gz
```

2.1.2 Getting Start with MobiVision vdj

The mobivision vdj command requires the path to the fastq files (-f), the folder path for the index (-i), the maximum number of threads (-t), and the output folder path (-o). After modifying the following code in red to actual parameters, run:

```
$ mobivision vdj -f /PATH_TO_FASTQ/230113-LYY-N7T-O04/convert/mobi \
-i /PATH_TO_REFERENCE/GRCh38_vdj/ \
-o /PATH_TO_OUTPUT \
```

Argument description:

	Argument	Description
Required	-f, --fastqDir	Path to the fastq files.
	-i, --referencePath	Path to the folder containing the MobiNova transcriptome reference files. For example: /home/refer/GRCh38_vdj/.
	-o, --outDir	Path to the output folder.
	-t, --threads	Number of threads to be used. Default value: 8.
Optional	-s, --sampleID	< specified sample name> User-specified sample ID for result files.
	-c, --chainType	Chain type to be designated: 'TR' for T cell receptors, 'IG' for B cell receptors, 'ALL' for both T cell receptors and B cell receptors, or 'auto' for auto-detection. Default: 'auto'.
	-p, --innerPrimerPath	Please specify the sequences of inner enrichment primers here as a text file with the sequence of one primer per line, if the primers are of a species other than human, mouse or rabbit provided in the MobiDrop kits.
Help	-h, --help	Show help information.

The analysis results of single TCR or single BCR data are as follows:

230113-LYY-N4T-O01	
├── _flagdone	Flag file indicating successful run
├── _log	Log file of the run
├── run_analysis_cmds.txt	Record of the command line
└── 230113-LYY-N4T-O01_outs	Root directory of the output result files
├── 230113-LYY-N4T-O01_airr_rearrangement.tsv	Annotation results in airr format
├── 230113-LYY-N4T-O01_all_contig_annotations.csv	Annotation results for all contigs in csv format
├── 230113-LYY-N4T-O01_all_contig.fasta	Sequence file for all contigs in fasta format
├── 230113-LYY-N4T-O01_clonotypes.csv	Clonotype result file
├── 230113-LYY-N4T-O01_filtered_contig_annotations.csv	Annotation results for filtered contig
├── 230113-LYY-N4T-O01_filtered_contig.fasta	Sequence file for filtered contigs in fasta format
├── 230113-LYY-N4T-O01_metrics_summary.csv	Summary file of the analysis in csv format
├── 230113-LYY-N4T-O01_Report.html	Quality Control report in html format
└── 230113-LYY-N4T-O01_Report.json	Quality Control report in json format

The analysis results of merged TCR/BCR data are as follows:

230113-LYY-N4T-O01	
├── _flagdone	Flag file indicating successful run
├── _log	Log file of the run
├── run_analysis_cmds.txt	Record of the command line
└── 230113-LYY-N4T-O01_outs	Root directory of the output result files
├── 230113-LYY-N4T-O01_metrics_summary.csv	Summary file of the analysis in csv format
├── 230113-LYY-N4T-O01_Report.html	Quality Control report in html format
├── 230113-LYY-N4T-O01_Report.json	Quality Control report in json format
└── IG_result	
│ ├── 230113-LYY-N4T-O01_airr_rearrangement.tsv	Annotation results in airr format
│ ├── 230113-LYY-N4T-O01_all_contig_annotations.csv	Annotation results for all contigs in csv format
│ ├── 230113-LYY-N4T-O01_all_contig.fasta	Sequence file for all contigs in fasta format
│ ├── 230113-LYY-N4T-O01_clonotypes.csv	Clonotype result file
│ ├── 230113-LYY-N4T-O01_filtered_contig_annotations.csv	Annotation results for filtered contig
│ ├── 230113-LYY-N4T-O01_filtered_contig.fasta	Sequence file for filtered contigs in fasta format
└── TR_result	
├── 230113-LYY-N4T-O01_airr_rearrangement.tsv	Annotation results in airr format
├── 230113-LYY-N4T-O01_all_contig_annotations.csv	Annotation results for all contigs in csv format
├── 230113-LYY-N4T-O01_all_contig.fasta	Sequence file for all contigs in fasta format
├── 230113-LYY-N4T-O01_clonotypes.csv	Clonotype result file
├── 230113-LYY-N4T-O01_filtered_contig_annotations.csv	Annotation results for filtered contig
└── 230113-LYY-N4T-O01_filtered_contig.fasta	Sequence file for all contigs in fasta format

3. Proteomics Analysis Module (CITE)

3.1 cite

"mobivision cite" is adapted to libraries and sequencing data prepared with the MobiCube® High-Throughput Single-Cell CITE-seq (A-Human) Kit. “mobivision cite” can be used for the quantification of gene expression and cell surface protein for each cell from data generated by Cellular Indexing of Transcriptomes and Epitopes by sequencing. Using the fastq file of the transcriptome and the fastq file of the protein tag as input, the final output can be html QC report, gene expression and protein quantification matrix, which can be used for downstream in-depth analysis.

3.1.1 Preparation of Fastq Data

The input fastq contains two sets of library data, transcriptome and proteome, the two types of data need to be placed in different folders and need to be named as follows.

[SampleName]_[ReadType].[Suffix]

For details,

[SampleName] should be composed of letters, numbers or underscore;

[ReadType] is recommended to be in one of the following formats: ‘R1’, ‘R2’, ‘R1_001’ or ‘R2_001’;

[Suffix] currently supports four formats: ‘fastq.gz’, ‘fq.gz’, ‘fastq’ or ‘fq’.

For example,

The transcriptomic data are as follows:

```
/data/Mobi/test_data/rna_data
├── R22001221-20220105-Emix-3-20220105-S-F5-2-2_R1.fq.gz
└── R22001221-20220105-Emix-3-20220105-S-F5-2-2_R2.fq.gz
```

The proteomic data are as follows:

```
/data/Mobi/test_data/adt
├── R22001221-20220105-ADT-3-20220105-S01_R1.fq.gz
└── R22001221-20220105-ADT-3-20220105-S01_R2.fq.gz
```

3.1.2 Getting Start with MobiVision cite

"mobivision cite" requires the path of input transcriptome fastq files(-f) , the path of input proteome fastq files(-fb) , the path of corresponding index folder(-i), the threads used for parallel computing(-t), the path of taglist file(-b) and the path for output files(-o). Replace the following code in red by actual parameters and run:

```
$ mobivision cite -f /data/Mobi/test_data/rna_data  \
-fb /data/Mobi/test_data/adt  \
-i /data/References/GRCh38-2023  \
-t 8 \
-b tag_ref50.csv \
-o output06
```

Argument description:

	Argument	Description
Required	-f, --fastqDir	The directory of transcriptome fastq files (e.g.,/media/mz-3/mzRaid10/Mobi_-data/Sample_R22014039-C07 .
	-fb,	The directory of proteome fastq files (e.g.,/media/mz-3/mzRaid10/Mobi_-data/Sample_T22014039-T01.
	-i, --indexDir	The path of folder containing mobivision-compatible transcriptome reference, requires index built with v2.2 and later. If the index file in tar.gz format is officially provided by MobiDrop, it is necessary to decompress the tar.gz file first, and then use the decompressed file path as the path of the -i parameter(e.g., /media/mz-3/db/gencode_hsa/GRCh38).
	-b, --tagBar-codeInfoFile	The path of barcode file, with header such as "id,sequence", tag sequence info as " CD56,TTTTAAATCGAT " and one line for each antibody barcode.
	-o, --outDir	The path of output files(e.g., /media/mz-3/analysis/R22014039-C07).
Optional	--intron	<IntronInclude_Option> Include introns of genes or not for analysis. Options are "excluded" and "includ-ed". The default setting is "included".
	-m, --mis-matchNum	<MISMATCH_N> The mismatch number for feature tag sequence align , the default value is 1.
	-p, --capture-Type	<CAPTURE_TYPE> The capture type of tag sequence, default: "Antibody". Options are "Antibody" and "Multiplexing".
	-t, --threads	The number of threads used by MobiVision(e.g., 12).
Help	-h, --help	Show help messages.

“mobivision cite” would quantify the genes including introns by default. Excluding introns option could be accomplished through setting “--intron excluded”. The default setting is “--intron included”.

The output path contains the following folders or files:

├── _flagdone	The flag for successful analysis
├── _log	The running log file
└── SAMPLEID_outs	The root of output results
├── filtered_cell_gene_matrix	The root of filtered matrix files
│ ├── barcodes.tsv.gz	Filtered barcodes file
│ ├── features.tsv.gz	Filtered gene list file
│ └── matrix.mtx.gz	Filtered count matrix file
├── SAMPLEID_Aligned.sort.bam	Mapping result file in bam format
├── SAMPLEID_Aligned.sort.bam.bai	Index file of bam file
├── SAMPLEID_filtered.h5ad	H5ad format of filtered matrix
├── SAMPLEID_Report.html	QC report in html format
├── SAMPLEID_Report.json	QC report in json format
├── SAMPLEID_summary.csv	Summary information in csv format
├── raw_cell_gene_matrix	The root of unfiltered matrix files
│ ├── barcodes.tsv.gz	Unfiltered barcodes file
│ ├── features.tsv.gz	Unfiltered gene list file
│ └── matrix.mtx.gz	Unfiltered count matrix file
├── result_mito_percentage.csv	Mitochondrial percentage information
└── run_analysis_cmds.txt	The command line records

4. Epigenomics Analysis Module (ChIP)

4.1 chip

"mobivision chip" is adapted to sequencing data from library cassettes prepared with MobiCube® high-throughput single-cell ChIP-seq kits. “mobivision chip” can be used for the analysis of gene regulatory loci for each cell from single-cell ChIP sequencing data. Using paired-end fastq files as input, it generates quality control summaries, peaks as well as fragments files for further analysis.

4.1.1 Preparation of Fastq Data

The input fastq files should be named as follows:

[SampleName]_[ReadType].[Suffix]

For detail,

[SampleName] should be composed of Letters, Numbers or Underscore;

[ReadType] is recommended to be in one of the following format: ‘R1’, ‘R2’, ‘R1_001’ or ‘R2_001’;

[Suffix] currently supports four formats: ‘fastq.gz’, ‘fq.gz’, ‘fastq’ or ‘fq’.

For example,

```
/Data/Sample_fastq/
├── R22001221-20220105-Emix-3-20220105-S-F5-2-2_R1.fq.gz
└── R22001221-20220105-Emix-3-20220105-S-F5-2-2_R2.fq.gz
```

4.1.2 Getting Start with MobiVision chip

“mobivision chip” requires the path of corresponding index folder(-i), the threads used for parallel computing(-t), the path of input fastq files(-f) and the path for output files(-o). Replace the following code in red by actual parameters and run:

```
$ mobivision chip -i /share/Reference/mobi_chip_ref/GRCh38 \
-t 12 \
-f /share/Data/Sample_SQ23009375-230428C-S-YXH-L06 \
-o /share/Outs/230428C-S-YXH-L06
```

Argument description:

	Argument	Description
Required	-f, --fastqDir	The directory of fastq files (e.g., /share/Data/Sample_SQ23009375-230428C-S-YXH-L06).
	-i, -- referenCePath	The path of folder containing mobivision-compatible ChIP reference. If the index file in tar.gz format is officially provided by MobiDrop, it is necessary to decompress the tar.gz file first, and then use the decompressed file path as the path of the -i parameter (e.g., /share/Reference/mobi_chip_ref/GRCh38).
	-o, --outDir	The path of output files (e.g., /share/Outs/230428C-S-YXH-L06).
	-t, --threads	The number of threads used by MobiVision (e.g., 12).
Optional	--peaktype	The type of peaks to be called. Options are "narrow" and "broad". The default setting is "narrow".
	--cellnumber	<TOP_CELL> Force cell number for cell filtering. If not designated, the cell number would be calculted by CR2.2 algorithm.
	--control	The directory of IgG sample fastq files.
Help	-h, --help	Show help messages.

The output path contains the following folders or files:

├── _flagdone	The flag for successful analysis
├── _log	The running log file
├── run_analysis_cmds.txt	The command line records
└── SAMPLEID_outs	The root of output results
├── filtered_cell_fragments_matrix	The root of filtered cell-fragment matrix files
│ ├── barcodes.tsv.gz	Filtered barcode file
│ ├── fragments.tsv.gz	Filtered fragment file
│ └── matrix.mtx.gz	Filtered cell-fragment matrix file
├── filtered_cell_peaks_matrix	The root of filtered cell-peak matrix files
│ ├── barcodes.tsv.gz	Filtered barcode file
│ ├── peaks.tsv.gz	Filtered peak file
│ ├── filtered.h5ad	H5ad format of filtered cell-peak matrix
│ └── matrix.mtx.gz	Filtered cell-peak matrix file

├── SAMPLEID.bam	Mapping result file in bam format
├── SAMPLEID.bw	Mapping result file in bw format
├── SAMPLEID.filtered.bed.gz	Filtered non-duplicated fragment file
├── SAMPLEID_Report.html	QC report in html format
├── SAMPLEID_Report.json	QC report in json format
├── summary.csv	Summary information in csv format
├── raw_cell_peaks_matrix	The root of unfiltered cell-peak matrix files
│ ├── barcodes.tsv.gz	Unfiltered barcode file
│ ├── peaks.tsv.gz	Unfiltered peak file
│ └── matrix.mtx.gz	Unfiltered cell-peak matrix file
├── result_mito_percentage.csv	Mitochondrial percentage information
├── fragmentsInCells.tsv.gz	Fragments belonging to filtered barcodes
├── fragmentsInPeaks.tsv.gz	Fragments overlapping with peaks
└── SAMPLEID.narrowPeak/broadPeak	Total library peak file

4.2 mk_chip_ref

"mobivision mk_chip_ref" could generate the MobiVision-compatible ChIP reference index using genome fasta. It requires the name of genome (-n), genome fasta file path (-f) and the generated index will is stored in a folder named by the argument "-n". The folder will be created in the path where you designated by the argument "-r". Replace the following code in red by actual parameters and run:

```
$ mobivision mk_chip_ref -n GRCh38 \
-f /media/db_out/GRCh38.primary_assembly.genome.fa.gz \
-r /share/Reference/chip
```

If you want to generate a reference index using two or more genomes, please specify -n\-f arguments two or more times. The name of genome and the genome fasta file specified by -n and -f must correspond one to one in order. For example,

```
$ mobivision mk_chip_ref -n GRCh38 \
-n GRCm39 \
-f /media/db_out/GRCh38.primary_assembly.genome.fa.gz \
-f /media/db_out/GRCm39.primary_assembly.genome.fa.gz \
-r /share/Reference/chip
```

Argument description:

	Argument	Description
Required	-n	Unique species name [a-zA-Z0-9_]+. If the index is created with multiple genomes argument multiple times. For example, if one spedifies -n <species1> and another -n <species2>, the output index folder will be named <species1>_and_<species2>.
	-f	Genome fasta file path. If the index is created with multiple FASTA files, please specify the -f argument multiple times.The supported formats of genome fasta files include '.fasta', '.fa.gz' and '.fna.gz'.
	-r	The output reference path.
Help	-h, --help	Show help messages.

The output path contains the following folders or files:

├── GENOME.1.bt2 (l)	The 1st level BWT of reference genome
├── GENOME.2.bt2 (l)	The 2nd level BWT of reference genome
├── GENOME.3.bt2 (l)	The 3rd level BWT of reference genome
├── GENOME.4.bt2 (l)	The 4th level BWT of reference genome
├── GENOME.chrom.sizes	The chromosome sizes of reference genome
├── GENOME.genome.fa	The reference genome fasta file
├── GENOME.genome.fa.fai	The index file of reference genome fasta
├── GENOME.reference.json	The json file with reference information
├── GENOME.rev.1.bt2(l)	The reverse complement of the 1st BWT
└── GENOME.rev.2.bt2(l)	The reverse complement of the 2nd BWT

REFERENCE INFORMATION

First, decompress mobivision-v3.0.tar.gz. Then, execute the source command in the Shell environment and the MobiVision v3.0 is ready to run. Run the 'source' command each time when you open a new terminal or new shell window.

Reference Name	Species	ref_version	Download URL
GRCh38	Human	Gencode_v38*	https://www.gencodegenes.org/human/release_38.html
		Ensembl_v104	https://ftp.ensembl.org/pub/release-104/fasta/homo_sapiens/dna/
GRCm39	Mouse	Gencode_vM27*	https://www.gencodegenes.org/mouse/release_M27.html
		Ensembl_v104	https://ftp.ensembl.org/pub/release-104/fasta/mus_musculus/dna/
GRCh38_and_GRCm39	Human and Mouse	Gencode_v38*	https://www.gencodegenes.org/human/release_38.html
		Gencode_vM27*	https://www.gencodegenes.org/mouse/release_M27.html
	Human and Mouse	Ensembl_v104	https://ftp.ensembl.org/pub/release-104/fasta/homo_sapiens/dna/
			https://ftp.ensembl.org/pub/release-104/fasta/mus_musculus/dna/
OryCun2	Rabbit	Gencode	-
		Ensembl_v109*	https://ftp.ensembl.org/pub/release-109/fasta/oryctolagus_cuniculus/dna/

Notes:
Reference versions ends with * are those provided by the MobiNova platform.

For more product information and operating instructions,

please follow Mobidrop Biosciences WeChat Official Account.

MOBIDROP

CONTACT US

Manufacturer: MobiDrop (Zhejiang) Co., Ltd.

Production Address: No.1888 Longxiang Road, Tongxiang, Zhejiang Province, China, 314500

Phone: 400-860-7763

Email: support@mobidrop.com

Website: www.mobidrop.com