Installation of TARDIS

1. **Install via pip**
   Open a terminal and run: `pip install tardis-em`. This command installs the TARDIS-em package, including the microtubule segmentation tool `tardis_mt`.

2. **Create a Conda Environment**
   Generate a conda environment to avoid dependency conflicts.

   ```
   conda create -n tardis python=3.10
   conda activate tardis
   pip install tardis-em
   ```

Using TARDIS to Segment

1. **Activate the TARDIS Environment**
   Before running any TARDIS commands, ensure your environment is active:

   ```
   conda activate tardis
   ```

2. **Organize your Files**
   Place your aligned, binned (and denoised) *.mrc tomograms from etomo in a directory named `tardis_inputs/`.

   Important: *In TARDIS-em v0.3.12 and earlier, the program expects a sub-folder called Predictions in the working directory. Create it once to avoid the missing prediction_log.txt error:*

   ```
   mkdir -p ./tardis_inputs/Predictions
   ```

3. **Running TARDIS on a Tomogram**
   Navigate to your project folder: cd /path/to/your/project. Run the segmentation by using the following template:

   ```
   tardis_mt \
     -dir ./tardis_inputs \
     -ct 0.10 \
     -cnn fnet_attn \
     -out mrc_None
   ```

   Ensure that you replace the placeholders. After completion, each tomogram directory inside Predictions/ will contain a file `semantic.mrc`.

4. **Converting Output to Amira-compatible File**
   The default TARDIS output file format is float-32, which creates problems when importing into Amira and other visualization tools for segmentation. We can solve this by converting the output semantic.mrc into a 16-bit integer volume file format.

   We do this using the IMOD newstack command.

   ```
   newstack -format 2 semantic.mrc semantic_int16.mrc
   ```

Example of a Full Run

Assume I `have Position5B_0602_deconv.mrc` `inside ~/relan/Cryo-EMData/tardis_inputs/` and I want the outputs `in ~/relan/Cryo-EMData/tardis_outputs/Position5B_0602/`:

The full commands would be:

```
cd ~/relan/Cryo-EMData
conda activate tardis

mkdir -p tardis_outputs/position5b_0602/Predictions

# Run segmentation

tardis_mt \
  -dir ./tardis_inputs \
  -ct 0.10 \
  -cnn fnet_attn \
  -out mrc_None \

newstack -format 2
./tardis_outputs/Position5B_0602/Predictions/semantic.mrc \
./tardis_outputs/Position5B_0602/Predictions/semantic_int16.mrc
```

You can also write the command in one line without the backslashes.

TARDIS takes roughly 3-5 minutes to segment a tomogram depending on CPU power.

Running a Batch Script on Multiple Tomograms

1. **Organize your Files with a Consistent Naming Pattern**
   We can call TARDIS once per file from inside a shell script instead of typing the command manually each time.

   Create a directory for your input files (e.g. tardis_inputs). Add all tomograms you want to segment in this folder. Ensure that they all follow some consistent naming pattern.

2. **Modify and Copy Script**
   Ensure that you modify your script to read the specific naming pattern you decide on. You can also edit the three variables (input directory, output base, and ckpt path) if necessary.

   The script should be copied into your input directory (the same directory as your tomograms you want to segment). Make sure to make the script executable by running:

   ```
   chmod +x run_tardis_batch.sh
   ```

3. **Navigate to the Directory and Run the Script**
   The script will first activate the tardis conda environment. Then, it will define *one input directory* and *one output base directory* where the results will be written. The shell script will loop over every file that matches the naming pattern you specify. For this example, we will follow the naming pattern:

   Position*_0602_deconv.mrc.

   The script will then create a matching sub-folder inside the output base folder and call tardis segment on the tomograms.

   Run the script with the command:

   ```
   ./run_tardis_batch.sh
   ```

Attributions:

TARDIS
Zhang, Y., et al. (2023). "TARDIS: Efficient Microtubule Segmentation in Cryo-ET." Nature Biotechnology.
GitHub: https://github.com/SMLC-NYSBC/TARDIS