Using TOPAZ particle picking through Relion 3.1 – a very basic instruction.

Goal: to get TOPAZ to run within Relion 3.1 and extract its picked particles.

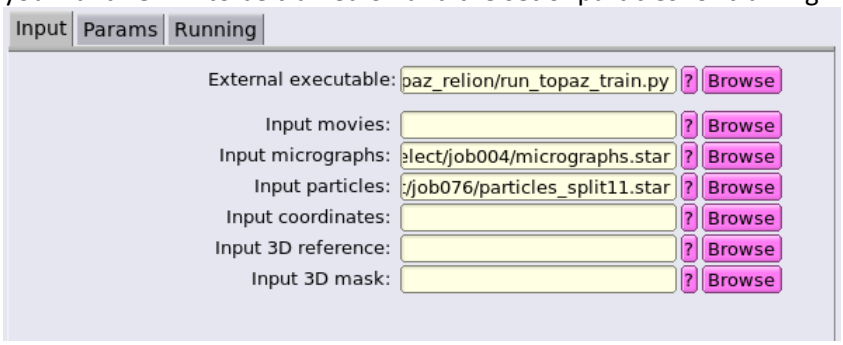
1. Install TOPAZ on your computer: https://github.com/tbepler/topaz. Depending on your CUDA version, you may need to replace PyTorch with a proper verion (https://pytorch.org/get-started/previous-versions/). Don't worry about this at the beginning.

2. Download the scripts for Relion integration from here: https://github.com/tbepler/topaz/tree/master/relion_run_topaz.

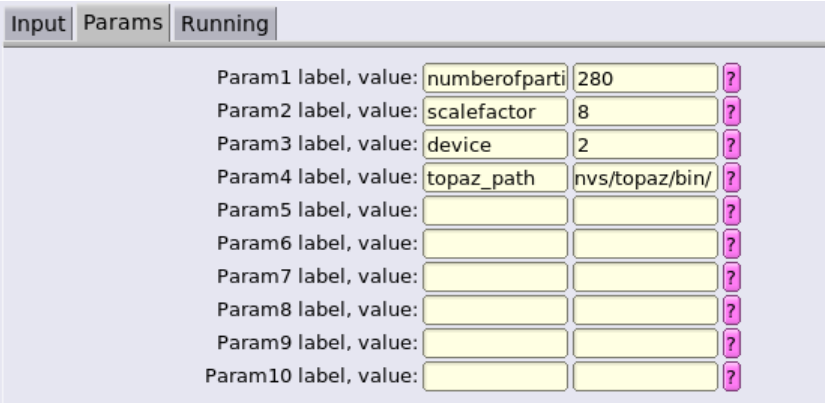
3. Assuming TOPAZ was installed using Anaconda (the recommended method), activate TOPAZ's environment by typing "*conda activate topaz*" on the command line.

4. Training TOPAZ in Relion 3.1.
    a. In your Relion project directory, launch Relion 3.1.
    b. Select "External" node on the left.

    c. In the "Input" tab, give Relion the script for training TOPAZ, as well as the micrographs you want TOPAZ to be trained on and the set of particles for training.



    d. In the "Params" tab, give Relion the parameters to run TOPAZ training, with flags in the left box and values in the right box.



Required parameters are numberofparticles and scalefactor. "device" is to choose which GPU card to run on. Seems like you can only specify one card at a time. And if your TOPAZ install path is not /usr/local/bin/topaz, you also need to supply with topaz_path. In the above example, the topaz installation path is /usr/local/anaconda3/envs/topaz/bin/. Refer to https://github.com/tbepler/topaz/tree/master/relion_run_topaz for more details.

e. You can also set the number of threads to run in "Running" tab. I have tested with 4 threads on a 2080Ti card, and it was OK.
f. Give a meaningful alias such as topaz_training and hit Run. TOPAZ will first downscale the micrographs and coordinates. Initially no log info will show up in Relion, but some information will show up in the error window. Don't worry. There are not errors.
g. To look at the training results, simple go to the folder: External/jobxxx/. .sav files are the trained models. The tutorial uses the 10th model (the last iteration) for picking. To check the quality of training, run "*cat model_plot.star | awk '{if ($3 == "test") {print}}'* ". Refer to https://github.com/tbepler/topaz/blob/master/tutorial/02_walkthrough.ipynb for how to interpret the results.

5. Picking with TOPAZ in Relion 3.1. This will be similar to the training process, simply replacing the training script with the picking script. Input micrographs in the "Input" tab, but no need to include the particles. Required parameters are: "scalefactor" should be the same as the trained model; "topaz_path" if your TOPAZ installation path is not default. Refer to https://github.com/tbepler/topaz/tree/master/relion_run_topaz for other parameters and if you want to use the pre-trained models. Again, 4 threads worked OK on a 2080Ti card.

   Note: Choosing thresholds for picking in TOPAZ is somewhat complicated. I would recommend skipping setting any threshold values for the first run. After inspecting the picks (which will be covered next), one may then select various threshold values to see the effect on the picking results (this will also be covered later).

6. To inspect particle picks, click cords_suffix_topazpicks.star from the Display dropdown menu of the External job, view the results in a default Relion fashion.

7. After inspecting the picks, go back to the TOPAZ picking External job and change "selectthreshold" value and repick. Be sure to also include "skip_pick" in the "Params" tab and give it the value "True". This will not re-run the picking, but only select coordinates according to the new select threshold. It will likely give you an error "mkdir: cannot create directory

'External/jobxxx/RawMovies/': File exists". Just ignore this as TOPAZ still runs and will update you in the output window once it finishes.

Refer to here https://github.com/tbepler/topaz/blob/master/tutorial/03_cross_validation.ipynb for more advanced approach to determine the parameters for training and picking through cross validation.

8. For particle extraction, use the cords_suffix_topazpicks.star file and follow normal particle extraction procedure in Relion.