IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.301

Volume 3, Issue 10, May 2023

Signature Verification using Convolutional Neural Networks for Forgery Detection

Dr. Reshma Banu[1], Monika H[2], M Saniya Sultana[3], Nisarga M[4], Kumar H[5]

Professor, Department of Computer Science and Engineering[1]

Students, Department of Computer Science and Engineering[2,3,4,]

Vidya Vikas Institute of Engineering and Technology, Mysore, Karnataka, India

Abstract*:* *Despite ongoing research in the field, the challenge of fully solving handwritten signature verification persists. Handwritten signatures play a vital role in authentication and proof within our social and legal spheres. Approval is granted only when a signature originates from the intended recipient, as it is highly unlikely for two signatures created by the same individual to be identical [7]. Even when the same person creates two signatures, numerous signature characteristics can undergo changes, making forgery detection a complex task. Signature verification systems aim to distinguish between genuine and counterfeit signatures. In this study, a proposed approach for signature verification utilizes convolutional neural networks (CNNs). By employing a CNN model, a more precise representation of image information can be extracted. The CNN model is trained on raw signature images to perform feature extraction and data augmentation, enabling judgments regarding the authenticity or falsification of a given signature. This software holds potential for verifying signatures across diverse platforms, including loan and application signings, as well as legal document authentication.*

Keywords: CNN, feature extraction, pre-processing, machine learning, RELU, and deep learning

I. INTRODUCTION

An individual's legal mark, used for authentication and performed by hand, can be referred to as a handwritten signature.

There are two primary categories of techniques and systems utilized for signature verification. The second category is based on the online signature verification method, which involves the use of additional hardware devices directly connected to a computer. On the other hand, the offline signature verification method relies on fewer hardware devices and utilizes images captured with a camera. In the case of offline verification, a reduced set of characteristics is employed. Throughout history, the utilization of signer signatures for identifying individuals has been a significant innovation. Biometric devices are typically categorized into two groups: verification and identification. Verification and identification processes are distinct from each other. Verification aims to determine whether a person's biometric truly belongs to them, while identification aims to recognize the person's biometric from a pool of potential candidates [1]. This essay explores the application of fingerprints for individual validation. The verification process involves two types of signature marks: genuine and fake.

Given the widespread use of signatures, there are many malicious actors attempting to forge them for personal gain.

Consequently, highly effective techniques for detecting signature forgery are necessary. The creation of a signature verification and detection system typically involves solving five sub-problems: data acquisition, pre-processing, feature extraction, the comparison process, and performance evaluation. In this paper, we propose an offline convolutional neural network (CNN) technique for verifying handwritten signatures. By utilizing a CNN-based approach and programming in Python with its libraries, we achieved successful detection of forged signatures [3]. The CNN model is trained using a dataset of signatures, and predictions are made based on information indicating the authenticity of a signature. Security systems deployed in public locations such as ATMs, official government buildings, colleges, legal organizations, etc., can be developed as applications or websites.

DOI: 10.48175/IJARSCT-10469

II. CONVOLUTIONAL NEURAL NETWORK

A deep supervised learning design employing a multi-layer convolutional neural network demonstrates the potential to independently extract features for classification. These networks find applications in medical image analysis, including picture classification, segmentation, and analysis. Within a CNN, there are two key components: an automated feature extractor and a trainable classifier [4]. The feature extractor utilizes convolutional filtering and down sampling techniques to extract relevant features from the input data.

In our proposed technique, we utilize a CNN as both a feature extractor and a classifier. We hypothesize that a CNN trained to classify genuine and forged signatures can effectively extract discriminative features related to forgery-related behavioural traits, such as hesitations and delays during signature creation. These complex signature components contribute to the creation of a feature vector using the CNN's feature extractor. Each input image undergoes a series of convolutional layers with filters (kernels), followed by a pooling layer, fully connected layers, and the SoftMax function to classify objects with probabilistic values ranging from 0 to 1. This training and testing process is conducted to evaluate the deep learning CNN model.

The following diagram illustrates the complete workflow of the CNN, processing an input image and classifying objects based on their assigned values.

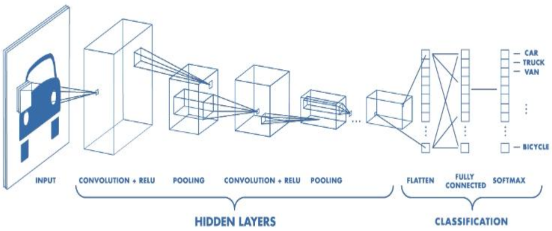


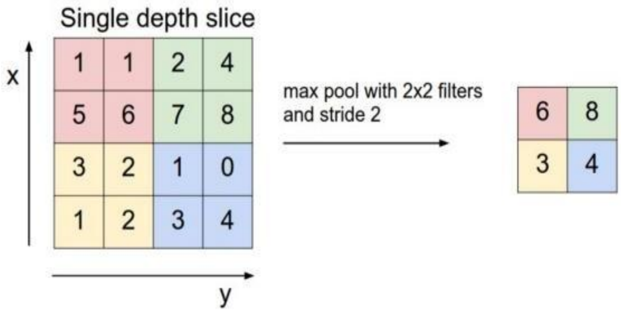
Fig: Layers of CNN

- **Convolutional layer:** The initial layer responsible for extracting features from an input image. It preserves the pixel relationships by learning image features through the use of small squares of input data. This mathematical operation involves the combination of an image matrix and a filter or kernel.
- **Stride:** Stride is a parameter of the neural network's filter that determines the amount of movement over the image. It defines the number of pixels the filter shifts across the input matrix. For instance, a stride of 1 moves the filters one pixel at a time, while a stride of 2 moves the filters two pixels at a time, and so on.
- **Padding:** In situations where the filters do not perfectly fit the input image, we have two options:
- **Zero padding:** The image is padded with zeros to accommodate the filter size.
- Valid padding: The part of the image where the filter does not fit is dropped, retaining only the valid portion of the image.
- **Non-Linearity (ReLU):** ReLU, short for rectified linear unit, is a non-linear operation that applies the activation function f(x) = max(0, x). It introduces non-linearity into our CNN, aiding in feature extraction and representation[5].
- **Pooling Layer:** The pooling layer aims to reduce the number of parameters when dealing with large images. Spatial pooling, also known as sub-sampling or down sampling, decreases the dimensionality of each feature map while retaining crucial information.

Different types of spatial pooling layers include:

- Max Pooling: Selects the maximum element from the rectified feature map.
- Average Pooling: Computes the average of elements in the feature map.
- Sum Pooling: Calculates the sum of all elements in the feature map.

DOI: 10.48175/IJARSCT-10469



Fully connected layer: After passing through multiple convolutional and max pooling layers, the final classification takes place using fully connected layers. In this step, the matrix is transformed into a vector format and then inputted into a fully connected layer, resembling a conventional neural network structure.

III. IMPLEMENTATION

The machine receives datasets containing image data of numerous signature examples. These datasets are used to train and test the samples. The public datasets consist of both authentic and fake signature. Before storing the images in separate directories, each picture is categorized as either authentic or fake.

Data Pre-processing:

To achieve the desired goal, data pre-processing is necessary since the pictures in the datasets are not identical and may not be oriented in the same way. Pre-processing involves enhancing crucial image features for subsequent dataprocessing and reducing unwanted distortions in the image data. The following pre-processing steps are applied:

- Conversion from RGB to grayscale: This conversion eliminates color information, transforming the images into a 3-pixel-deep matrix with X and Y dimensions. The average pixel value is computed for each RGB color, resulting in an approximate grayscale value.
- Noise Removal: Any known type of noise present in the pictures is sparsely added to the grayscale images to mimic real-world conditions. This step aims to remove unwanted noise and improve the overall process.
- Grayscale to bitmap conversion: Converting the images from grayscale to bitmap results in the creation of a matrix.
- Resizing: The matrix is resized to a standard image size.

Feature Extraction:

In this step, features are extracted from the images to facilitate the subsequent comparison process. Three types of features are generated: universal, mask, and grid. Global characteristics provide wavelet and Fourier coefficients, mask features provide information about signature line orientations, and grid features offer insights into the general appearance of the signature.

Data Splitting:

The data is divided into two or more subsets for evaluation and training purposes. One subset is used to train the model, while the other is utilized for testing. Typically, the training set is smaller, while the testing set is larger.

Cross-Validation:

To determine the authenticity of the signatures after training and testing, data cross-validation is employed. The images are compared, and their authenticity is assessed based on the extracted features.

Classification:

The classification process involves comparing the extracted features with the stored images obtained during the feature extraction stage. This comparison helps determine whether the characteristics of the signature are considered genuine or fake.

DOI: 10.48175/IJARSCT-10469

The test sets are chosen by split ratio. These processes are done using MATLAB.

After pre-processing the signature are stored in a file directory using the keras library. CNN is implemented using Keras with Tensor Flow backend in Python. These are used to train the patterns associated with signatures.

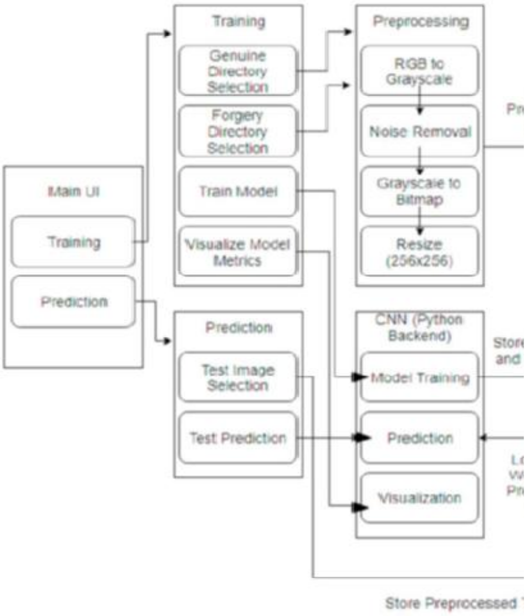


Fig: Architecture diagram

Then model will be verified using accuracy and loss metrics to check whether the prediction is correct or not.

In Front End development PyQt is a GUI widget. It is a Python interface for Qt, a widely used and feature-rich cross-platform GUI library that empowers developers to create robust graphical user interfaces.

QtCore module contains non-gui functionality which is used for working with file and directory.QtGui module contains all graphical controls. QtXml module used for working with XML.

IV. RESULT

The provided method successfully accomplished handwritten signature verification with enhanced efficiency and accuracy. It also demonstrated the capability to easily detect real and forgery signature. Python and its libraries were utilized in combination with a Convolutional Neural Network (CNN) to develop a solution for detecting signature forgery.

During the training phase, the proposed method would generate training and validation accuracy and loss.

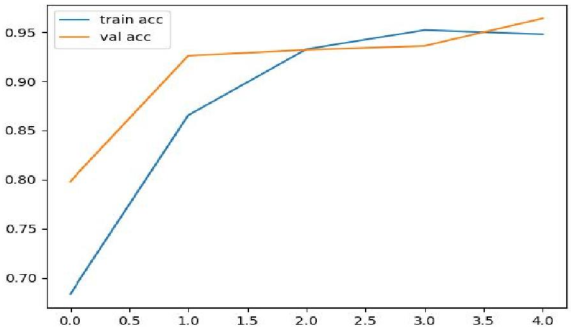


Fig: Training and validation accuracy

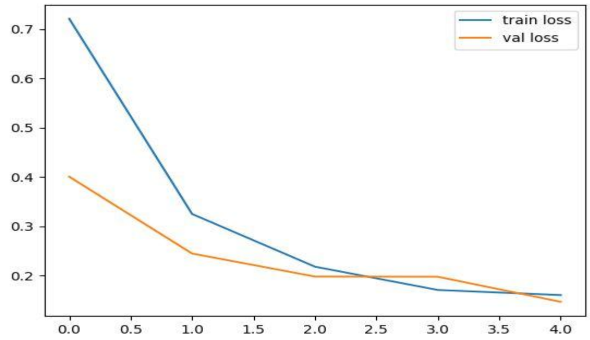


Fig: Training and validation loss

V. CONCLUSION

The proposed system has the capability to learn from signatures and provide forecasts for forgery detection. With the increasing digitalization in various aspects of daily life and emerging challenges in offices and agencies, there is a need for effective user verification techniques. The system can be utilized in any scenario where a signature serves as a means of authentication, including banks and educational organizations. Neural networks have proven to be successful in addressing various problems due to their ability to solve issues relatively easily. In this system, convolutional neural networks (CNNs), known as the most effective model for picture recognition and verification, are employed. When provided with real and fake signature examples from individuals whose signatures were encountered during training, the CNN performs exceptionally well in verifying signatures. It accurately classifies the input images as either genuine or fake, providing two class labels for the results

VI. REFERENCES

[1]. Gideon J., Kandulna A., Kujur AA., Diana A., Raimond K. Handwritten Signature Forgery Detection Using Convolutional Neural Networks 8th International Conference on Advances in Computing and Communication (ICACC-2018).

[2]. Poddar J, Parikh V, and Varti SK Offline Signature Recognition and Forgery Detection using Deep Learning The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40), Warsaw, Poland, April 6–9, 2020.

[3]. KshitijSwapnil Jain, UditAmit Patel, RushabKheni (2021). Handwritten Signatures Forgery Detection using CNN. International Research Journal of Engineering and Technology(IRJET), vol 08 issue:01| Jan 2021.

[4]. Handwritten Signature Verification using Local Binary Pattern Features and KNN 2019: TejasJadhav.

[5]. Collobert, Ronan; Weston, Jason A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning Proceedings of the 25th International Conference on Machine Learning.

[6]. Bin Xiao, Yang Wei, Xiuli Bi, and Weisheng Li (2020). Image Splicing Forgery Detection Combining coarse to redefined convolutional neural networks and adaptive clustering Information Science. 511:172–191.

[7]. Ruiz, V., Linares, I., Samchez, A., and Velez, J.E. (2020). Off-line Handwritten Signature Verification Using Compositional Synthetic Generation of Signatures and Siamese Neural Networks Neurocomputing. 374:30-41.

[8]. S Jerome Gideon, AnuragKandulna, AronAbhishekKujur, r, Diana, and KumudhaRaimond (2018) Handwritten Signature Forgery Detection Using Convolutional Neural Networks Procedia Computer Science. 143:978–987.

[9]. Hafemann L. G., Sabourine R., and Oliveria L. S., Learning feature for offline handwritten signature verification using deep convolutional neural networks and pattern recognition, volume 70, 2017, pages 163–176, ISSN 0031-3203.

[10]. Syed Faraz Ali Zaidi and Shahzaan Mohammed, "Biometric Handwritten Signature Recognition".

DOI: 10.48175/IJARSCT-10469

[11]. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a Convolutional Neural Network,",2017 International Conference on Engineering and Technology (ICET), pp. 1–6, 2017.

[12]. S. Sadak, N. Kahraman, and U. Uludag, "Handwritten signature verification system using sound as a feature," in Proceedings of the 2020 43rd International Conference on Telecommunications and Signal Processing (TSP), pp. 365–368, Milan, Italy, July 2020.

[13]. V. Bonde, P. Narwade, and R. Sawant, "Offline signature verification using a convolutional neural network," in Proceedings of the 2020 6th International Conference on Signal Processing and Communication (ICSC), pp. 119–127, Noida, India, March 2020.

[14]. H. A. B. Nehal and M. Heba, "Signature identification and verification systems: a comparative study on the online and offline techniques," Future Computing and Informatics Journal, vol. 5, no. 1, 2020.

[15]. A Comparative Study among HandwrittenSignature Verification Kancharla, K., Kamble, V., and Kapoor, M.: Handwritten signature recognition: a convolutional neural network approach. In: 2018 International Conference on Advanced Computation and Telecommunication (ICACAT), pp. 1–5. IEEE (2018).

[16]. Jadhav, T.: Handwritten signature verification using local binary pattern features and KNN. Int. Res. J. Eng. Technol. (IRJET) 6(4), 579–586 (2019).

[17]. Sam, S.M., Kamardin, K., Sjarif, N.N.A., and Mohamed, N.: Offline signature verification using deep learning convolutional neural network (CNN) architectures GoogLeNet inception-v1 and inception-v3. Procedia Computer Science, 161, 475–483 (2019).

[18]. Yapici, M.M., Tekerek, A., and Topaloglu, N.: Convolutional neural network-based offline signature verification application. In: 2018 International Congress on Big Data, Deep Learning, and Fighting Cyber Terrorism (IBIGDELFT), pp. 30–34. IEEE (2018).

[19]. Mohapatra, R.K., Shaswat, K., and Kedia, S.: Offline handwritten signature verification using CNN inspired by Inception V1 architecture. In: 2019 Fifth International Conference on Image Information Processing (ICIIP), pp. 263–267. IEEE (2019)

[20]. Sudharshan, D.P., and Vismaya, R.N.: Handwritten signature verification system using deep learning. In: 2022 IEEE InternationalConference on Data Science and Information Systems (ICDSIS), pp. 1–5. IEEE (2022)

[21]. Tamrakar, P., and Badholia, A.: Handwritten signature verification technology using deep learning—a review. In: 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 813–817. IEEE (2022)

[22]. Mosher, Q.S., and Hasan, M.: Offline handwritten signature recognition using a deep convolutional neural network. Eur. J. Eng. Technol. Res. 7(4), 44–77 (2022).

DOI: 10.48175/IJARSCT-10469