

Smartphone enabled Counterfeit Note Detection using Siamese Network

Dhanush C., Adith Kumar B. A., Ajay Umakanth, Ajay Deshpande, Bhavanishankar K.

Abstract: Counterfeit note has a disastrous impact on a country's economy. The circulation of such fake notes not only diminishes the value of genuine note but also results in inflation. The feasible solution to this burning issue is to create awareness about the counterfeit notes among public and to equip them with a technology to detect fake notes on their own. Though there exist numerous research articles on detection of fake notes, they are not handy. The reason for this could be the unavailability or unaffordability in acquiring the equipment for the same. This paper proposes an approach whose implementation can easily be deployed on a smart phone and hence anyone with access to them can use the application to detect the fake notes. The proposed approach consists of the processing phases including image procurement, pre-processing, data augmentation, feature extraction and classification. ₹500 notes are considered for experimentation analysis. Out of 17 distinctive features, 3 such from the obverse side are considered to evaluate the genuineness of the note. Siamese neural network is employed to build a model for effective classification of the notes. The performance of the proposed approach is evaluated at 85% with respect to accuracy.

Keywords: Contrastive loss, Counterfeit Note, Siamese Network, Smartphone.

I. INTRODUCTION

Currency plays a cardinal role in a country's economy. Banknotes are the primary form of currency which is pervasive in our Indian society. Although our country is progressing in terms of digitalization of money, there exists more than 20 Lakh crores worth of notes in circulation in India [1]. The focus on notes has reached the pinnacle after the nation was bewildered by the storm of demonetization. On 8 Nov 2016, the ₹500 and ₹1000 notes were rendered illegitimate and were replaced by the new ₹500 and ₹2000 notes. Eventually, new notes of ₹10, ₹20, ₹50, ₹100 and ₹200 denominations came into existence. As the new notes were circulated, a major problem accompanied it. The counterfeits of these new notes in circulation are disseminated in our society. The share of ₹500 and ₹2000 notes in this problem is about 15 lakh crore and the amount of counterfeit ₹500 notes has increased by 121% and ₹2000 by 20% over the last year [2].

Revised Manuscript Received on December 15, 2019.

Dhanush C, CSE Department, RNSIT, Bangalore, India. Email: actuallydc@gmail.com

Adith Kumar B A, CSE Department, RNSIT, Bangalore, India. Email: adithkumar.ba@gmail.com

Ajay Umakanth, CSE Department, RNSIT, Bangalore, India. Email: ajayunp@gmail.com

Ajay Deshpande, CSE Department, RNSIT, Bangalore, India. Email: ajay.d53@gmail.com

Dr. Bhavanishankar K, CSE Department, Professor RNSIT, Bangalore, India. Email: bsharsh@gmail.com

Counterfeiting doesn't have to be perfect; it must only be good enough to deceive the eyes of a common person. This is the reason for their inconspicuous existence. There are many measures taken by the government to validate notes, but there is no such option that enables a person to authenticate notes at any place at any time. Currently, it is required for us to go to a bank or an ATM for verification. This is not a feasible method at all instances and delayed detection will still lead to a loss for the person possessing the counterfeit note. This situation demanded a better, faster, and more portable solution with greater accessibility.

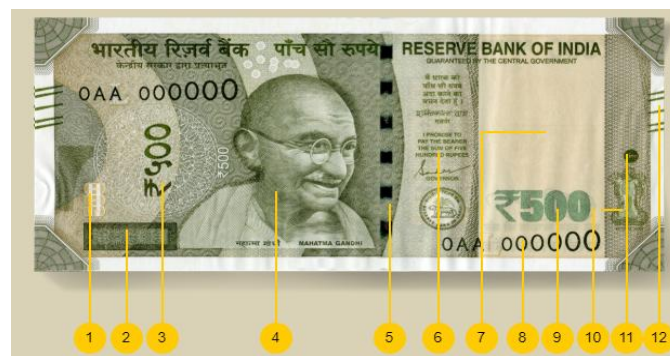


Fig 1.1 – RBI authorized features of the 500 note

With the latest advances in technology and the proliferation of high-resolution cameras in smartphones, more can be seen through images captured by our phones compared to naked eyes. This feature can be greatly advantageous in distinguishing the counterfeit notes from the genuine ones. The new notes have been designed with some distinctive features which help to identify a genuine note. Some features of the ₹500 note are shown in figure 1.1 and listed in Table I [3].

These features can be extracted from an image captured by a smartphone camera. The extracted features can be later compared with the dataset of features from the genuine note with Deep learning techniques to identify whether they match with enough consistency to be classified as genuine. Comparing individual features gives a result with far greater accuracy than comparing the whole Note with the dataset as the background information which might skew the efficiency of the model is removed.

Table I - Features of the note

1	See through register in denomination numeral	7	Portrait and electrotype watermark
2	The Latent image	8	Number panel with increasing font size
3	Denomination in Devanagari	9	Denomination in color-changing ink
4	Orientation and position of Mahatma Gandhi	10	Ashoka pillar emblem
5	Windowed security thread that changes color	11	Circle with 500 on raised print
6	Governor signature and RBI emblem	12	5 bleed lines on the side

Obtaining a fake note is next to impossible as these notes are expunged as soon as they are identified as fake. Hence any Neural Network may not give an efficient solution due to the absence of an extensive dataset. A Siamese Network is thus used which compares images without the requirement of a large dataset. To maximize accuracy, we extract important features and then pass them through the learning model instead of the whole picture. The implementation is explained in the later section. Out of all the features mentioned above, we selected the three most distinguishable features 2, 9 and 11 whose diminutive details can be captured through a smartphone camera. The paper is further organized as follows. Section 2 provides a literature review on various methods for detecting a fake note along with implementation details. The proposed Solution is explained with a detailed discussion on implementation in section 3. Section 4 briefs about the experimental results and observations. Section 5 concludes the paper with a summary of the observations and limitations along with a note on the future scope of the work.

II. LITERATURE REVIEW

Generative Adversarial Network was used by [4] which included two modules namely, Generative network and Discriminator network. The parameters used for augmentation are sample-wise center, feature-wise standard normalization, width shift range, shear range, zoom range, ZCA whitening, height shift range, ZCA Epsilon, rotation range, horizontal rescaling, channel shift, and vertical rescaling. The proposed approach yielded an accuracy of 80% with an overhead of augmentation. Images taken under UV light was used by [5] which were later converted into grayscale and segmented to reduce the dimensionality. The security features of the note such as Serial Number, Latent Image, Watermark, Identification mark, Security Thread were observed for classification. Soiled currency notes could not be recognized by this method which involved usage of expensive hardware like UV light emitter and digital camera. In [6], an extensive dataset of 200 images had been collected and perspective transformation and other common pre-processing techniques were applied. A VGGNet neural network with 16 layers was implemented using the Keras library with Theano as backend. The usage of most common and simple Adam optimizer and ReLU activation the proposed approach reached its apex at 55% accuracy. To secure a dataset with a resolution of 1200dpi expensive

scanners and many cameras were employed by [7]. The approach narrowly focuses on just the intaglio printing which are manually cropped. Many models are used for feature extraction and matching namely, feature similarity index model, Shift Invariant Wavelet Packet Transform and Scale Invariant Feature Transform, which adversely affects the run-time. At last the author uses a support vector machine for classification. The paper [8] makes use of multispectral images – images under IR light and visible light, which helps in recognizing age, soiling and patterns. The image is segmented and Bhattacharya difference method is used to select required segments. Co-relation and means are compared between genuine and fake notes by two models namely Fisher linear discriminant and Gaussian ML classifier, to produce result. Expensive IR emitter serves only to determine the age and soiling which are insignificant. The security features present most commonly on currency are detailed by [9], which are – See through Register, Intaglio printing, Latent image, Micro printing, Security Threads, Security links, Watermark, Paper Material, Dimensions, Hologram, Ultraviolet and infrared tests and Serial Numbers. The approach provides only a strategic solution or preventive measures but no failsafe methods to employ.

The image obtained by the scanner is made by [10] to undergo conversion to grayscale, removal of noise using median filter and enhancement by using imadjust function in MATLAB. Bit-plane slicing method is used to slice the image and select required planes to reduce dimensionality on which canny-edge detection is performed. Security features of the Kuwaiti note such as - Raised print of bank note value, falcon watermark, circles and wave shaped color changes are used for classification of note. In this approach [11] the input images are sharpened and then segmented based on similarity in intensity, color and texture. The texture is one main feature which is not considered by most other approaches. Clustering operation is done using the k-means algorithm which clusters the input images and then the Support Vector Machine classifies the currency as genuine or not. Input images are acquired by [12] under UV light and converted grayscale. A microcontroller is used to help transport the note through a conveyor belt for different units. One such unit is the fake not detection unit which uses MATLAB tools and compares the input with reference image.

This approach requires numerous hardware components and a micro controller to maintain the system. The dataset prepared using industrial camera for print inspection by the University of California Irvine machine learning repository has been used by [13]. The dataset is split and fitted to a Artificial Neural Network and which is also capable of detecting variance, skewness, curtosis, entropy and class. The model is fit for the images taken from this expensive high-grade camera and therefore will not work without that equipment. A different method of image segmentation has been made use by [14]. Once the image is taken from the digital camera the edge detection and removal of background noise is performed.

The image is then segmented based on Discontinuity and Similarity. Single features are extracted and compared with a matching algorithm. The approach [15] involves pre-processing using salt and pepper removal of noises on the dataset created using scanners. Canny-edge detection, grayscale conversion and extraction of features using Structure Similarity Index Method are performed on images. The features are then juxtaposed with reference images' feature and authenticity is predicted.

All the above approaches commonly make use of expensive hardware for image acquisition and cannot be used by a common person who is affected by this problem too. The extensive datasets used are difficult to be acquired as fake notes are not easily available. The approaches are resource intensive as huge data is used for training the model. These problems have been addressed in the proposed approach by making use of a simple handheld device for image acquisition. The approach involves a Siamese network with contrastive loss function and determines the dissimilarities between images and it will require a small dataset for training the model.

III. PROPOSED SOLUTION

A. Pre-Processing

The image of the note captured by the camera cannot be fed directly into the model for various reasons: the images might have different intensities, contain background noise or not be in a perfect birds-eye view format, which might render the model less efficient. So, the following steps are performed to standardize the input image as shown in Fig 3.2

A.1 Edge Detection

The image captured by the smartphone camera will contain background noise. So, Canny edge detection is used to find the edges of the note and crop out the note. Canny edge detection works in the following way:

- Apply Gaussian filter to smoothen the image by removing noise.
- Sobel kernel is then applied in the horizontal and vertical directions to get the first derivative in both the directions G_x and G_y respectively.

$$\text{Edge Gradient (G)} = \sqrt{G_x^2 + G_y^2} \quad (1)$$

$$\text{Angle } (\Theta) = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad (2)$$

- The angles are then rounded off to the horizontal, vertical or the two diagonals i.e. $0^\circ, 45^\circ, 90^\circ$ and 135° .
- The obtained image may contain edges that are uniformly thick or thin. Therefore, to thin out all the edges non-maximum Suppression Algorithm is used.

The previous step recognizes few parts of the images as an edge, which is actually not an edge (non-edge). Non-hysteresis thresholding is performed to detect and filter out these non-edges.

Two threshold values max and min are considered. Any edge with intensity greater than max is considered to be edges and

lesser than min is considered to be non-edge and discarded. Edges between these thresholds are considered only if they are attached to an edge that is detected before.

A.2 Contour Detection

The previous step detects many edges present inside the note and Contour detection is the principle of detecting part of the images with the same intensity. In the image of the note, the note is assumed to be the single largest feature with a rectangular edge. So, we make use of this fact for detection of the note. After contour detection, the largest contour with 4 points or corners is taken as the boundary of the note. The findContours function in OpenCV helps us with the objective of getting the different contour edges in the image.

Contours = cv2.findContours(image, cv2.RETR_LIST, cv2.CHAIN_APPROX_SIMPLE)

A.3 Perspective Transform

Using the four points obtained from the previous step, we now aim to transform the image into a birds-eye view. First, the corners are ordered in clockwise manner starting from the top-left corner. We do this making use of the following principles

- Smallest sum of x and y coordinate gives the Top-Left point.
- Largest sum of x and y coordinate gives the Bottom-Right point.
- Smallest difference of x and y coordinate gives the Top-Right point.
- Largest difference of x and y coordinate gives the Bottom-Left point.

The skewed image of the note when measured along all the different edges, gives us two heights and widths. The maximum width and maximum height from the two available widths and heights respectively from the image of the note are taken. Then, the perspective matrix is calculated using the 4 points using the getPerspectiveTransform in OpenCV.

p_matrix=cv2.getPerspectiveTransform(src_points,dest_points)

Then, using this matrix a perspective transformation is achieved using the OpenCV function (3) warpPerspective.

Result = cv2.warpPerspective(image, p_matrix, (max_width, max_height))

It uses the following formula

$$dts(x, y) = src \left(\frac{M_{11}x + M_{12}y + M_{13}}{M_{21}x + M_{22}y + M_{23}}, \frac{M_{21}x + M_{22}y + M_{23}}{M_{21}x + M_{22}y + M_{23}} \right) \quad (3)$$

Here, M is the perspective matrix.

A.4 Grey-scale Conversion

The RGB image format is now converted to a grey-scale format to reduce dimensionality and mitigate the adverse influence of intensity on the model.

The impact of different colors on the human eye is different, due to which same weightage to all Red, Green and Blue cannot be given. So the following function is used to give different weights to different colors.

$$Grey-scale = R * 0.3 + G * 0.50 + B * 0.11 \quad (4)$$

B. Data Augmentation

Now, the next step to be performed is data augmentation. Data augmentation is the process of converting existing images in the training set to new plausible images. This is done to increase the number of training examples in the dataset and to make the model more robust and accurate. This process can be done varying these parameters:

- Shifting
- Flipping
- Zooming
- Rotating
- Shearing

C. Template Matching



Fig 3.1 – Template matching

Table II - Features that are to be extracted

Feature	Description
	Green 500 print
	Circle with 500
	Latent Image of 500

For the next step, aforementioned features from the note are extracted to check if the note is genuine. The features chosen are the ones that are usually unclear in a fake note. Template matching is now performed so that these individual features of the note are detected and cropped out as shown in Fig 3.1. A template is a reference for image features. The whole image is scanned to search for the part that is similar to that of the template. After finding the feature, that part is cropped from

the image and sent to the further steps. We use scikit-image library’s match_template function to do the job.

$$result = match_template(image, feature)$$

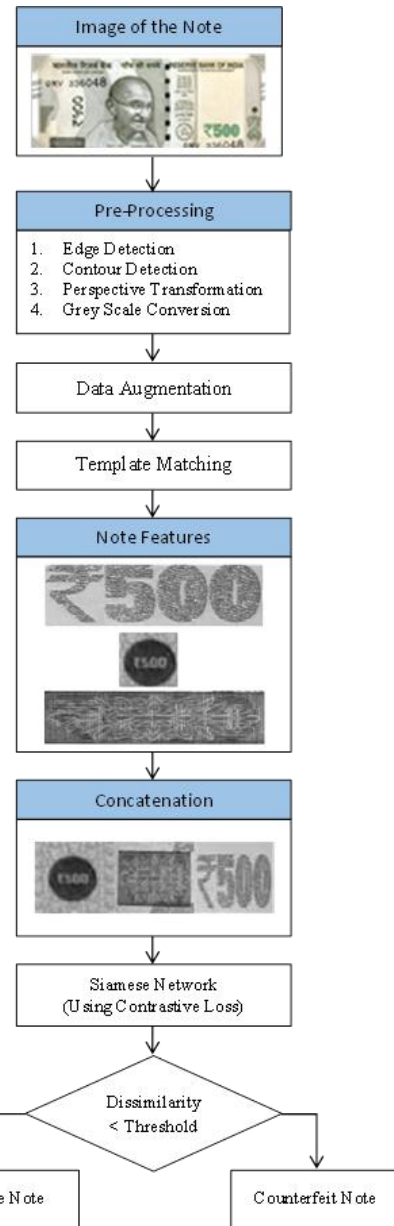


Fig 3.2 – Flow Diagram of the solution

D. SIAMESE NETWORK

The few-shot learning concept is implemented with the help of a Siamese Network. Few-shot learning uses few images of an entity to learn the similarity between images and can be used to compute dissimilarities between two images. Therefore, this model is best suited to overcome the limitation of the unavailability of an extensive dataset of fake notes.

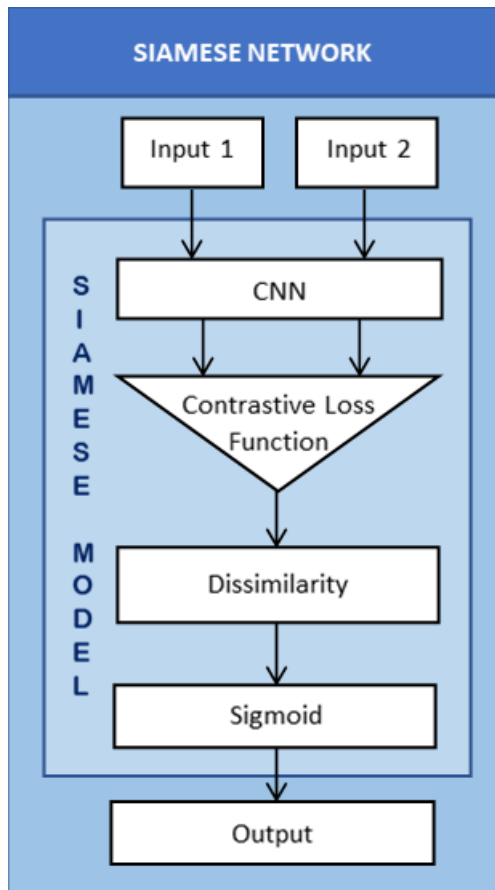


Fig 3.3 – Siamese network model

The difference between the two images is calculated using Contrastive Loss function. It is as follows:

$$L(W, X, \vec{X}_1, \vec{X}_2) = (1 - Y) \frac{1}{2} (D_w)^2 + (Y) \frac{1}{2} \{ \max(0, m - D_w) \}^2 \quad (5)$$

Where,

D_w - Euclidian Distance between the outputs of two images from Convolutional layer

m - margin

Y - 0 for images of same class otherwise 1

IV. EXPERIMENTAL RESULTS

The dataset is prepared by capturing images of genuine notes using a smartphone camera (16MP). The fake notes were generated by taking a photocopy of the note and then capturing its image through the smartphone camera. The dataset consisted a total of 160 genuine note images and 63 fake note images. The details of the dataset are summarized in the Table III.

Table III – Details of the dataset

Denomination	Resolution	Size	RGB
₹500	16 MP	4608x2592	Yes

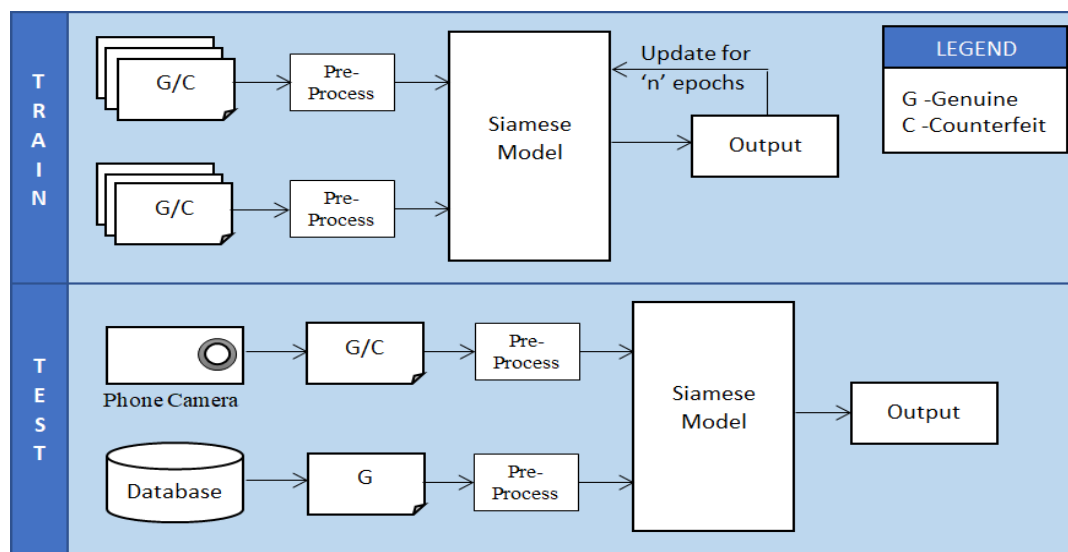


Fig 3.4 – Flow diagram for training and testing

Siamese network is a neural network variant which learns the similarities between two inputs. The neural network takes in different images that might belong to the same or different classes. The Siamese neural network receives two different images as inputs. After passing through the network, features are extracted for both images. The dissimilarities between these images is now calculated and fed back to the networks for further learning. Thus, the model learns the dissimilarities between the two images being fed to it.

The note is cropped along its edges using Canny edge detection. The data is converted into the standard dimension of 2825 x 1200 using openCV resize function. All images are converted to gray scale, using cv2.cvtColor function in openCV, for better feature extraction by stripping away illumination factor influencing the result.

To increase the dataset we employed several data augmentation techniques varying parameters such as width shift, shear range, height shift, and rotation range. Features are extracted using template matching function `match_template` in `skimage` library by matching against the standard feature images. These features are stacked horizontally. The Siamese model is fit by feeding permutations of genuine-genuine or genuine-counterfeit pairs of feature images which have been stacked horizontally. The testing is done by capturing an image of a note with a smartphone camera, preprocessed and then fed to the trained model with a genuine note as reference to classify the input image as genuine or fake. If the test image is genuine, then the positive class 1 is assigned otherwise negative class 0 is assigned.

The Siamese model involves a convolutional neural network (CNN) with 3 hidden layers each with rectified linear unit (ReLU) activation and one output layer with Sigmoid activation. The dissimilarity is obtained by the contrastive loss function on which Sigmoid activation is applied to squash output within a range of 0 to 1. To prevent over fitting of model to the training dataset we make use of Dropout regularization and provide probability of dropout as 0.25 for each hidden layer. Adam optimizer, a fast convergence optimizer, is used with learning rate of 0.00006. The confusion matrix for train and test dataset is shown in Table IV and Table V.

For 2000 epochs, with batch size of 1024, an accuracy of 85% was obtained and any increase to the number of epochs did not increase the accuracy. The evaluation of results has been depicted below by the confusion matrices in Table IV and V, and column graph with test accuracy against epochs where columns represent accuracy when dropout was included and excluded respectively (Fig 4.1).

The performance of proposed approach is evaluated against accuracy (6) and sensitivity (7) with obtained values 85%, 91% for test dataset and 87%, 92% for train dataset respectively (Fig 4.2).

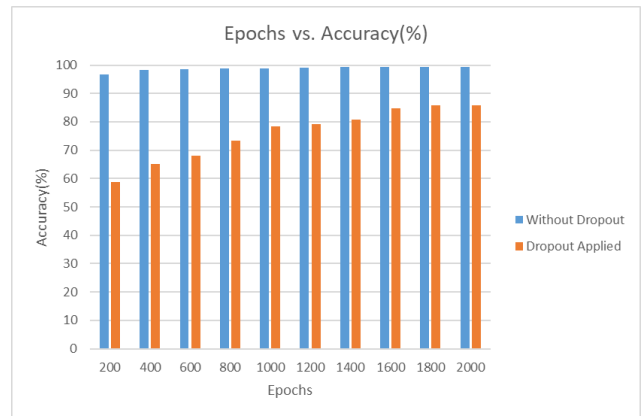


Fig 4.1 – Epoch vs Accuracy graph

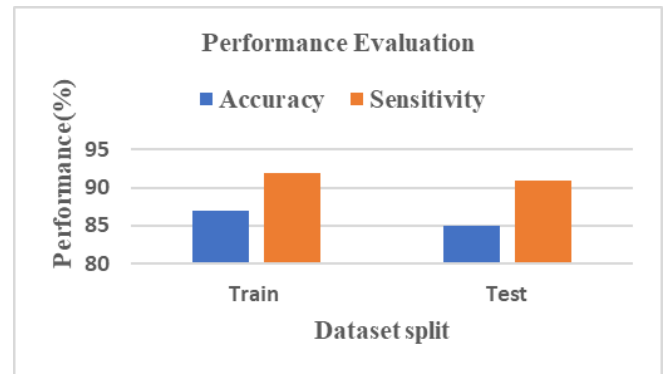


Fig 4.2 – Performance Evaluation

Table IV – Confusion Matrix Train Set

TP	FP
471	36
78.5%	6%
FN	TN
43	50
7.2%	8.3%

Table V – Confusion Matrix for Test Set

TP	FP
471	36
78.5%	6%
FN	TN
43	50
7.2%	8.3%

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \quad (6)$$

$$Sensitivity = \frac{TP}{TP+FN} \quad (7)$$

IV. CONCLUSION

Smartphone enabled fake note detection using Siamese network is proposed in this paper. Out of 17 distinct features of ₹500 currency 3 prominent features were chosen to build a model. The experimental analysis resulted in sensitivity and accuracy of 92% and 85% respectively. As a future work all distinct features from both obverse and reverse sides of various currencies can be evaluated to build more effective and highly predictive model and hence to classify any input image of a note as genuine or counterfeit.

REFERENCES

- <https://m.rbi.org.in/Scripts/AnnualReportPublications.aspx?Id=1263>
- <https://www.livemint.com/industry/banking/fake-500-rupee-notes-up-121-in-2018-19-says-rbi-1567136425194.html>
- <https://paisaboltahai.rbi.org.in/rupees-five-hundred.aspx>
- Ali, Toqeer, Salman Jan, Ahmad Alkhodre, Mohammad Nauman, Muhammad Amin, and Muhammad Shoaib Siddiqui. "DeepMoney: counterfeit money detection using generative adversarial networks." *PeerJ Computer Science* 5 (2019): e216.
- Agasti, Tushar, Gajanan Burand, Pratik Wade, and P. Chitra. "Fake currency detection using image processing." In *Materials Science and Engineering Conference Series*, vol. 263, no. 5, p. 052047. 2017.
- Navya Krishna G, Sai Pooja G, Naga Sri Ram B, Yamini Radha V, Rajarajeswari P. "Recognition of Fake Currency Note using Convolutional Neural Networks". In *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Volume-8 Issue-5 March, 2019
- Berenguel, Albert, Oriol Ramos Terrades, Josep Lladós, and Cristina Cañero. "Banknote counterfeit detection through background texture printing analysis." In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pp. 66-71. IEEE, 2016.
- Kang, K., and C. Lee. "Fake banknote detection using multispectral images." In *2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pp. 1-3. IEEE, 2016.
- Upadhyaya, Akanksha, Vinod Shokeen, and Garima Srivastava. "Counterfeit Currency Detection Techniques." In *2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 394-398. IEEE, 2018.
- Alshayegi, Mohammad H., Mohammad Al-Rousan, and Dunya T. Hassoun. "Detection method for counterfeit currency based on bit-plane slicing technique." *International Journal of Multimedia and Ubiquitous Engineering* 10, no. 11 (2015): 225-242.
- Kamble, Achal, and Mrudula Nimbarte. "Fake Currency Detection using Clustering and SVM Classification." (2018).
- Darade, Sonali R., and G. R. Gidveer. "Automatic recognition of fake Indian currency note." In *2016 International Conference on Electrical Power and Energy Systems (ICEPES)*, pp. 290-294. IEEE, 2016.
- Mohamad, Nur Syuhada, Burairah Hussin, A. S. Shibghatullah, and A. S. H. Basari. "Banknote authentication using artificial neural network." In *Proceedings of the International Symposium on Research in Innovation and Sustainability, Malacca, Malaysia*, pp. 15-16. 2014.
- Murthy, Sahana, Jayanta Kurumathur, and B. Roja Reddy. "Design and implementation of paper currency recognition with counterfeit detection." In *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, pp. 1-6. IEEE, 2016. Kndlksands,ds
- Roy, Vidhi, Gangey Mishra, Rahul Mannadiar, and Sushant Patil. "Fake Currency Detection Using Image Processing." (2019).



Dhanush C – Obtained his Bachelor of Engineering degree in Computer Science and Engineering from RNSIT . his areas of focus are Computer vision, Natural language processing and Data science.



Dr. Bhavanishankar K– Working as an Asst. Prof. at RNS Institute of Technology in Department of CSE. His research area of interests includes image processing, machine learning, artificial intelligence, algorithms and data structures. He has published 8 research articles in International journals. He is passionate about guiding creative technical/research projects.

AUTHORS' PROFILE



Adith Kumar B A – Completed his B.E in Computer Science and Engineering from RNS Institute of Technology, Bangalore. He has secured 25th Rank for the state in National Talent Search Examination 2012 (NTSE). His research interested include Machine learning and Financial engineering



Ajay Deshpande – Obtained his B.E in Computer Science and Engineering from RNS Institute of Technology, Bangalore. His area of interests includes data science, machine learning and mathematical theories.



Ajay Umakanth – B.E in Computer Science and Engineering from RNS Institute of Technology, Bangalore. His area of interests includes data science, machine learning, app development and IOT