

VeriFace: Face Recognition Engine

Nayan Patel*, Sailee Gayke**, Prof. Ruchita Sharma***

*(Computer Science Department, MIT ADT University, Pune
Email: nayannpatel2003@gmail.com)

** (Computer Science Department, MIT ADT University, Pune
Email: sailee.gayke2003@gmail.com)

*** (Computer Science Department, MIT ADT University, Pune
Email: ruchita.sharma@mituniversity.edu.in)

Abstract:

Face liveness detection systems are commonly trained on real-world facial images, where genuine faces and spoof attempts often appear quite similar. However, there has been limited research on identifying live faces using a blend of real facial photos and synthetic images generated by deep convolutional neural networks (CNNs). As facial recognition becomes more widely used for biometric authentication, it's crucial that these systems not only recognize genuine users but also resist spoofing attempts—such as using printed images, digital displays, or replayed videos.

To enhance spoof prevention, this approach introduces a liveness detection method that integrates a CNN-based classifier with an activity-based analysis module. Specifically, it monitors facial behaviors like eye blinking and lip movement to detect signs of life. These features are essential because conventional spoofing methods struggle to replicate such subtle, spontaneous actions.

The solution is built using Python and combines two modules: one powered by CNN for feature classification, and another that tracks eye and mouth movements. Together, they form a robust defense against various spoofing attacks, including those involving posters, silicone masks, or smartphone screens.

Moreover, the study proposes an adaptive fusion layer that harmonizes convolutional features from both real faces and CNN-generated synthetic faces during the training process. This fusion enables the model to generalize better across different scenarios.

Extensive evaluations using prominent face anti-spoofing datasets—such as CASIA, OULU, and Replay-Attack—demonstrate that the proposed method achieves superior performance in both intra-database and cross-database testing environments, outperforming several existing state-of-the-art approaches.

Keywords- Face Recognition, Convolutional Neural Network (CNN), Liveness Detection, Face Anti-Spoofing, Biometric Authentication

I. Introduction

In the rapidly evolving digital era, biometric authentication has emerged as a secure and user-friendly method for verifying identity. Among various biometric modalities, facial recognition has gained prominence due to its ease of use and high precision. From unlocking smartphones to verifying identities at airports, facial recognition is now integrated into many aspects of everyday life. However, this growing adoption has also made it a prime target for spoofing attacks, where attackers attempt to deceive the system using printed photos, high-definition videos, or even sophisticated 3D masks—often sourced from social media profiles, making the

threat more accessible than with other biometric data like fingerprints or iris scans.

Spoofing attempts can generally be classified into two major types: **static attacks**, which involve still images or masks, and **dynamic attacks**, which use video replays. To counter these threats, **liveness detection** techniques have been introduced to determine whether the detected face belongs to a live person. For example, detecting involuntary eye blinks—typically occurring 5 to 10 times per minute and lasting approximately 250–300 milliseconds—can help differentiate a real user from a static spoof. Lip movement detection is another dynamic cue that helps verify liveness. However, such behaviors can sometimes be mimicked using high-quality video replays,

limiting the effectiveness of these methods when used in isolation.

To strengthen anti-spoofing defenses, **challenge–response mechanisms** are often employed, requiring users to perform random facial gestures like turning their head, smiling, or blinking on command. Although effective, these methods can reduce user convenience and may not be suitable for all scenarios. More advanced solutions, including the use of **3D imaging** which tracks subtle blood flow changes—offer improved accuracy by leveraging depth and texture information. However, these technologies require specialized hardware and are often incompatible with standard mobile or embedded camera systems.

Another class of anti-spoofing techniques involves analyzing **facial texture and reflectance properties** using hand-crafted features like **Local Binary Patterns (LBP)**, **Histogram of Oriented Gradients (HOG)**, or **color space variations** (RGB, HSV). These are often combined with traditional classifiers such as **Support Vector Machines (SVM)** or **k-Nearest Neighbors (k-NN)**. While these approaches can be effective in controlled environments, they tend to degrade in performance under varying lighting conditions or poor camera quality, limiting their robustness in real-world settings.

With the advancement of **deep learning**, particularly **Convolutional Neural Networks (CNNs)**, more adaptive and data-driven solutions have emerged. CNNs can automatically learn discriminative features from large datasets, capturing subtle differences between live and spoofed faces that hand-crafted methods may miss. Their ability to generalize across different spoofing techniques and environmental conditions makes them suitable for real-time, scalable security systems.

This work proposes a comprehensive anti-spoofing framework that combines the strengths of CNN-based classification with **behavioral liveness cues** such as **eye blinking** and **lip movement** to improve detection accuracy. Furthermore, the system introduces a **facial region segmentation** approach—dividing the face into zones like the eyes, nose, lips, and forehead—to enhance both spoof detection and **criminal identification**. This modular breakdown aids law enforcement in eyewitness-based searches by allowing users to focus on specific identifiable features, improving search relevance and reducing false positives.

Overall, the proposed system addresses the limitations of existing techniques by integrating multiple layers of detection, offering a robust, real-time solution for both personal authentication and broader surveillance applications.

II. Literature Survey

- Arpita Nema et al. [1] A desktop application was designed to detect liveness using the blink of an eye. The system utilizes a camera to regularly capture images and monitor eye blinks, helping to distinguish real users from spoofing attempts using photos or videos. If no blinking is detected, the system either logs out the user or captures an image of the person for security purposes. This approach enhances face recognition systems by adding a simple yet effective liveness check.
- Mehmet Killioglu et al. [2] A liveness detection technique was developed that uses pupil tracking to enhance anti-spoofing in face recognition systems. The process begins with detecting the eye region using a Haar-Cascade Classifier, followed by the Kanade-Lucas-Tomasi (KLT) algorithm to stabilize and track the pupil area. An Arduino-controlled LED prompts the user to look in specific directions, and the system checks if the pupil follows the LED, ensuring that a live person is interacting with the system.
- Yuming Li et al. [3] A unified approach for face liveness detection and recognition based on shearlet transform features was introduced. These features are extracted and processed using stacked autoencoders along with a softmax classifier to identify spoofing. The method is tested on the CASIA Face Anti-Spoofing database, showing strong performance in distinguishing real faces from fake ones, like printed photos or screen displays.
- Junyan Peng et al. [4] A technique for liveness detection using high-frequency descriptor analysis was proposed. The method compares facial images under different lighting conditions—one with additional illumination and one without. The energy difference in high-frequency components between these images helps to distinguish real faces from spoofing mediums like photos or digital displays. This enhances the system's reliability in detecting liveness.
- C. Yuan et al. [5] A fingerprint liveness detection method was developed using Deep Convolutional Neural Networks (DCNNs) and Deep Residual Networks (DRNs). Optimization challenges in training deep networks were addressed by proposing adaptive residual structures that monitor parameter stability. A Region of Interest (ROI) extraction method and Local Gradient Pattern (LGP) feature analysis are used to improve texture recognition, making the model more robust in detecting live fingerprints.

III. Proposed Methodology

This research introduces a multi-module anti-spoofing model that comprises three core components: face spoof detection, liveness detection, and criminal identification using a Convolutional Neural Network (CNN) classifier. The workflow of this model is streamlined and sequential. Initially, the face spoof detection module processes the input image or

video to identify fraudulent attempts such as printed photos, masks, or digital displays. Upon detecting a face, the system passes the data to the CNN classifier, which categorizes the input as genuine or spoofed.

Subsequently, the input undergoes liveness verification, where two distinct signs of life—eye blinks and lip movements—are analyzed. If both the spoof detection and liveness modules validate the input, the system confirms it as a real face. The process then advances to the final module: criminal identification. This stage uses datasets like CASIA, OULU, NUAA, and Replay-Attack to match the verified face against a database of known individuals, determining if the person has a criminal profile.

The development of CNN-based classifier modules follows a series of common steps: data acquisition, data preprocessing, model training, model validation, and testing. The liveness detection module includes two sub-components—blink detection and lip motion analysis.

For detecting lip movement, the system incorporates the lip-movement-net framework [22], which relies on a Recurrent Neural Network (RNN) algorithm. This model assesses one second of video footage to detect speech by tracking the distance between the upper and lower lips. It uses a filtering mechanism to identify lip locations and evaluates motion using real-time video or camera input.

Blink detection utilizes an approach derived from earlier studies [23], where a filter is applied to locate the eye region in the input image. After identifying the eye area, the system determines eye openness using a classifier that evaluates the probability of the eye being open. This method calculates the variation between maximum and minimum openness across frames. A significant variation implies a blink, indicating the presence of a live person. To train this module, separate datasets for open-eye and closed-eye images were compiled.

Data collection involves capturing face images or video streams through cameras or similar devices. These data are then preprocessed through operations such as cropping, resizing, and normalization to ensure suitability for training the deep learning models.

For performance evaluation, statistical analysis is conducted using metrics like true positive rate, false positive rate, precision, and recall. Additionally, Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) values may be employed to measure the effectiveness and reliability of the neural network in classification tasks.

A. Face Liveness Detection:

Face recognition is a biometric authentication approach that identifies individuals by comparing their facial features with

those stored in a known database. Over time, researchers have introduced numerous techniques to enhance facial recognition, tackling challenges like varying facial expressions, angles, and lighting conditions. This technology has gained widespread use in the past decade and is now applied in areas such as attendance tracking, secure transactions, and mobile device login systems [1]. It is also commonly utilized in forensic investigations and secure access control [2].

However, one major issue with facial recognition is face spoofing—where attackers use methods such as printed photographs, pre-recorded videos, or 3D masks to deceive recognition systems. Figure 1 illustrates these spoofing techniques, which allow unauthorized individuals to bypass facial security without the subject’s consent.

To address this vulnerability, face liveness detection has emerged as a crucial countermeasure. This technology determines whether the face presented to the system belongs to a living person, adding an extra layer of defense. Unlike passwords or fingerprints, face liveness detection is relatively recent but increasingly necessary, especially for businesses aiming to secure their systems against impersonation using synthetic face media. It is vital for detecting and blocking spoofing attempts using static images or realistic masks, ensuring only genuine, live users gain access.

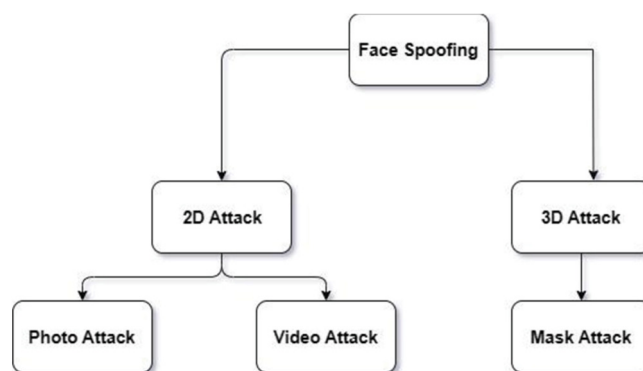


Figure 1: Face Spoofing Types

B. Face Recognition System:

Modern face recognition systems face growing threats from attackers who exploit images, videos, or 3D masks to fool identity verification processes. While facial recognition accurately identifies a person’s face, it lacks the capability to distinguish between live and fake inputs, making it susceptible to spoofing threats.

To mitigate these risks, face anti-spoofing technologies are employed. These tools are specifically designed to verify the

authenticity of a presented face, ensuring that it originates from a real, live individual. Developing a robust and reliable anti-spoofing mechanism is critical to protecting facial recognition systems from malicious exploitation. High accuracy and strong generalization across diverse environments are essential features for such solutions to be effective in maintaining secure and trustworthy authentication systems.

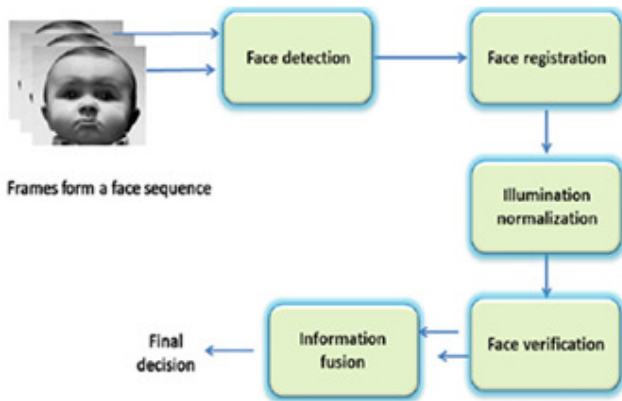


Figure 2: Face Recognition System

C. Criminal Identification:

The rising crime rates and the growing number of offenders have significantly heightened security concerns. Law enforcement agencies face critical challenges in both deterring criminal activities and accurately identifying suspects. Despite their responsibility to safeguard life and property, police departments often operate with limited personnel, making crime management more difficult.

To address this issue, an automated facial recognition system is proposed that leverages a popular Convolutional Neural Network (CNN) classifier. This approach enables real-time face detection and recognition from a criminal database, thereby assisting authorities in faster and more efficient identification. Accurately detecting facial features remains a complex task; however, methods like the Viola-Jones algorithm are widely adopted for recognizing faces and other objects in images. Tools such as OpenCV offer accessible, community-supported face detection classifiers that enhance the development process.

This system is designed to deliver a comprehensive solution for facial identification and recognition based on images. It emphasizes improved detection precision, faster response times, and serves as a foundational step for deploying video-based surveillance systems in law enforcement applications.

D. Deep Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are a widely used type of deep learning model, especially in the fields of image classification and pattern recognition. They play a key role in tasks such as object detection, facial recognition, and emotion analysis. A CNN works by taking an input image and processing it through several layers to classify or identify features within the image.

CNN stands for a neural network that includes one or more convolutional layers, which are specifically designed to automatically detect patterns like edges, shapes, and textures in images.

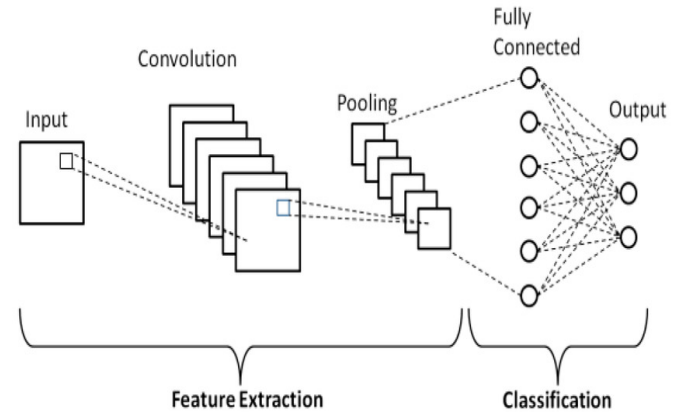


Figure 3: Basic Architecture of a CNN

[1] Pseudocode of the CNN Algorithm:

- Step 1:** Load the dataset, which contains input images and corresponding reference labels or frames.
- Step 2:** Import required libraries (e.g., TensorFlow, Keras, NumPy) and define the CNN model structure.
- Step 3:** Use convolutional layers to scan each image pixel-by-pixel and extract key visual features.
- Step 4:** Convert the image data into matrix form (of size $M \times N$) for easier processing.
- Step 5:** Apply max pooling to the matrix—this operation selects the highest value from each region, reducing dimensionality while keeping important information.
- Step 6:** Normalize the data by converting all negative values to zero, ensuring that the input to the next layers remains positive.
- Step 7:** Use Rectified Linear Unit (ReLU) activation, which further filters the data by replacing any negative value with zero and allowing positive values to pass through.
- Step 8:** Assign weights to the inputs from the visible layer. These weights are adjusted to maximize the likelihood of correctly predicting the output in the hidden layers.

IV. Contributions of the Proposed Work

- 1. This research aims to mitigate spoofing attacks by analyzing facial liveness indicators such as eye

blinks and lip movements. However, traditional methods fall short against video-based replay attacks. To address this, the proposed approach integrates a Convolutional Neural Network (CNN) classifier with advanced face liveness detection mechanisms.

2. A comprehensive performance evaluation will be conducted to assess the system's effectiveness in countering spoofing attacks, with experiments carried out on both intra-database and cross-database scenarios.
3. The third module focuses on criminal identification, utilizing CNN-based facial recognition techniques to aid in crime prevention and the identification of offenders.
4. The proposed system is also designed to enhance both the accuracy and overall performance of face liveness detection in real-world applications.

V. Performance Analysis

In studies like this, performance evaluation generally focuses on how accurately the deep neural network distinguishes between real and spoofed faces. This is usually assessed through metrics such as true positive rate (TPR), false positive rate (FPR), precision, recall, and the F1-score. TPR represents the proportion of actual genuine faces that are correctly recognized, whereas FPR indicates the proportion of spoofed faces that are mistakenly classified as genuine.

Beyond these common metrics, some analyses also include additional indicators like the area under the ROC (Receiver Operating Characteristic) curve or the Detection Error Tradeoff (DET) curve. These help provide a broader understanding of model performance by considering the balance between TPR and FPR across various threshold settings.

While the exact performance outcomes may differ depending on the task and the neural network architecture used, most studies follow similar evaluation methods. They rely on standard accuracy metrics and often supplement them with ROC or DET curves for a deeper performance insight.

Acknowledgement

I would like to sincerely thank the researchers and publishers for making their resources available. I am also grateful to my guide and the reviewers for their valuable suggestions and support throughout this work.

References

[1] A. Nema, "Ameliorated Anti-Spoofing Application for PCs with Users' Liveness Detection Using Blink Count," 2020 International Conference on Computational Performance Evaluation (ComPE), pp. 311-315, July 2020.

[2] R. B. Hadiprakoso, H. Setiawan and Girinoto, "Face Anti-Spoofing Using CNN Classifier & Face liveness Detection," 2020 3rd International Conference on Information and Communications Technology (ICOIACT), (Yogyakarta, Indonesia November 2020), pp. 143-147

[3] Y. Li, L. Po, X. Xu, L. Feng and F. Yuan, "Face liveness detection and recognition using shearlet based feature descriptors," 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), (Shanghai, China, March 2016), pp. 874-877.

[4] A. K. Singh, P. Joshi and G. C. Nandi, "Face recognition with liveness detection using eye and mouth movement," 2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014), (Ajmer, India, July 2014), pp. 592-597

[5] F. Ullah et al., "Cyber Security Threats Detection in Internet of Things Using Deep Learning Approach," in IEEE Access, Vol. 7, pp. 124379-124389, August 2019.

[6] A. Kumar T.K., R. Vinayakumar, S. Variyar V.V., V. Sowmya and K. P. Soman, "Convolutional Neural Networks for Fingerprint Liveness Detection System," 2019 International Conference on Intelligent Computing and Control Systems (ICCS), (Madurai, India, Ma 2019), pp. 243-246

[7] P. Zhang et al., "FeatherNets: Convolutional Neural Networks as Light as Feather for Face Anti-Spoofing," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), (Long Beach, CA, USA, June 2019), pp. 1574-1583

[8] W. Jian, Y. Zhou and H. Liu, "Densely Connected Convolutional Network Optimized by Genetic Algorithm for Fingerprint Liveness Detection," in IEEE Access, Vol. 9, pp. 2229-2243, December 2021.

[9] R. F. Nogueira, R. de Alencar Lotufo and R. Campos Machado, "Fingerprint Liveness Detection Using Convolutional Neural Networks," in IEEE Transactions on Information Forensics and Security, Vol. 11, no. 6, pp. 1206-1213, June 2016.

[10] T. Alipourfard, H. Arefi and S. Mahmoudi, "A Novel Deep Learning Framework by Combination of Subspace Based Feature Extraction and Convolutional Neural Networks for Hyperspectral Images Classification," IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, pp. 4780-4783, July 2018.