# Fingerprint Presentation Attack Detection using Referential Quality Metrics and Minutiae Count

**Akhilesh Verma**
Ajay Kumar Garg Engineering College
**Anshadha Gupta**
Ajay Kumar Garg Engineering College
**Mohammad Akbar**
Ajay Kumar Garg Engineering College
**Arun Kumar Yadav**
National Institute of Technology Hamirpur
**Divakar Yadav** ( dsy99@rediffmail.com )
National Institute of Technology https://orcid.org/0000-0001-6051-479X

**Research Article**

**Keywords:** FPAD, Liveness, k-NN, SVM, neural network, referential image quality metrics

**Posted Date:** September 21st, 2021

**DOI:** https://doi.org/10.21203/rs.3.rs-792415/v1

# Fingerprint Presentation Attack Detection using Referential Quality Metrics and Minutiae Count

[1]Akhilesh Verma, [2]Anshdha Gupta, [3]Mohammad Akbar, [4]Arun Kumar Yadav, [5]Divakar Yadav[*]

[1, 2, 3]Department of Computer Science & Engineering, AKG. Engineering College, Ghaziabad (UP), India

[4, 5]Department of Computer Science & Engineering, National Institute of Technology, Hamirpur (H.P), India

[1]akhilesh.verma@hotmail.com; [2]anshdhagupta94@gmail.com; [3]mohdakbar1971997@gmail.com; [4]ayadav@nith.ac.in; [5]dsy99@rediffmail.com

ORCID: https://orcid.org/0000-0001-6051-479X

**Abstract:** The fingerprint presentation attack is still a major challenge in biometric systems due to its increased applications worldwide. In the past, researchers used Fingerprint Presentation Attack Detection (FPAD) for user authentication, but it suffers from reliable authentication due to less focus on reducing the 'error rate'. In this paper, we proposed an algorithm, based on referential image quality (RIQ)-metrics and minutiae count using neural network, k-NN and SVM for FPAD. We evaluate and validate the error rate reduction with different machine learning models on the public domain, such as LivDet crossmatch dataset-2015 and achieved an accuracy of 88% with a neural network, 88.6% with k-NN and 88.8% using SVM. In addition, the average classification error (ACE) score is 0.1197 for ANN, 0.1138 for k-NN and 0.1117 for SVM. Thus, the results obtained show that it was achieved a reasonable accuracy with a low ACE score with respect to other state-of-the-art methods.

**Keywords:** FPAD, Liveness, k-NN, SVM, neural network, referential image quality metrics

## 1. INTRODUCTION

Nowadays biometrics is parlance, as it provides a way for identification and authentication of individuals. Among all the biometrics, fingerprints are the most reliable and useful biometrics system[1]. In the paper, [2] authors commented that fingerprint-based biometric system is less expensive as compared to face and iris-based systems. In the past, many researchers commented on applications of fingerprints and their usability but it laid down in its objective due to presentation attacks [3].

In continuation of resolving security threats in fingerprint, various researches have been carried out for FPAD at both hardware as well as at software level. Hardware-based approaches of FPAD use fingerprint readers along with sensors to analyse living attributes of persons like odour, blood pressure, skin distortion, etc. Two types of fingerprint features are generally used in software-based approaches, first is *dynamic features* like skin colour change due to skin elastic properties, pressure etc. and second is *static features* like ridges and valley features, sweat pores, perspiration, etc.

The FPAD are further grouped under open-set and closed-set solution types. Open-set solutions are based on limited information about spoof materials during the training phase and testing is done with a different set of spoof materials [4], [5]. Further, closed-set solutions are based on full information about spoof materials during training and testing is done only with known spoof materials. The major challenges of FPAD in close-set solutions is that of over-fitting and poor generalization. Many of the researchers commented that fingerprints spoof can be fabricated using numerous materials so an open-set generalized solution is a very difficult task and dependent on the fingerprint sensors.

The remaining article is organized as follows; section-2 explains the related research done in the field of FPAD, the significant finding (RQ) of literature and the process of solving the RQs. Section-3 discuss the proposed method based on RQs identified in the literature review. Section-4 discusses the experimental evaluation of the proposed work. Finally, in section-5 we conclude and put forwards lights for future works.

## 2. RELATED WORK

The magnitude of the presentation attack is posing security flaws existing in fingerprint recognition systems. So automated spoof detection techniques are developed in past years. The spoof artifacts or spoof attack materials come in large variation because of sensors optical design and materials mechanical properties. Spoof detectors suffer from unseen spoof samples because the machine learning model uses limited spoof samples for training.

The FPAD approaches are mainly based on *local features that* include Local Binary Pattern (LBP), Local Phase Quantization (LPQ), Binarized Statistical Image Features (BSIF) [6] and *global features* that are based on deep learning methods [7]. In a recent work, researchers used a middle approach of FPAD using minutiae-centred patches with deep learning [8]. The state of art prescribing above feature-based solution are discussed in the current section. In paper, authors proposed a convolution neural network (CNN) based approach using local patches to identify features around fingerprint minutiae. These local patches are then trained by the Inception v3- CNN model to generate a global spoofed score to discriminate between fake and live fingerprints. The suggested method for spoof detection has reduced average classification error up to 69% under both: unknown and known spoof materials for dataset 'LivDet 2015'.

R. F. Nogueira et al. in [9] explained the use of principal component analysis (PCA) and support vector machine (SVM) to identify the features of liveness. They also used the combination of machine learning model on LivDet dataset 2009, 2011 and 2013. Further, they compared the results of different models on the mentioned datasets. The concluding remarks of authors said that ConvNet + PCA + SVM and AUG + ConvNet + PCA + SVM showed best result for LivDet 2013 data set. A. Rattani et al. in [10] adopted a fingerprint spoof detector using W-SVM approach. This scheme has achieved an average true detection rate of up to 70% for LivDet 2011 dataset. In another research, Y. Ding et al. [11], suggested multiple one-class SVM (OC-SVM) classifiers with local textural features using GLCM, LBP, BSIF, BGP, and LPQ. Each OC-SVM uses various set of features and some fake fingerprints for refining the decision boundary. This technique requires lesser spoof samples at the time of training and has stable performance across different fabrication materials. They obtained the Correct Detection Rate (CDR) of 87%, which was better than B-SVM (binary SVM) [12] on LivDet dataset 2011.

In another approach, given by C. Yuan et al. [13] suggested fingerprint liveness detection based on feature extraction technique using CNN with PCA reduction and SVM classifier. This method performed well in liveness detection. They calculated the average classification error (ACE) scores on LivDet 2009, LivDet 2011 and LivDet 2013 datasets. In another study, E. Park et al. [14], proposed a CNN model on fingerprint patches to calculate ACE scores. The advantage of using patches was to increase the dataset size for the training. In their implementation, they used databases of LivDet 2011, 2013 and 2015 and obtained an average classification error (ACE) rate of 1.35%. In [15], R. Gajawada et al. proposed an approach named Universal Material Translator (UMT) for cross material fingerprint spoof detection using style transfer. This technique enhances the generalization performance on novel spoof materials while preserving high performance for known materials. They used local patches instead of the whole image of LivDet 2015. A detailed survey has been performed by E. Marasco et al. in [16] that specifies different methodologies used for liveness detection.

After studying the available literature, the following significant points have been identified. Firstly, in the previous research a bounded approach (shape, size, location, and device-dependent) was used for fingerprint presentation attack detection ((RQ1)). However, this approach suffers from high computation overhead, requires device-dependent algorithm and variety in large

numbers in live samples to gives better results. Secondly, researchers used different data sets and algorithms to improve accuracy, but they did not focus much on reducing the error rate that is more important in spoof detection for PA (RQ2).

To examine RQ1, we used non-hand-crafted feature (Image Quality Assessment (IQA)-Referential Quality Metrics (SSIM, MSE and PSNR) and Minutiae Count) to reduce the computation overhead (with limitation to test on multiple sensors). To examine RQ2, we verified the consistency of error rate on different machine learning algorithms and checked the performance with average classification error (ACE) score and accuracy. In the next section, we discuss the proposed method for spoof detection to obtain a good trade-off between accuracy and ACE score.

## 3. PROPOSED WORK

This section describes the proposed work based on issues identified in the literature. In the past, many researchers worked on data sets: LivDet 2009, 2011, 2013, 2015, 2017 and 2019 [23] for FPAD. In this study, we appraised the proposed approach on the LivDet 2015 of CrossMatch dataset. This dataset contains 500 live fingerprint images and 502 spoof fingerprints (152-Body double, 150-Ecoflex, 200-Play Doh) images.

Figure-1 shows that identification of 'Live' and 'Spoof' fingerprint images is difficult. Hence, some methodologies are required to detect these spoof fingerprints as they are a threat to security. In this work, we proposed a scheme to detect fingerprint authentication in the aspect of live or spoof for a secure mechanism in today's biometric verification scope. Figure 2 explains the basic architecture of the proposed work. The methodology is mainly divided into three stages Pre-processing phase, image quality assessment phase and minutiae extraction, training phase.

(a) Live fingerprint (b) Spoof fingerprint

Figure 1**.** Live and Spoof fingerprint [20]

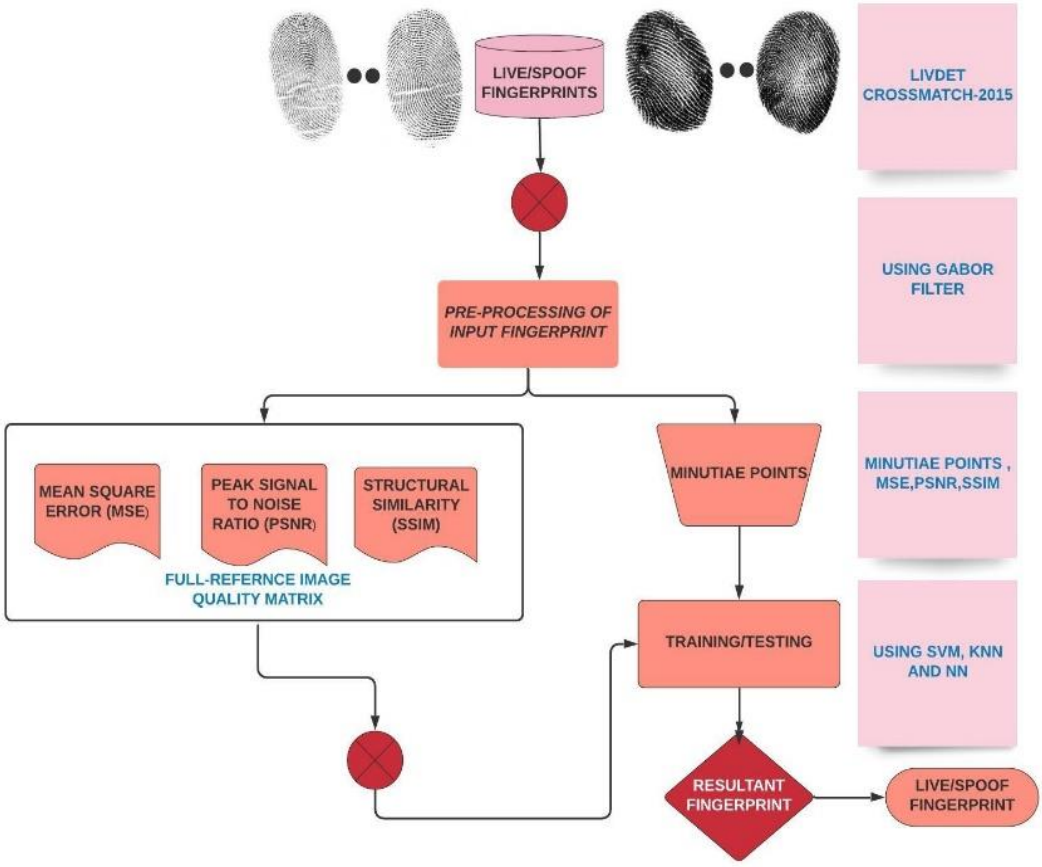



**Figure 2.** Flow chart for proposed methodology

### 3.1. Pre-processing

The captured fingerprint images are of high quality and full of noise during image recording, due to dryness or wetness of the skin. To enhance the captured images Gabor filter is used for filtering using specific wavelength and orientation. The grayscale image is transformed into its threshold image, it is assigned as black otherwise white. The ridges in fingerprints of binary images are then thinned to one-pixel width to ease the task of fingerprints minutia extraction.

### 3.2. Image Quality Assessment

Image quality assessments are made using three types of full-reference image quality matrices namely: Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). MSE and PSNR are error-sensitive measures and the SSIM is a structural similarity measure. These matrices are further explained as follows:

In this work, MSE finds an average grey value of image-spoof and real fingerprints. Hence two images were being used in which one image was the enhanced fingerprint image obtained from the first stage and the other was a plain white image as referenced image. Hence, the filtered image is given by equation 1.

$$MSE = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[A(x,y) - A'(x,y) \quad \ldots\ldots\ldots \quad (1)$$

Where, A(x, y) is enhanced image, A'(x, y) is referenced image and M, N is resolution.

In PSNR, an error is measured using peak value i.e., calculating the ratio of the maximum power of image and power of noise in an image. The more value of PSNR better the quality of an image. The general form is given by equation 2.

$$PSNR = 20 * \log_{10}(\frac{255}{sqrt(MSE)}) \quad \ldots\ldots\ldots \quad (2)$$

The SSIM is a perception-based model that mainly considers the degraded image and hence measures the perceptual difference between two identical images. SSIM utilises luminance, contrast and structure to compare local patterns of pixel intensities. The formula for it is as given in equation 3, where $\mu_x$ = Average of enhanced fingerprint image and $\mu_y$ = Average of the reference image.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x{}^2\,\mu_y{}^2 + C_1)(\sigma_x{}^2 + \sigma_y{}^2 + C_2)} \quad \text{--------} \quad (3)$$

Where, $c_1$ and $c_2$ are included to avoid instability when $\mu_x^2 + \mu_y^2$ *is very small and close to zero*. $c_1$ and $c_2$ are given as: $c_1 = (K_1 L)^2$ where $K_1 = 0.01$; $c_2 = (K_2 L)^2$ *where* $K_2$= 0.03; L = dynamic range of the pixel values.

### 3.3. Minutiae points extraction

Before extracting minutiae points, it is important to perform ridge orientation since it is necessary for describing, matching, and detecting minutiae. First-a-fall image gradient is calculated from the original image by convolving it with some filter. In this work, we used a Gaussian filter. The gradient image is calculated for two directions 'x' and 'y'. With the help of these two gradient images, covariance data is calculated. After smoothing the covariance data, a weighted summation of the data is performed. Sine and Cosine functions are applied to the principal direction of gradient image which is further smoothened that helps in calculating the reliability of moment of orientation. Hence, the moment of inertia is calculated around the orientation axis (i.e., minimum inertia) along with an axis perpendicular to orientation (i.e., maximum inertia). Therefore, the reliability factor is calculated as 1–(minimum inertia/maximum inertia). If (minimum inertia/maximum inertia) ratio is near to one, then it will depict little orientation information. At last, the mask is applied to reliability measures to exclude those areas which makes the orientation in the denominator small. Therefore, in this experiment, we used the mask value 0.001. After calculating ridge orientation, minutiae points are extracted [29]. Minutiae points refer to some specific points in the fingerprint and consists of many features such as ridge bifurcation or ridge ending. In this work, all the ridge bifurcation having crossing number three are detected and some of these are weeded and trimmed which are of no use and thus, we get a final set of minutiae points.

### 3.4 Training and Testing of Models

In this phase, we trained the model with three types of training methodologies: Neural Network (NN), SVM (Support Vector Machine) and k-NN (k-Nearest Neighbour) along with the comparison between their results. The input set for them contains 4 features- MSE, PSNR, SSIM and minutiae points.

#### *3.4.1 Neural Network*

A neural network is a sequence of algorithms that aims to recognize patterns for a set of data in a way the human brain works. Hence it refers to a collection of multiple neurons either in artificial or organic nature which widely helps in the classification of data.

In this work, we used in total 1002 images of live and spoof fingerprints out of which 70% were used as training data i.e., 702 samples, 15% as the validation set i.e., 150 samples and remaining 15% as the testing data i.e., 150 samples. Data division for training and testing was random. The number of hidden neurons used in the NN model was 9 and the activation function used was sigmoid which returns a value in the range of 0 to 1. The number of hidden neurons was estimated by training the net with different neurons as shown in Table 1. We have chosen neuron 9 as it attained the best result for both testing and overall accuracy.

Table 1. Different combinations of hidden neurons

| Method | Hidden neurons | Testing Accuracy (%) | Overall Accuracy (%) |
|---|---|---|---|
| Neural Network | 6 | 88 | 87.9 |
| | 7 | 88.7 | 88 |
| | 8 | 86.7 | 87.5 |
| | **9** | **89.3** | **88** |
| | 10 | 85.3 | 87.7 |
| | 11 | 87.3 | 88.1 |
| | 12 | 88 | 86.1 |

*3.4.2 SVM*

In this work, Support vector machine (SVM) is experimented with all kinds of SVM (linear SVM, Quadratic SVM, Cubic SVM, Fine Gaussian SVM, medium SVM, coarse Gaussian SVM) and got the best outcome for Fine Gaussian SVM. For this, we set cross-validation folds equal to 5. The kernel function chosen for this experiment was Gaussian (evaluated by experimenting performance with other kernel functions too and got best result for Gaussian). The two hyperparameters i.e. kernel scale and box constraint were also used. These two parameters were gauge by experimenting with different combinations of values as shown in table 2 and observed that if box constraint increases then kernel scale had to decrease for getting better results. In addition, box constraint should not be lower as it leads to overfitting problems and increases support vectors. The best accuracy was 88.8% which was received for kernel scale-1 and box constraint-5. The multiclass method used here was one-vs-one as default.

Table 2. Different combinations of hyperparameters for SVM

| Method | Kernel Scale | Box Constraint | Accuracy (%) |
|---|---|---|---|
| FINE GAUSSIAN SVM | 0.5 | 1 | 88.1 |
| | 1 | 6 | 88.6 |
| | 1 | 8 | 88.5 |
| | 1 | 7 | 88.5 |
| | 1 | 5 | 88.8 |
| | 1 | 4 | 88.6 |
| | 1 | 3 | 88.5 |
| | 2 | 2 | 88.2 |
| | 2 | 4 | 88.2 |
| | 3 | 2 | 87.4 |
| | 3 | 3 | 87.8 |
| | 3 | 2 | 87.4 |
| | 3 | 1 | 86.7 |
| | 3 | 2 | 87.4 |

*3.4.3 k-NN*

K-nearest neighbour (k-NN) is a part of supervised machine learning algorithms widely used for classification problems and wield features similarity to anticipate the new data point values, which means that these data points are assigned values based on how scrupulously points are matched in the training set. In this work, we used Medium k-NN. There are three hyperparameters used: number of neighbours, distance metric and distance weight. These values were set by examining different combinations of them as shown in table 3. It was observed that, if neighbours are very small then it leads to more noise in data. Hence, we need to choose an optimal value for this parameter. Therefore, we did not experiment with several neighbours less than 5. The best

result was achieved when the number of neighbours were 5, the distance metrics were Chebyshev and the distance weight was inverse**.**

Table 3. Different combinations of hyperparameters for k-NN

| Method | No. of Neighbours | Distance Metrics | Distance Weight | Accuracy (%) |
|---|---|---|---|---|
| Medium K-Nearest Neighbour (KNN) | 5 | Euclidean | Equal | 87.7 |
| | 5 | Euclidean | Inverse | 87.8 |
| | 5 | Chebyshev | Equal | 88.1 |
| | 5 | Chebyshev | Inverse | 88.6 |
| | 7 | Euclidean | Equal | 87.2 |
| | 7 | Euclidean | Inverse | 88.3 |
| | 7 | Chebyshev | Equal | 87.3 |
| | 7 | Chebyshev | Inverse | 88.1 |
| | 9 | Euclidean | Equal | 86.4 |
| | 9 | Euclidean | Inverse | 88.1 |
| | 9 | Chebyshev | Equal | 86.6 |
| | 9 | Chebyshev | Inverse | 87.8 |
| | 11 | Euclidean | Equal | 87.3 |
| | 11 | Euclidean | Inverse | 87.9 |
| | 11 | Chebyshev | Equal | 86.9 |
| | 11 | Chebyshev | Inverse | 87.6 |

# 4 EXPERIMENTAL EVALUATION AND DISCUSSION

In section 3, we evaluated and discussed the testing accuracy of all models (NN, SVM and k-NN) from features evaluated from LivDet 2015. This section discussed the benchmarking of the ACE score of FPAD using different machine learning models.

## 4.1 Results analysis using Neural Network

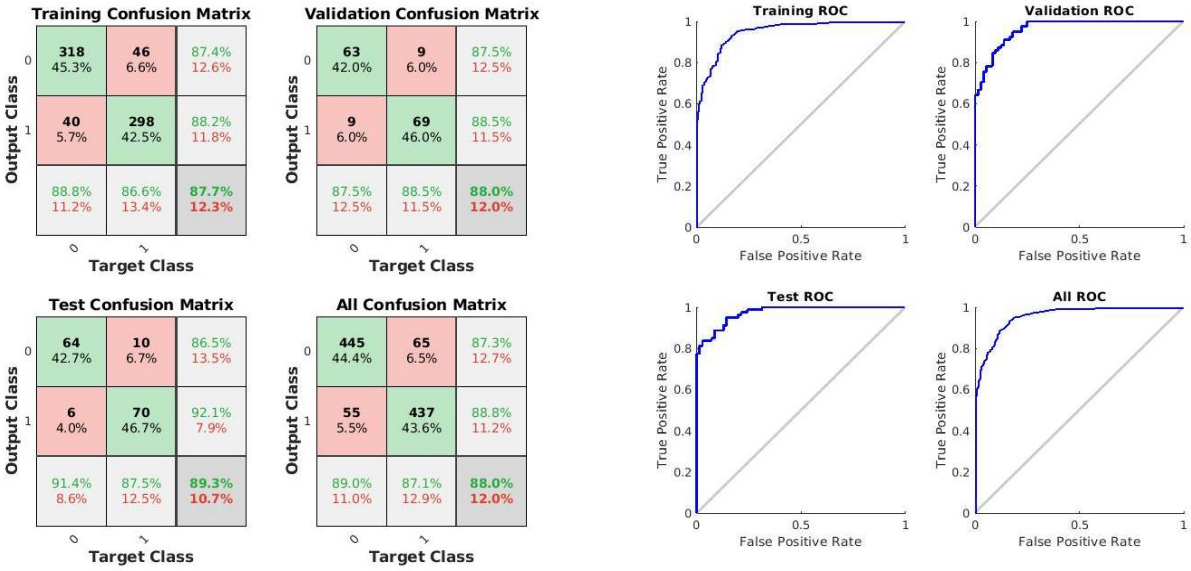
In section 3, table 1 gives the best accuracy of neural network at 9-hidden neurons with 88%. The confusion matrix and receiver operating characteristic (ROC) is shown in figure 4. In the confusion matrix, '0' describes *live* fingerprints and '1' depicts *spoof* fingerprints. Out of 40 epochs, the best validation result was obtained at epoch 34 with validation performance 0.24575 and gradient value was 0.022848 at epoch 40. According to the confusion matrix shown in figure 3(a), the false-negative rate (***FerrFake***) for spoof is small for all 4 matrices i.e., training, validation, testing and when combined together. We can see, for spoof detection algorithms the (***FerrFake)*** should be less than (***FerrLive)*** values to have better authentication. Also, the ROC curve as shown in figure 3(b), the curve is showing false-negative rate remains at '0' when the true positive rate was 0.65.

## 4.2 Results analysis using k-NN

In section 3, table 3 gives the best accuracy for k-NN as 88.6%. The confusion matrix is as shown in figure 4(a). False-negative rate (FNR) is smaller for spoof i.e., 8.8% as compared to live fingerprints, which are 14%. This means there are only 44 spoof images, which were being misclassified as live, and as this factor should be small hence, we worked upon showing a low FNR rate for spoof as compared to the FNR rate for live fingerprints. According to AUC shown in figure 4(b), our classifier model is showing the most optimal result at (0.09, 0.86) i.e., for TPR 0.86, and FPR is 0.09. In this case, false-positive rate is less than the true positive rate. As we move towards right of optimal value, both TPR and FPR will increase which is not good for the model as the value for FPR should be as low as possible and if we move towards the left of optimal value, then both the values decrease. We obtained AUC = 0.95 which is comparably good as the maximum value of an area for the model can be 1.

## 4.3 Results analysis using SVM

In section 3, Table 2 gives the best accuracy of SVM (FINE GAUSSIAN SVM) as 88.8%. The confusion matrix as shown in figure 5(a), out of 500 live images 440 were classified correctly as live and 60 misclassified as fake. Out of 502 spoof fingerprints, 450 were classified correctly as spoof and 52 were misclassified as live. Like k-NN and NN, we can observe that the false-negative rate for spoof is 10.4% which is less than false-negative rate for live which is 12.0%. According to the AUC parameter shown in figure 5(b), the SVM classifier model is showing the optimal result on (0.10, 0.88), thus TPR is 0.88 and FPR is 0.10. In this case, false-positive rate is less than the true positive rate. As we move towards the right of optimal value both TPR and FPR increase which is not good for a model as the value for FPR should be as low as possible and if we move towards the left of optimal value, both the values decrease. We obtained AUC = 0.95 which is quite good as the maximum value of an area for the model can be 1.




(a) (b)

Figure 3. Confusion matrix-(a) and ROC curve-(b)

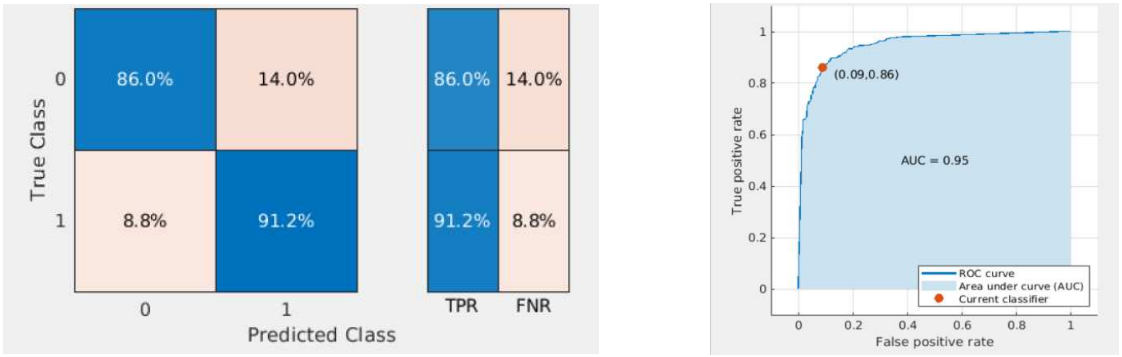
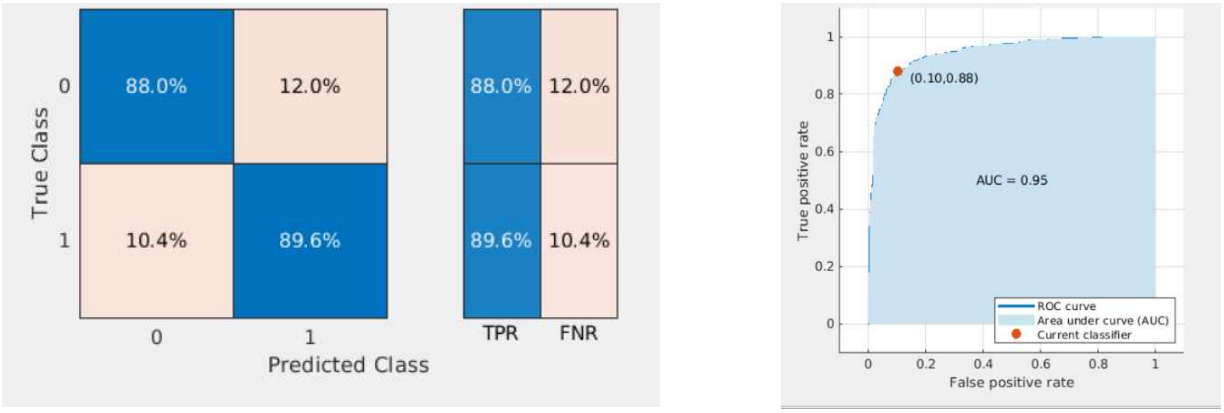



Figure 4. Confusion matrix-(a) and AUC (Area under curve) curve-(b) obtained for K-NN.




(a) (b)

Figure 5. Confusion matrix-(a) and AUC curve-(b) obtained for SVM.

### 4.4 Performance comparison of ACE score with other state of art method

In table 5, the comparison between all three methodologies i.e., k-NN, NN and SVM based on accuracy and ACE score is shown. The results obtained with the SVM is found to be the best in terms of lowest ACE score and highest accuracy. The ACE score is calculated by the formula **ACE = (*Ferrlive* + *FerrFake*) / 2** where '***FerrFake***' is calculated as total count of misclassified

spoof fingerprints divided by the total count of spoof fingerprints. '***FerrLive***' is given by the total count of misclassified live fingerprints divided by the total count of live fingerprints. Lower the ACE score better the model for the FPAD system.

Type-I and Type-II error needs to be minimized for raising the confidence on FPAD system. Therefore, we compare the performance of our approach in terms of an ACE score with other state of art. We evaluated the proposed approach with three classification methods as described above, which reflect an algorithm's robustness against existing spoof materials, in close-set environments. We observed that the models trained on referential quality metrics and minutiae count of the entire image has achieved a significantly higher reduction in average classification error as compared to the existing methods that use complex feature set as shown in table 6.

Table 5. ACE score and Accuracy of all three methods

| **Methodology** | **ACE** | **Accuracy (in %)** |
|---|---|---|
| Neural Network | 0.119780 | 88 |
| KNN | 0.113824 | 88.6 |
| SVM | 0.111792 | 88.8 |

Table 6. Performance comparison in terms of average classification error (ACE) (in %) with existing methods.

| **YEAR** | **TECHNIQUE** | **ACE** |
|---|---|---|
| 2013 | Weber local descriptor. (WLD + LPQ) [21] | 7.87 |
| 2014 | Convolution neural network (CNN) [9] | 6.45 |
| 2014 | Local Contrast phase descriptor (local contrast and LPQ) [22] | 5.70 |
| 2015 | DCNN + Voting strategy [23] | 3.50 |
| 2015 | Quality Features [24] | 2.10 |
| 2016 | Gradient-based texture features [25] | 6.63 |
| 2016 | Random Sample patches + CNN [26] | 3.42 |
| 2017 | CNN + PCA [13] | 4.50 |
| 2017 | Worked on BISF (binarized statistical image features) [27] | 3.03 |
| 2017 | CNN Patch-based voting approach [28] | 1.90 |
| 2020 | The proposed method: (RIQ metrics with minutiae count) | 0.111792 (SVM)<br>0.113824 (KNN)<br>0.119780 (NN) |

## 5 CONCLUSION AND FUTURE SCOPE

Fingerprint spoof detection is a challenging task as differentiating the live and spoof fingerprints are difficult. In this work, we tried to address issues identified in the literature. The proposed method solved the issues using features of RIQ metrics and minutiae count. These features were evaluated and verified with ANN, k-NN, and SVM machine learning models. We observed that the accuracies and ACE scores obtained by all three methods were approximately similar. SVM provides an accuracy of 88.8% and ACE = 0.111792 while neural network provides 88% accuracy and ACE=0.119780. Similarly, k-NN also showed good accuracy of 88.6% with ACE= 0.113824. Even though the accuracies are in good trade-off with the other state-of-arts, ACE scores are quite low. Hence, we can conclude that, the image quality matrices with minutiae count provide a significant improvement in error rates. This paper uses the whole images for extracting quality metrics, in future we can work on parts or patches of fingerprints. To improve results towards accuracy, other image quality matrices like a full reference or non-reference measures can be utilized. The input image can be divided into R, G, B components and by using combinations of R, G, B images as input images and rest as the reference images. With the above two proposal, the proposed method may further enhance the training process and thus increase the performance as well.

## 6. DECLARATIONS

**Funding:** No funds, grants, or other support was received.

**Conflicts of interest/Competing interests:** We declare that none of the authors' has any conflicts of interest involved.

**Availability of data and material:** Not applicable

**Code availability:** We will provide the codes if desired by publication after acceptance.

**Ethics approval:** We declare that none of the authors' violated any ethical principle.

**Consent to participate:** Not applicable.

**Consent for publication:** We provide our consent to the journal for the publication.

## References


[1] S. Guennouni, A. Mansouri, and A. Ahaitouf (2020). Biometric Systems and Their Applications. in *Visual Impairment and Blindness - What We Know and What We Have to Know*, IntechOpen.

[2] V. Rai, K. Mehta, J. Jatin, D. Tiwari, and R. Chaurasia (2020). Automated Biometric Personal Identification-Techniques and Applications. in *Proceedings of the International Conference on Intelligent Computing and Control Systems, ICICCS 2020*, 1023–1030.

[3] J. Galbally-Herrero, J. Fierrez-Aguilar, J. D. Rodriguez-Gonzalez, F. Alonso-Fernandez, J. Ortega-Garcia, and M. Tapiador (2014). On the vulnerability of fingerprint verification systems to fake fingerprints attacks. in *Proceedings - International Carnahan Conference on Security Technology*, 130–136.

[4] T. Chugh, K. Cao and A. K. Jain (2018). Fingerprint Spoof Buster: Use of Minutiae-Centered Patches. in *IEEE Transactions on Information Forensics and Security*, 13(9). 2190-2202, doi: 10.1109/TIFS.2018.2812193.

[5] A. Verma, V. K. Gupta, and S. Goel (2020). Fingerprint Presentation Attack Detection in Open-Set Scenario using Transient Liveness Factor. *Recent Adv. Comput. Sci. Commun.*, vol. 13, 2020.

[6] L. Ghiani, G. L. Marcialis, and F. Roli (2012). Fingerprint liveness detection by local phase quantization. in *Proceedings - International Conference on Pattern Recognition*. 537–540.

[7] A. S. Ahmad, R. Hassan, and R. M. Othman (2017). An investigation of fake fingerprint detection approaches. in *AIP Conference Proceedings*, vol. 1891, 1-7. https://doi.org/10.1063/1.5005353

[8] T. Chugh, K. Cao, and A. K. Jain (2018). Fingerprint spoof detection using minutiae-based local patches. in *IEEE International Joint Conference on Biometrics, IJCB 2017*, 581–589.

[9] R. F. Nogueira, R. De Alencar Lotufo, and R. Campos MacHado (2016). Fingerprint Liveness Detection Using Convolutional Neural Networks. *IEEE Trans. Inf. Forensics Secur.*, 11(6), 1206–1213.

[10] A. Rattani, W. J. Scheirer, and A. Ross (2015). Open set fingerprint spoof detection across novel fabrication materials. *IEEE Trans. Inf. Forensics Secur.*, 10(11), 2447–2460.

[11] Y. Ding and A. Ross (2017). An ensemble of one-class SVMs for fingerprint spoof detection across different fabrication materials. in *8th IEEE International Workshop on Information Forensics and Security, WIFS 2016*. 1-6. *doi: 10.1109/WIFS.2016.7823572*

[12] K. P. K. and P. S. Aithal (2017). Literature Review on Fingerprint Level 1 and Level 2 Features Enhancement to Improve Quality of Image. *Int. J. Manag. Technol. Soc. Sci.*. 2(2), 8–19.

[13] C. Yuan, X. Li, Q. M. J. Wu, J. Li, and X. Sun (2017). Fingerprint liveness detection from different fingerprint materials using convolutional neural network and principal component analysis. *Comput. Mater. Contin.*, vol. 53(4), 357–371.

[14] E. Park, X. Cui, W. Kim, J. Liu, and H. Kim (2018). Patch-based fake fingerprint detection using a fully convolutional neural network with a small number of parameters and an optimal threshold. *arXiv*., 1–12.

[15] R. Gajawada, A. Popli, T. Chugh, A. Namboodiri, and A. K. Jain (2019). Universal material translator: Towards spoof fingerprint generalization. *arXiv*., 1–8.

[16] E. Marasco and A. Ross (2014). A survey on antispoofing schemes for fingerprint recognition systems. *ACM Comput. Surv.*, 47(2), 1-36. https://doi.org/10.1145/2617756

[17] S. S. Marcel Mark Nixon Stan Z Li (2019). *Handbook of Biometric Anti-Spoofing*. Springer International Publishing, 2019.

[18] V. Mura *et al. (2018).* LivDet 2017 fingerprint liveness detection competition 2017. in *Proceedings - 2018 International Conference on Biometrics, ICB 2018*, 297–302.

[19] G. Orrù *et al.*, (2019). Livdet in action-fingerprint liveness detection competition 2019. ICB 2019, *arXiv*. 2019. doi: 10.1109/ICB45273.2019.8987281

[20] B. Tan (2008). Novel methods for fingerprints image analysis detect fake fingers. *SPIE Newsroom*. 10.1117/2.1200805.1171

[21] D. Gragnaniello, G. Poggi, C. Sansone, and L. Verdoliva (2013). Fingerprint liveness detection based on Weber Local image Descriptor. in *2013 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications, BioMS 2013 - Proceedings*, 46–50.

[22] D. Gragnaniello, G. Poggi, C. Sansone, and L. Verdoliva (2015). Local contrast phase descriptor for fingerprint liveness detection. *Pattern Recognit.*, vol. 48(4), pp. 1050–1058.

[23] J. Yang, J. Yang, Z. Sun, S. Shan, W. Zheng, and J. Feng, (2015). Biometric recognition: 10th Chinese conference, CCBR 2015 Tianjin, China, november 13-15, proceedings," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9428. 241–249.

[24] G. Arunalatha and M. Ezhilarasan (2015). Fingerprint spoof detection using quality features. *Int. J. Secur. its Appl.*, vol. 9(10), 83–94.

[25] Z. Xia, R. Lv, Y. Zhu, P. Ji, H. Sun, and Y. Q. Shi (2017). Fingerprint liveness detection using gradient-based texture features. *Signal, Image Video Process.*, vol. 11(2), 381–388.

[26] E. Park, W. Kim, Q. Li, H. Kim, and J. Kim (2016). Fingerprint liveness detection using CNN features of random sample patches: Liveness detection using CNN features. in *Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI)*, P-260.

[27] L. Ghiani, A. Hadid, G. L. Marcialis, and F. Roli(2017). Fingerprint liveness detection using local texture features. *IET Biometrics*, vol. 6(3), 224–231.

[28] A. Toosi, S. Cumani, and A. Bottino (2017). CNN patch-based voting for fingerprint liveness detection. in *IJCCI 2017 - Proceedings of the 9th International Joint Conference on Computational Intelligence,* 158–165.

[29] Bartunek and J. Strom (2005). Minutiae Extraction from Fingerprint with Neural Network and Minutiae based Fingerprint Verification Josef Str. om Bart ˚ ek," *Master's Thesis MEE*.