BioIT guidelines

Bioinformatics guide for using DeMuxnanoT to demultiplex nano-tRNAseq sequencing datasets. (NTPRO-12; RTPRO-12, NTRSQ-12, RLNTR-12)

FOR RESEARCH USE ONLY

Supplemental product information and tips for success

Required hardware:

- Linux operating system
- CPU with 4 cores
- 16 GB RAM memory
- Nvidia GPU with at least 8 GB vRAM
- 500 GB SSD storage

Required software:

- cuda (version 11.3 or newer)
- docker (version 23 or newer)
- nvidia-container-toolkit
- dorado (v0.7.2 or newer)
- minimap2 (version 2.24 or newer)
- samtools (version 1.15 or newer)
- bwa

Steps to perform before running the DeMux nanoT application:

Step 1. Basecalling:

- LC_ALL=en_US.UTF-8 dorado basecaller -r -x cuda:all rna004_130bps_hac@v5.0.0 --emit-moves pod5_direcotory > bam_file.bam

Parameter	Definition
pod5_direcotory	Pod5 output directory from MinKNOW
bam_file.bam	Output bam file

Step 2. Mapping:

- samtools fastq -T ts,mv bam_file.bam | minimap2 -y -t8 -ax splice -k7 -w3 -n1 -m13 -s30 -A2 -B1 -O1,32 -E1,0 fasta_tRNA_database.fa - | samtools sort -@8 --write-index -o mapped.bam

Parameter	Definition
bam_file.bam	Output bam file from dorado
fasta_tRNA_database.fa	FASTA database file containing tRNA sequences. For downloading mature tRNA sequences to use as a reference database, you can refer to the following website: https://gtrnadb.org/ . Please ensure to include the following adapter sequences in the mature tRNA FASTA file: 5' Splint: CCTAAGAGCAAGAAGAAGCCTGGN 3' Splint: GGCTTCTTCTTGCTCTTAGGAAAAAAAAAA It is recommended to retain only tRNA sequences with sequence homology below 87-90% to reduce multimapping reads.
mapped.bam	Output BAM file containing aligned reads.

Step 3. Demultiplexing bam file with DeMuxnanoT app:

To activate the DeMuxnanoT application, please send an email to bioinformatics@immaginabiotech.com .

After receiving a confirmation email from the bioinformatics group, the application will be activated and *will remain active for 45 days*.

Before launching DeMuxnanoT, create a input_directory folder containing the following:

- the pod5 folder generated by MinKNOW. Ensure that the output folder from MinKNOW is named "pod5." If it is not, please rename the folder to "pod5."
- the BAM file generated in step 2.

To launch the application, follow the command provided below:

- ./DeMuxnanoT-[number] --inputDir input_directory --bamName bam_name --outputDir output_directory --outputName exp_name

Parameter	Definition
DeMuxnanoT-[number]	DeMuxnanoT is the binary file located on the USB drive sent by Immagina Biotechnology.
input_directory	Path to input folder containing pod5 folder and bam file obtained in Step 2

bam_name	Name of the bam file present in input_directory (Example: test.bam)
output_directory	Path to output folder
exp_name	Name of the experiment (Example: test)

During the demultiplexing step, reads are filtered using a **baseQ > 50** threshold.

If you need to modify this filtering parameter, you can manually adjust the filtering using the file located in the demux output folder:

pod5.demux.tsv.gz

This file contains all reads and their associated metadata, including read ID, barcode assignment, base quality, and mapping quality.

Column description

Column	Description
read_id	Unique identifier for each nanopore read (UUID format).
adapter_end	Position (in bases) where the adapter sequence ends on the read. This indicates where the biological sequence begins.
barcode	Barcode assigned to the read.
mapQ	Mapping quality of the barcode assignment. Higher values indicate greater confidence in the assignment.
baseQ	Mean base quality of the read in Phred scale. Higher values indicate fewer sequencing errors.

You can filter this table according to your specific requirements (for example using different baseQ or mapQ thresholds).

After filtering the table, you can use the resulting read IDs to filter and split the BAM file generated by minimap2, retaining only the reads that pass your chosen criteria.

DeMuxnanoT Troubleshooting Guide

1. Handling "Install failed" Errors

If you encounter an **"Install failed"** error, verify that Docker is functioning correctly by running the following commands:

- docker pull ubuntu:noble-20250127
- docker save ubuntu:noble-20250127 > repotag.tar
- docker load -i repotag.tar

2. Handling "An error occurred while processing" Errors

If you receive an **"An error occurred while processing"** message, check the following:

- Ensure that Docker is correctly utilizing the GPU by running:

    - docker run --rm --runtime=nvidia --gpus all ubuntu nvidia-smi

For more details, refer to the NVIDIA documentation.

- Verify that the **input directory** contains the required files and ensure you provide the **full path**. The directory should include:
    - **Folder:** pod5
    - **BAM file:** test.mapped.bam

3. Cleanup After Errors

After encountering an error, always check and remove any folders that were created in the **output directory** before retrying.

Step 4. Counts tRNA:

For the analysis of tRNA abundances, basecalling errors, and the generation of plots for visualizing these results, you can use the scripts available on GitHub: https://github.com/novoalab/Nano-tRNAseq/tree/main/src

4.1 The expression of tRNAs can be estimated as follows:

```
get_counts.py -o counts.tsv -i /path/*bam_file.bam > stats.tsv
```

Parameter	Definition
/path/*bam_file.bam	Path to bam files obtained in Step 3
counts.tsv	output file with counts for each sample and each tRNA
stats.tsv	output file with statistics on bam files

4.2 Scatterplot of tRNA count for all samples using:

```
plot_scatter.py -i counts.tsv -o scatter.pdf
```

Parameter	Definition
counts.tsv	Input file with counts for each sample and each tRNA
scatter.pdf	Output file name

4.3 Generate differences in sum of basecalling errors between WT and some other sample(s)/condition(s) as follows:

```
get_sum_err.py -o heatmap/sample1 -f fasta_tRNA_database.fa -i /path/*bam_file.bam
```

Parameter	Definition
fasta_tRNA_database.fa	FASTA database file containing tRNA sequences.

/path/*bam_file.bam	Path to bam files obtained in Step 3. The first BAM file will be used as WT.
heatmap/sample1	Output name

4.4 Plots heatmaps of basecalling errors:

plot_heatmap.py -i heatmap/sample1.err_diff.tsv.gz -f fasta_tRNA_database.fa -a fasta_tRNA_database.aln.fa

Parameter	Definition
heatmap/sample1.err_diff.tsv.gz	File gz obtained in Step 4.3
fasta_tRNA_database.fa	FASTA database file containing tRNA sequences.
fasta_tRNA_database.aln.fa	FASTA file containing alignments of tRNA sequences.

Contacts

Info

info@immaginabiotech.com

Sale support (quoting, ordering, and order status update)

orders@immaginabiotech.com

Technical service (technical inquiries and quality complaints)

techsupport@immaginabiotech.com

Viale Dell'industria, 47, 38057, Pergine Valsugana (TN), ITALY

https://www.immaginabiotech.com/

+39 04611787270

Notes: