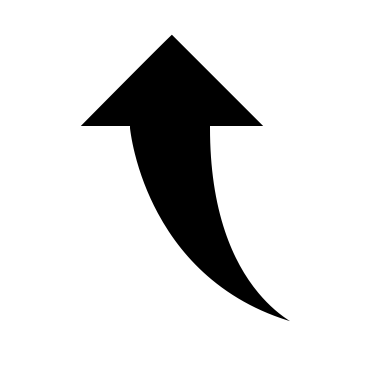
**Full Pipeline How To:**

**\*Pipeline is designed to run on MacOS and is optimized for a M1 Chip\***

**Step 1. Add all relevant files in a single path in MATLAB**

* This means downloading the zip file from GitHub and putting all of the relevant data and MATLAB files into their own folder to be called into the MATLAB workspace. This folder should include:

1. The live script “Full\_Pipeline\_Final.mlx”
2. The trained neural network “netUpdated.mat”,
3. The preprocessing function for experimental images “preprocessingTEM.m”
4. The image cropping function for implementation of experimental images into the neural network “boundingBoxTEM.m”
5. The folder containing all of your TEM images
6. The directory as an excel file “Directory.xls”

**Graphical user interface, application, Teams

Description automatically generated**

* Double clicking on the live script should open MATLAB with the script in Live Editor with the correct path open, but you can double check the current folder on the left of the MATLAB screen and manually add anything that is missing. Everything listed above should be open in the current folder and added to the path if it is not already (left click > Add to path > Selected folders and subfolders).

**Graphical user interface, text, application

Description automatically generated**

* The current path is indicated in the white bar next to the folder icon.

**Step 2. Add your TEM images to the directory**

**Calendar

Description automatically generated with medium confidence**

* Column A should have the name of your data set – something that will enable you to easily differentiate between what is what.
* Column B should be the relative path of the folder that holds the images within the folder that you made in step 1, apart from the specific image name.
* Column C should hold the common filename of all of the images of a specific material that you are analyzing. If you want to analyze images with different filenames all at once, change the image names to contain the same filename.
* Column D should contain all of the different image file types present in the specific folder of images being analyzed, separated by commas.
* Columns E and F together define the range of numbers for the images with the common filename indicated.
* Column G is the numbers of any missing images in the above-defined range, or images you want to leave out.
* Column H is the format of the numbers, which is usually as indicated in the example.

**Diagram

Description automatically generated**

**Logo

Description automatically generatedStep 3. Run the script**

* Press the green “Run” arrow

**Step 4. Answer the questions in the command window**

Graphical user interface, text, application

Description automatically generated

* Running the script will prompt you to input answers to a series of questions in the command window one by one, as seen above.
* The reference number is the row number of the images you want to run from the directory excel sheet minus 2, because the headings take up the first two rows (this becomes useful when your directory is very full). For example, to analyze CsPbBr3 Hot Injection batch from the example directory in step 2, it’s the first entry and it’s in row 3, so you would type “-rn 1” (3-2 = 1) as the answer to the prompt (typed exactly as shown above, with a space between “-rn” and the reference number).
* Unless doing an alternative study, the images need to be processed separately for the full pipeline to work, so the answer to that prompt should be “0”.

**Step 5. Wait for the output.**

* The pipeline should completely run in under 5 minutes for samples with less than 10,000 particles on a Mac M1 chip. If it is taking a long time, try closing all other running programs and relaunching MATLAB.
* If there is an error when running the code, make sure all of the previous steps are correctly executed.
* The code itself has guiding comments double commented out (%%) in green. These prompt personalization in the output and help explain which sections of code are doing what.
* Green code that is single commented out (%) are optional outputs that can be uncommented if desired.

A picture containing logo

Description automatically generated