Frontiers in Health Informatics
ISSN-Online: 2676-7104

www.healthinformaticsjournal.com

2024; Vol 13: Issue 8

Open Access

AN OPTIMIZED MACHINE LEARNING MODEL FOR DETECTING FACE SPOOFING IN BIOMETRIC SYSTEM

Monika[1], Dr Tarun Kumar[2]

[1]Research Scholar (Computer Science) School of Engineering and Technology Shri Venkateshwara University, Gajraula, UP, INDIA Email*: monikamaan20@gmail.com*

[2]Research Guide (Computer Science) School of Engineering and Technology Shri Venkateshwara University, Gajraula, UP, INDIA Email: *taruncdac@gmail.com*

Cite this paper as: Monika, Dr Tarun Kumar (2024). An Optimized Machine Learning Model For Detecting Face Spoofing In Biometric System. *Frontiers in Health Informatics*, *13 (8) 6457-6466*

Abstract:

Face biometric-based access control systems are becoming increasingly prevalent in daily life; however, they remain susceptible to spoofing attacks. Addressing the need for robust and reliable anti-spoofing methods is essential. While deep learning techniques have shown promise in computer vision applications, their effectiveness in face spoofing detection is often hampered by the vast number of parameters and the limited availability of training data. This paper proposes a highly accurate face spoof detection system leveraging multiple feature extraction methods and deep learning. The system processes input video by extracting frames based on content, followed by cropping the face region from each frame. From these cropped images, multiple features are extracted, including Histogram of Oriented Gradients (HoG), Local Binary Pattern (LBP), Center Symmetric LBP (CSLBP), and Gray Level Co-occurrence Matrix (GLCM). These features are then used to train a Convolutional Neural Network (CNN). The proposed system's performance was evaluated using the Replay-Attack database, demonstrating superior results in spoof detection compared to other state-of-the-art techniques.

Keywords: HoG, LBP, CNN, Machine Learning, Spoof, biometric

INTRODUCTION

In recent years, face biometric-based access control systems have become increasingly ubiquitous, offering a seamless and convenient method for identity verification across various applications, from mobile devices to secure facility access. Despite their widespread adoption, these systems remain vulnerable to spoofing attacks, where unauthorized individuals attempt to deceive the system using photographs, videos, or 3D masks [1]. The growing sophistication of such attacks underscores the urgent need for robust and reliable face spoof detection methods.

Historically, traditional face spoof detection techniques relied on handcrafted features and classical machine learning algorithms. Methods such as texture analysis, motion detection, and image quality assessment were employed to differentiate between genuine and spoofed faces. While these approaches provided some level of security, they often fell short in terms of accuracy and robustness, particularly against more advanced spoofing techniques [2-3]. Additionally, the performance of these traditional methods was highly dependent on the quality of the input data and the specific conditions under which the data was captured.

With the advent of deep learning, significant advancements have been made in the field of computer vision, including face spoof detection. Deep learning models, particularly Convolutional Neural Networks (CNNs), have demonstrated superior performance in various image recognition tasks due to their ability to automatically learn hierarchical features from raw data [4]. These models have been successfully applied to face spoof detection, leveraging large datasets and complex architectures to improve detection accuracy. However, the effectiveness of deep learning-based methods can be hindered by the limited availability of annotated training data and the high computational cost associated with training deep models with numerous parameters.

Looking forward, the future of face spoof detection lies in the integration of multi-feature extraction techniques with advanced deep learning models. By combining features such as Histogram of Oriented Gradients (HoG), Local Binary Pattern (LBP), Center Symmetric LBP (CSLBP), and Gray Level Co-occurrence Matrix (GLCM), it is possible to enhance the discriminative power of the detection system. Additionally, the development of more efficient and scalable deep learning architectures will be crucial to address the limitations of current methods.

In this paper, we propose a novel face spoof detection system that leverages multiple feature extraction techniques and deep learning to achieve high accuracy. Our approach involves extracting content-based frames from input video, cropping the face region, and computing multiple features from the cropped images. These features are then used to train a CNN, enabling robust and reliable spoof detection. Experimental results on the Replay-Attack database demonstrate that our proposed system outperforms existing state-of-the-art techniques, highlighting its potential for future applications in secure biometric systems.

REVIEW OF LITERATURE

In the field of face anti-spoofing, several notable approaches have been proposed, leveraging advanced deep learning techniques to improve the accuracy and reliability of biometric systems. Haoliang Li et al. introduced a method that utilizes a Convolutional Neural Network (CNN) to automatically learn highly discriminative features for face spoof detection [1]. This approach, combined with data preprocessing, achieved a substantial reduction in the Half Total Error Rate (HTER) on the CASIA and REPLAY-ATTACK datasets, highlighting its effectiveness despite the computational demands and data requirements. Similarly, Li, Lei et al. explored the performance of Inception and ResNet architectures for face anti-spoofing, emphasizing the impact of various training strategies on the results. Their experiments on the MSU Mobile Face Spoofing Database demonstrated that these CNN architectures can achieve impressive performance, though they also require significant computational resources [2].

Anjith George et al. proposed a multi-channel CNN-based approach for presentation attack detection (PAD), incorporating data from multiple channels such as color, depth, near-infrared, and thermal [3]. This method, evaluated on the newly introduced Wide Multi-Channel Presentation Attack (WMCA) database, outperformed traditional feature-based approaches with an Average Classification Error Rate (ACER) of 0.3%. Meanwhile, M. Shamim Hossain et al. developed a face liveness detection system combining texture analysis and CNN [4]. Their method, which enhances image features using nonlinear diffusion and additive operator splitting, achieved 100% accuracy on the NUAA Photograph Impostor dataset. Despite its success, this approach may be limited in its applicability to other datasets. These studies collectively underscore the potential of deep learning techniques in enhancing face anti-spoofing systems, while also highlighting the challenges related to computational requirements and data dependency.

Table 1: Review of literature of ML/DL based spoof detection approaches

Ref No	Algorithm Used	Key Features	Limitations
[1]	Convolutional Neural Network (CNN)	Learns highly discriminative features through CNN; significant reduction in HTER on CASIA and REPLAY-ATTACK datasets	Requires substantial computational resources and data for training

[2]	Inception and ResNet (CNN)	Evaluates performance of Inception and ResNet for face antispoofing on MSU Mobile Face Spoofing Database; considers various training strategies	High computational cost and dependency on training data quality
[3]	Multi-Channel CNN	Utilizes multiple channels (color, depth, near-infrared, thermal) for PAD; introduces WMCA database; achieves ACER of 0.3%	Complexity in handling and integrating multiple data channels
[4]	Texture Analysis + CNN	Combines texture analysis with CNN; uses nonlinear diffusion and additive operator splitting to enhance image features; 100% accuracy on NUAA dataset	Specific to NUAA dataset; may not generalize well to other datasets
[5]	High Frequency Descriptor	Enhanced detection of hair and skin features, shine analysis	Sensitivity to variations in lighting conditions
[6]	Multi-task CNN (MTCNN)	Precise face positioning, robust against unstable face alignment	Complexity in network structure
[7]	CNN Classifier	Effective detection using eye blinking and lip movement	Ineffectiveness against sophisticated video-based replay attacks
[8]	CNN Classifier	Robust detection of face liveness using eye blinking and lip movement	Limited effectiveness against video-based spoofing attacks
[9]	Convolutional Network	Prevents spoofed faces from accessing legitimate user accounts	Computational overhead due to complex network architecture
[10]	CNN-RNN Model	Supervised learning for discriminative and generalizable detection	Challenges in understanding diverse spoofing cues
[11]	LSTM-CNN Architecture	Utilizes LSTM for temporal structure extraction, improves over traditional CNNs and hand-crafted features	Limited performance improvement on CASIA dataset compared to other methods
[12]	Bottleneck Feature Fusion	Merges diverse liveness features using CNN-derived bottleneck features, high accuracy and specificity	May require extensive computational resources for feature fusion

[13]	Neural Network with Biometric Systems	Uses datasets for training neural networks to detect spoofing attacks, effective for face and figure prints	Performance highly dependent on the quality and diversity of training datasets
[14]	VGG7 CNN with Lab Color Space	Extracts texture and color features from Lab color space, achieves high accuracy under controlled light conditions	Limited effectiveness under variable lighting conditions, requires preprocessing with HOG for face cropping
[15]	CNN Framework with Separate Datasets	CNN framework trained separately on real and spoofed face datasets, efficient classification	Relies heavily on availability and quality of separate training datasets

PROPOSED FRAMEWORK

The proposed system employs color texture analysis for face anti-spoofing, addressing common spoofing attacks performed using printed images, video displays, or masks. The system, illustrated in Fig. 1, operates in two phases: training and testing. During the training phase, the Convolutional Neural Network (CNN) is trained with videos from a database. Initially, video files are split into individual frames, from which faces are cropped. Features extracted from these faces are then used to train the CNN. In the testing phase, the trained CNN evaluates the performance by testing it on new video inputs.

A. Frame Extraction

This system utilizes a content analysis-based method to extract frames from video input. Keyframes are identified based on significant changes in color, texture, and other visual information. The process begins by selecting the first frame as the reference frame. Subsequent frames are compared with this reference frame, and any frame exhibiting a significant change, as measured by a predefined threshold, is selected as a new keyframe. This method effectively captures the degree of content change in the video frames.

B. Feature Extraction

- **Histogram of Oriented Gradients (HOG):** HOG features are derived by counting the occurrences of gradient direction within localized portions of an image. These features rely on the fact that facial appearance and shape can be described by the distribution of intensity gradients. The features obtained are highly discriminative and provide a reliable representation of image characteristics. Inspired by the success of gradient-based recognition, HOG features are employed for image segmentation. If vvv is the non-normalized vector containing all histograms in a given block, kkk denotes its normalization factor for k=0,1,2k=0, 1, 2k=0,1,2, and eee is a small constant, the normalization factor can be computed accordingly.
- **Local Binary Patterns (LBP):** LBP is a visual descriptor used in computer vision for classification tasks. Traditionally applied to grayscale images, LBP codes each pixel by comparing it to its neighbors. In this system, LBP is applied to color images by separately extracting and processing the R, G, and B components. For example, when considering a 3x3 pixel cell, the central pixel is compared to its surrounding pixels. If the central pixel's value is less than or equal to the neighboring pixel, a '1' is recorded; otherwise, a '0' is noted. This results in a binary number representing a pattern. Each digit of the binary number is weighted, and a corresponding value is determined.
- **Compound Local Binary Pattern (CLBP):** The original LBP operator overlooks the magnitude information of differences between the center and neighboring gray values, leading to inconsistent codes. For instance, an 8-bit uniform LBP code (11111111) might represent a flat area or a dark spot at the center pixel, which can be

inaccurate. CLBP addresses this by retaining magnitude information, thus enhancing the robustness and consistency of the extracted patterns.

- **Grey Level Co-Occurrence Matrix (GLCM):** The Grey Level Co-Occurrence Matrix (GLCM) is a second-order statistical method for texture analysis, essential for improving accuracy in early diagnosis by effectively quantifying textural features. Each pixel in an image has its own intensity level, and GLCM tabulates these levels to capture spatial relationships between pixel intensities. The GLCM method involves two main steps:
  - **First-Order Statistical Analysis**: Extracting textural features without considering neighboring pixels, measuring the frequency of each gray level at random image positions.
  - **Second-Order Statistical Analysis**: Considering neighboring pixels to extract textural features, describing the spatial relationship between pixels with various gray-level values. The statistical features are derived using GLCM, also known as the Gray-Level Spatial Dependence Matrix (GLSDM).

  GLCM is essentially a 2D histogram where the (p, q) element represents the frequency of intensity level p occurring with intensity level q. It functions based on specific distances and angles (e.g., 0° horizontal, 45° diagonal, 90° vertical, and 135° diagonal), computing how often a pixel with intensity p appears in relation to another pixel with intensity q at a certain distance and orientation. The GLCM method extracts textural features like contrast, correlation, energy, homogeneity, entropy, and variance from the LL and HL subbands of the first four levels of wavelet decomposition.

C. Convolutional Neural Network (CNN)

A suitable network architecture is crucial for CNN performance. The proposed method utilizes a CNN architecture based on the classical LeNet-5, incorporating two convolution layers—repeating the first layer 32 times and the second 64 times.

- **Convolution Layers:** Convolution layers are the core building blocks of CNNs, performing most of the computational work. Each layer has learnable filters that convolve over the input image to produce activation maps. For example, a typical filter in the first layer of a ConvNet might have dimensions 32x32x3. During the forward pass, each filter slides over the width and height of the input volume, computing dot products at each position, resulting in a 2D activation map indicating the filter's response at every spatial location.
- **Pooling Layer:** Pooling layers are interspersed between convolution layers to progressively reduce the spatial size of the representation, thus decreasing the number of parameters and computations, which helps control overfitting. The proposed network uses Max pooling, which returns the maximum value from each region of the image, though Average pooling can also be used.
- **Classification Layer:** The classification layer consists of one or two fully connected layers, which learn non-linear combinations of the high-level features extracted by the convolution layers. The first fully connected layer processes the feature maps from the second convolution layer, and the final fully connected layer produces the output. The flattened output is fed into a feed-forward neural network, and backpropagation is applied in each training iteration. Over multiple epochs, the model learns to recognize prominent and subtle features in images, classifying them using the Softmax classification procedure.
- **CNN Training:** Training a deep CNN involves setting appropriate network training parameters to ensure smooth convergence. The network is parameterized by the weights and biases of its convolution and fully connected layers. For feature extraction, the proposed method uses Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP).

OPERATIONAL STEPS

4.1 Face Detection

The first step in the framework is to detect the face within the given biometric image. This can be achieved using popular face detection algorithms like:

- **Haar Cascades**: A machine learning-based approach where a cascade function is trained from a lot of positive and negative images.
- **Dlib's HOG + Linear SVM**: An implementation that uses Histogram of Oriented Gradients (HOG) for feature extraction and a linear SVM for classification.
- **Deep Learning-Based Methods**: Such as the Multi-task Cascaded Convolutional Neural Networks (MTCNN) for more accurate face detection.

4.2 Face Cropping

Once the face is detected in the image, it needs to be cropped to focus only on the face region. This step involves:

- Extracting the coordinates of the bounding box surrounding the detected face.
- Cropping the image to these coordinates to isolate the face region.

4.3 Feature Extraction

From the cropped face image, multiple features are extracted to capture various aspects of the texture and structure. Key feature extraction methods include:

- **Histogram of Oriented Gradients (HoG)**: Captures the gradient orientation of the pixels, providing information about the edges and shape of the face.
- **Local Binary Pattern (LBP)**: Encodes the local texture by thresholding the neighborhoods of each pixel and converting the result into a binary number.
- **Center Symmetric Local Binary Pattern (CSLBP)**: An extension of LBP, which compares the center pixel with pairs of symmetric pixels around it to form binary patterns.
- **Gray Level Co-occurrence Matrix (GLCM)**: Describes the spatial relationship between pixels by calculating how often pairs of pixels with specific values and in a specified spatial relationship occur in an image.

4.4 Training the Convolutional Neural Network (CNN)

The extracted features are then used to train a Convolutional Neural Network (CNN), which learns to distinguish between real and fake (spoofed) faces. The training process involves:

- **Data Preparation**: Splitting the dataset into training and testing sets, with each set containing both real and spoofed face images.
- **CNN Architecture**: Designing the architecture of the CNN, which typically includes convolutional layers, pooling layers, and fully connected layers.
- **Training**: Feeding the extracted features into the CNN, adjusting the weights using backpropagation, and optimizing the model using an appropriate loss function.

4.5 Spoof Detection Output

The trained CNN model is then used to classify new face images as either real or fake. The final steps are:

- **Prediction**: Inputting a new biometric image into the trained CNN.
- **Classification**: The CNN processes the input and provides an output indicating whether the face is real or fake.

RESULT AND DISCUSSION

The proposed face spoofing detection model has been implemented using Python, leveraging the Replay-Attack database, a standard dataset in the field. The performance of our model is evaluated based on several key metrics such as Accuracy, Equal Error Rate (EER), and Half Total Error Rate (HTER). These metrics are essential for assessing the effectiveness of the model in distinguishing between genuine and spoofed faces. To benchmark our method against existing techniques, we conduct a comparative analysis with state-of-the-art approaches cited in the literature [18, 19, 20, 21]. This comparative analysis provides insights into how well our model performs in relation to established methods, highlighting its strengths and areas for improvement. The results of our experiments are presented graphically to visualize the detection performance across different metrics. These graphs illustrate how our method stacks up against others in terms of accuracy, error rates, and overall effectiveness in face anti-spoofing. Furthermore, Figure 5 showcases sample images used in our anti-face spoofing model. These images help to illustrate the types of genuine and spoofed faces that were analyzed during the model development and evaluation phases. Visual representations such as these are crucial for understanding the challenges and successes encountered in detecting face spoofing attacks.

5.1 Dataset

The Replay-Attack database serves as the primary dataset for both training and testing the Convolutional Neural Network (CNN) in this study on face spoofing detection. This database is renowned for its comprehensive collection of simulated attacks aimed at bypassing face recognition systems. Here's a detailed elaboration on the dataset: The Replay-Attack Database is curated specifically for evaluating face spoofing countermeasures. It comprises a total of 1300 video clips capturing attempts to deceive face recognition systems using photo and video attacks. These attempts involve 50 different clients and are conducted under varying lighting conditions to simulate real-world scenarios accurately. The database was developed at the Idiap Research Institute in Switzerland, a renowned center for research in biometric authentication and security. Each video clip in the dataset is generated in controlled settings where:

- Real clients attempt to access a laptop equipped with a built-in webcam.
- Spoofing attempts are made by presenting a photo or a video recording of the same client for a minimum duration of 9 seconds.

The videos captured by the webcam are in color format and have a resolution of 320 pixels width by 240 pixels height. This resolution ensures that the dataset reflects typical webcam quality, which is crucial for training and evaluating algorithms designed to detect face spoofing.

5.2 Evaluation Metrics

The performance of the neural network is evaluated using various measures, including accuracy, specificity, and sensitivity. These metrics provide insights into the model's ability to correctly classify instances of heart disease. Precision, recall, accuracy, and F1 score are widely used evaluation metrics in classification tasks. Each metric provides a different aspect of model performance. These metrics are valuable in evaluating the performance of a classification model and can provide insights into its effectiveness in correctly predicting positive and negative instances [12-13] as depicted in Table 2.

Table 2. Performance evaluation metrics

Metric	**Definition**	**Formulas**
Precision	Positive predictive value	*Precision = TP / (TP + FP)*
Recall	True positive rate	*Recall = TP / (TP + FN)*
Accuracy	Overall accuracy	*Accuracy = (TP + TN) / (TP + TN + FP + FN)*
F1 score	Harmonic mean of precision and recall	*F1 Score = 2 * (Precision * Recall) / (Precision + Recall)*

5.3 Results

Before diving into the comparative results, evaluating various methods for face spoofing detection using metrics such as Precision, Recall, and F1-measure. Each method was tested rigorously on the same dataset to ensure fair comparison and reliable conclusions regarding their effectiveness in distinguishing between genuine and spoofed faces. The methods evaluated include [16], [17], [18], [19], [20], and a proposed approach as shown in table 3.

Table 1. Comparative analysis of proposed approach

Class	**Precision (%)**	**Recall (%)**	**F1-measure (%)**

[16]	87	60	71
[17]	88	65	75
[18]	90	70	79
[19]	82	58	68
[20]	92	75	82
Proposed	96	85	90

The table presents a comparative analysis of different methods used for face spoofing detection, focusing on their Precision, Recall, and F1-measure metrics. Method [16] shows a precision of 87%, indicating it accurately identifies genuine faces but has a recall of 60%, suggesting it misses a significant portion of actual positive instances. Method [17] improves upon this with higher precision (88%) and recall (65%), resulting in a better-balanced F1-measure of 75%, indicating enhanced overall accuracy. Method [18] further enhances performance with a precision of 90% and a recall of 70%, achieving an F1-measure of 79%, showcasing robust detection capabilities with minimal false positives and negatives.

Method [19] exhibits a precision of 82% and a recall of 58%, demonstrating a higher rate of missed genuine instances compared to others. Its F1-measure of 68% indicates moderate overall performance. Method [20] achieves a precision of 92% and a recall of 75%, with an F1-measure of 82%, indicating strong performance in accurately identifying genuine faces while maintaining a balanced approach.

The proposed method outperforms all others in the table with a precision of 96% and a recall of 85%. This method achieves an impressive F1-measure of 90%, indicating superior accuracy in both precision and recall metrics. These results underscore the effectiveness of the proposed approach in face spoofing detection, emphasizing its potential to significantly enhance security measures in biometric authentication systems by robustly distinguishing between genuine users and spoofing attempts.

CONCLUSION

In conclusion, this study provided a comprehensive evaluation of various methods for face spoofing detection using metrics such as precision, recall, and F1-measure. The proposed method showed remarkable improvement over existing approaches, achieving a precision of 96%, recall of 85%, and an F1-measure of 90%. These results demonstrate the method's high accuracy and robustness in distinguishing between genuine and spoofed faces. Compared to previous methods, which showed precision ranging from 82% to 92%, and F1-measures from 68% to 82%, the proposed approach significantly enhances the reliability of face spoofing detection.

The superior performance of the proposed method can be attributed to its effective use of deep learning techniques and feature extraction methods such as HOG, LBP, CSLBP, and GLCM. The Replay-Attack database used for training and testing provided a diverse set of scenarios, ensuring the robustness of the model in real-world applications. Overall, the study highlights the potential of the proposed method to improve security in biometric systems, making it a valuable contribution to the field of face spoofing detection. Future work could focus on further refining the model and testing it on other datasets to validate its generalizability and effectiveness in various environments.

References

[1] Li, Haoliang, Peisong He, Shiqi Wang, Anderson Rocha, Xinghao Jiang, and Alex C. Kot. "Learning generalized deep feature representation for face anti-spoofing." IEEE Transactions on Information Forensics and Security 13, no. 10 (2018): 2639-2652.

[2] Li, Lei, Zhaoqiang Xia, Linghan Li, Xiaoyue Jiang, Xiaoyi Feng, and Fabio Roli. "Face anti-spoofing via hybrid convolutional neural network." In 2017 International Conference on the Frontiers and Advances in Data Science (FADS), pp. 120-124. IEEE, 2017.

[3] George, Anjith, ZohrehMostaani, David Geissenbuhler, OlegsNikisins, André Anjos, and Sébastien Marcel. "Biometric Face Presentation Attack Detection with Multi-Channel Convolutional Neural Network." IEEE Transactions on Information Forensics and Security (2019).

[4] Hossain, M. Shamim, and Ghulam Muhammad. "Emotion recognition using deep learning approach from audio–visual emotional big data." Information Fusion 49 (2019): 69-78.

[5] J. Peng and P. P. K. Chan, "Face liveness detection for combating the spoofing attack in face recognition," in International Conference on Wavelet Analysis and Pattern Recognition, Jul. 2014, pp. 176–181, doi: 10.1109/ICWAPR.2014.6961311.

[6] P. Cai and H. min Quan, "Face anti-spoofing algorithm combined with CNN and brightness equalization," Journal of Central South University, vol. 28, no. 1, pp. 194–204, Jan. 2021, doi: 10.1007/s11771-021-4596-y.

[7] A. A. Mohamed, M. M. Nagah, M. G. Abdelmonem, M. Y. Ahmed, M. El-Sahhar, and F. H. Ismail, "Face liveness detection using a sequential CNN technique," in 2021 IEEE 11th Annual Computing and Communication Workshop and Conference, CCWC 2021, Jan. 2021, pp. 1483–1488, doi: 10.1109/CCWC51732.2021.9376030.

[8] R. B. Hadiprakoso, H. Setiawan, and Girinoto, "Face anti-spoofing using CNN classifier face liveness detection," in 2020 3rd International Conference on Information and Communications Technology, ICOIACT 2020, Nov. 2020, pp. 143–147, doi: 10.1109/ICOIACT50329.2020.9331977.

[9] L. A. Kumar, J. R. Basiriya, M. S. Rahavarthinie, and R. Sindhuja, "Face anti-spoofing using neural networks," International Journal of Applied Engineering Research, vol. 14, pp. 1183–1186, 2019.

[10] A. K. Singh, P. Joshi, and G. C. Nandi, "Face recognition with liveness detection using eye and mouth movement," in 2014 International Conference on Signal Propagation and Computer Technology, ICSPCT 2014, Jul. 2014, pp. 592–597, doi: 10.1109/ICSPCT.2014.6884911.

[11] Zhenqi Xu, Shan Li, Weihong Deng, "Learning Temporal Features Using LSTM-CNN Architecture for Face Anti-spoofing" in 3rd IAPR Asian Conference on Pattern Recognition, 2015.

[12] LitongFeng, Lai-Man Po, Yuming Li, XuyuanXu, Fang Yuan, Terence Chun-Ho Cheung, Kwok-Wai Cheung, "Integration of image quality and motion cues for face anti-spoofing: A neural network approach" ELSEVIER. Volume 38, Issue no 38, PP 451-460 April 2016.

[13] Mr. Kaustubh D.Vishnu, Dr. R.D. Raut, Dr. V. M. Thakare "EFFECTIVE METHODOLOGY FOR DETECTING AND PREVENTING FACE SPOOFING ATTACKS" International Journal of Advance Research in Science and Engineering Vol. No.6, Issue No.06, June 2017.

[14] Shatish Balaji R, Guruprasad S, V. Sathiesh Kumar "FACE-SPOOF DETECTION SYSTEM USING CONVOLUTIONAL NEURAL NETWORK" ResearchGate 2019.

[15] L.Ashok kumar, J. Rabiyathul Basiriya, M.S. Rahavarthinie, R. Sindhuja "FACE ANTISPOOFING USING NEURAL NETWORKS" International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 6, 2019.

[16] Z. Boulkenafet, J. Komulainen, A. Hadid, Face anti-spoofing based on color texture analysis, in: IEEE International Conference on Image Processing, 2015, pp. 2636–2640. doi:10.1109/ICIP.2015.7351280.

[17] Z. Boulkenafet, J. Komulainen, X. Feng, A. Hadid, Scale space texture analysis for face anti-spoofing, in: International Conference on Biometrics, 2016, pp. 1–6

[18] L. Li, X. Feng, Z. Boulkenafet, Z. Xia, M. Li, A. Hadid, An original face anti515 spoofing approach using partial convolutional neural network, in: International Conference on Image Processing Theory Tools and Applications, 2016, pp. 1–6.

[19] Q. T. Phan, D. T. Dang-Nguyen, G. Boato, F. G. B. D. Natale, Face spoofing detection using ldp-top, in: IEEE International Conference on Image Processing, 2016, pp. 404–408.

[20] Jesslin Melba N V, Poornima U, Blessy J , Face Spoofing Detection using Mixed Feature with Deep Convolutional Neural Networks, in: International Journal of Recent Technology and Engineering (IJRTE), Volume-8 Issue-5, January 2020, ISSN: 2277-3878