Enhancing Face Recognition Security with CNN-Based Liveness Detection

Wan-Xuan Kow2, a), Siew-Chin Chong1, 2, b), Lee-Ying Chong1, 2, c), Kuok-Kwee Wee1, 2, d)

1Centre for Advanced Analytics, CoE for Artificial Intelligence, Multimedia University, Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia.

2Faculty of Information Science & Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia.

*b)* *Corresponding author:* [*chong.siew.chin@mmu.edu.my*](mailto:chong.siew.chin@mmu.edu.my)

*a) 1211307650@student.mmu.edu.my*

*c) lychong@mmu.edu.my*

*d) wee.kuok.kwee@mmu.edu.my*

**Abstract—** Face recognition Face recognition is among the most popular biometric authentication systems in use, as it is convenient, non-intrusive, and highly accurate. Nevertheless, increasing complexity of spoofing attacks, including printed photographs, video replay attacks and 3D masks, makes it insecure, breaking its reliability. Liveness detection is essential in traditional face recognition systems because they do not typically have an effective anti-spoofing system. This research paper suggests an improved face recognition system with eye-blinking liveness detection based on a CNN-DenseNet model. It is trained and tested on the ROSE-Youtu Face Liveness Detection Database to differentiate live face inputs and spoofing attempts. Dense connections in DenseNet result in better feature reuse and gradient propagation, resulting in increased detection of spoofing artifacts with efficiency and robustness compared to the benchmark method in this paper, ResNet.

# Introduction

The security of the traditional authentication methods such as passwords and PINs is still at risk of attacks, stimulating the wider use of facial recognition systems. Nevertheless, these biometric systems are vulnerable to photo, videos, and 3D mask spoofing attacks [1,2]. Convolutional Neural Networks (CNNs) have been used to increase the recognition accuracy, but they are not effective in high-security scenarios due to the lack of effective liveness detection. The current ongoing project solves these drawbacks by combining eye-blink detection and behavioral features with CNN-based systems to perform live user vs. spoofing attack classification [3], and additionally, using preprocessing and data augmentation tricks to keep high accuracy in varying lighting conditions and angles. The improved system shows specific potential to be used in banking and healthcare, where authenticating the user is especially important, taking the ease of use of facial recognition and enhancing it through increased anti-spoofing protections to produce a more secure biometric system.

# Literature review

## Convolutional-Neural Network (CNN)

A convolutional neural network (CNN) consists of several layers arranged in a sequential way. The central component of a CNN is the Convolutional Layer that consists of a number of convolutional filters, or kernels. The Convolutional Layer has a number of filters or kernels which are matrices with numerical values. These take the input image or feature map of the previous layers and combine them to give a new feature map. One of the basic tasks of this layer is the convolution, i.e. the multiplication of two matrices and the addition of their elements but preserving the spatial relations between pixels with the help of small receptive fields. The filters are connected to particular parts of the input called receptive fields and they adjust in the learning process to better extract features [4].

The size of the result produced by the convolutional layer depends on a few parameters, such as depth (number of filters), stride (displacement made by each filter) and zero padding (which enables spatial manipulations). The result of this convolution process yields an activation map that is subsequently passed on to later layers for further analysis, enabling the CNN to learn and capture hierarchical patterns inherent in the data.

## Face Spoofing Attack Method (FSA Method)

Face spoofing attacks are classified into static, dynamic, and 3D demonstration attacks [3]. Static attacks use non-interactive artifacts like photos or masks, while dynamic attacks involve motion, such as video replays or sequences of images, to mimic real facial behaviour. 3D demonstration attacks are more advanced and expensive, using realistic three-dimensional face replicas, like sculptures, to deceive recognition systems.

1. Static 2D Demonstration Attacks: Use printed photographs or 2D masks (flipping, rotating, or bending the photo) to spoof systems. Example: printed photo, 2D display mask.
2. Dynamic 2D Demonstration Attacks: Employ video replays or AI-generated content to mimic natural movements, increasing detection difficulty. Example: replay videos, AI-generated videos.
3. 3D Demonstration Attacks: Use silicone/plastic masks/sculpture with depth and texture that duplicates the face and circumvents 2D-based defenses. Examples: 3D masks, sculptures of faces.

Although static (printed photos) and dynamic (video replays) 2D attacks are trivial to construct with ordinary devices, 3D masks attack is harder since it is more complex, expensive, and needs specialized craftsmanship [5]. Such a threat scenario warrants countermeasures, especially liveness detection systems, which we discuss in the next section.

## Face Liveness Detection Techniques (FLD Techniques)

Face presentation attack detection (Face Liveness Detection), is an important security countermeasure used in facial recognition systems to distinguish between real human faces and spoofed faces (photos, videos, 3D masks, etc.). FLD operates by examining the facial features and behaviors in a number of major ways:

1. Feature Extraction – Analysis of features such as skin texture, depth, or thermals signals.
2. Behavioural Analysis – Observation of natural behavioral responses (blinking, head movement).
3. Challenge Response Mechanisms – Requesting the user to undertake actions (e.g., smile, blink) to prove liveness.

Depth-based approaches shed light on 2D attacks and are ineffective against 3D masks. Behavioral cues (e.g., blinking, lip movement) together with CNNs can enhance detection under complicated spoofing attacks. Major techniques; feature extraction, behavioral analysis, challenge-response, and data fusion can be classified in to two subsets; hardware-based and data-driven techniques. These categories are further discussed as follows.

1. Hardware-based techniques: These methods rely on specialized hardware (e.g., face recognition sensors) to capture facial features by analyzing differences in reflectivity between real faces and fake ones. Common approaches include:
2. Sensor Features: Infrared (IR), polarization, and UV sensors.
3. Blinking Detection: The eye area is observed to determine whether the target is blinking using an eye area filter. For example, once the target's eye area is captured, the first step is applying detection whether the eyes is opening. By using the filter to classify the eye openness level. Through this classification, he probability of the eyes being open is generated based on the input image. This probability is then analyzed by comparing the values between the maximum and minimum eye openings. If the result shows a significant difference, it indicates at least one transition between open and closed eyes, allowing us to conclude that the target is blinking [3,6].
4. Lip Movements: Observing actions to confirm the presence of a living person. For example, detecting whether the target is speaking by analyzing their lip movements in a one-second video. This involves identifying lip movement using a filter to determine the location of the upper and lower lips and then calculating the distance of lip separation [7,8].
5. Challenge response: User performs a specific task that prompts by the FLD system.

However, these methods rely heavily on specialized hardware, which is often costly, and their performance can be significantly impacted by lighting conditions [3]. In general, primary limitation of hardware-based methods is the high cost of equipment, its limited accessibility, and the needs for additional installation process.

1. Data-driven based techniques: Data-driven methods rely on training datasets (real vs. fake) and classifiers, with performance depending on dataset diversity. They fall into two categories:
2. Handcrafted feature-based approaches: This method determines the live cues in face images by using analysing the texture features of face images, for example Histogram of Orientation Gradients (HOG), SHEARLET, and Local Binary Patterns (LBP) [9] to detect spoofing cues. They are computationally efficient and perform well in in-database training by using static and dynamic features extracted from common sensors.
3. Deep-learning based approaches: First introduced through AlexNet [6], deep learning methods like CNNs, RNNs, and DNNs have significantly advanced face liveness detection (FLD). Unlike traditional methods, they automatically learn features from raw data, improving accuracy and adaptability to different conditions [6,10]. These models generalize better to variations in lighting, pose, and facial attributes. However, a challenge remains in the lack of clearly interpretable features, as CNNs rely on subtle patterns [3]. Combining CNNs with detection techniques like blinking and lip movements can enhance detection performance.

## Advanced CNN Architectures for Liveness Detection

The basic structure of Convolutional Neural Networks (CNNs) that has been outlined above performs effectively. Researchers have however noted that as these networks get deeper, that is additional layers added, performance can actually degrade. This degradation comes about since it gets hard to effectively pass information present in the previous layers to the subsequent layers during training. To overcome this fact, new CNN structures were designed. The more modern versions also featured so-called shortcut or skip connections, which can be thought of as information highways that allow data to skip several layers and travel more directly through the network. This simple, but effective idea has helped to train very deep and high accuracy models. Residual Networks (ResNet) and Densely Connected Convolutional Networks (DenseNet) are two notable architectures that rely on the specified approach and have been chosen as the participants of this research.

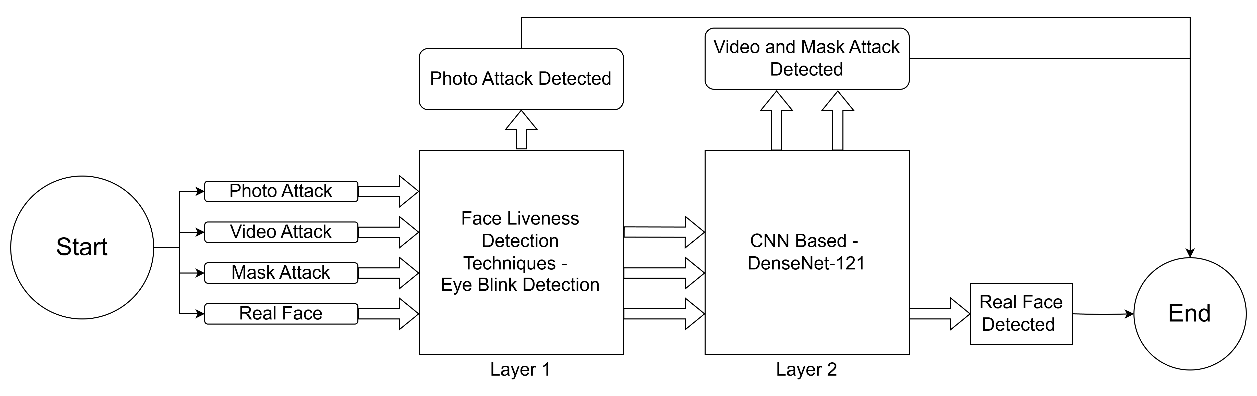
Firstly, DenseNet utilizes a unique and powerful form of skip connection called dense connectivity. In this architecture, each layer accepts the feature maps from all preceding layers and passes its own feature maps to all subsequent layers [11]. Unlike the additive method used in ResNet, DenseNet combines these features by concatenating them. This architecture is designed to enhance feature propagation and efficiency [12]. There are two primary benefits of this architecture, such as improved feature reuse. DenseNet connects layers differently than ResNet. Instead of adding features, it combines them by stacking. This lets the network reuse simple patterns (like edges or textures) many times. As a result, the network works more efficiently. Moreover, it also improved gradient flow, because every layer can see all the earlier features directly. This helps the training signals (gradients) flow better through the whole network. With better gradient flow, the network learns more easily.

Additionally, Residual Network (ResNet), represents a significant advancement in neural network architecture by introducing the concept of “residual learning”. The core principle underlying ResNet is its implementation of skip connections. In a standard deep network, information passes through one layer after another, with each layer transforming the data it receives. However, in a ResNet, the input to a block of layers can also “skip” over that block and be added directly to its output. This shortcut ensures that the original information is not lost as it travels through the network [13]. This makes the network’s job easier, as it only needs to learn the additional, or “residual”, information needed to improve the features. Because of this innovation, ResNet models can be built with a very large number of layers (e.g., 50, 101, or more) and still be trained effectively, leading to excellent performance on complex tasks.

# Proposed implemetation

This research utilizes the ROSE-Youtu Face Liveness Detection Database [14], containing 3,350 videos of 20 individuals captured across five devices (iPhone 5S, Hasee/Huawei/ZTE phones, and iPad). The dataset includes: genuine samples ('G') and seven spoofing attack types: still printed paper ('Ps'), quivering printed paper ('Pq'), Lenovo LCD video ('Vl'), Mac LCD video ('Vm'), cropped-eye paper mask ('Mc'), full paper mask ('Mf'), and split-upper paper mask ('Mu'). Figure 1illustrates sample images from different categories in the ROSE-Youtu dataset.

The proposed methodology consists of two main layers. The first layer is a face liveness detection technique, it employs an eye blink detection technique to determine whether the detected face exhibits natural blinking behavior. However, it is important to note that while this layer is effective against static attacks, such as printed photos, it has limitations when dealing with video-based spoofing. To address the limitations of the first layer, the second layer employs a Convolutional Neural Network (CNN) to analyze the face of liveness. CNN is trained to differentiate between real and fake faces, by examining intricate patterns and features, including those that bypass the first layer by simulating blinking. The flow of the proposed methodology is illustrated in Figure 2, which outlines the interaction between Layer 1 and Layer 2 for comprehensive face liveness detection.

**FIGURE 1.** Sample of different image of different categories from ROSE-Youtu dataset

**FIGURE 2.** Flow of two-layered methodology for face liveness detection

The foundation of this project is the two-stage liveness detection system, which first checks for motion (eye blinks) and then applies a deep learning model for detailed analysis. A critical decision in our proposed method was how to prepare the dataset to train the deep learning component (Stage 2) in the most robust way possible. While the original research paper suggested excluding static photo attacks (“printed paper”, “quivering printed paper”, etc.) from training, our methodology takes a more comprehensive **“**defense-in-depth” approach. We made the deliberate choice to include all seven spoofing categories in the training data.

The purpose for this is to build a more resilient system. By training the deep learning model that include the dataset of static photos in addition to video and mask attacks, it becomes a powerful second line of defense. If a photo attack were somehow bypassing the first-stage blink detector, our model is still able to recognize it as a spoof, closing a potential security vulnerability. Therefore, all available attack types were curated and consolidated into a single “fake” class to train a more generalized and secure classifier. This image dataset was then consolidated into two classes (“real” and “fake”) and split into a training set (80%), a validation set (10%), and a testing set (10%).

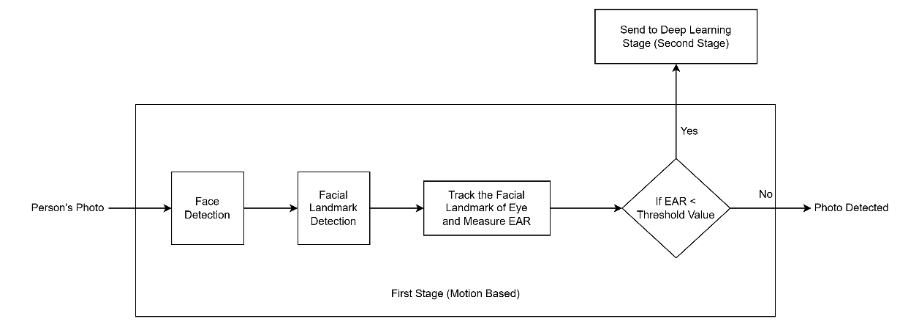
In this project, the pre-trained DenseNet-121 architecture, initially trained on the ImageNet Dataset [15], is utilized. The model is adapted for this project by adding a custom fully connected layer to replace the original fully connected layer of DenseNet-121. Furthermore, the last 15 layers of the network are unfrozen. They are fine-tuned using the ROSE-Youtu dataset. This enhances the network's ability to distinguish between real and spoofed faces. The model is then deployed for a real-time face liveness detection system.

## First Stage – Motion Based

At first stage, the system will detect the eye blinking from the real-time video captured by webcam. To achieve this, dlib-based facial landmark detection is utilized. This step identifies and tracks the landmarks of the face, including the eyes, which are then used to detect blinking. The real-time eye blinking is detected by measuring the Eye Aspect Ratio (EAR). The EAR is a geometric-based metric that calculates the relationship between the height and width of the eye. It provides a reliable indication of whether the eye is open or closed. The EAR can be computed using the Equation (1) and understood easily from Figure 3, where are the coordinates of the six key landmarks around the eye:

|  |  |  |
| --- | --- | --- |
|  |  | (1) |
| **FIGURE 3.** Coordinates of six key landmarks |  |  |

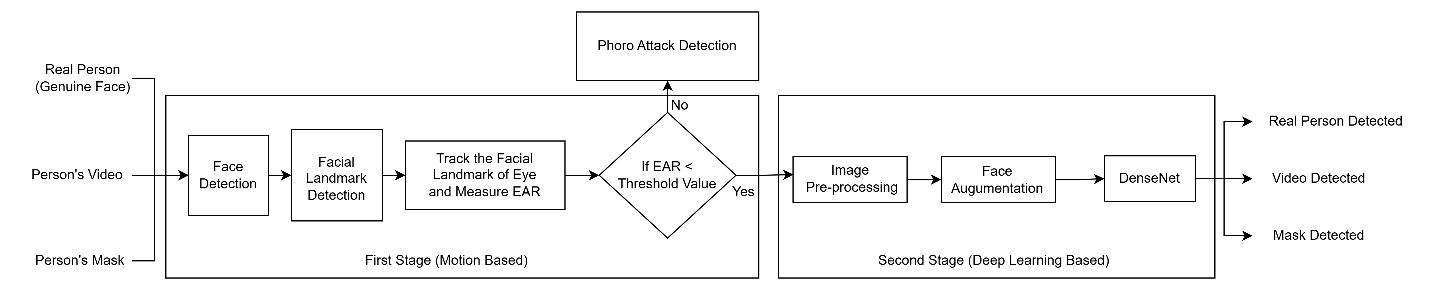
By continuously monitoring the EAR in real-time, the system can accurately determine whether the eyes are opened or closed. A significant drop in EAR value was interpreted as a blink action.

In summary, when a real-time video is captured by the webcam, the system first detects the face as the Region of Interest (ROI). It then checks for eye blinking action. If no blinking action is detected, the system classifies the face as fake face. However, if blinking action is detected, then the face will pass to deep learning-based stage for further analysis. The entire process is visually represented in the following Figure 4, which provides a clear understanding of the first stage of the proposed methodology.

**FIGURE 4**: Algorithm for detecting photo of person

## Second Stage – Deep Learning Based

After the detected blinking face passes through the DenseNet model, which is our proposed model, it undergoes a pre-processing stage that includes operations such as resizing and cropping. Through these steps, it can increase the size and diversity of the dataset, ensure that the input is consistent and optimized for the neural network. To further enhance the dataset, Face Augmentation techniques such as horizontal flipping, zooming in, and zooming out are applied. This approach enhances the dataset by creating more samples, enabling the model to learn more effectively. Therefore, the more data the network is exposed to, the better it generalizes to unseen inputs during evaluation. As previously mentioned, the model is trained on three categories: real faces, masked faces, and video-based spoofing faces. During the evaluation, the model classifies the input face into one of these categories: 1. Real Face (Genuine Face): If the input face is classified as real, the system confirms it as a genuine face, or 2. Fake Face (Spoofing Attack): If the input face is identified as a spoofing attack, such as a masked face or video-based spoofing, the system interprets it as fake. The workflow of detecting real individuals and distinguishing between video-based and mask-based spoofing attacks is visually illustrated in Figure 5 for better understanding.

**FIGURE 5.** Algorithm for detecting the real person, his video and mask

# experimental result

The experimental evaluation demonstrates that the proposed model (DenseNet-121) achieved a superior final test accuracy of 95.55%, outperforming the ResNet-50 benchmark, as summarized in table 1. A detailed analysis of the performance parameters, shown in table 2, reveals the specific trade-offs between the two models. Based on the result, the ResNet-50 model achieved a perfect recall of 1.0000, which means that it correctly identified every genuine user, but it did so at the cost of a higher False Positive Rate (6.86%). While DenseNet-121 model provided a more secure system with a significantly lower False Positive Rate of 4.27% and a higher precision of 0.6250. This indicates that while ResNet is less likely to inconvenience a real user, DenseNet is substantially better at the primary security task of rejecting spoofing attacks.

|  |  |
| --- | --- |
| **TABLE 1.** Final test accuracy on both model. | **TABLE 2.** Calculation of performance parameters. |
| |  |  |  | | --- | --- | --- | | **Experiments** | DenseNet-121 | ResNet-50 | | **Accuracy (%)** | 95.55% | 93.63% | | |  |  |  | | --- | --- | --- | | **Name of the parameters** | **DenseNet-121** | **ResNet-50** | | Error Rate (ERR) | 4.45% | 6.37% | | Precision | 0.6250 | 0.5268 | | Recall | 0.9322 | 1.0000 | | False Positive Rate | 4.27% | 6.86% | |

# CONCLUSION

In the current implementation, a two-stage face liveness detection system was successfully implemented and evaluated, comparing DenseNet-121 and ResNet-50 architectures on the ROSE-Youtu dataset. The experimental results conclusively demonstrate that the DenseNet-121 model provides a more effective and secure solution, achieving a final accuracy of 95.55% with a corresponding error rate of 4.45%. The system's primary strength lies in the DenseNet model's ability to minimize critical security failures, evidenced by its low False Positive Rate of 4.27%. This finding validates the hypothesis that DenseNet's architecture is better suited for capturing the subtle, complex artifacts that distinguish real faces from spoofed ones.

# Acknowledgments

This work is supported under the grant provided by the Malaysia’s Fundamental Research Grant Scheme

(FRGS/1/2023/ICT02/MMU/03/2, 2023).

# References

[1] S. Policepatil and S.M. Hatture, “Face Liveness Detection : An Overview,” Int J Sci Res Sci Technol, 22–29 (2021).

[2] X. Tu, H. Zhang, M. Xie, Y. Luo, Y. Zhang and Z. Ma, “Enhance the Motion Cues for Face Anti-Spoofing Using CNN-LSTM Architecture,” (2019).

[3] R.B. Hadiprakoso, H. Setiawan and Girinoto, “Face Anti-Spoofing Using CNN Classifier Face Liveness Detection,” In Proceedings of the 2020 3rd International Conference on Information and Communications Technology, ICOIACT 2020, Institute of Electrical and Electronics Engineers Inc., 143–147 (2020).

[4] R.E. Saragih and Q.H. To, “A Survey of Face Recognition Based on Convolutional Neural Network,” 4 (2022).

[5] Y.A.U. Rehman, L.M. Po and J. Komulainen, “Enhancing Deep Discriminative Feature Maps via Perturbation for Face Presentation Attack Detection,” Image Visual Computing, 94 (2020).

[6] R. Cai, H. Li, S. Wang, C. Chen and A.C. Kot, “DRL-FAS: A Novel Framework Based on Deep Reinforcement Learning for Face Anti-Spoofing,” IEEE Transactions on Information Forensics and Security 16, 937–951 (2021).

[7] M. Aydin, M. Taskiran, N. Kahraman, and H.V. Dudukcu, “A Fusion-Based Deep Neural Networks Approach for Face Liveness Detection,” In Proceedings of the 17th International Conference on Innovations in Intelligent Systems and Applications, INISTA 2023, Institute of Electrical and Electronics Engineers Inc., (2023).

[8] R. Cai, Z. Li, R. Wan, H. Li, Y. Hu and A.C. Kot, “Learning Meta Pattern for Face Anti-Spoofing,” IEEE Transactions on Information Forensics and Security, 17, 1201-1213 (2022).

[9] Y.A.U. Rehman, L.M Po and M. Liu, “SLNet: Stereo Face Liveness Detection via Dynamic Disparity-Maps and Convolutional Neural Network,” Expert System Appl, 142 (2022).

[10] Z. Ikram, “Hybrid Deep Neural Network for Face Liveness Detection in Real-Time Video,” In Proceedings of the SIST 2024 - 2024 IEEE 4th International Conference on Smart Information Systems and Technologies, Institute of Electrical and Electronics Engineers Inc., 188–193 (2024).

[11] G. Huang, Z. Liu, L. van der Maaten and K.Q. Weinberger, “Densely Connected Convolutional Networks,” CVPR, (2017).

[12] I.P.G. Yoga Pramana Putra, Ni Wayan Jeri Kusuma Dewi, Putu Surya Wedra Lesmana, I Gede Totok Suryawan and Putu Satria Udyana Putra, “Comparison of ResNet-50 and DenseNet-121 Architectures in Classifying Diabetic Retinopathy,” Indonesian Journal of Data and Science, 6, 65–73 (2025).

[13] K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition,” In Proceedings of the Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 770–778 (2016).

[14] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li, “A Face Anti Spoofing Database with Diverse Attacks,” in Proc. 5th IAPR Int. Conf. Biometrics (ICB), New Delhi, India, 26–31 (2012).

[15] M. M. Hasan, M. S. U. Yusuf, T. I. Rohan, and S. Roy, "Face Anti-spoofing Using Texture-based Methods: A Review," in Proc. 4th Int. Conf. on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh, 1–6 (2019).