Deep Learning Based Handwritten Signature Recognition

Suresh Pokharel [a], Santosh Giri [b], Subarna Shakya[c]
[a, b, c] *Department of Electronics and Computer Engineering, Pulchowk Campus, Nepal*
Corresponding Email : [a] suresh.pokharel@pcampus.edu.np, [b] santoshgiri@ieee.org, [c] drss@ioe.edu.np

***Abstract*— A handwritten signature commonly practiced route for confirming the authenticity of legal documents. The verification of the signature is critical as it varies every time and may change with age, behavior, and environment. This paper presents a Deep learning model based on the CNN architecture to verify the signature. For experimental purpose, the feature extraction portion of the GoogleNet model has been used to transfer value calculation and the classification layer was retrained using back propagation with the concept of transfer learning. The classification layer of the Deep learning model was retrained with 25 classes of signature image dataset with each class consisting of 85 signatures. After training, the model was evaluated with a testing dataset of 15 signatures from each class. The mean testing precision of the neural network architecture with signature dataset was found to be 95.2 %.**

Keywords—Signature Verification, Deep Learning, GoogleNet, Transfer Learning, Back propagation

I. INTRODUCTION

A signature is a person's name written in a distinctive way as a form of identification in authorizing a cheque or document or concluding a letter [1]. The handwritten signature of a person is commonly accepted as a means of verifying the legality of documents such as certificates, checks, drafts, letters, approvals, visa, passport etc. and is indispensable in countering the forgery and falsification of such documents in diverse financial, legal, bureaucratic, academic, and other commercial settings. In the banking system, whenever cashier receives a cheque from the client, the cheque is verified using signature. The cashier compares the signature written on the cheque with some stored record of genuine signatures before proceeding with any legal transaction.

In the context of Nepal, this convention of using the signature as the route for confirming the authenticity of documents has been followed from medieval time to present and will continue in future. Such authentication with signature is at times very critical and crucial in the legal scenario. A signature in any contracts has a vital role to indicate the identity of the person of interest and also to provide evidence of intent and informed consent. Any falsification and fraudulent of a signature may result in severe damages in persons lives and assets. In such cases, a systematic approach to verifying the signature is very necessary to prevent such forgery. Traditionally, authentication of specimen signature is achieved by person: comparing and evaluating the specimen with copies of genuine signature specimens acquired previously and with the help of some sort of witness. In the case of Nepal in banking sector, signature verification is a critical subject in banking transaction and approvals processes. But this simple approach may not be sufficient in all cases as various advanced forgery and falsification techniques are emerging. This paper tries to assist and improve the verification process of a human’s handwritten signature using machine learning.

II. RELATED WORKS

The field of Signature Verification has been broadly researched in the last decades and still remains an open research problem. In [2], the authors use the multiply blend mode, which multiplies the check images by the signatures to obtain the synthetic signature database for gray level distortion in signatures. Otsu’s threshold algorithm [3] was used for binarization and Hough transform [4] was used to detect the beginning and end of each line. Local binary pattern operator [5] is used to find local patterns of the signature image. And finally, nearest neighbor classifier [6] and SVM classifier [7] was used for final classification. For training the model, 10 images per class was used and remaining are used for testing purpose.

In [8], image pre-processing was performed by using different image processing techniques like color inversion, image filtering, and finalization. The next step is feature extraction in which different five geometrical features: Area, Centroid co-ordinates, Eccentricity, Kurtosis and Skewness are used to distinguish signatures. Training was done using Trainlm [9] and for model performance evaluation, 100 signatures from 3 users were tested. The network found to be capable of classifying signatures with the classification ratio of about 93%.

In [10] paper presents a recognition system for offline signature using Discrete Cosine Transform [11] and Hidden Markov Model [12]. In the image pre-processing step, the smoothed image was converted into a binary image by using morphological operation. Discrete Cosine Transform was used for the feature extraction from signature images and for verification Hidden Markov Model [12], a probabilistic pattern matching technique that has the ability to absorb both the variability and similarities between signature images was used. A set of five signature image from each class was used for training the HMM model. Parameters for training are chosen based on the maximum likelihood criteria [13].

III. METHODOLOGY

A) Data collection and Pre-processing

In order to train and to validate the classification model, signature image data was used. In the data collection step, Images of signature specimens of 25 different persons was collected, with 100 signatures per class. Among the 100 signatures, 85 signatures are taken as training set and 15 are used as the test set. The data in hard copy was converted into an image file using a mobile camera and scanned into the computer. Few samples of the collected signatures are presented in fig 1.

Fig. 1: Samples of the collected signatures

In image pre-processing, the different techniques like cropping the image, scaling to appropriate dimension (224pixels×224pixels), giving a proper name and putting them in the separate directory was performed. The average size of an image after preprocessing is about 40 KiloBytes.

B) Implementation of Convolutional Neural Network and Training

Convolutional neural network for this experiment is based on the GoogleNet [14] model originally trained on Disbelief platform. Originally, the model is 22 layer deep Convolutional neural network developed by Google. The architecture details are given in Table 1. According to [14] Training a CNN network from scratch is a computationally intensive task and depending on computer setup it takes even weeks which is not possible with limited resources. To overcome this problem transfer learning [15] was adopted and retraining was performed on the GoogleNet [14] model, which is trained on ImageNet [16] dataset. The GoogleNet model has two parts; a classification layer and a feature extraction layer. The parameters on the classification layer are removed and trained with the transfer values from the feature extraction layer of the model.

TABLE I: THE GOOGLENET ARCHITECTURE

type	patch size/ stride	output size
convolution	$7\times7/2$	$112\times112\times64$
max pool	$3\times3/2$	$56\times56\times64$
convolution	$3\times3/1$	$56\times56\times192$
max pool	$3\times3/2$	$28\times28\times192$
inception (3a)		$28\times28\times256$
inception (3b)		$28\times28\times480$
max pool	$3\times3/2$	$14\times14\times480$
inception (4a)		$14\times14\times512$
inception (4b)		$14\times14\times512$
inception (4c)		$14\times14\times512$
inception (4d)		$14\times14\times528$
inception (4e)		$14\times14\times832$
max pool	$3\times3/2$	$7\times7\times832$
inception (5a)		$7\times7\times832$
inception (5b)		$7\times7\times1024$
avg pool	$7\times7/1$	$1\times1\times1024$
dropout (40%)		$1\times1\times1024$
linear		$1\times1\times1000$
softmax		$1\times1\times1000$

IV. EXPERIMENTAL SETUP

The pre-trained CNN model, GoogleNet [14] is used for experiment and the platform used here is tensorflow [17] and the hardware used is DELL: Intel i5, 1.7 GHZ processor with 7.7GiB Memory. To test the model on signature image dataset, 25 classes of signature images with each class consisting 100 images were used. The signatures images were divided into training, validation and testing sets. Amongst 100 images 85 images were used for training and validation of the model, and remaining 15 images were used for testing purpose. While training the model, transfer learning [15] is adopted in pre-trained GoogleNet model. By using the transfer learning [15], classification layer of the GoogleNet is retrained by signature image dataset.

The classification layer of GoogleNet was re-trained by using backpropagation algorithm [18] and weights of the classification layer are adjusted by using cross-entropy cost function. The parameters used for training the model are training steps (5000), learning rate (0.01) and training interval (1).

V. RESULTS

A) *Training*

The training, validation, and cross entropy graph on the signature image data is given in fig 2 and fig 3. fig 2 represents the training and validation accuracy on signature dataset. the model was trained using back propagation algorithm, during the training phase, initially the training and validation accuracy were at around 96 % and 90% respectively. As the training iteration increases, the training and validation accuracy got improved. the randomness in training and validation accuracy is due to dissimilarities of data in original model and our data.

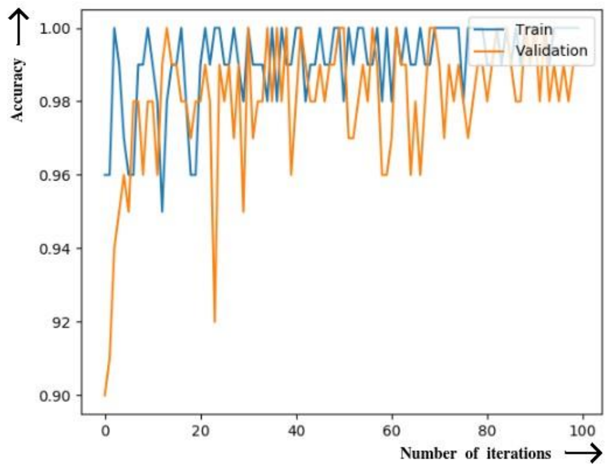

Fig.2: Training and Validation Accuracy graph.

Figure 3 shows the cross entropy error of the model during training and validation. Initially cross entropy was high and as the training and validation accuracy improved, the error begins decreasing.

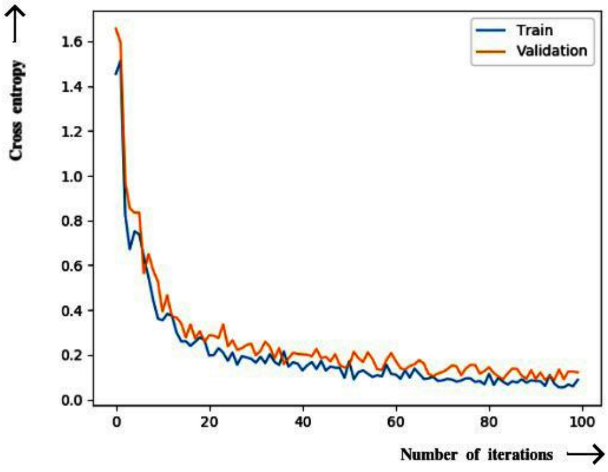

Fig.3: Cross-entropy error graph.

B) *Testing and Evaluation*

The performance evaluation of the model after testing it on signature image data is given in Fig 4. It illustrates the actual result of the experiment. The result is presented in confusion matrix. output parameters are presented in the table 1. Among the total of 275 images from 25 classes the total positive obtained was 257 and False Positive was 18. Other details are given in the confusion matrix.

TABLE II: CONFUSION MATRIX REPRESENTING RESULT OF THE EXPERIMENT.

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19	c20	c21	c22	c23	c24	c25
c1	14								1																
c2		15																							
c3			14					1																	
c4				15																					
c5					14										1										
c6						14							1												
c7							15																		
c8			1					14																	
c9									15																
c10										15															
c11							3				11	1													
c12												15													
c13													13							2					
c14														15											
c15					1										14										
c16																15									
c17						2											13								
c18																		15							
c19											1								14						
c20																				15					
c21							1														14				
c22																						15			
c23																							15		
c24		1																						14	
c25														1											14

VI. CONCLUSION

In this paper, the classification layer of pre-trained GoogleNet model was re-trained successfully with the collected signature data set by using transfer learning mechanism based on CNN architecture. The experimental modal described in this paper gives the precision of 95.2 % on primary signature image data.

ACKNOWLEDGMENT

We would like to thank the people who helped us by providing their handwritten signature for this research work.

REFERENCES

[1] O. E. Dictionary, "Oxford english dictionary," *Simpson, JA & Weiner, ESC,* 1989.

[2] M. A. Ferrer, J. F. Vargas, A. Morales, and A. Ordonez, "Robustness of offline signature verification based on gray level features," *IEEE Transactions on Information Forensics and Security,* vol. 7, no. 3, pp. 966-977, 2012.

[3] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics,* vol. 9, no. 1, pp. 62-66, 1979.

[4] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern recognition,* vol. 13, no. 2, pp. 111-122, 1981.

[5] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis & Machine Intelligence,* no. 12, pp. 2037-2041, 2006.

[6] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE transactions on systems, man, and cybernetics,* no. 4, pp. 580-585, 1985.

[7] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning,* vol. 20, no. 3, pp. 273-297, 1995.

[8] A. Karouni, B. Daya, and S. Bahlak, "Offline signature recognition using neural networks approach," *Procedia Computer Science,* vol. 3, pp. 155-161, 2011.

[9] J. J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory," in *Numerical analysis*: Springer, 1978, pp. 105-116.

[10] S. A. Daramola and T. S. Ibiyemi, "Offline signature recognition using hidden markov model (HMM)," *International journal of computer applications,* vol. 10, no. 2, pp. 17-22, 2010.

[11] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE transactions on Computers,* vol. 100, no. 1, pp. 90-93, 1974.

[12] E. L. Sonnhammer, G. Von Heijne, and A. Krogh, "A hidden Markov model for predicting transmembrane helices in protein sequences," in *Ismb,* 1998, vol. 6, pp. 175-182.

[13] G. N. Murshudov, A. A. Vagin, and E. J. Dodson, "Refinement of macromolecular structures by the maximum-likelihood method," *Acta Crystallographica Section D: Biological Crystallography,* vol. 53, no. 3, pp. 240-255, 1997.

[14] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition,* 2015, pp. 1-9.

[15] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering,* vol. 22, no. 10, pp. 1345-1359, 2009.

[16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition,* 2009, pp. 248-255: Ieee.

[17] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16),* 2016, pp. 265-283.

[18] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural computation,* vol. 1, no. 4, pp. 541-551, 1989.