

This report is presenting two approaches to Optical Character Recognition (OCR). The approaches applied are k -Nearest Neighbor Classifier and Linear Support Vector Classifier. A general overview of the implementation is shown in Figure 1. The data is loaded from the Char74k-Lite dataset and divided, at random, into two databases. 80 % is selected as the training set, and 20 % is the test set. The data is then preprocessed, as described in ???. A model is trained on the training set, before the model is passed to the classifier. The classifier is described in ??. The system output is the classifier error, which is given by

$$E = \frac{N_{\text{fail}}}{N_{\text{test}}} \quad (1)$$

where N_{fail} is the number of wrong classifications in the test set and N_{test} is the total number of samples in the test set.

We implemented the Linear Support Vector Classifier in both Python and Matlab. The Python version was used to optimize detection parameters with grid search and calculate the overall error E_{LSVC} , while the Matlab version uses inherent high level Matlab features for displaying data, and its superb matrix handling.

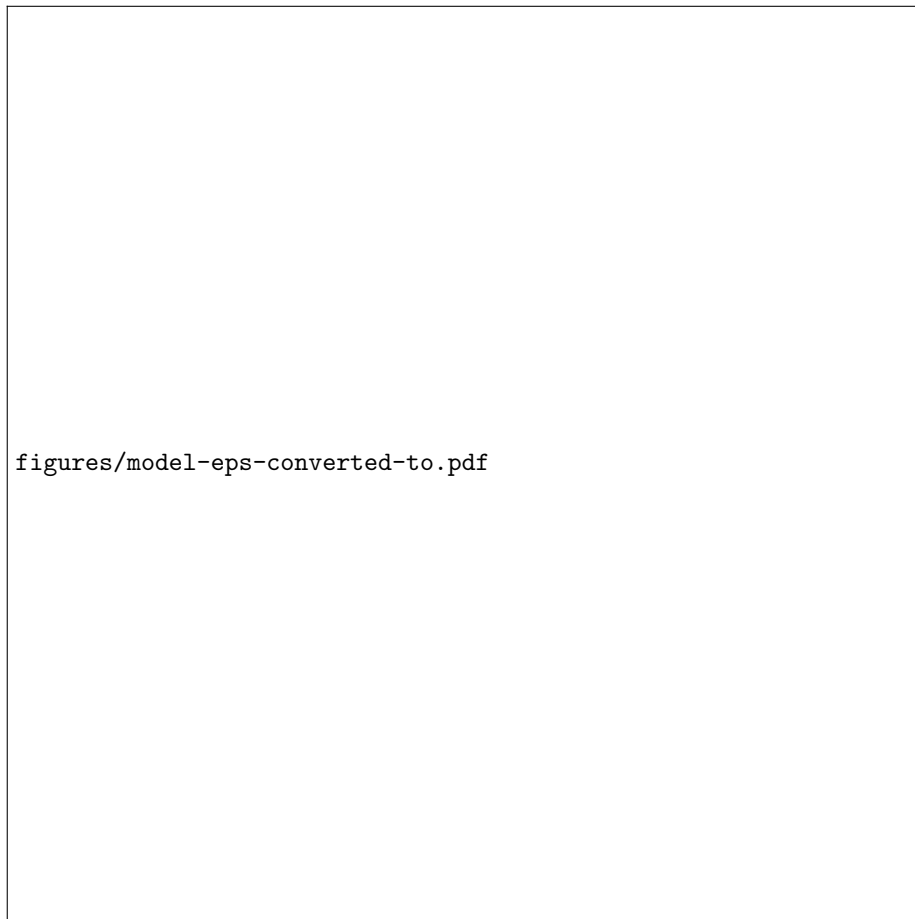
The Python program mainly used two packages: Scikit-image and Scikit-learn. Scikit-image is a opensource image processing library, that was mainly used for preprocessing. Scikit-learn is a library that can be used for classification, regression, clustering, dimensional reduction, model selection, and also preprocessing. It was heavily applied during this project. It is hard to have a complete overview of A complete list of all libraries used in the python implementation are shown in ??. It is hard to have a complete overview of Matlab toolboxes, as namespacing is all but apparent in Matlab. However, the two main toolboxes used in this project were Computer Vision System Toolbox and Statistics and the Machine Learning Toolbox.

The software uses UNIX convention for file paths, so this might not work on Windows. To start the python program: type

```
$ python3 main.py
```

into your terminal. To run the Matlab program, type

```
$ matlab -nodesktop -nosplash
>> run SVM4OCR
```



figures/model-eps-converted-to.pdf

Figure 1: Model of OCR system.