An Offline Signature Verification and Forgery Detection Method based on a a Single Known Sample and an Explainable Deep Learning Approach

Authors: Hsin Hsiung Kao and Che-Yen Wen

In this paper, Authors proposed an off-line handwritten signature verification method based on an explainable deep learning method (deep convolutional neural network, DCNN) and unique local features extraction approach to tackle the need of deep learning neural networks hunger for data which in turn increases its accuracy.

POINTERS –

1) Deep Convolutional Neural Network (convolutional layer, pooling layer, and fully-connected layer) was used.
2) Signatures of a particular user also have high intra-class variance and also high inter-class variance (between a genuine and a forged signature)
3) To tackle the disadvantage of having a single sample for reference and the above-mentioned issue, each image was converted into overlapping sub images since a lot of features are scattered, which were then fed separately into the DCNN. The convolutional, pooling and FC layers extract all the minute features and those features which are spread over a large distance.
4) Transfer Learning and per training methods were used. Architectures were VGG-19 and Inception V3.
5) This is a binary classification problem (genuine or forged signature). Sigmoid activation function was used in the output layer.
6) A saliency map was generated to display the points or areas which were sensitive to the final prediction of the DCNN. Binary Cross Entropy (BCE) was the Loss or Cost Function. Stochastic Gradient Descent was used as an Optimizer to reduce the loss function during backpropagation. False Rejection Rate and False Acceptance Rate to evaluate the performance of the network.
7) CDAR 2011 SigComp dataset is used for signature verification competition and contains both online and offline signature samples. The offline section of this dataset has different sample sizes of skilled forgeries for each genuine author. 12 experiments with two different network architectures, two genuine authors, and three types of forgery sample size were designed.

DATA PRE-PROCESSING –

1) Raw images were converted into a grayscale image and then saved as a 24-bit BMP file.
2) A block-based method was used for data augmentation to divide the images to multiple sub-images. A window was used to get sub-images block from the original image. Size of the window depends on the architecture of DCNN used (Here in this paper Inception V3 and VGG-19 were used). Window is shifted by 20px everytime. The process is repeated from left-right and then top-bottom.
3) To reduce paper texture and background noise caused by the optical scanning unit, authors brightened each sub-image by 7.5%, and then multiplied each RGB pixel values by 1.075 directly.
4) The genuine signatures used for training is rotated by 10 degrees each time. The forged signature used for training is rotated by 20 degrees each time, and the rest of the sub-datasets are only used for verification and testing, so it is set to 60 degrees to save the computation cost. The genuine signatures are rotated by 10 degrees only so as to tackle the lack of samples. Rotations helps the DCNN adapt to signatures having high intra-class variability.
5) 250 is set as the threshold grayscale value. Any pixel whose grayscale exceeds the threshold is a valid pixel. Then, if a sub-image block has over 7.5% valid pixels, it is classified as a valid sample otherwise it is discarded.
6) All the sub-images are then fed into the network.

PERFORMANCE –

Training Accuracy – Ranges from 99.67% to 100% (For various architectures, genuine and forged samples)

Validation Accuracy – Ranges from 84.58% to 100% (For various architectures, genuine and forged samples)

Testing Accuracy – Ranges from 48.38% to 99.97% (For various architectures, genuine and forged samples)

FUTURE PROSPECTS –

1) Voting Method to be used to increase the accuracy.