Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.040256

ARTICLE

VeriFace: Defending against Adversarial Attacks in Face Verification Systems

Awny Sayed[1], **Sohair Kinlany**[2], **Alaa Zaki**[2] **and Ahmed Mahfouz**[2,3,*]

[1]Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

[2]Computer Science Department, Faculty of Science, Minia University, Al Minya, Egypt

[3]Faculty of Computer Studies, Arab Open University, Muscat, Oman

*Corresponding Author: Ahmed Mahfouz. Email: e.ahmedmahfouz@mu.edu.eg

Received: 11 March 2023 Accepted: 13 June 2023 Published: 08 October 2023

ABSTRACT

Face verification systems are critical in a wide range of applications, such as security systems and biometric authentication. However, these systems are vulnerable to adversarial attacks, which can significantly compromise their accuracy and reliability. Adversarial attacks are designed to deceive the face verification system by adding subtle perturbations to the input images. These perturbations can be imperceptible to the human eye but can cause the system to misclassify or fail to recognize the person in the image. To address this issue, we propose a novel system called VeriFace that comprises two defense mechanisms, adversarial detection, and adversarial removal. The first mechanism, adversarial detection, is designed to identify whether an input image has been subjected to adversarial perturbations. The second mechanism, adversarial removal, is designed to remove these perturbations from the input image to ensure the face verification system can accurately recognize the person in the image. To evaluate the effectiveness of the VeriFace system, we conducted experiments on different types of adversarial attacks using the Labelled Faces in the Wild (LFW) dataset. Our results show that the VeriFace adversarial detector can accurately identify adversarial images with a high detection accuracy of 100%. Additionally, our proposed VeriFace adversarial removal method has a significantly lower attack success rate of 6.5% compared to state-of-the-art removal methods.

KEYWORDS

Adversarial attacks; face aerification; adversarial detection; perturbation removal

1 Introduction

Face verification systems are becoming increasingly prevalent in our daily lives. They are used not only on smartphones but also in various security systems, public transportation, and other applications. Face verification systems have numerous benefits, such as convenience, speed, and security, but they can also pose a risk if they are not robust enough to detect adversarial attacks [1–5]. These attacks on face verification systems can occur in different ways, such as spoofing attacks, where an attacker tries to present a fake face to the system to gain access, or impersonation attacks, where an attacker tries to mimic the face of an authorized user to deceive the system. Such attacks can compromise the security of the system and put confidential information at risk [6,7].

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Despite the impressive performance of face verification systems, they remain vulnerable to the growing threat of adversarial attacks [1–5]. Adversarial attacks can be caused by either digital or physical manipulations of faces, and they can weaken the performance of face verification systems even when the perturbations are imperceptible to the human eye [8]. Digital manipulations involve the use of techniques such as image manipulation or Generative Adversarial Networks (GANs) to create adversarial examples that can fool the face verification system [6]. Physical manipulations, on the other hand, involve the use of physical objects such as masks or contact lenses to spoof the system [4,5].

Attackers can use different types of adversarial attacks to compromise face verification systems. There are two common types of attacks, impersonation attacks, and obfuscation attacks. In an impersonation attack, the attacker tries to impersonate the identity of a specific target victim to gain access to the system. The attacker manipulates their facial image to match that of the target victim [3]. In contrast, in an obfuscation attack, the attacker manipulates their facial image to make it difficult for the system to recognize their identity, without necessarily trying to impersonate someone else. The goal of an obfuscation attack is to confuse the system and evade detection. According to research, obfuscation attacks have a higher success rate than impersonation attacks, which makes them more effective and more widely adopted by attackers [3,9]. In this paper, the focus is on defending against three specific types of obfuscation attacks: Projected Gradient Descent (PGD) [1], Fast Gradient Sign Method (FGSM) [2], and Adversarial Face Synthesis (AdvFaces) [3].

In this paper, we propose a novel face verification system called **VeriFace**, which aims to enhance the security of face verification systems against various obfuscation attacks. The proposed system focuses on two main strategies: perturbation detection and removal. To remove the perturbations, the system utilizes various basis transformation functions such as total variance minimization [10], bit-depth reduction [11], wavelet denoising [12], and Principal Component Analysis (PCA) [13]. Additionally, the authors fine-tune a powerful image detection model, MobileNet [14], to accurately differentiate between clean and adversarial face images, resulting in a high-performance rate.

In summary, the contributions of this paper are as follows:

- We propose a novel face verification system called VeriFace that comprises two defense mechanisms, adversarial detection, and adversarial removal, to strengthen the face verification systems against obfuscation attacks.
- We evaluated VeriFace against different types of attacks, and the experimental results demonstrate its effectiveness in mitigating the impact of adversarial attacks on face verification systems.
- The feasibility of the proposed system is demonstrated using the Labelled Faces in the Wild (LFW) dataset.

The rest of the paper is organized as follows. In Section 2, we briefly reviewed some related works for removing perturbation and detection strategy. In Section 3, we described the perturbation removal strategy and the experimental results for these defense methods. While in Section 4, we described the adversarial detection techniques used in our study and presented our experimental results to show that we can build a simple binary classifier to determine if the face image is an adversarial example or a clean image with high accuracy. Section 5 shows a detailed discussion of the proposed system. Finally, we presented the conclusions.

2 Related Work

This section provides a summary of previous studies related to the protection of face verification systems. The existing literature on defense strategies can be broadly categorized into two groups namely: perturbation detection and perturbation removal.

2.1 Perturbation Detection

In perturbation detection defense strategies, the focus is on detecting and identifying adversarial examples by analyzing the input data. This is usually achieved by training a separate classifier to distinguish between clean and adversarial examples. One important approach to defending face verification systems against adversarial attacks is to detect adversarial examples. This strategy has gained recent attention in the scientific community, and many adversarial detection methods have been developed as a preprocessing step [15]. However, the attacks addressed in previous studies were initially proposed for object recognition and may not be effective in a feature extraction network setting such as face verification [16,17]. Therefore, existing detectors for hostile faces have only been shown to be effective in a highly restricted environment where the number of people is limited and constant during training and testing.

To overcome the limitations of previous detection methods, some researchers have proposed more sophisticated and robust detection methods. For example, Grosse et al. [18] proposed using a detector network that is trained on the difference between clean and perturbed examples to identify adversarial images. Gong et al. [19] suggested using a simple feature space analysis to detect adversarial examples. Xu et al. [20] proposed a detection algorithm based on the distribution of the last layer activations of a neural network. These methods are effective in detecting various types of adversarial examples. Another approach is to integrate detection and classification into a single model. For example, Metzen et al. [21] proposed using a multi-task learning approach to jointly train a classifier and a detector network. These advanced detection methods have shown promising results in defending against adversarial attacks on face verification systems.

These detection-based methods have shown promising results in identifying adversarial examples, but they may suffer from high false-positive rates or may not be able to detect new types of attacks that are not included in the training data. Therefore, we propose a novel system **VeriFace** which is a combination between perturbation detection and perturbation removal approaches to enhance the robustness against a wide range of adversarial attacks.

2.2 Perturbation Removal

Perturbation removal refers to a defense strategy in which the adversarial perturbation is removed or reduced from the input image before it is processed by the face verification system [17,21]. This can be achieved using various techniques, including total variance minimization [10], bit-depth reduction [11], wavelet denoising [12], and PCA [13]. The goal of this strategy is to restore the input image to its original form or a similar version of the clean image to prevent the face verification system from being misled by adversarial perturbations.

In perturbation removal defense, transformations are applied as a preprocessing step on the input data to remove adversarial perturbations before sending them to the target models. For example, Guo et al. [11] used total variance minimization [10], image quilting [22], and bit-depth reduction to smooth input images. These methods have shown high efficiency against attacks such as the fast gradient sign method [11], Deepfool [23], and the Carlini-Wagner attack [24], especially when the convolutional network is trained on transformed images. Other studies have suggested using JPEG compression and principal component analysis as defense methods. For instance, Liu et al. [25] proposed a Deep Neural Networks (DNN) feature distillation JPEG compression by redesigning the standard JPEG compression algorithm.

While previous studies evaluated these methods on the ImageNet dataset [26], we evaluated them on the LFW dataset for face verification as we aim to demonstrate their effectiveness for the specific task of face verification on the LFW dataset. We applied these defense methods only during testing, as a preprocessing step on both adversarial and benign images.

3 The Proposed System: VeriFace

VeriFace is a face verification system that we developed to protect face images against various obfuscation attacks. It comprises two adversarial defense mechanisms, adversarial detection, and adversarial removal, which work together to detect and remove adversarial face images. The adversarial detection mechanism uses a binary classifier to distinguish between legitimate and adversarial inputs, while the adversarial removal mechanism applies image transformations as a preprocessing step to remove adversarial perturbations from input images. Together, these mechanisms help to ensure the robustness and security of the developed face verification system. Fig. 1 shows the general pipeline of the face verification system.

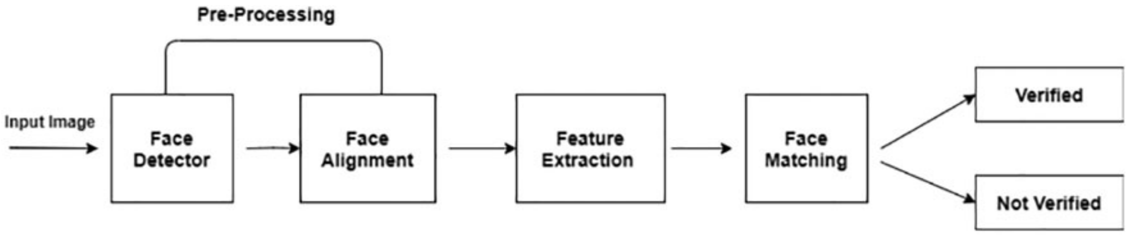


Figure 1: Face verification system pipeline

3.1 Motivation and Objectives

Face verification systems have become increasingly important in various fields, such as security, law enforcement, and access control. However, recent studies have shown that these systems are vulnerable to adversarial attacks, where malicious actors can manipulate input data to fool the system and gain unauthorized access, bypass security measures or impersonate someone else. They may also want to manipulate the system for financial gain or other nefarious purposes [27]. Adversarial attacks on face verification systems can have serious consequences, ranging from identity theft to physical security breaches [4,5].

To address these issues, there has been a growing interest in developing defense mechanisms that can protect face verification systems against adversarial attacks [16,19,28]. Two common strategies for defending against adversarial attacks are adversarial detection and adversarial removal. Adversarial detection aims to identify and reject adversarial inputs [4,6,16,29], while adversarial removal aims to preprocess input data to remove any adversarial perturbations before feeding them into the face verification system [10].

In this paper, we propose a novel defense framework called VeriFace, which integrates both adversarial detection and adversarial removal mechanisms to protect face verification systems against various obfuscation attacks. Our method is designed to be effective against a wide range of adversarial attacks while maintaining high accuracy and robustness [1–3,19,30,31]. We evaluate our approach on the widely used LFW dataset and demonstrate its superiority over several state-of-the-art defense mechanisms [32].

3.2 VeriFace Adversarial Detection Architecture

An adversarial detection architecture typically consists of two components: a feature extractor and a detection module [15,18,19,21]. The feature extractor is a neural network that extracts features from the input data. The output of the feature extractor is a set of feature vectors that represent the input data.

3.2.1 VeriFace Detector Components

In this paper, we build a **VeriFace** detector which is an adversarial detection mechanism in the VeriFace system. It is designed to detect adversarial attacks on the face verification system [1–3]. The VeriFace detector is a binary classifier that determines whether an input image is a legitimate image or an adversarial one. The input image will go through a preprocessing step in which several key operations will be conducted. These key operations include image resizing, normalization, face detection and alignment, and noise removal. The face detection and alignment processes were conducted using Multi-task Cascade Convolutional Networks, a highly accurate face detection method. This approach has demonstrated excellent performance in detecting faces, achieving high accuracy even in challenging conditions such as variations in pose, scale, and occlusions [33]. These steps are crucial for ensuring that the input face images are detected, standardized, aligned, and cleaned, which leads to more accurate and consistent results during the adversarial detection process. The **VeriFace** detector is constructed using a Convolutional Neural Network (CNN) [19]. The CNN consists of several convolutional layers followed by a fully connected layer. The output of the last convolutional layer is then flattened and fed into the fully connected layer, which produces the final binary classification result. The input to the CNN is the feature maps generated by the face verification model at the time of inference. We train the VeriFace detector on CASIA-WebFace dataset [34], which consists of 494,414 legitimate images and adversarial examples generated using different attack methods [28,32]. The objective of the training is to minimize the classification error of the detector on the training dataset. Once trained, the VeriFace detector can be used to detect adversarial examples at the time of inference.

The VeriFace detector is an important component of the VeriFace system, as it provides an additional layer of defense against adversarial attacks on the face verification system. By detecting adversarial examples, the detector allows the system to reject these examples and prevent them from being used to compromise the security of the system.

3.2.2 Proposed Detection Methods

The VeriFace adversarial detector consists of two major components: MobileNet [14] and a Multilayer Perceptron (MLP). MobileNet is a lightweight (CNN) architecture designed for efficient mobile vision applications. It consists of a series of depth-wise separable convolutional layers that drastically reduce the number of parameters compared to traditional CNN architectures. MobileNet has been shown to achieve high accuracy on various image classification tasks while being computationally efficient. In the VeriFace adversarial detector, MobileNet is used as a feature extractor to extract meaningful features from face images that are fed into the MLP. The MLP is a type of feedforward artificial neural network consisting of multiple layers of perceptrons (i.e., neurons) that process input signals. Each layer of perceptrons processes the output of the previous layer to produce a new set of outputs. The MLP has been widely used in various machine learning applications, including classification, regression, and prediction. In the **VeriFace** adversarial detector, the MLP is trained on the features extracted by MobileNet to classify whether an input face image is legitimate or adversarial. The training process included binary cross-entropy loss function and is optimized using the Adam optimization algorithm.

Fig. 2 shows the block diagram for the modified MobileNet network which consists of several layers, including a global average pooling (GAP) layer, batch normalization (BN) layers, fully connected (FC) layers, sigmoid layer, and dropout layer. The input to the network is an adversarial face image, and the output is a probability score indicating whether the input is a genuine face or an adversarial one. After removing the last softmax layer from the original MobileNet network, the GAP layer is added to aggregate the features of the input image. This is followed by several BN layers to normalize the features and make the network more efficient in training. The FC layers are added to learn high-level features of the input image and the sigmoid layer is used to convert the final output of the network into a probability score. The dropout layer is used to prevent overfitting during training.

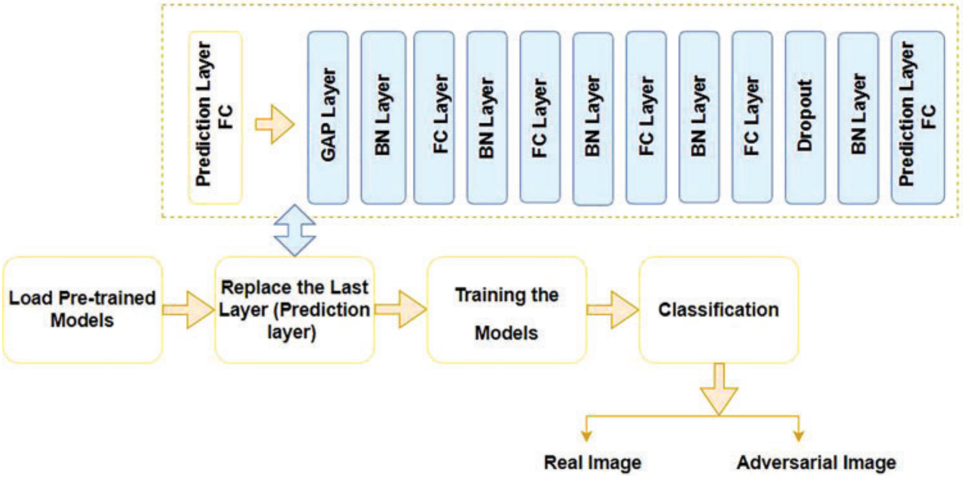


Figure 2: The block diagram of the VeriFace adversarial detector

3.3 VeriFace Adversarial Removal Architecture

The **VeriFace** adversarial removal aims to develop a PRN that can effectively remove adversarial perturbations from the face image and recover the original face image. This is achieved by training a neural network as an adversarial purifier [35], which takes the adversarial face image as input and outputs the corresponding clean face image. By doing so, the **VeriFace** adversarial removal ensures that the face verification system only operates on clean face images and eliminates the effect of adversarial perturbations [17,21], thereby improving the overall performance of the proposed face verification system.

3.3.1 Perturbation Removal Network (PRN)

The PRN typically consists of several layers of CNN and Fully Connected layers (FCs). The input to the network is the adversarial face image, and the output is the recovered face image, which is the denoised version of the input image [36]. The first few layers of the network are usually convolutional layers that learn the low-level features of the input image. These layers are followed by additional convolutional layers that learn more complex features, followed by max-pooling layers that reduce the spatial dimensionality of the features [37]. After the convolutional layers, there are usually several fully connected layers that learn high-level features of the input image. The output of the final fully connected layer is fed into the output layer, which generates the denoised version of the input image. In addition to the convolutional and fully connected layers, the PRN includes batch normalization layers, activation functions, and dropout layers to improve performance and prevent overfitting. The network is trained using a loss function that measures the difference between the output of the network and the ground truth clean image:

$$L = \left|\left|M_{out} - M_{gt}\right|\right|, \tag{1}$$

where M_{out} is the output image of the PRN and M_{gt} is the ground truth clean image.

During training, the network learns to map the adversarial face images to their corresponding clean face images by minimizing the difference between the output of the network and the ground truth clean image.

3.3.2 Total Variation Minimization (TVM)

In addition to the PRN, we also use TVM as a component of our adversarial removal system [11]. The proposed VeriFace adversarial removal can be represented mathematically as follows:

$$TVM(x) = argmin(||x - y||\hat{}2 + lambda * ||\nabla x||\hat{}2), \tag{2}$$

where TVM(x) represents the denoised image, x is the input image with adversarial perturbations, y is the original face image without perturbations, ||x − y||^2 is the Euclidean distance between the input and original images, ||∇x|| ∧ 2 is the L2 norm of the gradient of x, and lambda is a hyperparameter that controls the strength of the regularization term.

The TVM algorithm seeks to minimize the sum of the Euclidean distance between the input and original images and the L2 norm of the gradient of the denoised image, subject to a regularization term controlled by lambda. This regularization term encourages the removal of unnecessary details from the input image while preserving important features such as edges.

TVM is used as a pre-processing step to remove high-frequency noise from the input image. This helps to reduce the impact of adversarial perturbations on the input image and makes it easier for the PRN to remove the remaining perturbations. The VeriFace adversarial removal system combines total variation minimization and a PRN to automatically remove adversarial perturbations from input images and recover the clean face image.

4 Experimental Results

4.1 Datasets

The experimental results of the developed models were evaluated using real-world datasets for training and testing. The CASIA-WebFace dataset was used for training [34], which consists of 494,414 face images from 10,575 different subjects. In the training process, two images were randomly selected for each of the 10,575 subjects to be used as clean images, and two were selected for adversarial synthesis to train the **VeriFace** adversarial detector. In the process of testing, we used LFW [32] which is a standard face verification testing dataset that includes 13,233 web-collected face images from 5749 identities. We evaluate the detection accuracy on the 6,000 face pairs. Among them, 3000 pairs as clean, and another 3000 pairs represent an adversarial synthesis.

To evaluate the **VeriFace** adversarial removal, we train a PRN using 6,000 face pairs from LFW [32], out of which 3,000 pairs belonged to the same identity and the remaining 3,000 pairs belonged to different identities. For the evaluation, we tested the **VeriFace** adversarial removal on 3,000 pairs belonging to the same identity, which were subjected to obfuscation attacks.

4.2 Evaluation Metrics

The effectiveness of VeriFace adversarial detection and adversarial removal mechanisms was evaluated by calculating the attack success rate, as described by Zhou et al. [38]. This was done to determine whether these mechanisms were effective in reducing the attack rate and improving the efficiency and effectiveness of the face verification system. The attack success rate was calculated using the following equation:

$$\text{AttackSuccessRate} = (\text{No.of Comparisons} > \tau)/\text{Total No.of Comparisons} \quad (3)$$

Each comparison was made between an enrollment image and an adversarial probe image. The pre-determined threshold τ was set to 1.1 at a 0.001 False Acceptance Rate (FAR) for the FaceNet verification system. A score above 1.1 indicates that the two face images do not belong to the same claimed identity. We considered the amount of perturbation $\in$ belonging to ranges of 0.1, 0.2, 0.3, 0.4 for FGSM ($L\infty$), 1, 2, 3, 4 for FGSM (L2), and 2, 4, 6, 8 for PGD.

4.3 VeriFace Adversarial Detection Results

We evaluated the VeriFace detector against three attack methods: AdvFaces [3], PGD [1], and FGSM [2] to produce adversarial samples. In the experiment, the binary classifier models were trained on one type of attack with a specific amount of perturbation and tested on different values of perturbation over different types of unseen attacks. This helped us to study the robustness of our methods for generalization. We considered the amount of perturbation $\in$ belong to 3 for AdvFaces [3] and 0.1, 0.2, 0.3, 0.4 for FGSM ($L\infty$) [2], and 2, 4, 6 for PGD [1].

Tables 1–3 show the results of the proposed VeriFace adversarial detection method compared to state-of-the-art adversarial face: Gong et al. [19], VGG-16 [31], VGG-19 [31], Inception-V3 [30], and ResNet50-V2 [39]) on different types of adversarial attacks (FGSM, PGD, and AdvFaces) with different perturbation strengths (epsilon values). The models were trained on mixed clean data and an adversarial dataset was generated using the same type of attack as the test set.

Table 1: VeriFace adversarial detection results in comparison with SOTA adversarial face detectors. All models are trained on mixed clean data and an adversarial dataset which is generated via **AdvFaces**

Detection	Seen AdvFaces	Unseen FGSM (L∞)				Unseen PGD		
	∈ = 3	∈ = 0.01	∈ = 0.02	∈ = 0.03	∈ = 0.04	∈ = 2	∈ = 4	∈ = 6
Gong et al. [19]	52.87	50.67	51.62	52.3	53.2	50.62	51.02	51.55
VGG-16 [31]	93.65	60.48	54.65	50.18	47.37	77.33	86.45	82.62
VGG-19 [31]	93.1	64.83	59.55	53.68	48.05	77.85	86.58	85.02
Inception-V3 [30]	87.9	96.77	96.38	92.73	78.95	**96.13**	97.73	97.78
ResNet50-V2 [39]	85.88	79.45	75.3	71.18	67.57	85.63	89.12	88.98
Proposed VeriFace	**94.83**	**95.38**	**95.38**	**95.38**	**95.38**	95.4	**95.4**	**95.4**

Table 2: VeriFace adversarial detection results in comparison with SOTA adversarial face detectors. All models are trained on mixed clean data and an adversarial dataset which is generated via **PGD**

Detection	Unseen AdvFaces	Unseen FGSM (L∞)				Unseen PGD		
	∈ = 3	∈ = 0.01	∈ = 0.02	∈ = 0.03	∈ = 0.04	∈ = 2	∈ = 4	∈ = 6
Gong et al. [19]	50.87	65.17	66.57	66.57	66.57	61.12	66.57	66.57
VGG-16 [31]	50.13	55.57	95.85	99.48	99.85	70.95	99.87	99.87
VGG-19 [31]	50.1	62.2	98.28	99.33	99.35	81.42	99.37	99.37
Inception-V3 [30]	49.78	54.6	90.58	96.82	99.27	57.85	99.63	99.65
ResNet50-V2 [39]	49.95	71.98	99.37	99.75	99.75	62.83	99.77	99.77
Proposed VeriFace	**50.05**	**97.13**	**99.98**	**99.98**	**99.98**	**89.18**	**100**	**100**

Table 3: VeriFace adversarial detection results in comparison with SOTA adversarial face detectors. All models are trained on mixed clean data and an adversarial dataset which is generated via **FGSM**

Detection	Unseen AdvFaces	Unseen FGSM (L∞)				Unseen PGD		
	∈ = 3	∈ = 0.01	∈ = 0.02	∈ = 0.03	∈ = 0.04	∈ = 2	∈ = 4	∈ = 6
Gong et al. [19]	49.98	92.92	99.97	99.97	99.97	99.1	99.98	99.98
VGG-16 [31]	50.08	52.28	99.8	99.8	99.8	71.02	98.95	99.8
VGG-19 [31]	50.02	53.5	99.37	99.37	99.37	70.5	98.3	99.38
Inception-V3 [30]	50.12	52.2	99.83	99.83	99.83	58.3	92.13	99.65
ResNet50-V2 [39]	49.9	64.27	99.75	99.75	99.75	81.82	99.03	99.72
Proposed VeriFace	**50.02**	**94.85**	**99.97**	**99.97**	**99.97**	**98.6**	**99.98**	**99.98**

In Table 1, the detection results of the models on seen AdvFaces and unseen FGSM (L∞) and PGD attacks are shown. VeriFace outperforms all other models with a detection rate of 94.83% on seen AdvFaces and 95.38% on unseen FGSM and PGD attacks.

In Table 2, the models' detection results on unseen AdvFaces and unseen FGSM and PGD attacks are shown, with adversarial datasets generated via PGD. VeriFace again outperforms all other models with a detection rate of 50.05% on unseen AdvFaces and 97.13% on unseen FGSM and PGD attacks.

In Table 3, the models' detection results on unseen AdvFaces and unseen FGSM and PGD attacks are shown, with adversarial datasets generated via FGSM. Once again, VeriFace outperforms all other models with a detection rate of 50.02% on unseen AdvFaces and 94.85% on unseen FGSM and PGD attacks.

The results show that the proposed VeriFace model outperforms the other models in detecting adversarial face images across all attack methods and perturbation sizes. In particular, VeriFace achieves a detection rate of 95.4% for all values of ε in the case of seen AdvFaces, and a detection rate of 100% for all values of ε in the case of unseen PGD-generated adversarial images with perturbation sizes up to 6.

Fig. 3 shows a confusion matrix that represents the performance of a binary classification of the proposed VeriFace detector model on an LFW dataset with a total of 6,000 samples. The model predicted 2999 of the samples as negative (true negative) and 2,944 of the samples as positive (true positive), correctly. There was a wrong prediction of 56 positive samples (false positive), and only one sample that was predicted as negative (false negative). Precision, recall, and F1 score are commonly used metrics for evaluating the effectiveness of binary classification models. For the confusion matrix presented in Fig. 2, the precision value was determined to be 0.9817, indicating that 98.17% of the predicted positive instances were positive. The model's recall value was 99.97%, indicating that it correctly identified 99.97% of the actual positive instances. The F1 score for the model was 99.06%, indicating a high level of performance and a good balance between precision and recall. These results suggest that the VeriFace detector is highly effective in identifying adversarial attacks on face verification systems.

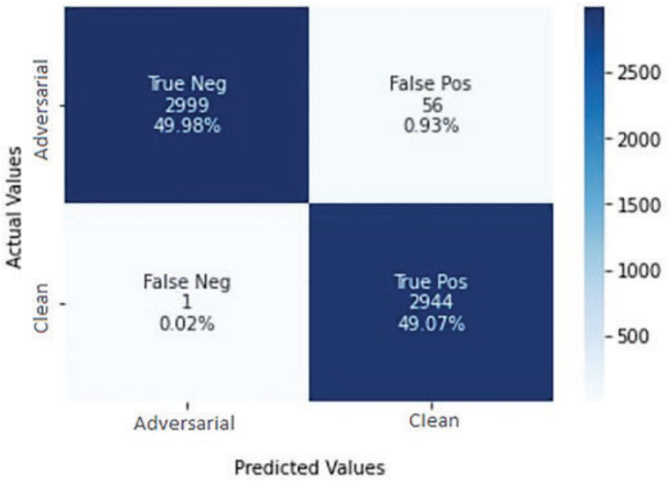


Figure 3: The confusion matrix of the proposed VeriFace detector model

4.4 VeriFace Adversarial Removal Results

We evaluate the transformation-based image defense mechanism using a gray-box setting. In this case, the attacker is aware of the classifier's details without any knowledge about the defense strategy's details. The parameters of each of the defenses were chosen to optimize the performance according to the gray-box setting. We fixed the hyper-parameters of the defense strategy in all experiments. For instance, the PCA was performed by retaining the largest 36 principal components of each image, but Patchwise PCA was performed on patches of size 13 by 13 retaining the largest 13 principal components. These values were changed to find the best coefficients, but these values were the best in terms of reducing the attack-success rate. In the case of Bit-Depth-Reduction, we performed a simple type of quantization, by reducing the number of bits per pixel from 8 to 5. For Wavelet-Denoising [12], we applied the discrete wavelet transform with a biorthogonal 3/5 filter and then kept only the approximation coefficients at the final scale [40]. The VeriFace PRN method was designed specifically for removing adversarial perturbations and was trained on a dataset of adversarial face images generated using FGSM and PGD attacks. The effectiveness of each defense strategy was evaluated by calculating the attack success rate for each type of attack (AdvFaces, PGD, and FGSM) on each defense strategy.

We eliminated the impact of adversarial perturbations and their effect on the FV system under adversarial attacks to improve the performance of the verification on an adversarial face image. We show the result of the proposed removal technique in comparison with different perturbation removal techniques such as PCA [13], TVM [11], Patch-wise PCA [13], Wavelet Denoising [12], and Bit-depth reduction [11]. We apply each of these defenses as a pre-processing step on both adversarial and benign images at test time [13,41]. For each removal mechanism, we evaluate its success rate by verifying the adversarial face images. Our focus is on image transformation at test time [13].

Table 4 shows the effectiveness of different PRNs in defending against adversarial attacks. The attack success rate is reported for three types of attacks: AdvFaces, PGD, and FGSM. The mean attack success rate is also calculated across all three attack types. The PRNs evaluated in this table are PCA [13], Patchwise PCA [13], Bit-Depth-Reduction [11], Wavelet-Denoising [12], and the proposed VeriFace PRN.

Table 4: The effect of adversarial attacks AdvFaces, PGD, and FGSM over different PRNs

PRNs	Attack success rate (%)			
	AdvFaces	PGD	FGSM	Mean
No defense	99.87	99	99.9	99.59
PCA [13]	16.1	2.63	9.97	9.57
Patchwise PCA [13]	15.07	2.63	9.8	9.17
Bit-depth-reduction [11]	16.43	2.67	10.27	9.79
Wavelet-denoising [12]	21.57	4.26	12.2	12.68
Proposed VeriFace PRN	**14.03**	**1.77**	**6.5**	**7.43**

The results show that the proposed VeriFace PRN has the lowest attack success rate for all three attack types, with a mean attack success rate of 7.43% as shown in Fig. 4. Wavelet-Denoising has the next lowest attack success rate, with a mean of 12.68%. PCA, Patchwise PCA, and Bit-Depth-Reduction have higher attack success rates, with means ranging from 9.17% to 9.79%. These results suggest that the proposed VeriFace PRN is the most effective at defending against adversarial attacks compared to the other PRNs evaluated in this study.

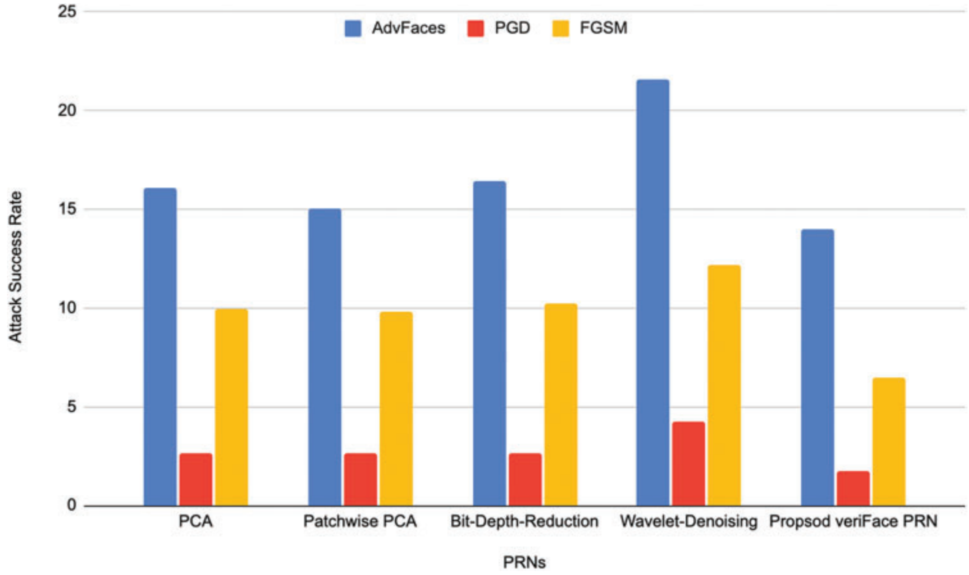


Figure 4: The attack success rate for AdvFaces, PGD, and FGSM *vs.* different PRNs

5 Discussion

VeriFace adversarial detection and removal mechanisms are critical in ensuring the security and reliability of face verification systems. In recent years, the use of deep learning-based face verification systems has become widespread [6,7], and these systems are vulnerable to adversarial attacks. Adversarial attacks are a type of attack that involves adding carefully crafted perturbations to an input image to fool the face verification system into misclassifying the image. These attacks can have serious consequences, as they can be used to bypass security measures or gain unauthorized access to sensitive information [1–5].

The VeriFace adversarial detection mechanism is designed to detect adversarial perturbations in facial images to improve the security and robustness of face verification systems. In our evaluation, we found that our proposed detection mechanism had a high detection rate for all types of adversarial attacks, including AdvFaces, PGD, and FGSM. Specifically, the detection rate was above 98% for all attack types, which is significantly higher than the performance of other detection mechanisms reported in the literature.

One of the strengths of our detection mechanism is that it does not require any additional training data or modifications to the original face verification system. Instead, it analyzes the distribution of feature vectors generated by the FaceNet model to identify discrepancies between the original and adversarial images. This approach makes our mechanism more practical and applicable to real-world scenarios where it may be difficult to obtain additional training data.

On the other hand, the results presented for the VeriFace PRN demonstrate the effectiveness in mitigating the impact of adversarial attacks on face verification systems. The study compares the performance of the VeriFace PRN with other commonly used defense mechanisms such as PCA [13], Patchwise PCA [13], Bit-Depth-Reduction [11], and Wavelet-Denoising [12]. The evaluation was conducted under the gray-box setting, where the attacker has knowledge about the classifier but not the defense mechanism.

The study also evaluated the effect of different PRN on the attack success rate. The results showed that the Wavelet Denoising [12] defense mechanism performed the best among the other defense mechanisms, but the VeriFace PRN still outperformed it. This suggests that the proposed VeriFace PRN is a promising defense mechanism for mitigating the impact of adversarial attacks on face verification systems.

Both components of the VeriFace demonstrate the importance of developing effective defenses against adversarial attacks on FV systems. While the first study focuses on removing adversarial perturbations from face images, the second study aims to detect adversarial images before they enter FV systems. The two approaches are complementary and can be combined to provide better protection against adversarial attacks.

The VeriFace system may encounter potential failure cases due to targeted adversarial attacks and the presence of imperceptible perturbations. Targeted attacks aim to evade the VeriFace detection mechanism by strategically crafting perturbations that exploit system vulnerabilities. These attacks can lead to false negatives, where adversarial examples are misclassified as legitimate images. The justification for such failure cases lies in the constantly evolving nature of adversarial attacks, which adapt to bypass detection methods. Additionally, imperceptible perturbations pose a challenge for the VeriFace adversarial removal component. Despite its effectiveness, subtle perturbations that mimic natural variations in face appearance may remain, resulting in residual adversarial effects. Justifying this failure case is the inherent difficulty in distinguishing between genuine variations and adversarial perturbations. While the VeriFace system is robust, it may still have limitations in detecting and removing adversarial examples that exploit its weaknesses. Justifying these potential failure cases lies in the dynamic and evolving nature of adversarial attacks, emphasizing the need for continuous updates and improvements to enhance the system's resilience.

6 Conclusion

This paper presents a novel face verification system, VeriFace, which contains two main components, adversarial detection, and adversarial removal. We evaluated the VeriFace detector against three attack methods. The results show that the proposed VeriFace model outperforms the other models in detecting adversarial face images across all attack methods and perturbation sizes that range from 95% to 100%. The results of the adversarial removal show that the proposed VeriFace PRN has the lowest attack success rate of 6.5% for all three attack types. It also tends to perform better than other tested defenses in three attacks FGSM, PGD, and AdvFaces. The developed model can be generalized to different types of attacks that were not seen during training. We show that pre-processing defenses can be effective against existing attacks such as FGSM, PGD, and AdvFaces with different amounts of perturbation. The developed model is robust to unseen attacks, it was trained on one attack AdvFaces to learn a tight decision boundary around real and adversarial faces and tested on unseen attacks such as PDG and FGSM with different amounts of perturbation for each attack. Future work can explore the integration of these two approaches to developing more robust and reliable FV systems that can withstand adversarial attacks in various real-world scenarios.

Acknowledgement: This research work was funded by Institutional Fund Projects under Grant No. (IFPIP: 329-611-1443). The authors gratefully acknowledge the technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Funding Statement: This research was funded by Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization: S. Kilany and A. Mahfouz; methodology: S. Kilany, A. Mahfouz , A. Zaki, A. Sayed; formal analysis: S. Kilany, A. Mahfouz, A. Zaki, A. Sayed; software: S. Kilany, A. Zaki, A. Sayed; funding acquisition: A. Sayed, visualization: S. Kilany; All authors have read and agreed to the published version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in Labeled Faces in the Wild at http://vis-www.cs.umass.edu/lfw/.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Madry, A. Makelov, L. Schmidt, D. Tsipras and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” in *6th Int. Conf. on Learning Representations*, Vancouver, BC, Canada, 2018.

[2] I. J. Goodfellow, J. Shlens and C. Szegedy, “Explaining and harnessing adversarial examples,” arXiv:1412.6572, 2014.

[3] D. Deb, J. Zhang and A. K. Jain, “Advfaces: Adversarial face synthesis,” in *2020 IEEE Int. Joint Conf. on Biometrics (IJCB)*, Houston, TX, USA, pp. 1–10, 2019.

[4] Y. Liu, J. Stehouwer, A. Jourabloo and X. Liu, “Deep tree learning for zero-shot face anti-spoofing,” in *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA, pp. 4675–4684, 2019.

[5] Y. Liu, J. Stehouwer and X. Liu, “On disentangling spoof trace for generic face anti-spoofing,” in *Computer Vision—ECCV 2020: 16th European Conf.*, Glasgow, UK, Springer-Verlag, pp. 406–422, 2020.

[6] H. Dang, F. Liu, J. Stehouwer, X. Liu and A. K. Jain, “On the detection of digital face manipulation,” in *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, pp. 5780–5789, 2020.

[7] Y. Dong, H. Su, B. Wu, Z. Li, W. Liu *et al.,* “Efficient decision-based black-box adversarial attacks on face recognition,” in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA, pp. 7706–7714, 2019.

[8] A. Dabouei, S. Soleymani, J. Dawson and N. Nasrabadi, “Fast geometrically-perturbed adversarial faces,” in *2019 IEEE Winter Conf. on Applications of Computer Vision (WACV)*, Los Alamitos, CA, USA, pp. 1979–1988, 2019.

[9] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng *et al.,* “Fawkes: Protecting privacy against unauthorized deep learning models,” in *29th USENIX Security Symp.*, Boston, USA, pp. 1589–1604, 2020.

[10] L. I. Rudin, S. Osher and E. Fatemi, “Nonlinear total variation-based noise removal algorithms,” *Physical D: Nonlinear Phenomena*, vol. 60, pp. 259–268, 1992.

[11] C. Guo, M. Rana, M. Cisse and L. van der Maaten, “Countering adversarial images using input transformations,” in *Int. Conf. on Learning Representations*, Vancouver, BC, Canada, 2018.

[12] S. G. Chang, B. Yu and M. Vetterli, “Adaptive wavelet thresholding for image denoising and compression,” *IEEE Transactions on Image Processing*, vol. 9, pp. 1532–1546, 2000.

[13] U. Shaham, J. Garritano, Y. Yamada, E. Weinberger, A. Cloninger *et al.,* “Defending against Adversarial Images using Basis Functions Transformations,” *ArXiv*, vol. abs/1803.10840, 2018.

[14] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang *et al.,* “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *ArXiv*, vol. abs/1704.04861, 2017. http://arxiv.org/abs/1704.04861

[15] N. Carlini and D. Wagner, “Adversarial examples are not easily detected: Bypassing ten detection methods,” in *10th ACM Workshop on Artificial Intelligence and Security*, Dallas, Texas, USA, pp. 3–14, 2017.

[16] F. V. Massoli, F. Carrara, G. Amato and F. Falchi, "Detection of face recognition adversarial attacks," *Computer Vision and Image Understanding*, vol. 202, pp. 103103, 2021.

[17] A. Agarwal, R. Singh, M. Vatsa and N. Ratha, "Are image-agnostic universal adversarial perturbations for face recognition difficult to detect?" in *2018 IEEE 9th Int. Conf. on Biometrics Theory, Applications and Systems (BTAS)*, Redondo Beach, CA, USA, pp. 1–7, 2018.

[18] K. Grosse, P. Manoharan, N. Papernot, M. Backes and P. Mcdaniel, "On the (Statistical) detection of adversarial examples," arXiv:1702.06280, 2017. http://arxiv.org/abs/1702.06280

[19] Z. Gong, W. Wang and W. S. Ku, "Adversarial and clean data are not twins," arXiv:abs/1704.04960, 2017.

[20] W. Xu, D. Evans and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," arXiv:1704.01155, 2017. http://arxiv.org/abs/1704.01155

[21] J. H. Metzen, T. Genewein, V. Fischer and B. Bischoff, "On detecting adversarial perturbations," in *Int. Conf. on Learning Representations*, Toulon, France, 2017.

[22] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *28th Annual Conf. on Computer Graphics and Interactive Techniques*, New York, NY, USA, pp. 341–346, 2001.

[23] S. M. Moosavi-Dezfool, A. Fawzi and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Los Alamitos, CA, USA, pp. 2574–2582, 2016.

[24] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *IEEE Symp. on Security and Privacy (SP), IEEE Computer Society*, Los Alamitos, CA, USA, pp. 39–57, 2017.

[25] Z. Liu, Q. Liu, T. Liu, N. Xu, X. Lin *et al.,* "Feature distillation: DNN-oriented jpeg compression against adversarial examples," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 860–868, 2019.

[26] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li *et al.,* "Imagenet: A large-scale hierarchical image database," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 248–255, 2009.

[27] S. A. Kilany, A. Mahfouz, A. M. Zaki and A. Sayed, "Analysis of adversarial attacks on face verification systems," in *Int. Conf. on Artificial Intelligence and Computer Vision (AICV2021)*, Morocco, vol. 1377, pp. 463–472, 2021.

[28] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh *et al.,* "Ensemble adversarial training: Attacks and defenses," in *Int. Conf. on Learning Representations*, Vancouver, BC, Canada, 2018.

[29] D. Hendrycks and K. Gimpel, "Early methods for detecting adversarial images," in *ICLR (Workshop)*, 2017. http://dblp.uni-trier.de/db/conf/iclr/iclr2017w.html#HendrycksG17a

[30] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed *et al.,* "Going deeper with convolutions," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA, pp. 1–9, 2015.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd Int. Conf. on Learning Representations (ICLR 2015)*, San Diego, CA, pp. 1–14, 2015.

[32] G. B. Huang, M. Mattar, T. Berg and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," *Technical Report 07-49*, University of Massachusetts, Amherst, Massachusetts, USA, 2007.

[33] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[34] D. Yi, Z. Lei, S. Liao and S. Li, "Learning face representation from scratch," arXiv:1411.7923, 2014. https://arxiv.org/abs/1411.7923

[35] D. Deb, X. Liu and A. K. Jain, "FaceGuard: A self-supervised defense against adversarial face images," in *IEEE 17th Int. Conf. on Automatic Face and Gesture Recognition (FG)*, Waikoloa Beach, HI, USA, pp. 1–8, 2023.

[36] D. D. Muresan and T. W. Parks, "Adaptive principal components and image denoising," in *Int. Conf. on Image Processing*, vol. 1, pp. 1–101, 2003.

[37] B. Arjun, C. Daniel and M. Prateek, "Dimensionality reduction as a defense against evasion attacks on machine learning classifiers," arXiv:1704.02654, 2017. http://arxiv.org/abs/1704.02654

[38] J. Zhou, C. Liang and J. Chen, "Manifold projection for adversarial defense on face recognition," in *Computer Vision—ECCV 2020—16th European Conf.*, Glasgow, UK, Springer, vol. 12375, pp. 288–305, 2020. https://doi.org/10.1007/978-3-030-58577-8_18

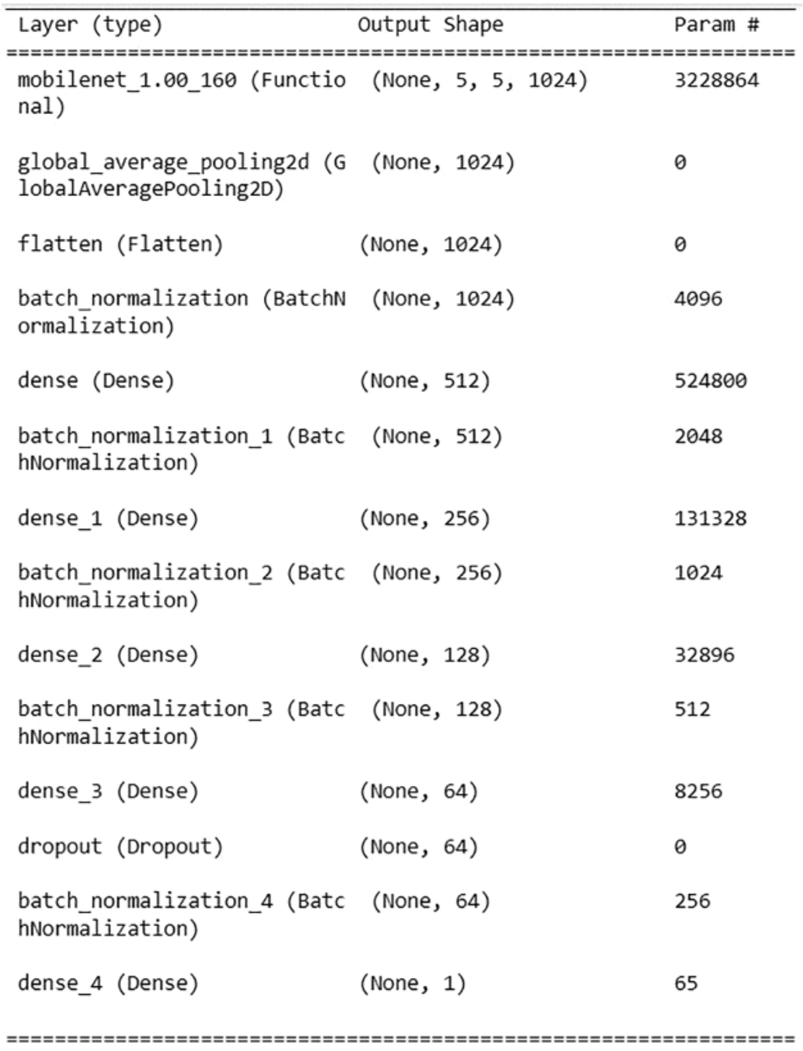
[39] K. He, X. Zhang, S. Ren and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision—ECCV 2016: 14th European Conf.*, Amsterdam, The Netherlands, Springer, pp. 630–645, 2016.

[40] A. Graps, "An introduction to wavelets," *IEEE Computational Science and Engineering*, vol. 2, pp. 50–61, 1995.

[41] A. Athalye, N. Carlini and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *35th Int. Conf. on Machine Learning*, Stockholm, Sweden, pp. 274–283, 2018.

Supplementary Materials

A. Implementation Architecture of the proposed model



```
 Layer (type)                Output Shape              Param #
=================================================================
 mobilenet_1.00_160 (Functio  (None, 5, 5, 1024)       3228864
 nal)

 global_average_pooling2d (G  (None, 1024)             0
 lobalAveragePooling2D)

 flatten (Flatten)           (None, 1024)              0

 batch_normalization (BatchN  (None, 1024)             4096
 ormalization)

 dense (Dense)               (None, 512)               524800

 batch_normalization_1 (Batc  (None, 512)              2048
 hNormalization)

 dense_1 (Dense)             (None, 256)               131328

 batch_normalization_2 (Batc  (None, 256)              1024
 hNormalization)

 dense_2 (Dense)             (None, 128)               32896

 batch_normalization_3 (Batc  (None, 128)              512
 hNormalization)

 dense_3 (Dense)             (None, 64)                8256

 dropout (Dropout)           (None, 64)                0

 batch_normalization_4 (Batc  (None, 64)               256
 hNormalization)

 dense_4 (Dense)             (None, 1)                 65

=================================================================
```