Neural Networks Project Report

ID	Name	Sec	BN
9210684	Ali Mohamed Farid	1	25
9213035	Walid Osama Khamees	2	31
9213347	Mohamed Maher Hasan Amin	2	12
9211164	Mostafa Elsayed	2	20

Project Pipeline

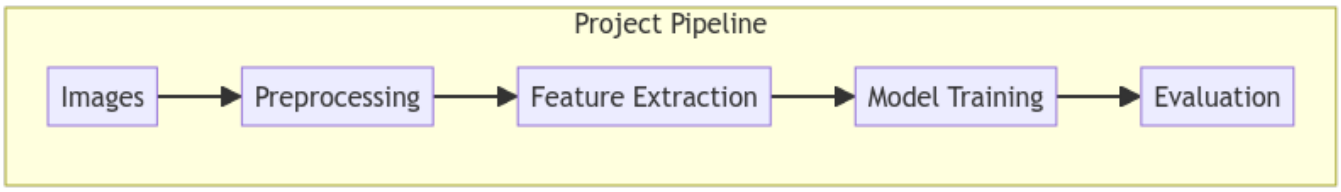

Figure 1: Project Pipeline

The project takes a dataset of images and pre-processes it to be used for the training of the model, the preprocessing stage is described in detail in the next section. The preprocessed data is then into the feature extraction module which extracts the features from the images. The extracted features are then fed into the model which is a regression forest model. The model is trained on the training data and then tested on the testing data.

Preprocessing

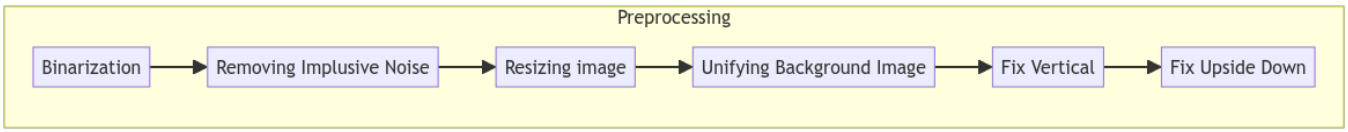

Figure 2: Preprocessing

The preprocessing stage is as follows, first the image is read and binarized then impulsive noise is removed from the image using a median filter. Then the image is resized to 300x300 pixels. Then we detect if the image has a white background if it does then we invert the image. Otherwise leave it as is. Then we detect if the image is rotated 90 degrees if it is then we rotate it back to the correct orientation. Then we check if the image is upside down or not if it is then we rotate it 180 degrees.

Unifying Background Color

We check each of the corners of the image and take the most frequent color to be the background color.

Rotating the Image

We apply closing operation on the image to detect the lines in the image. After we get the largest of the contours we get its bounding box and and calculate the aspect ratio of the bounding box. If the aspect ratio is less than 0.1 then the image is rotated 90 degrees.

Upside Down Detection

We apply a closing, erosion and dilation operations on the image to detect the lines in the image. Then on one of the lines we split it in half if the bottom half has more light pixels then the image is not upside down. Otherwise it is upside down.

Feature Extraction

1. **Gabor filter** We use Gabor filters with two rotation angles and two sinusoidal wavelengths to analyze the image's texture. By aggregating the filter responses using standard deviation and mean calculations, we extract important texture features.

2. **Spectral analysis** We convert the image to the frequency domain using the Fourier Transform. We apply passband filters to analyze the frequency behavior in the text image, helping us identify differences in frequencies across different fonts.

Model Selection

After trying different models we found that the regression forest model gave the best results. The model is trained on the training data and then tested on the testing data.

Performance Analysis

We have done a train test split of 80% training and 20% testing. After this we have done a cross validation of 5 folds. The model has an accuracy of 96%.

Future Work

We plan to improve the model by segmenting the text into words, we will then extract features from the words, instead of extracting the features from the whole paragraph.

Test Command

```
curl -X POST -F "image=@YOUR-IMAGE-PATH" \
https://arabic-font-recognition.onrender.com/predict
```

Work Distribution

Walid Osama Khamees

- Preprocessing
- Model Training

Ali Mohamed Farid

- Feature Extraction
- Model Evaluation

Mohamed Maher Hasan Amin

- Feature Extraction
- Model Evaluation

Mostafa Elsayed

- Preprocessing
- Model Training