IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

Signature Matching using Key Point Descriptors

Prof. Chethan H V, Abhilash PS, Bandlapalli Tanishq, Dugamari Sandeep Reddy
Department of Information Science and Engineering
SJC Institute of Technology, Chikkaballapura, Karnataka, India

Abstract: *This report presents the implementation of a CNN which aims to classify and verify the authenticity of the offline signatures, using the writer-independent method based on ORB features. In order to develop this work, 2 databases (training/validation and testing) were built manually, i.e., the manual collection of the signatures of the 3 users as well as forged signatures made by people not belonging to the base and altered by the same users were done, and signatures of another 115 people were used to create the category of non-members. Once the network is trained, its validation and subsequent testing is performed, obtaining overall accuracies of 85.4%, showing the features learned by the network and verifying the ability of this configuration of neural network to be used in applications for identification and verification of offline signatures.*

Keywords: Signature Matching

I. INTRODUCTION

Signatures, apart from biometric identifications such as fingerprints, are person authentication systems that are currently accepted in most social contexts as a means of identifying and validating identity. However, the forgeries of signatures is a problem immersed in this system that represents a big problem which has been tried to solve or mitigate in different ways, using different methods of verification of the authenticity of a signature. In the first place, it must be taken into account that there are 2 ways of obtaining data from a signature, where the first one is offline, signatures that are obtained from documents already signed, which are scanned in order to acquire a digitalization of the same [1], either by passing the image in gray scale or by binarization. While the second is online, where data is acquired during the process of making the signature [2, 3], taking information such as the start and end position, number of times the pen is lifted, speed, pressure, etc. Comparing the use of online or offline signatures, in general, the use of online methods has a greater degree of accuracy, however, their problem is that the data can only be obtained through special devices and at the time of signing, so it can be a problem when they are signatures already made in documents or bank checks, for this reason, the use of methods with offline signatures that validate them, is necessary.

One of the most important causes for the methods of verification of offline signatures does not have the degree of accuracy of online, is the loss of dynamic information, which makes the problem more difficult [4]. On the other hand, within offline systems, there are two methods, writer-dependent (WD), which is to perform a verification system for each user individually, and writer-independent (WI), where a single system covers the verification of all registered users [5], nevertheless, a WI system represents a greater challenge, since it is possible not to focus the system on recognizing the specific details of a user, but to generalize characteristics [6], as well as erroneously to say that the signature of one user belongs to another.

Due to the above, several techniques have been implemented for the offline signature verification, such as the one presented in [7], in which both the WI and WD methods are used to generate a hybrid classifier, reaching accuracies from 86.04% up to 94.62%. Likewise, thanks to the high performance of neural networks in the classification of users using handwritings employing the WI method, as can be seenin [8] and [9] with accuracies of up to 99%, have begun to be used to implement systems for verifying offlinesignatures. An example of this is described in [10], where it is built a neural network architecture which, afterbeing trained with a database of 111 users, obtained a 96.3% in its overall accuracy.

With the rise of the Deep Learning techniques [11], more robust networks, such as Convolutional Neural Networks (CNN) [12], have been started to be used in recognition of patterns in images. Those networks are able to extract higher level features, allowing its use in complex developments related to letters and words, such as in the recognition of license plates of cars [13] and word spotting applications [14, 15], among others. The CNN has also demonstrated

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

to have better performance than the common artificial neural networks, surpassing the accuracy in more than 10% [16]. The advantage that the CNNs have is that they do not only focus their learning on finer details but on general appearance of the signatures [17]. This capability has given the possibility of applying them in works like the one developed in [18], where a Deep CNN is used in order to discriminate which signatures belonged to which user using the WI method, and then move on to another phase, where WD is used to classify the signature as genuine or not.Other similar work is described in [19], where a hybrid WI-WD model is implemented, using a 2-channel CNN with a channel for a query signature and other for a reference signature, additionally, adding an WD stage with a support vector machine (SVM) trained with the CNN features of the last down sampling layer.

In [20], in contrast to the aforementioned works, the authors propose 3 stages, that were called "feature learning", where a pre-trained residual CNN was used to learn the characteristics of the signatures, then, 2 additional stages "active learning" and "final verification" are implemented with SMV, the first one tofinally separate the genuine signature from the forgeries, and the other to finally verify the authenticity. However, those development needs at least 2 stages to finally verify if the signature is genuine or not, hence, the idea of our work is to combine the two stages in only one network capable of identify a user signature and verify its authenticity and also discard the signatures no belonging to the users registered. A similar approach to our work is presented in [21], where the WI and WD stages are combined into only one Siamese CNN, nevertheless, it needs to compare the similarity between a pair of signatures, i.e. a sample of a genuine signature is needed. In our work, no sample to compare is needed in order to verify the signature entered to the network.

In order to advance in the development of the techniques used for offline signature verification, this work presents the use of a DAG-CNN, a network that consists of a CNN configured with a structure typeDirected Acyclic Graph (DAG) Network [22], which has as advantage over a CNN configured in a linearway the use of multiple paths or branches in order to allow the network to increase its depth (number of convolution layers used) without having to make it longer. Also, each path can be configured in different ways, providing the network with the possibility to learn different features directly from the original input. With this network, we seek to verify the authenticity of different signatures using the WI method for an application where a registration of 3 users is done, which are accepted by means of their signature, additionally, the network must be able to know if the signature that is made belongs or not to the peoplein registered.

This work is divided into 5 sections, where section 2 describes the database built for the implementation. Section 3 presents the proposed architecture together with its respective training and validation. Section 4 presents the results obtained through a new database and the analysis of the behavior of the network. Finally, section 5 gives the conclusions reached.

II. METHODOLOGY

2.1 Database

Because this work is focused on the verification of signatures of specific users, we have downloaded the standard dataset from the Kaggle web site. This database consists of 50+ genuine signatures and 50+ forgeries made manually by users.

2.2 Database Collection

The collection of the genuine signatures is done with each of the users, writing it on a white sheet, without restriction of size and direction, i.e., as the person does not always sign in the same way, the signaturecan be made larger or smaller, just as it may have variations in its inclination. In addition to this, the collection of all the signatures is done in a single day, in other words, people start with the rested hand, but while they advance in the writing, they are going to feel annoyance or fatigue when writing, doing that they have major variations in the elaboration of the signature, this to have major variation in the genuine signatures. In total, 339 genuine signatures are obtained. The forgeries consist of 2 types, some elaborated by the same user, so that he makes slight variations where he/she is able to identify that is not his/her signature, and made by third parties (people not belonging to the database), collecting a total of 260 forged signatures. A sample of the database can be seen in Figure 1.

2.3 Preprocessing

Two phases of preprocessing were carried out: the first phase was done manually, where each of the signatures was digitalized with a scanner of 300 dpi resolution, saved in JPG format, followed by this, eachof the signatures was cut to a predefined size, taking into account the largest signature, which was 570×920 px, without scaling, i.e., with its

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

original digitization size, and finally, the residues not belonging to the signatures, such as stains or dirt, were eliminated. In the second phase, the improvement of the image is performed, which consists of passing it to grayscale and making a contrast adjustment to highlight the details of the signatures using (1).

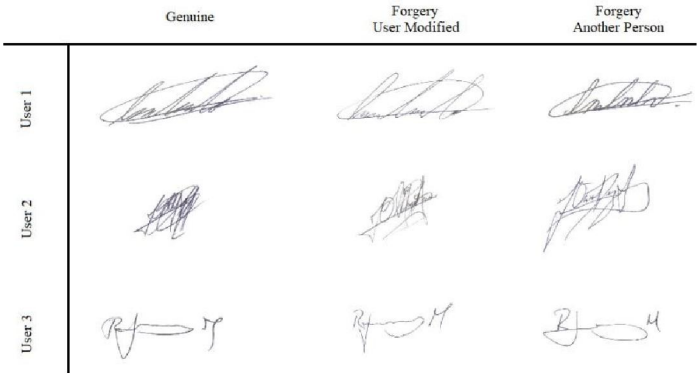


Figure 1. Samples of each user

2.4 Data Augmentation

To increase the database without needing to require more signatures of the users and in turn to make changes in the positions of the signatures, it is decided to carry out a data augmentation. In order to do this augmentation, a location algorithm is applied using morphological operations [24] and a conditionalof area, i.e., the image is binarized with a fixed threshold, where pixels with values greater than 252 become 1, otherwise, become 0, to then applying a close operation (dilation followed by erosion) followed byan open operation (erosion followed by dilation), in order to eliminated the noise that was not erased in the first phase of the preprocessing. With this, a conditional is applied over the remaining areas, where if they have less than 500 px, they are erased, in order to avoid taking into account residual stains, since the resultingareas are used to generate the bounding box that will contain the signature. To the bounding box, 20 px are added to each side, in order to cover possible parts of the traces that might be out from it, and then, the box is located on the original image. This process can be seen in Figure 2.

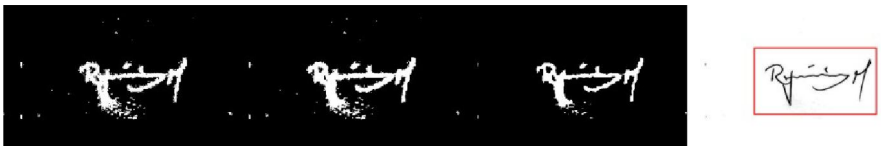


Figure 2. Process to locate the region of interest of the signature in the image

In this way, the region of interest of each signature is found, from which the values of location ofthe vertices of the bounding box are obtained, comparing what vertex is more separated from the edge of the image. Taking into account this separation, a signature shift is performed, whose value is random between the point of the vertex and half the distance from the farthest edge, both in the height and in the width of the image. This process is performed at least 4 times in each category, resulting in a total of 1356 genuine and 1381 forgeries, adding 461 signatures not belonging to the 3 users. From that database, 50 are randomly extracted from each category to carry out the validation of the network, obtaining a dataset of 2,848 signatures for training and 350 for validation.

2.5 SIFT, Surf and ORB Features

A. Scale invariant feature transform (SIFT) Descriptor

In 2004, SIFT [6], [7] was proposed by D. Lowe an invariant feature detector. SIFT uses a cascade filtering concept to detect the features and convert image data into scale-invariant features. SIFT detects localfeatures which are robust against illumination changes, minor changes in viewpoint, and noise. In general, SIFT consist of four main stages: scale-space detection, key-points localization, orientation assignment, and extraction of the key-point descriptor.

In the scale-space detection stage, SIFT decomposes the original image using a Gaussian pyramid, which has multiple

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

levels called octaves. Each octave is also decomposed into multiple sub-levels through convolving the original image with Gaussian filters with different scales. Each pixel DoG is compared withits eight neighbors; when the pixel has the maximum or the minimum value among all the eight neighbors' pixels, it is considered as a *key-point*. SIFT uses the quadratic Taylor expansion of the DoG scale-space function. Around the key-point, the direction and the magnitude of the gradient are calculated for each pixel and the orientation histogram is formed. Once this process is completed, the highest value is considered asthe orientation of the key-point.

B. Speeded-up robust features (SURF) Descriptor

In 2006, Herbert Bay *et al.* presented SURF [8] algorithm. This algorithm contains four main steps: interest point detection, location and scale-space representation of interest points, local neighborhood description, and key-points matching. To detect the interest points, as a first step, SURF uses square-shaped filters to compute Gaussian approximation after the image was already cropped and discretized. Then, the Hessian blob detector [9] is used, which computes the determinant of the Hessian matrix around each point. The point that gets the highest determinant is considered as an interest point.

The determinant is used to compute the location and the scale-space representation of interest points, SURF applies different filter sizes to represent the scale-space representation, then the highest determinant of the Hessian matrix is added in the image space and scaled as Brown et al. proposed [9].

In order to identify the rotational invariance, the orientation of interest points is found. After SURF computes the Haar wavelet [8], we collect responses in the circular neighborhood around the interest point and weighting them by a Gaussian function. In order to evaluate the primary orientation, all responses are calculated within a sliding window of $\pi/3$ size, and the sliding window's size is chosen carefully to maintain a balance between angular resolution and robustness. SURF is widely used in image matching and recognition systems, including steganography [10], face liveness detection or face anti-spoofing [11].

C. Oriented FAST and rotated BRIEF (ORB) Descriptor

In 2011, Rublee proposed oriented FAST and rotated BRIEF (ORB) that is built on FAST key-point detector and BRIEF descriptor. These two algorithms are attractive because of their superior performance and low time requirements [12], [13]. FAST detector [14], [15] is a technique that finds key-points in real time that match specific visual features [16].

ORB can match signature images using low-power devices without the use of GPU acceleration. Therefore, it performs as well as SIFT and better than SURF with almost two orders of magnitude [17]. Image patches are sets of binary intensity tests that BRIEF descriptor [20] makes a bit-string description of these patches. Afterward, Gaussian distribution is performed around the center of the image's patch. In ORB, in order to use BRIEF descriptor on the orientation of key-points, an efficient method is performed to steer BRIEF regarding the orientation. For n binary tests, a feature set at (xi, yi) can be represented as 2×n a matrix as (6):

While SURF and SIFT algorithms are based on histograms of gradients, ORB is a binary descriptor that is based on image intensity comparisons to encode patch's information as a binary string; which makes it relatively faster. ORB can match two images in a single instruction by using the hamming distance only.

III. METHOD

The proposed ORB algorithm process is divided into several steps as depicted in Figure 1, more details about these steps are given in the following sub-sections. We start with acquiring offline signatures from our users, then we proceed to apply several pre-processing steps to normalize the system inputs and remove any unnecessary data features. Afterwards, we apply several features extraction techniques, (SIFT, SURF, and ORB) to extract the signatures features, and then we perform the features matching comparison toevaluate the proposed system performance.

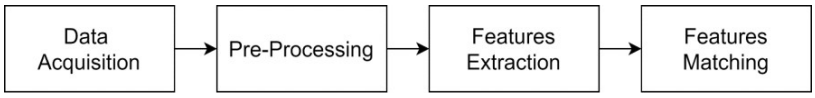


Figure 1. Proposed system flow chart

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

3.1 Feature Extraction

In the feature extraction stage, the system applies ORB, SURF, and SIFT algorithms to extract the signatures features and saves them in two byte-arrays: the serialized image as a byte array, and its associated features also as a byte array. These arrays are then stored into a custom database to facilitate storing and retrieving them. Figure 5 depicts the steps of the feature extraction stage

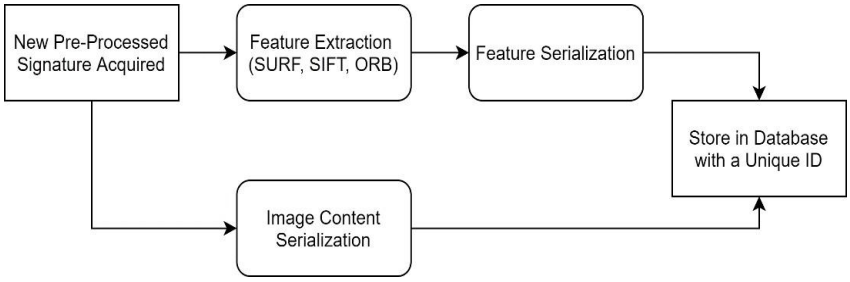


Figure. The process of features extraction process using SIFT, SURF, and ORB

3.2 Proposed Algorithm (pseudo code)

Step1: collect the genuine & forged signatures images
Step2: apply pre-processing operation to remove the noise in the images
Step3: split the data set as training & testing data set
Step4: apply ORB feature extraction algorithm to extract the features
Step5: build CNN model architecture & train the model with the training data
Step6: validate the model
Step7: input genuine & supposed to be forged signature image from the user
Step8: apply pre-processing operation
Step9: extract features and pass it to the model
Step10: Model will predict if the signature is genuine or forged.
Step11: stop

3.3 DAG-CNN

A. Architecture

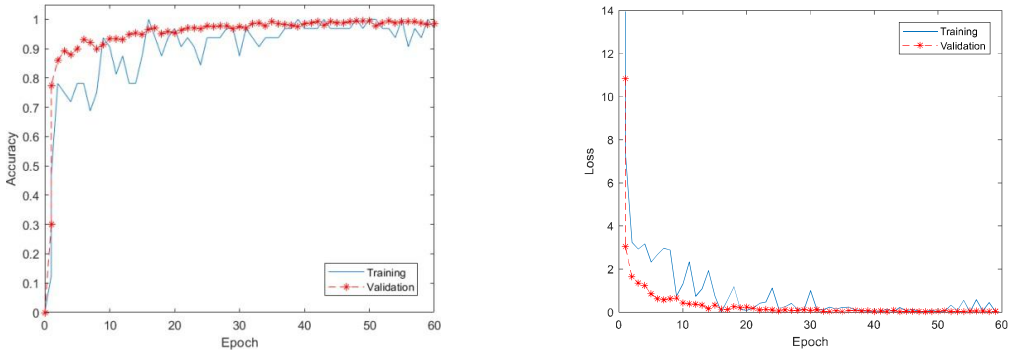
To perform the verification of each signature, a DAG-CNN with its own bilinear architecture is used, in such a way that a very deep CNN can be achieved but divided into two sections focused on learning different characteristics. As shown in Table 1, each division consists of a different configuration, wherepath 1, by the sizes of its filters, is focused on learning details of the signatures, for this reason, it also consists of a large number of filters in all layers to be able to learn enough features. On the other hand, path 2has fewer filters with larger sizes in each layer, which allows this division to learn more general characteristics of the signatures, such as its dimensions and sections by larger details. Due to the original size of the images of the dataset, the computational cost would be very large, for this reason, when entering the network, it was reduced to a size where the characteristics of the signature were not lost, being resized to 30% of its size, remaining with dimensions of 171×276 px at the entrance of the network. In general, the architecture must be able to classify 7 different categories, of which are: a category of genuine signatures and one of forgeries for each user, having 6 categories in total, plus an additional one of signatures not belonging to the database.

B. Training Process

For the execution of the training, batches of 32 signatures were used per iteration, in other words, to complete an epoch, 89 iterations were needed, since if a larger amount of data were used per batch,the computational cost would be very high. Due to this, we chose to use a learning rate of 10^{-4} for the network to be able to learn the characteristics more generally and prevent the network from falling into astate of overfitting, in addition, the order of the signatures changes randomly in each epoch, in order not to bias the network to learn data sorted in a specific order. With these parameters, the DAG-CNN was trained for 60 epochs, showing the behavior of Figure 3, obtaining an accuracy of 100%

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

with the training dataset and99.43% with the validation, which is a very reliable accuracy in terms of identification of forgeries, i.e. of the350 validation signatures, it only had misclassification in 2 signatures, one given as genuine when it was forgery for user 1, and one recognized as forgery when it was genuine for user 3, as shown in the confusion matrix of Figure 4. Likewise, all the signatures of the 115 users not belonging to the verifiable ones were correctly classified.



(a) (b)

Figure 3. Training and validation behavior of the DAG-CNN a) Accuracy, b) Loss

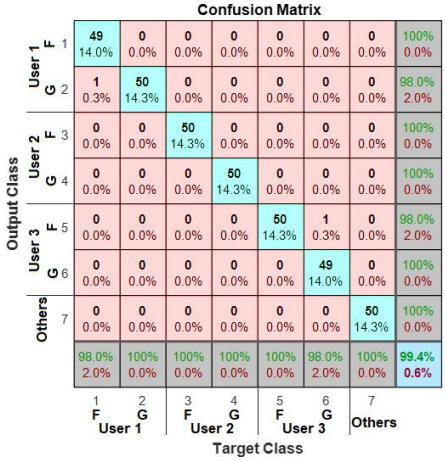


Figure 4. Confusion Matrix of the last epoch trained of the network using the Validation DatasetF are the Forgery and G are the Genuine Signatures.

IV. RESULTS AND DISCUSSIONS

To test the network, an additional database is built, called Test dataset, which consists of 23 genuine signatures and 23 forgeries of each user to which a data augmentation was performed with changes of random position, i.e., in the same way to the augmentation done in section 2, obtaining a total of 345 signatures per category (2415 in total of the database). For the category of others, the 115 users of the UT Sig Dataset were used, but choosing 3 different samples to those already used in the previous databases.

This database is entered to the trained network to perform the test, in which 99.3% of overall accuracy is obtained, as shown in Figure 5, where it can be seen that the genuine signatures of all users were recognized with percentages greater than 98%. However, some forgeries managed to pass as genuine, but their correct classification was higher than 98.5% in all cases, demonstrating the effectiveness of the network in verifying the users registered, even having forgeries with great similarity (those elaborated by the same user with slight changes). To understand the internal operation of the network, the strongest activations of each convolution are extracted after the application of ReLU (Rectified Linear Unit, whose function is to eliminate the negative values of the output volume), in such a way that it can be observed how the network behaves with the image through it.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

In Figure 6, it can see the different activations of the two divisions of the network.Path 1 shows activations mainly focused on the internal details of the signature, contour and shape in the last two convolutions, while path 2 focuses largely on the general form of the signature, empty sections withinthe contour and the boundaries of possible signature location (convolution 2).

In a more detailed way to visualize the activations over the signature, heat maps overlapped on the original image are used. In those maps, the strongest activations are represented by red color and the weakest ones by dark blue. For this, the best seven activations obtained in the first layer of each path are shown in Figure 7. Within the path 1, the filters were able to learn finer details of the signature, not only related to the shape but of different key parts, such as the first letter of the signature, as can be seen in the last activation of the bottom, and specific traces that can differentiate a forgery from a genuine, such as the third activation at the top and the bottom. On the other hand, in path 2, as expected, generalize the "appearance" of the signature, activating independently the lower, center and top part of the signature, also it activates empty regions as in the second convolution layer, as shown in Figure 6. The above shows that the proposed architecture obtained the behavior that was desired, described in section 3.

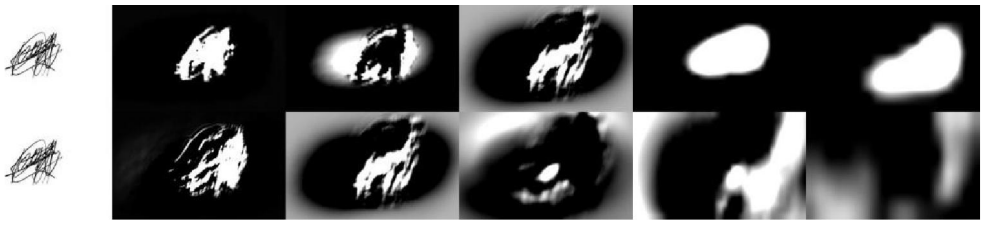


Figure 6. Activations of each convolution. At the top, the path 1; at the bottom, the path 2.

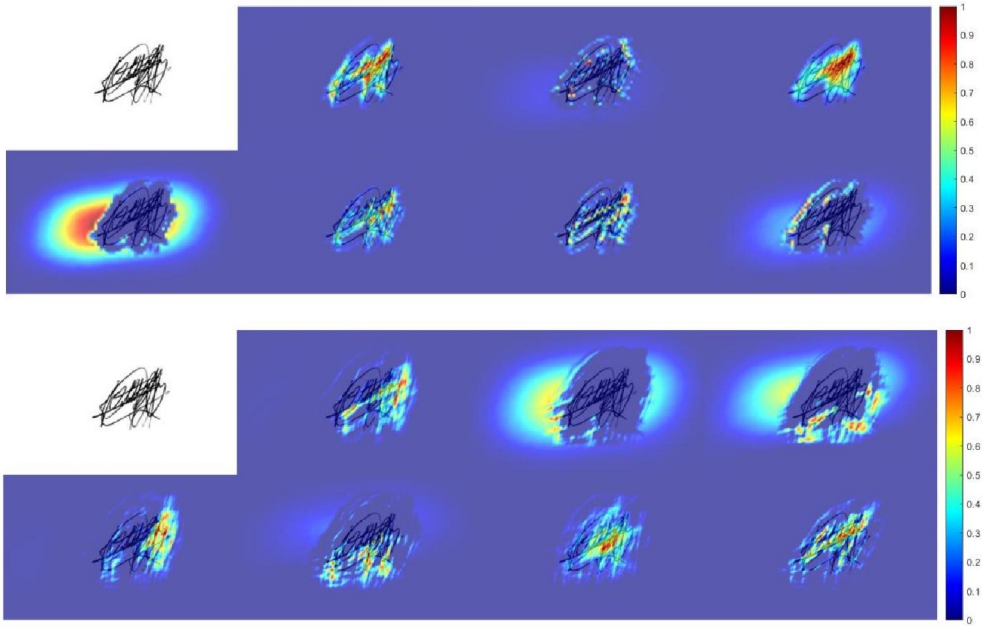


Figure 7. Activations of the first convolution layer using overlapped heat maps, a) Path 1, b) Path 2

V. CONCLUSION

This paper presented the application of a DAG-CNN focused on the offline signature verification, as an advance in the use of Deep Learning pattern recognition techniques in this type of task, giving a network capable of classifying and authenticate at the same time, showing the performance of this type of network. Using a DAG-CNN allowed it to learn different characteristics of the signatures, as shown in Figure 6 and Figure 7, where, depending on the type of configuration that is set in each branch, the network focused on certain sections and details of the signature in order to be

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 5, June 2022
Impact Factor: 6.252

able to differentiate, first, whether it belonged or not to registered users, and then verify the genuineness of it. The results obtained and the analysis of the behavior of the trained network, where an accuracy greater than 99% was obtained, allow us to verify the efficiency of the CNN, in this case of a DAG-CNN, for the recognition of the genuineness of signatures offline. On the other hand, to achieve even a higher percentage, it can be added online data collection methods and analysis of these, making real-time verification much more reliable, i.e., at the time of signing, however, it requires the use of specialized devices to perform said data acquisition, which can be applied in future work.

REFERENCES

[1]. D. Impedovo and G. Pirlo, "Automatic signature verification: The state of the art," EEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 38, no 5, pp. 609-635, 2008. DOI: 10.1109/TSMCC.2008.923866

[2]. A. Kholmatov and B. Yanikoglu, "Identity authentication using improved online signature verification method," Pattern recognition letters, vol. 26, no 15, pp. 2400-2408, 2005. DOI: 10.1016/j.patrec.2005.04.017

[3]. J. Bromley, I. Guyon, Y. LeCun, E. Säckinger and R. Shah, "Signature verification using a "siamese" time delay neural network," Advances in Neural Information Processing Systems, pp. 737-744, 1994.

[4]. L. G. Hafemann, R. Sabourin and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," Pattern Recognition, vol. 70, pp. 163-176, 2017. DOI: 10.1016/j.patcog.2017.05.012

[5]. S. N. Srihari, A. Xu and M. K. Kalera, "Learning strategies and classification methods for off-line signature verification," in Frontiers in handwriting recognition, 2004. IWFHR-9 2004. Ninth international workshop on, pp. 161-166, 2004. DOI: 10.1109/IWFHR.2004.61

[6]. A. Soleimani, B. N. Araabi and K. Fouladi, "Deep multitask metric learning for offline signature verification,"Pattern Recognition Letters, vol. 80, pp. 84-90, 2016. DOI: 10.1016/j.patrec.2016.05.023

[7]. G. S. Eskander, R. Sabourin and E. Granger, "Hybrid writer-independent–writer-dependent offline signature verification system," IET biometrics, vol. 2, no 4, pp. 169-181, 2013. DOI: 10.1049/iet-bmt.2013.002

[8]. H. C. Fu, H. Y. Chang, Y. Y. Xu, and H. T. Pao, "User adaptive handwriting recognition by self-growing probabilistic decision-based neural networks," IEEE Transactions on neural networks, vol. 11, no 6, pp. 1373-1384, 2000. DOI: 10.1109/72.883451

[9]. L. Xing and Y. Qiao, "Deepwriter: A multi-stream deep CNN for text-independent writer identification," in Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on, 2016, pp. 584-589. DOI: 10.1109/ICFHR.2016.0112

[10]. A. McCabe, J. Trevathan and W. Read, "Neural network-based handwritten signature verification," Journal of computers, vol. 3, pp. 9-22, 2008. DOI: 10.4304/jcp.3.8.9-22

[11]. Y. LeCun, Y. Bengio and G. Hinton, "Deep learning," Nature, vol. 521, no 7553, pp. 436-444, 2015. DOI: 10.1038/nature14539

[12]. M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in European conference on computer vision, Springer, Cham, pp. 818-833, 2014. DOI: 10.1007/978-3-319-10590-1_53.