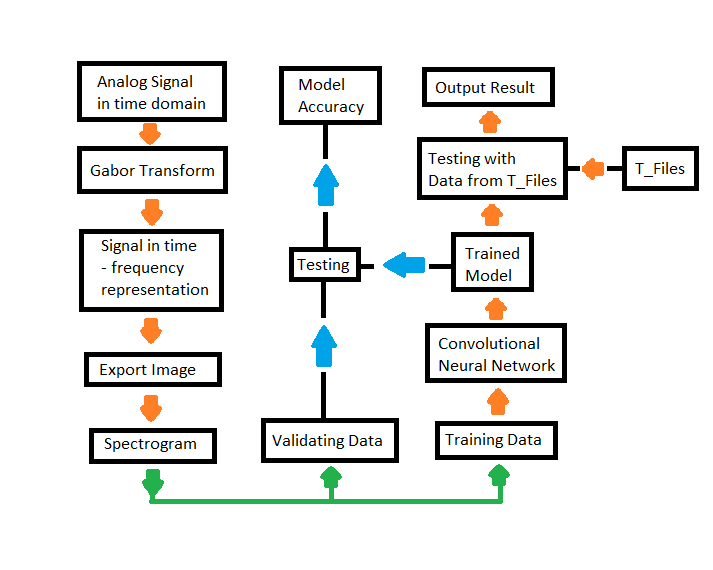
1. Overiew:

The problem of the project is how to classify time-series reflected signal of different objects. To solve this problem, a series of step was laid out in form of a project pipeline, which is presented above in this text.



In Summary, the code for the project was divided into 3 stages. Stage one involves processing of the analog time signal sample into spectrogram via Gabor Transform, see Literature Review for details of Gabor Transform. Stage two trains the Convolutional Neural Network (CNN) via adjusting the weights of the initial network to fit the training set. Stage two ends with validating the learning by feeding the validation set to the trained network. In stage three, the trained network was used to predict unlabeled data from the test file set T\_Files.

1. About the Dataset:

The data was issued from Professor Pech in .xlsx files specifically for the Subject Computational Intelligence at Frankfurt University of Applied Sciences. There are two types of datasets in the provided files from the subject:

1. Training Files:

The training set includes 3 analog time readings samples of 3 different Objects: ‘Data Object1’, ‘Data Object 2’, ‘Data Object 3’. They are located under Matlab/dataset/. Each Row, from the 7th column in each files represents a time-series sample of the object. So, when the samples are read from the training data, the rows are read from the 7th column to the end of each row.

Table

Description automatically generated

The samples in each training files:

|  |  |
| --- | --- |
| Data Object 1 | 315 Samples |
| Data Object 2 | 200 Samples |
| Data Object 3 | 400 Samples |

During the training, the number of samples used for training and validation were chosen via a split of data from the original training files. The current approach used the same number of training samples for each type of Object. Therefore, the number of validation samples left from ‘Data Object 2’ will be less than that of the others. This is the reason that the training data samples is currently set to 200, the maximum number of samples from ‘Data Object 2’.

Due to a current lack of training data, the scope of this Projects focused more on developing a pipeline for dealing with reflected time series signal of objects rather than reaching a good result of training.

1. Test Files:

Test Files are structured the same way as the training Files, but they have no label, there are twelve test files in total, each has 50 data samples.

For the scope of this project, these test files were used as result of the model. They were fed into the trained model after the training session ends for the prediction of the associated label that belongs to the test file. Tests File are located under Matlab/dataset/ and share the same form as “T File <No>” with 12 >= No >= 1.

The documented output of the test file ‘s label is written under Matlab/Result/<netName> with netName is the name of the used trained network.

1. Gabor Transform - Code:

To do have a mathematically description of Gabor Transform, please refer to the literature review above. The creation of Spectrograms involves plotting and export the potted graph into .jpg, so during the creation of the pictures, a plot window will pop up and close for each data sample.

The code explanation below is applied for doing Gabor Transformation in 2 stages of the Experiment. One at the creation of the training data, reading the training set and one at the preprocessing of the test files. It serves the purpose of transforming time-series signal samples into time frequency representations (TFRs).



In Figure <>, the data samples, in time series form of the data was read as a table. In the second row, the data\_range extract the table for only the data from column 7th to end for each data samples.



Because there are only one files for one label, dataLabel for the learning was chosen as the filename. In this work, the file names are ‘Data Object1’, ‘Data Object 2’ and ‘Data Object 3’. Consequently, the respective dataLabel are ‘Data Object1’, ‘Data Object 2’ and ‘Data Object 3’. So at this stage we have classification of multiple objects.

A screenshot of a computer

Description automatically generated with medium confidence

We then proceed to read the length of data\_range (extracted data from the original files, column 7th –> end ) which is 3400 time stamps for every data samples. A variable ‘t’ was used here for the indexing of the timestamps for upcoming iteration. Then a folder was made in trainingData/<dataLabel> to save the output spectrogram.

The Next session discuss about the iteration that simulated Gabor transform in the project. The instruction video of **<insert 45’ guy youtube name>** demonstrated the process beautifully. The code for Gabor Transform in this project is based on his Matlab demonstration.



tslide was used to take the indexes which belongs to the start of each window used for Gabor transform. To recall, Gabor transform makes use of a sliding window in the time domain, then Fourier transform the signal in each window respectively. In this experiment, the window moves 20 samples at a time.

In figure **<No>**The iteration of the Transform based on the index j, running from 1 to the end of tslide’s length.

Chart

Description automatically generated