

Signature Classification using CNN

September 2024

1 Introduction of Signature Detection Task

I have to classify signatures of different students using deep learning methodologies. We have 184 students and corresponding to each student we have 4 signatures of each. We have to classify them to their respective ID that is present on the input images provided.

2 Methodology

2.1 Pre-Processing

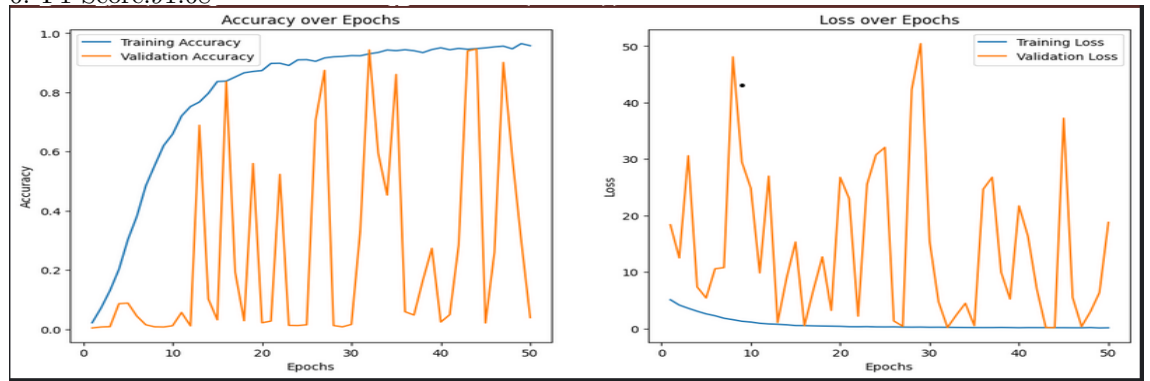
1. Defined a function read and scale image which will resize all images to 888px width and 1212px height. It will also scale the image to 50 percent
2. Defined an extract signatures function that will extract all the signatures and their IDs from images and place those signatures in the folders that have their respective IDs
3. Defined a data augmentation function that will augment all images so that data quantity can increase

2.2 Model Architecture

1. 2 Conv2D layers of 64 filters of size 3x3 with activation relu. In between batch Norm to ensure smooth gradient flow and a final max pool layer to reduce output
2. 2 Conv2D layers of 128 filters of size 3x3 with activation relu. In between batch Norm to ensure smooth gradient flow and a final max pool layer to reduce output
3. 2 Conv2D layers of 256 filters of size 3x3 with activation relu. In between batch Norm to ensure smooth gradient flow and a final max pool layer to reduce output
4. 2 Conv2D layers of 512 filters of size 3x3 with activation relu. In between batch Norm to ensure smooth gradient flow and a final max pool layer to reduce output
5. GlobalAveragePooling2d instead of flatten layer
6. Dense layer with 1024 neurons
7. Dense layer with 512 neurons
8. Final dense layer with number of neurons equal to number of classes and softmax activation function.

3 Results

1. Train Accuracy:94
2. Test Accuracy:91.81
3. Train Loss:0.22
4. Precision:93.35
5. Recall:91.81
6. F1 Score:91.68



4 Discussion

1.Data for training was quite low so I had to augment the data 2.Total classes were 184 so model needed to be very complex in order to capture relationship 3.On training the model converges and then after another epoch it gets far away from optimum. The model does not stay in lowest gradient once it reaches therefore I stopped the model when validation accuracy crossed 90