# Maintenance Manual

Marcell Veiner

May 2021

To set up the project on a home computer, it is easiest to clone the repository git clone `https://github.com/Vejni/Repository`, and download the datasets from Google Drive, logged in as the user described in the User Manual, Step 2. Alternatively one can download the original datasets and recreate the split and patches using the *datasets.py* script. All project dependencies are in the *requirements.txt* and can be installed via pip.

The main folder contains project files, such as the submit file *task.sh* for Maxwell, *.gitignore, README.md* for git and the launch file *main.py*. The *data* folder contains the datasets (not uploaded to git). The *docs* contain files for the project report, while the *models* folder contains saved models and their checkpoints during training. The *src* folder contains the source code for the project. The overall structure is the following:

- Root: The root folder contains project speficic files, but also the main script which can be run locally.
- src: Contains the source code, such as definitions for the patch- and image-wise networks, training loops and methods for dataset manipulation. Additionally, it contains truncated code for the 3 types of capsules used in the project: DynamicCaps, VarCaps and SRCaps, which can be found in the appropriate folders.
- models: This folder contains saved models, as well as saved checkpoints. Additionally the outs folder contains outputs from experiments, which were then used to create plots. These files are NOT unaltered, as some have been trimmed or put together for ease of plotting.
- docs: Contains the presentation, report and poster files.
- data: Folder for datasets. It is recommended to create the datasets here using the appropriate script.

Description of the files within, can also be found in Chapter 4, but briefly, the *datasets.py* script contains methods to recreate and test the datasets. The *model.py* script defines the basics for all networks used in the project, and patch-wise, image-wise networks and mixed models inherit from the base model. All of these classes can be found in the appropriately named scripts. The folders

*DynamicCaps, SRCaps, VarCaps* contain source code for the capsule networks used in this project. They original sources are listed in previous sections as well as within the script. The files within the folders have been cut to the necessities for our project.

To test the networks on a new dataset, it is recommended to download the dataset into the *data* folder, then using the *datasets.py* script to recreate the patches and the test-train-validation split. It is recommended to select the correct resizing, and patching strategy, which should depend on the dataset itself. Running the script will create two folders, one for patch-wise and one for image-wise training. The only things left are to point the models to the new folder, and specify the correct number of classes, in the arguments.