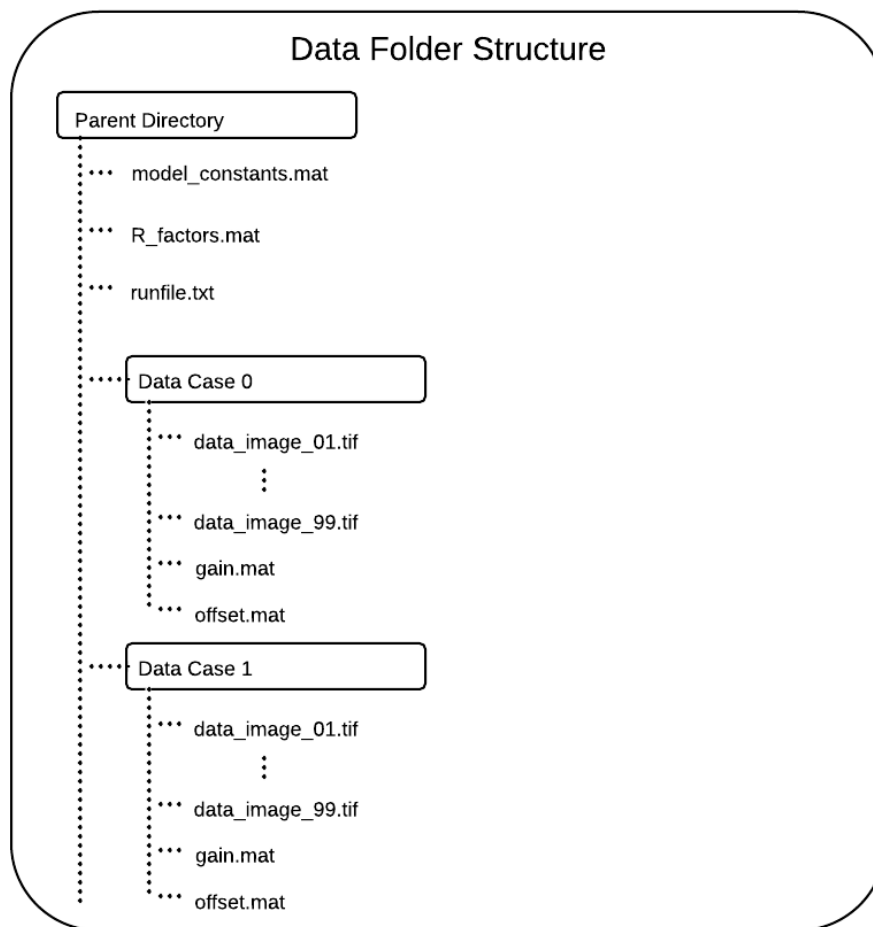


PRE-PROCESSING

The data processing routine requires a specific folder structure and a collection of .mat files containing various constants. The matlab scripts should be stored in a separate directory outside this. Directory and data names do not need a specific name, but all .mat and .tif files must have the names shown.

Folder Structure



Everything should be contained in some overall parent directory (of any name). Directly in the parent directory you should have model_constants.mat, R_factors.mat, and runfile.txt (see instructions for generating these below). You should then package each data case into a directory with a meaningful name (e.g. Wind16) and place all tif data images, the respective gain.mat, and the respective offset.mat into the directory. Repeat this for each data case you wish to process.

In all of the following scripts, the only variable you have to change is ***parent_dir***. This is defined as follows:

parent_dir: full directory path to parent directory as explained above
e.g. 'C:\Users\Steve\SIT_Data\' (single quotes necessary)

gain.mat, offset.mat, model_constants.mat

These should all be provided to you in the sample directory. Simply place them as explained above.

R_factors.mat

This is fairly large file, and must be generated. To do so point matlab to the scripts directory and run the following command:

```
R_factor(1024,1280,.0002898,parent_dir)
```

Check the parent directory to make sure R_factors.mat exists once matlab finishes.

runfile.txt

This will create an index of all data images in the parent directory and subsequent sub-directories. Assuming matlab is still pointing to the scripts directory, run the following command:

```
CreateRunfile(parent_dir, 'tif')
```

Check the parent directory to make sure runfile.txt exists. You can open it in notepad and should see a list of the full path to all data images to be processed and a 0 next to each one.

DATA PROCESSING

Start Data Processing

Once the above steps are completed you are ready to begin processing data. Pre-processing steps should never have to be repeated for this machine.

Run the following command in matlab:

```
ElevationBatch_global(parent_dir)
```

This will start a single session of data processing, but you can launch additional sessions on the same system if you have enough memory. Each process will top out at 8GB of RAM usage. If you have enough memory for a second session (remembering to leave some open memory for the OS) you can start it by simply opening another matlab window and running the same command as above. Be careful not to run over the available memory as once the system hits the pagefile, processing will greatly slow down.

Check Processing

When data processing is running smoothly you should see regular updates in the matlab command window. You should also see new files being created in the Retrieval directory which is automatically created as a sub-directory of the parent.

If a program happens to crash, you can restart it by running the same command as above.

Finalize Processing

The scripts keep track of progress in runfile.txt using the numeric flags next to each filename.

‘0’ – data has not been processed yet

‘1’ – data has successfully finished processing

‘2’ – data is currently processing or failed during processing (e.g. program crashes)

ElevationBatch_global.m will output “Analysis Complete” when there are no more images. Once all instances report this, run the following matlab command to check to make sure all data was processed successfully:

```
CheckRunfile(parent_dir)
```

You should see some output message. If it says that processing is incomplete, re-run ElevationBatch_global as above and then run CheckRunfile again.