International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 06 | June - 2022 Impact Factor: 7.185 ISSN: 2582-3930

VERIFICATION AND VALIDATION OF HANDWRITTEN SIGNATURE

Shyam Sundar S
(Student)
Jeppiaar Engineering College

Sri Rithik Guru R
(Student)
Jeppiaar Engineering College

Vignesh S
(Student)
Jeppiaar Engineering College

Dr. P. Jesu Jayarin
Professor
Jeppiaar Engineering College

Abstract --- As we all know the most important form of biometric authorization are the handwritten signatures which is used globally among people to verify and legalize a document. They are the most vulnerable form of metric because they can be easily forged and leading to identity theft, Real estate theft, information theft etc. Which in turn paved way for the need to identify the genuineness of a signature resulting in the invention of signature recognition systems. The problems faced by these signature recognition systems is the intra-class variation (i.e.) an individual's original sign may not be the same each time since there are a few external and internal factors that affect it, such as, the pen used , the surface on which the sign is signed on , etc. Hence we consider two factors of signature validation, the speed of the writer may affect the sign, hence the sign is identified as an image and then the image is classified by the neural network (Siamese neural network). This will increase the accuracy of our signature validation drastically.

Keywords--- Bio-metric Authorization, Forged, Genuineness, Siamese.

I.INTRODUCTION

It is well known that doesn't exist two signatures equals of the same person. Successive signatures by the same person will differ, may also differ in scale and orientation. Some researchers suggests that a signature has at least three attributes, form, movement and variation, and since signatures are produced by moving a pen on paper, movement perhaps is the most important part of a signature. We also come across Forging signatures and stealing various personnel belongings like identity, property etc

Thus this arise the need to verify the signature and the need to develop validation systems. In this project we have to compare two identical images and find the genuineness of the signature from the image. In order to do so we are employing image comparison algorithms to process and compare images. The comparison is done by taking Colorpixels into account. If the color of each pixel of both images coincides, TestComplete considers the two images to be identical. The comparison engine has a lot of parameters.

Pixel Tolerance - The pixel tolerance is an arbitrary number of how many pixels of motion we choose to tolerate in our image. A pixel tolerance of 7 pixels means that the stars in the frame will move up to 7 pixels of distance for the recommended shutter speed.

Color Tolerance - The color difference is represented as an integer value within the range 0…255 that specifies an acceptable difference for each color component (red, green, and blue) of the compared pixels

II. MODEL DEPENDENT FROM THE WRITER

A. PRE-PROCESSING

At First images are pre-processed using the OpenCV library and the following are the steps: 1. Resizing: Bilinear interpolation is used to resize the photos .2. Gray Scale Conversion: colored images are converted to grayscale. 3. Image thresholding and binary inversion: this method determines the best threshold value for inverting an image.4. Normalization: the images are normalized.

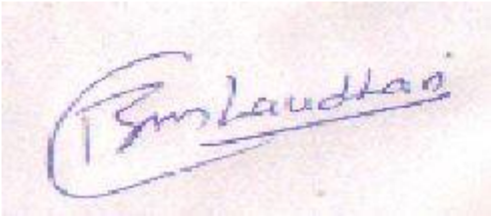


Figure 1 Original Signature

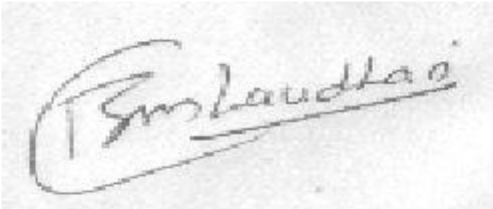


Figure 2 Grayscale Image

B. FEATURE EXTRACTION

Feature extraction is a part of the dimensionality reduction process, in which, an initial set of the raw data is divided and reduced to more manageable groups. So when you want to process it will be easier. The most important characteristic of these large data sets is that they have a large number of variables. These variables require a lot of computing resources to process. So Feature extraction helps to get the best feature from those big data sets by selecting and combining variables into features, thus, effectively reducing the amount of data. These features are easy to process, but still able to describe the actual data set with accuracy and originality.

> Ratio

It's the ratio of the size of a signature image to the size of a signature within a bounding box. The number of pixels that make up a signature's area.

> Centroid

In a binary image, you must calculate the average coordinate to identify the centroid of a group of pixels. In the case of a grayscale image, the weighted average location can be calculated using the grey values of the pixels.

> Eccentricity

Eccentricity is defined as the shortest path length from one vertex v to any other vertex w in a connected network.

It converts the graph's connection structure into a set of values when computed for each vertex v. The neighbourhood graph and the supplied metric are used to define a linked region of a digital image.

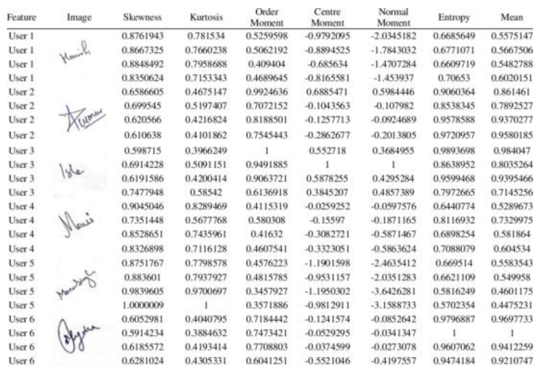
> Solidity

When compared to its convex hull, solidity is a fraction of the region's area.

The solidity of your territory is measured as a percentage of its total area.

> Kurtosis

It's the standardised data's average or expected value raised to the fourth power.



Feature	Image	Skewness	Kurtosis	Order Moment	Centre Moment	Normal Moment	Entropy	Mean
User 1		0.8761943	0.781534	0.5259598	-0.9792095	-2.0345182	0.6685649	0.5575147
User 1		0.8667325	0.7660238	0.5062192	-0.8894525	-1.7843032	0.6771071	0.5667506
User 1		0.8848492	0.7958688	0.409404	-0.685634	-1.4707284	0.6609719	0.5482788
User 1		0.8350624	0.7153343	0.4689645	-0.8165581	-1.453937	0.70653	0.6020151
User 2		0.6586605	0.4675147	0.9924636	0.6885471	0.5984446	0.9060364	0.861461
User 2		0.699545	0.5197407	0.7072152	-0.1043563	-0.107982	0.8538345	0.7892527
User 2		0.620566	0.4216824	0.8188501	-0.1257713	-0.0924689	0.9578588	0.9370277
User 2		0.610638	0.4101862	0.7545443	-0.2862677	-0.2013805	0.9720957	0.9580185
User 3		0.598715	0.3966249	1	0.552718	0.3684955	0.9893698	0.984047
User 3		0.6914228	0.5091151	0.9491885	1	1	0.8638952	0.8035264
User 3		0.6191586	0.4200414	0.9063721	0.5878255	0.4295284	0.9599468	0.9395466
User 3		0.7477948	0.58542	0.6136918	0.3845207	0.4857389	0.7972665	0.7145256
User 4		0.9045046	0.8289469	0.4115319	-0.0259252	-0.0597576	0.6440774	0.5289673
User 4		0.7351448	0.5677768	0.580308	-0.15597	-0.1871165	0.8116932	0.7329975
User 4		0.8528651	0.7435961	0.41632	-0.3082721	-0.5871467	0.6898254	0.581864
User 4		0.8326898	0.7116128	0.4607541	-0.3323051	-0.5863624	0.7088079	0.604534
User 5		0.8751767	0.7798578	0.4576223	-1.1901598	-2.4635412	0.669514	0.5583543
User 5		0.883601	0.7937927	0.4815785	-0.9531157	-2.0351283	0.6621109	0.549958
User 5		0.9839605	0.9700697	0.3457927	-1.1950302	-3.6426281	0.5816249	0.4601175
User 5		1.0000009	1	0.3571886	-0.9812911	-3.1588733	0.5702354	0.4475231
User 6		0.6052981	0.4040795	0.7184442	-0.1241574	-0.0852642	0.9796887	0.9697733
User 6		0.5914234	0.3884632	0.7473421	-0.0529295	-0.0341347	1	1
User 6		0.6185572	0.4193414	0.7708803	-0.0374599	-0.0273078	0.9607062	0.9412259
User 6		0.6281024	0.4305331	0.6041251	-0.5521046	-0.4197557	0.9474184	0.9210747

Figure 3 Extracted Features

III. INDEPENDENT MODEL FROM THE WRITER

A. DEVELOPING THE MODEL

When compared to existing systems, this model was constructed using Tensor Flow with minor changes to the model, resulting in higher accuracy. The model accepts input, and both images are supplied to the model, which splits them into left and right images and are trained individually to extract the features using the following layers:

1. A Model is created by grouping a linear stack of layers into a Sequential. On this model, Sequential provides training and inference features.

2. Convolutional neural networks use a filter to produce a feature map, which summarises the existence of observed features in an input.

3. By obtaining the greatest value over an input window, the input is down sampled along its spatial dimensions height and breadth.

4. The Dropout layer, which helps minimise overfitting, sets input units to 0 at random with a rate frequency at each step during training time.

5. A flatten operation on a tensor reshapes the tensor into a shape equal to the number of elements in the tensor, excluding the batch dimension.

6. Rectified linear unit (relu) activation is used to connect the Dense Layer to the preceding layer.

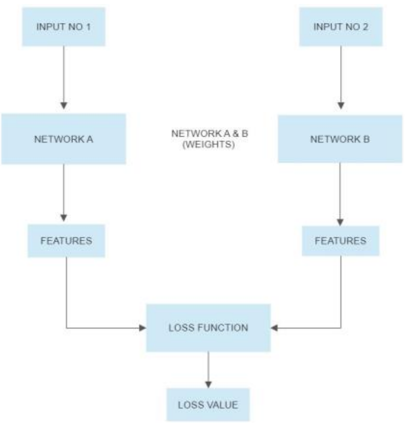


Figure 4 Siamese Model

B. TRAINING THE MODEL

The Loss function used in this model is contrastive Loss function. Its goal is to reduce dimensionality by learning an invariant mapping that generates high-to-low-dimensional space maps that transfer similar input vectors to nearby locations on the output manifold and dissimilar vectors to distant place.

Contrastive loss calculates the distance between a positive example and another example of the same class and compares it to the distance between negative instances. Positive samples encoded to comparable representations and negative examples encoded to dissimilar representations result in a modest loss. For two inputs X1 and X2, contrastive loss function is calculated as such:

Softmax Loss

$$\mathcal{L}_S = -\sum_{i=1}^{m}\log\frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}}$$

Contrastive Loss

$$(1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{max(0, m-D_W)\}^2$$

Center Loss

$$\mathcal{L}_C = \frac{1}{2}\sum_{i=1}^{m}\|x_i - c_{y_i}\|_2^2$$

Figure 5 Contrastive Loss Function

RMS Prop is utilised as the model's optimizer. RMS prop's main goal is to:

- Maintain a moving average of gradients squared.

- Divide the gradient by the average's root. Simple momentum is used in this RMSprop implementation.

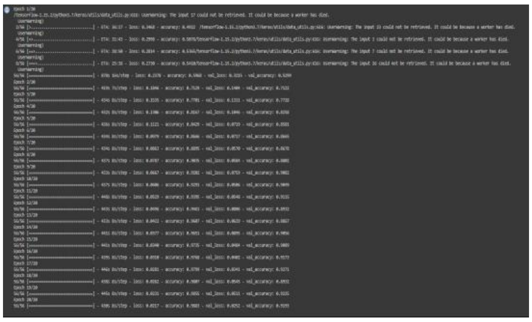


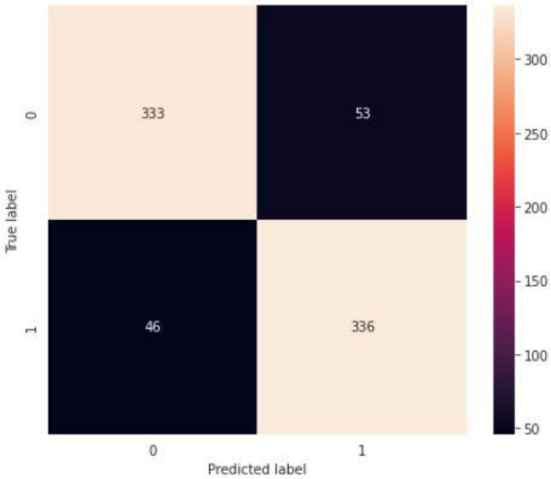
Figure 6 Training the datasets

C. EVALUATING PERFORMANCE

Confusion Matrix---A confusion matrix is a table that is often used to describe the performance of a classification model on a

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 06 | June - 2022 Impact Factor: 7.185 ISSN: 2582-3930

set of test data for which the true values are known. The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.




```
Accuracy 0.87109375

Classification Report
              precision    recall  f1-score   support

           0       0.88      0.86      0.87       386
           1       0.86      0.88      0.87       382

    accuracy                           0.87       768
   macro avg       0.87      0.87      0.87       768
weighted avg       0.87      0.87      0.87       768
```

Figure 7 Confusion Matrix

IV. MODULE DESCRIPTION

A.CAMERA MODULE

In this module the user or the client can upload the image. After the upload the image is then passed down to processing module. In which the image is processed as per requirements.

B.PROCESSING MODULE

In this module the image uploaded is processed as per the requirements, such as grayscale, image resize and so. After this module the processed image is 34 then passed to the comparison module. These different processing of the image is explained below.

Image Scaling --- In computer graphics and digital imaging, image scaling refers to the resizing of a digital image. In video technology, the magnification of digital material is known as upscaling or resolution enhancement.

When scaling a vector graphic image, the graphic primitives that make up the image can be scaled using geometric transformations, with no loss of image quality.

Grayscale --- In digital photography, computer-generated imagery, and colorimetry, a grayscale image is one in which the value of each pixel is a single sample representing only an amount of light; that is, it carries only intensity information. Grayscale images, a kind of black-and-white or gray monochrome, are composed exclusively of shades of gray. The contrast ranges from black at the weakest intensity to white at the strongest. Grayscale images are distinct from one-bit bi-tonal black-and-white images, which, in the context of computer imaging, are images with only two colors: black and white (also called bilevel or binary images).

C.COMPARISON MODULE

The comparison module will compare the uploaded image which is processed to analyze based on certain factors and verify it

D. VERIFICATION MODULE

In this module the result of the validation after the comparison module is verified and displayed to the user.

Plot Accuracy

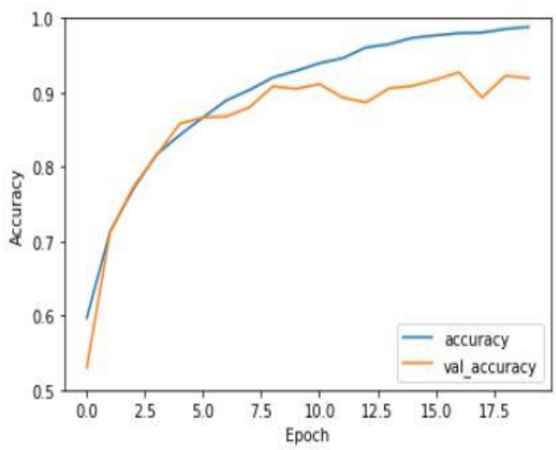


Figure 8 Training Curve

V.RESULT ANALYSIS

In machine learning, a learning curve (or training curve) plots the optimal value of a model's loss function for a training set against this loss function evaluated on a validation data set with same parameters as produced the optimal function. It is a tool to find out how much a machine model benefits from adding more training data and whether the estimator suffers more from a variance error or a bias error. If both the validation score and the training score converge to a value that is too low with increasing size of the training set, it will not benefit much from more training data.

A.RESULT

Thus after the verification we can now upload the pair and continue waiting for the result.

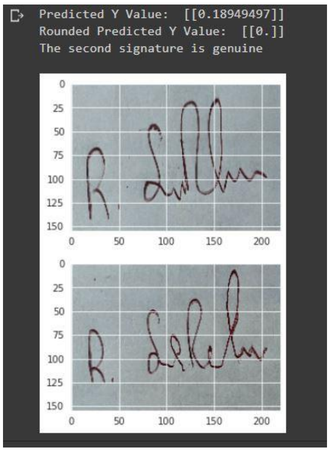


Figure 9 Result Genuine

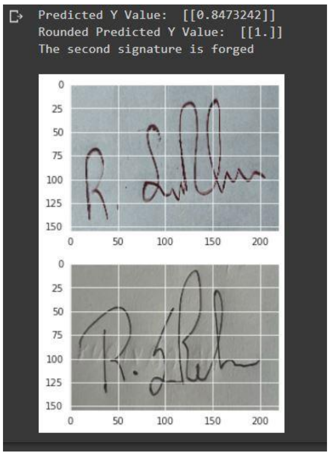


Figure 10 Result Forged

VI.CONCLUSION AND FUTURE SCOPE

A.CONCLUSION

Since we all know that Signature is most important form of all biometric authorizations. Thus there are chances of forging signature of an individual this arises the theft of identity, property and so. This in turn paved way for the need to develop the verification systems and validation systems of signatures. The main goal of the proposed system is to validate if the given signature is genuine or forged. The system works with the writer-independent model to validate. It can be concluded that the proposed system of the writer-independent model provides better accuracy and precision compared to the existing system. Each verification method has advantages and disadvantages, it is very difficult to determine the best method, and every method uses different parameters. However, if viewed from the ease of implementation and performance of verification, artificial neural networks (Convolutional Siamese Neural Network) is the right choice this time. Over the last decade, researchers have proposed a large variety of methods for Signature Verification. While distinguishing genuine signatures and skilled forgeries remains a challenging task, error rates have dropped significantly in the last few years, mostly due to advancements in Deep Learning applied to the task. In the way of plying this using Artificial Neural Network (ANN) we can accomplish various prediction systems. In which our model is considered to be one of the prediction systems and hence helps us in verifying the signatures and distinguish them from Genuine and forged.

B.FUTURE SCOPE

The adaptability and the main aspect or goal of all prediction and verification systems is to have a good accuracy. Thus inspired by our model results the future scope is always to focus on the accuracy and the adaptability of the datasets used in our model. The availability of more data the accuracy can be more precise compared to the model that we have now.

Yet that is not all as we all know that signature verification is not available for everyone, since it is exclusive or either used only by certain organizations, bank, register office etc. We could possibly upload this model as a bank end and develop it into a working, public website for the freedom for anyone to be able to us it.

VII.REFERENCES

1. Ismail, M.A.; Gad, Samia (Oct 2000). "Off-line arabic signature recognition and verification". Pattern Recognition. 33 (10): 1727–1740.

2. "Explainer: Signature Recognition | Biometric Update". www.biometricupdate.com. 2016-01-11. Retrieved 2021-04-03.

3. Houmani, Nesmaa; A. Mayoue; S. Garcia-Salicetti; B. Dorizzi; M.I. Khalil; M. Mostafa; H. Abbas; Z.T. Kardkovàcs; D. Muramatsu; B. Yanikoglu; A. Kholmatov; M. Martinez-Diaz; J. Fierrez; J. Ortega-Garcia; J. Roure Alcobé; J. Fabregas; M. Faundez-Zanuy; J. M. Pascual-Gaspar; V. Cardeñoso-Payo; C. Vivaracho-Pascual (March 2012). "BioSecure signature evaluation campaign (BSEC'2009): Evaluating online signature algorithms depending on the quality of signatures". Pattern Recognition. 45 (3): 993–1003.

4. Faundez-Zanuy, Marcos (2007). "On-line signature recognition based on VQ-DTW". Pattern Recognition. 40 (3): 981–992.

5. Chapran, J. (2006). "Biometric Writer Identification: Feature Analysis and Classification". International Journal of Pattern Recognition & Artificial Intelligence. 20 (4): 483–503.

6. Yeung, D. H.; Xiong, Y.; George, S.; Kashi, R.; Matsumoto, T.; Rigoll, G. (2004). "SVC2004: First international signature verification competition". Lecture Notes in Computer Science. LNCS-3072. 3072: 16– 22.

7. Ortega-Garcia, Javier; J. Fierrez; D. Simon; J. Gonzalez; M. FaúndezZanuy; V. Espinosa; A. bimodal 50 biometric database". IEE Proceedings - Vision, Image, and Signal Processing.

8. Chicco, Davide (2020), "Siamese neural networks: an overview", Artificial Neural Networks, Methods in Molecular Biology, vol. 2190 (3rd ed.), New York City, New York, USA: Springer Protocols, Humana Press, pp. 73–94.

9. Bromley, Jane; Guyon, Isabelle; LeCun, Yann; Säckinger, Eduard; Shah, Roopak (1994). "Signature verification using a "Siamese" time delay neural network" (PDF). Advances in Neural Information Processing Systems 6: 737–744.

10. Chopra, S.; Hadsell, R.; LeCun, Y. (June 2005). "Learning a similarity metric discriminatively, with application to face verification". 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). 1: 539–546.

11. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. (June 2014). "DeepFace: Closing the Gap to Human-Level Performance in Face Verification". 2014 IEEE Conference on Computer Vision and Pattern Recognition: 1701– 1708.

12. Chatterjee, Moitreya; Luo, Yunan. "Similarity Learning with (or without) Convolutional Neural Network" (PDF). Retrieved 2018-12-07.

13. Chandra, M.P. (1936). "On the generalized distance in statistics" (PDF). Proceedings of the National Institute of Sciences of India. 1. 2: 49–55.

14. Zhu, Zheng; Wang, Qiang; Li, Bo; Wu, Wei; Yan, Junjie; Hu, Weiming (2018). "Distractor-aware Siamese Networks for Visual Object Tracking". arXiv:1808.06048 [cs.CV].