Iris Liveness Detection using MobileNetV3 and EfficientNet using LivDet Dataset

Manimozhi. G
M. Tech. CSE
Pondicherry University,
Pondicherry, India
ganapathymanimozhi@gmail.com

Sukhvinder Singh
Faculty of Computer Science Department,
Pondicherry University,
Pondicherry, India
sukh1in@gmail.com

***Abstract*--Nowadays Iris attacks have become a common thing, even though the iris has unique features to differentiate each person, hackers manipulate the biometric authentication system using various types of fake iris images such as printed iris images, contact lens images, patterned iris images, and cadaver iris images. There are a number of techniques or models that have already been proposed for Iris Liveness Detection using Deep Learning Networks. In this paper, a three-class scenario is suggested that uses a novel serial architecture for Presentation Attack Detection based on MobileNetV3 and EfficientNetB0 to improve the efficiency, and accuracy and reduce the relevant error rates of the system by making it lighter. The MobileNetV3 and EfficientNet will be trained from scratch to distinguish between presentation attack images and genuine iris images. The bonafide class comprises live iris pictures, while the assault show instrument class comprises of printed, and patterned riris pictures for three species. The result from the MobileNetV3 and EfficientNetB0 will be compared with other existing deep learning networks such as VGG-19 and MobileNetV2. From the experimental results and analysis, we can see that the MobileNetV3 performs better than the other deep learning networks by providing improved accuracy, efficiency with less time. The proposed models gives out 0.24% APCER and 0.12% BPCER for MobileNetV3 and 0.38% APCER and 0.23% BPCER for EfficientNet. MobileNetV3 takes 19 milliseconds for each step whereas EfficientNet takes 34 milliseconds and VGG-19 takes 85 milliseconds.**

Keywords--Biometric authentication system, Iris Liveness Detection, Presentation Attack Detection, deep learning networks, confusion matrix.

I. INTRODUCTION

Iris Liveness Detection systems have been widely used today for distinguishing between live iris and fake iris images. These systems are widely used in applications where security is paramount, including financial institutions, border control system, hospitals and clinics, immigration at the airports, to unlock various devices and smart phones, aviation security, various government agencies, and many more. Infrared sensors and lightings are used for detecting the originality of the iris in Iris recognition systems [1]. However, this technology is not immune to security breaches, especially when attackers attempt to present false images of iris patterns to trick the system, a type of attack known as presentation attack. To counteract presentation attacks, iris liveness detection systems based on presentation attack instruments [2] have been developed to ensure the iris images presented are from live sources and not fake or manipulated ones. Iris liveness detection systems based on presentation ttack instruments use various techniques and methods to detect the presence of presentation attacks, including analysing the dynamic features of the live iris, modelling the characteristics of iris images, and differentiating between live and manipulated images.

Neural Networks play a vital role in iris recognition systems. Recently, the utilization of deep learning models such as MobileNetV3 and EfficientNet for liveness detection has been on the rise due to their high accuracy and speed. In this context, the Iris Liveness Detection System using MobileNetV3 and EfficientNet is a cutting-edge solution that presents a highly accurate and efficient way to authenticate individuals through iris scanning. Iris Liveness Detection System using MobileNetV3 and EfficientNet is a state-of-the-art biometric authentication system that ensures the highest level of security for applications that require reliable identity verification.

Iris scanning is considered to be one of the most reliable biometric parameters for identity verification, however, various liveness attacks such as printing iris images, using contact lenses or even video replay attacks are common issues in iris authentication systems. To overcome these challenges, the Iris Liveness Detection System utilizes deep learning models such as VGG-19, MobileNetV2, MobileNetV3 [3], EfficientNet [4], etc., which are highly accurate and efficient in detecting such attacks. The system captures an iris image of the user, processes it using these deep learning models and detects whether the iris is real or not. Compared to traditional liveness detection methods, this system provides a highly accurate and efficient authentication solution that is resistant to even the most sophisticated liveness attacks.

In this paper, MobileNetV3 and EfficientNet are implemented using LivDet-2017 dataset [5] and compared with VGG-19 and MobileNetV2 and among each other to see which one gives better accuracy and has high efficiency with low error rates. The main involvement of this work can be summarized as follows:

A. Architecture:

Here a novel architecture is proposed where MobileNetV3 and EfficientNet models were trained from scratch to differentiate between bonafide presentation and presentation attack images such as pattern (contact lens) and printed images from Attacks by imposters involve determining the actual origin of the image. MobileNetV2 and VGG-19 architecture are used to compare with the proposed model.

B. Network Inputs:

The inputs given here to the networks include bonafide images, printed and patterned images. Also, using one of these three sorts of input, distinct and thorough tests were carried out, and the outcomes were examined.

C. Weights:

Balanced weights are for representing the number of images per class. The weights of the databases can be mostly be unbalanced, without balancing the databases prediction won't be accurate. So, in order to avoid errors balancing of the dataset is a must to get realistic results.

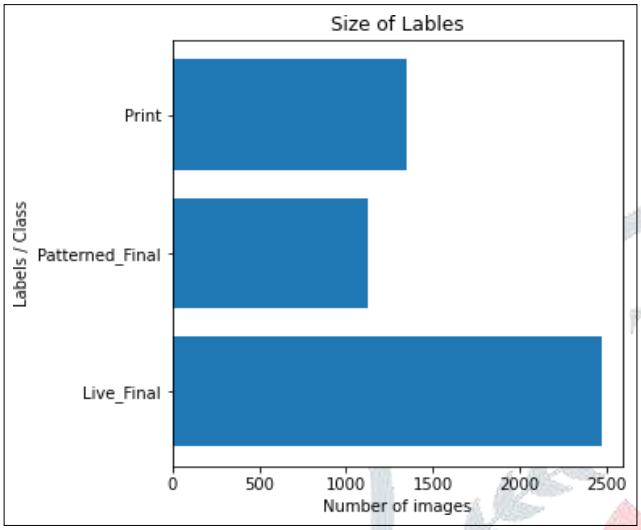


Figure 1. Number of images in each class

D. Database:

This system is trained from scratch and implemented using the Clarkson Dataset for LivDet-Iris 2017 Database. A total of 4937 images are in this dataset with a resolution of 128*128 pixels. The number of live images is 2469, printed images is 1122 and pattern images is 1346. The dataset is then divided as training and testing dataset.

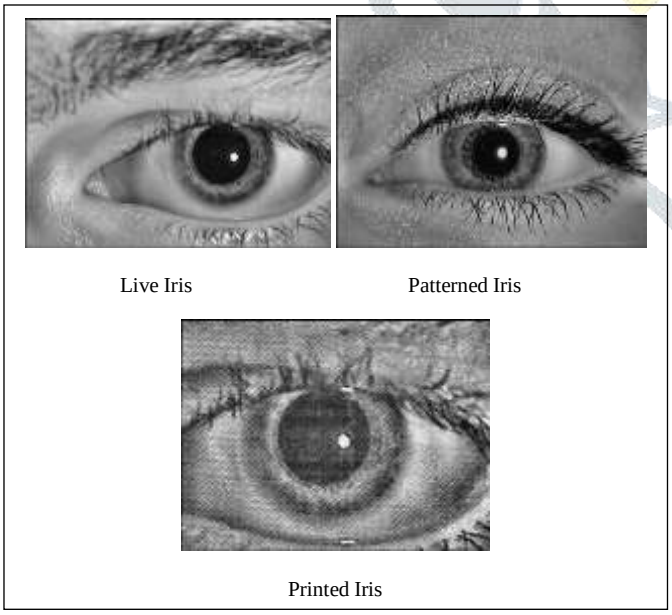


Figure 2. Types of Iris images from LivDet-Iris 2017 Dataset

The above figure shows the types of Iris images from LivDet-Iris 2017 dataset which are live iris, patterned iris and printed iris images.

E. Algorithm:

Adam optimization algorithm is used here for training the deep neural networks. It is a combination of Adaptive Gradient and Root Mean Square Propagation algorithms [6] to provide an optimization algorithm that can handle sparse gradients on noisy problems.

The remaining of the article is organized as follows: Section II summarizes the existing methods of the iris liveness detection systems and the various methods used. Section III demonstrates the metrics that are used to measure the error rates and accuracy of the models. Section IV shows the methodology that have been used in this paper. Section V has experimental results and comparison of the implemented method and existing methods. Finally, the whole work is concluded in Section VI.

II. LITERATURE SURVEY

Iris liveness detection systems based on neural networks have received growing attention in recent years, as they have shown promising results in detecting presentation attacks in iris recognition systems. This section provides a literature review of previous studies on iris liveness detection using neural networks.

A study by Luiz et al. [5] proposed an iris liveness detection system based on deep neural networks. Their proposed system used the ResNet-50 convolutional neural network (CNN) architecture to classify iris images as real or fake. The system was tested on the UBIRIS database from NICE.II competition and achieved an Equal error rate (EER) of 13.98%.

An iris presentation attack detection CNN model is presented in [6], where the cross-dataset capabilities of CNNs are also shown. Most of the models used in iris liveness detection systems are convolutional neural network models. There are various layers in CNN models which are trained for better accuracy.

In [7], the MVANet utilizes the deep convolutional neural network architecture for presentation attack detection. It is trained from scratch and compared with the other baseline neural networks. This network requires an average of 6 minutes and 47 seconds per epoch. It required less time to train when compared with the other baseline networks such as DenseNet, ResNet, etc.

Meenakshi *et al.* [8] proposed an ensemble e of Customized DenseNet and SVM approach by using an Iris Contact Lens Detection and Classification problem. Here, SoftMax classifier has been replaced with SVM, which has given a better performance. They have used IIIT-Delhi Contact Lens and Notre Dame Contact Lens 2013 Database; this model has improved Correct Classification Rate (CCR) up to 4%.

Meiling *et al.* [9] proposed a system for detecting iris presentation attacks that especially for detecting contact lenses based on the standard multiple micro stripes. The authors also analyse the effect of different parameters on the performance of the approach, such as the size of the micro stripes and the number of stripes used for analysis. They show that the approach is relatively robust to these parameters and can achieve high accuracy with a small number of micro stripes.

In [10], a two-head 'compression development' convolutional brain organization (CNN) design for strong show assault discovery is proposed. The findings demonstrate that, in terms of accuracy and robustness to various attacks, the proposed method outperforms current methods by 5.3% and 10%, respectively. The creators likewise dissect the impact of various boundaries on the exhibition of the methodology, for example, the number of convolutional layers and the size of the channels utilized in the organization. They demonstrate that the method can achieve high accuracy with a limited number of convolutional layers and is relatively resistant to these parameters..

Federico Pala and Bir Bhanu [11] proposed an approach based on comparing the relative distances between different regions of the iris image. The model is evaluated on three different iris databases. The results show that the proposed approach outperforms existing models. The proposed model is also computationally efficient since it does not require complex feature extraction or classification algorithms.

Iris liveness detection using regional features [12], in this paper an approach is proposed based on extracting regional features from the iris image using a grid-based approach. The authors analysed the effect of different parameters on the performance of the approach, such as the size of the grid used for feature extraction and the number of regions used for classification. They show that the approach is relatively robust to these parameters and can achieve high accuracy with a small number of regions. The error rate of the proposed method is generally higher than state-of-the-art features, but here the advantage of using regional features with respect to the corresponding low-level features is shown. The proposed approach is also computationally efficient since it does not require complex feature extraction or classification algorithms.

Mateusz *et al.* [13] proposed an approach based on a deep learning-based image segmentation method that can accurately segment the iris from the surrounding tissues. They have said that accurate segmentation is crucial for post-mortem iris recognition since the iris texture can change after death. When compared to the 16.89% and 5.37 percent of EER that were observed for OSIRIS and Iri Core, respectively, the proposed method achieves less than 1% of the EER for samples collected up to 10 hours after death. The proposed method has an EER of 21.45% for samples taken up to 369 hours after the death, while OSIRIS and Iri Core have EERs of 33.59% and 25.38 percent, respectively.

In [14], the work is done using residual images obtained from the difference between the original iris image and its smoothed version. The residual images are then transformed using BSIF to extract binary features that can distinguish between live and non-live iris images. Clarkson contact lens database was used for evaluation in both modes segmented and unsegmented eye images. The results were promising in the unsegmented scenario and the three filters enhanced in the results with 8.6667%, 10%, and 18.3333%. The best overall result was obtained by our proposed method using the first filter in segmented mode with Correct Classification Rate (CCR) of 93.3333%.

In order to enhance iris liveness detection and contact lens identification, multiple binarized statistical image feature (BSIF) and Dual Closed-loop Network (DCLNet) score-level fusion is used in [15]. During the testing period, this model uses three feature extraction steps: one for DCLNet and two for MBSIF feature extraction from raw and normalized iris pairs. These require a limited quantity of test time to group the validation endeavour as live or assault.

Fusion of discrete cosine transforms (DCT) and Zernike moments for Iris Liveness Detection utilizing upgraded off-angle iris data set [16]. Zernike moments on the extracted region of interest are separately calculated using the discrete cosine transform and statistical features derived from it. An outrageous learning machine classifier is utilized to compute liveness identification precision, from which equivalent blunder rate is determined, and region under bend is determined. DCT and Zernike moment features are combined by selecting maximum values for statistical features following all individual testing. Execution of the framework is determined as equal error rate (EER) and area under the curve (AUC) from collector working attributes. The higher the EER system's performance, the lower its value. In this model, the pre-handling time is decreased for improved proficiency in the framework.

In [17], an algorithm is made by encoding the textural differences between real and attack iris images by combining VGG features (MHVF) with local and global Haralick texture features in the multi-level Redundant Discrete Wavelet Transform domain. This calculation is tried on a huge iris dataset containing in excess of 270,000 genuine and went after iris pictures and yields a complete blunder of 1.01%. The dataset for this is developed by combining multiple databases. The result of the MHVF is more accurate because of the fusion than when the MH and VGG are tested alone. Each and every dataset are tested with all the three models, in which MHVF gives out the better result on the largest dataset that's been developed by them.

In conclusion, the use of neural networks, specifically CNNs, have shown to be effective in iris liveness detection systems based on presentation attack instruments. These systems have shown superior performance compared to traditional methods and possess the potential to enhance the security of iris recognition systems.

III. METRICS

In this section, the metrics used in this paper are discussed which are APCER and BPCER. The ISO/IEC 30107-3 standard provides guidelines on how to assess the effectiveness of Presentation Attack Detection (PAD) algorithms used in biometric systems. Two of the metrics defined in this standard are the APCER (Attack Presentation

Classification Error Rate) and BPCER (Bonafide Presentation Classification Error Rate). To understand this, first confusion matrix should be understood. Confusion matrix is a 2×2 for binary classification matrix with actual values on one axis and predicted on another [22].

A. APCER:

The Attack Presentation Classification Error Rate measures the percentage of attack presentations, for each specific Presentation Attack Instrument (PAI), that are incorrectly classified as genuine (bonafide) presentations. In simpler terms, it quantifies the rate at which fake or manipulated biometric samples are mistakenly identified as real ones by the PAD algorithm.

The APCER metric calculates the percentage of attacks that are mistakenly categorised as bonafide (live) presentations for each unique PAI. For the purpose of calculating this measure for each PAI, the worst-case situation is taken into account.

$$APCER_{PAIS} = 1 - (\frac{1}{N_{PAIS}}) \sum_{i=1}^{NPAIS} RES_i \quad (1)$$

Where, the value of N_{PAIS} corresponds to the number of attack presentation images, where RES_i for the *i*th image is 1 if the algorithm classifies it as an attack presentation (patterned and printed iris), or 0 if it is classified as a bonafide presentation (live iris).

B. BPCER:

The Bonafide Presentation Classification Error Rate metric measures the proportion of bonafide (live images) presentations incorrectly classified as attacks presentations to the iris liveness detection, or the ratio between wrong rejection to total honest attempts.

$$\text{BPCER} = \frac{\sum_{i=1}^{N_{BF}} RES_i}{N_{BF}} \quad (2)$$

Equation 2 describes how to calculate the BPCER metric. N_{BF} stands for the number of bonafide (live) presentation images, and RES_i uses the same values as the APCER metric.

IV. METHODOLOGY

In this section, the networks and algorithm used in this paper will be discussed in detail and also about the process involved in implementing the paper.

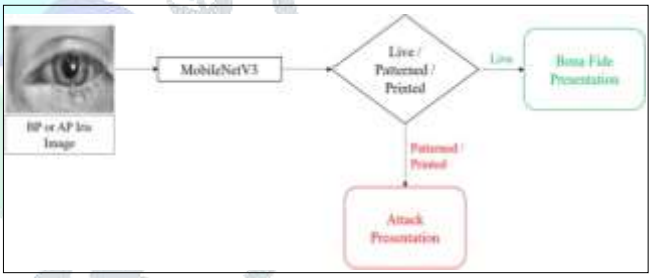
A. Networks:

1) MobileNetV3: It is a convolutional neural network that is tuned to cell phone computer chips through a mix of hardware-aware network architecture search (NAS) supplemented by the NetAdapt algorithm, and afterward further developed through novel design progresses. MobileNetV2 with the hard wash initiation and Crush and Excitation modules is the MobileNetV3 design [3]. It is fundamentally utilized for driving the picture examination abilities of numerous well-known mobile applications. MobileNetV3 is 4.6% more exact and 25% quicker than MobileNetV2 [23]. MobileNetV3 consists of 28 layers including deep convolution layer, 1 × 1 point convolution layer, batchnorm [24], ReLU, average collecting layer and SoftMax.

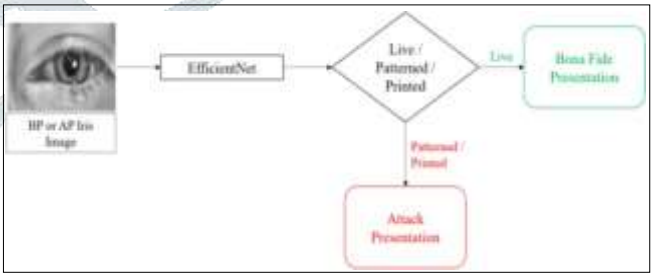
NetAdapt algorithm is a network shrinking algorithm to adapt a pretrained network to a mobile platform given a real resource budget. NetAdapt can incorporate direct metrics, such as latency and energy, into the optimization to maximize the adaptation performance based on the characteristics of the platform. MobileNetV3 uses Network Architecture Search (NAS), which uses the NetAdapt algorithm for refining the layers in the model.

2) EfficientNet: A compound coefficient is used to uniformly scale all depth, width, and resolution dimensions in this convolutional neural network architecture and scaling method [4]. In addition to squeeze-and-excitation blocks, the base EfficientNet-B0 network is built on MobileNetV2's inverted bottleneck residual blocks. Automated machine learning and model scaling, it is used to increase accuracy and efficiency.

EfficientNetB0 has 237 layers with 5 modules. Module 1 is the beginning stage for the sub-blocks. Module 2 is utilized as a beginning stage for the primary sub-block of the multitude of 7 principal blocks with the exception of the first one. Module 3 is utilized for interfacing a skip association with all the sub-blocks. Module 4 is for joining the skip association in the main sub-blocks. Furthermore, module 5 is utilized for associating each sub-block to its past sub-block in a skip association and they are joined utilizing this module. These modules are further combined to form sub-blocks which will be used in a certain way in the blocks. EfficientNet has convolution layer, mobile inverted bottleneck convolution layer, average pooling, flatten layer, dense layer, and SoftMax.



(a)



(b)

Figure 3. Proposed novel serial framework for Presentation Attack Detection (a) MobileNetV3, (b) EfficientNet

In this paper, two additional network architectures are used such as MobileNetV3 and EfficientNet. MobileNetV3 and EfficientNet were trained from scratch, based on bonafide and fake species images such as patterned (contact lens), and printed images.

B. Class Weights:

A weight factor has been calculated for each class of images according to the number of images of the class, which helps in balancing the database. By rescaling the gradient steps during training, class weights are applied to the loss function, favoring under-representation and penalizing over-representation of classes. The below equation is for calculating the weights of each class of images.

$$\text{Weight}_i = \frac{\text{Nsamples}}{\text{Nclasses} \times \text{samples}_i} \quad (4)$$

Where Weight_i is the weight for class i, Nsamples is the total number of images in the database, Nclasses is the total number of classes in the database, and samplesi is the number of samples of class i. The weight values associated to each class are the following:

- Class 0, Bonafide (Live): 0.5001
- Class 1, Pattern: 0.2272
- Class 2, Print (Contact Lens): 0.2726

C. Networks Parameters:

The number of trainable boundaries and a number of duplicate extras can be altered by utilizing MobileNetV3's alpha parameter, which increments/diminishes the number of channels in each layer of the organization. This alpha worth is known as the profundity multiplier in the first MobileNet execution, the alpha values can be at three potential values, for example:

- If Alpha = 1.0, the default number of filters from the real MobileNet paper are used at each layer.
- If Alpha < 1.0, the number of filters in each layer is proportionally decreased.
- If Alpha > 1.0, the number of filters in each layer is proportionally increased.

For the experiments here we have taken the alpha parameter as 1.0 maximum and 0.0 minimum. All experiments used a limit of 30 epochs to train the networks. Here, the loss function (error) was found using sparse categorical cross-entropy.

D. Adam Optimization Algorithm:

It is a replacement optimization algorithm for stochastic gradient descent which is used for training deep learning neural network models [25]. The Adam optimization algorithm uses the following functions, *f(θ)* is noisy objective function, g_t denote the gradient, *t* is the timestamp, β1 and β2 are the hyper parameters, m_t and v_t are the moving averages of the gradient and squared gradient respectively. α is the step size or learnilng rate. The learning rate given here for the MobileNetV3 is 0.0001and for EfficientNet is 0.00001.

This algorithm combines the best properties of the Adaptive Gradient and Root Mean Square Propagation algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems. The advantages of using Adam optimization are that it has faster computation time, it works well with sparse gradients, and the number of parameters required for tuning is less. In most cases, the results of Adam optimization are better than other optimization algorithms.

E. Algorithm for the Proposed Work:

Input: Images from LivDet-Iris 2017 Dataset.

Output: Live or Patterned or Printed Iris.

1. Balancing the dataset using class weight (Live, Patterned and Printed Iris).
2. Reading the dataset.
3. Normalization with 3 channels:
   a. Live = 0
   b. Patterned = 1
   c. Printed = 2
4. Splitting the dataset using stratify:
   a. Train and Test dataset
   b. x_train = 3702 ; x_test = 1235 are the number of inputs
   c. y_train = 3702 ; y_test = 1235 are the number of outputs
5. MobileNetV3 Architecture:
   a. 30 Epochs using the test dataset.
   b. Model accuracy and model loss is shown in the form of graph with respect to epochs.
   c. Confusion matrix is given for the MobileNetV3.
6. EfficientNet Architecture:
   a. 30 Epochs using the test dataset.
   b. Model accuracy and model loss is shown in the form of graph with respect to epochs.
   c. Confusion matrix is given for the EfficientNet.
7. Inference: Output is given.

Algorithm 1. Iris Liveness Detection using MobileNetV3 and EfficientNet

The above algorithm shows the implementation flow of the proposed work; first the dataset is balanced using class weight, there is a total of 3 classes which are live, patterned and printed iris images. Then the dataset is read after which normalization is done to give labels for each class, where live is 0, patterned is 1 and printed is 2. In step 4, the dataset is divided into train and test dataset using stratify then MobileNetV3 and EfficientNet architecture are implemented and 30 epochs are taken to train the dataset. Finally, in the inference the output of the models are obtained.

V. EXPERIMENTAL RESULTS

In this paper, the attack presentation images, and the number of images per class are taken into account. These images present a problem for the classifiers such as SGD classifier, RandomForest classifier and Kneighbors classifiers because the PAI species are not equally represented. This unbalanced dataset needs to be balanced to prevent false detections. Our network MobileNetV3 and EfficientNet is trained on three classes such a live, printed and pattern (contact lens) images. To study the limitations and to improve the performance of the network models' various experiments are conducted in order to analyse the best hyper-parameter configuration of MobileNetV3 and EfficientNet. The experiments are done on VGG-19 [26], MobileNetV3 and EfficientNet and then the results are compared to determine which neural network works the best and provides better accuracy and efficiency. All the networks were trained with a limit of 30 epochs. The LivDet-Iris 2017 dataset is used with an input size of the image as 128 × 128 pixels. The number of images used in each experiment was the same. A series of experiments were conducted using MobileNetV3 and EfficientNet and they are evaluated using the metrics APCER (Attack Presentation Classification Error Rate) and BPCER (Bonafide Presentation Classification Error Rate) Google Colaboratory and they are discussed below.

A. Experiment 1:

The experiment is done on VGG-19, MobileNetV2, MobileNetV3, and EfficientNet architectures which were trained with fine-tuning techniques. Numerous numbers of tests were performed on these architectures. For this experiment, the images were assembled into two classes: Bonafide and Fake. All of the various species of Presentation Attack Instruments are included in the Fake dataset: Printout, Patterned (Contact Lenses).

B. Experiment 2:

A changed MobileNetV3 and EfficientNet networks were prepared without any preparation. For this analysis, the pictures were again gathered in two classes: Bonafide and Presentation attack with different PAI. The AP dataset is contained all Presentation Attack Instrument classes: Contact Lenses (Patterned), Printout.

C. Experiment 3:

For this experiment, a modified MobileNetv3 and EfficientNet networks were trained from scratch. The images were grouped in three classes this time: Live, Contact lenses (patterned) and Printouts.

D. Experiment 4:

When these PAI species were not included in the PAD algorithm's training set, our proposed novel method was tested in this experiment to see if it would work with these species. Using a cross-validation strategy of leave-one-PAI-species-out, two networks with two stages were trained. Printed and bonafide images were used to train the first model, and patterned contact lenses from PAI species were used to evaluate it. Additionally, printed PAI species were used in the evaluation of the second model, which was trained with genuine and patterned contact lenses. To find the loss function (error), all networks also go through Sparse categorical cross-entropy.

E. Results:

In this section, the results for the implemented methods are explained in detail. Adam optimizer is used with a learning rate of 0.0001 for VGG-19 and MobileNetV3 and 0.00001 for EfficientNet. The model accuracy and model loss are shown in the form of graph for the 30 epochs, the minimum value is 0.0 and the maximum value is 1.0 for the alpha parameter (α). Global max pooling is performing better compared to global normal pooling. An alpha value of 1.0 performed better with the three class situations, with an input picture size of 128 × 128. The input will be as (3702, 128,128, 3) and (1235, 128, 128, 3) where 3702 and 1235 indicates the number of images in train and test dataset, 128, 128 means the height and width of the image and 3 means channels such as Blue, Green, Red, and the output will be in the form of (3702) and (1235) which indicates that the output will be from that particular number of images.

Figure 4 is about VGG-19 where the graph for model accuracy and model loss for the 30 epochs are shown and figure 5 shows the confusion matrix of the VGG-19 architecture with the live iris image and PAI species images such as contact lens (patterned) and printed iris images from the LivDet-Iris 2017 dataset, the values in the confusion matrix denote the number of instances that belong to the particular class, for example, 6.2e+02 means 6.2*10^2 = 620.

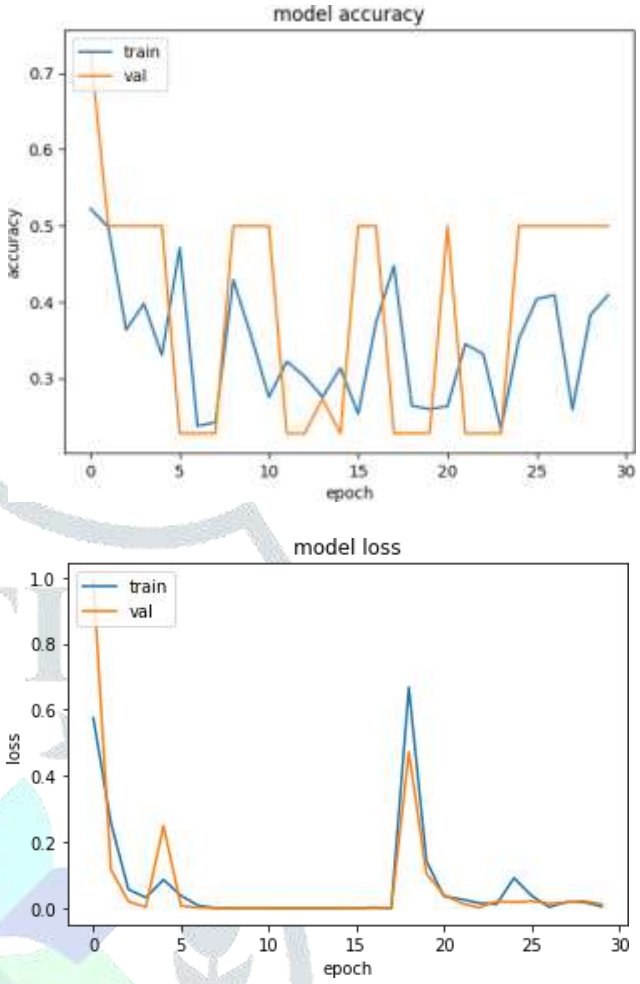


Figure 4. The Accuracy and Error in the VGG-19 architecture using the LivDet-Iris 2017 dataset is shown in the graph

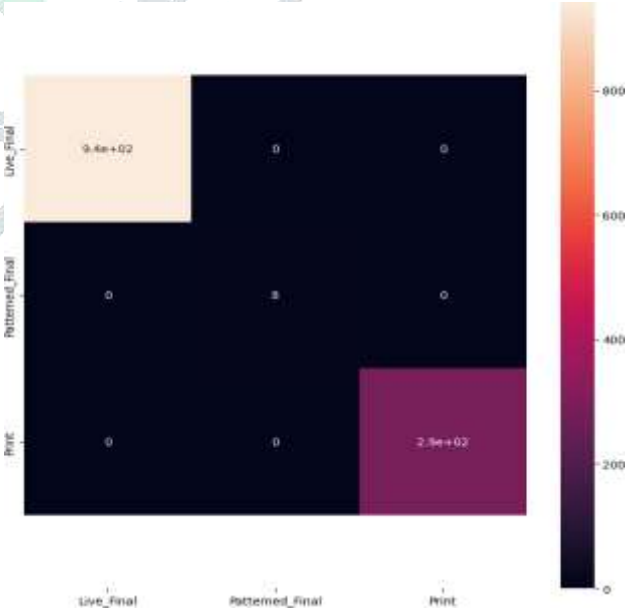


Figure 5. Confusion matrix for the three-class test in VGG-19 architecture

The below figure 6 and 7 shows the graph for MobileNetV2 architecture's model accuracy and model loss for the 30 epochs and the confusion matrix of the MobileNetV3 architecture with the live iris image and PAI species images such as contact lens (patterned) and printed iris images from the LivDet-Iris 2017 dataset.

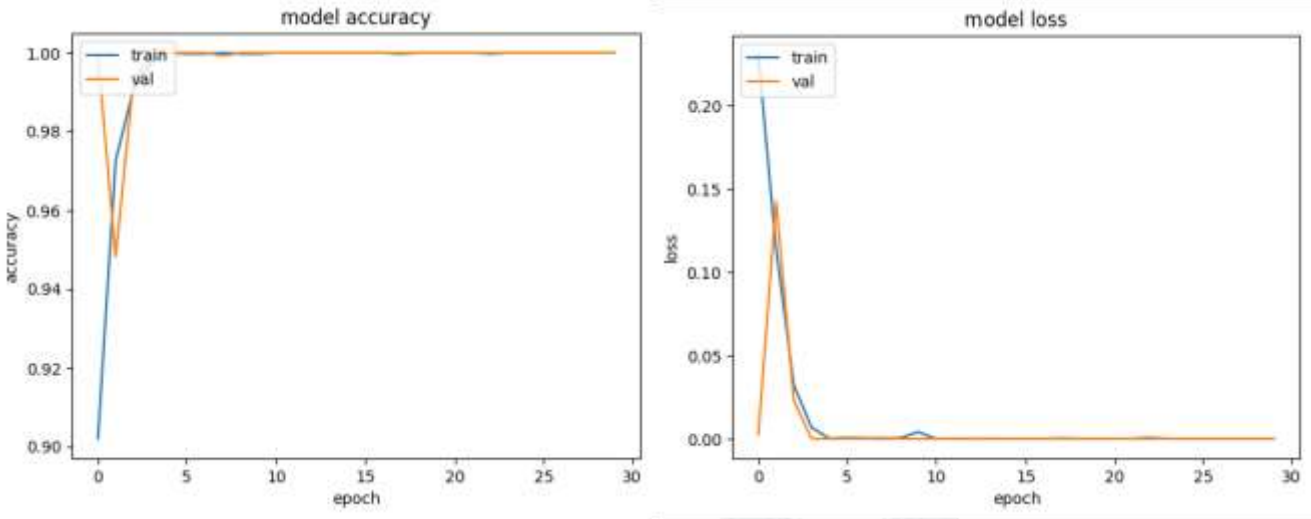


Figure 6. The Accuracy and Error in the MobileNetV2 architecture using the LivDet-Iris 2017 dataset is shown in the graph

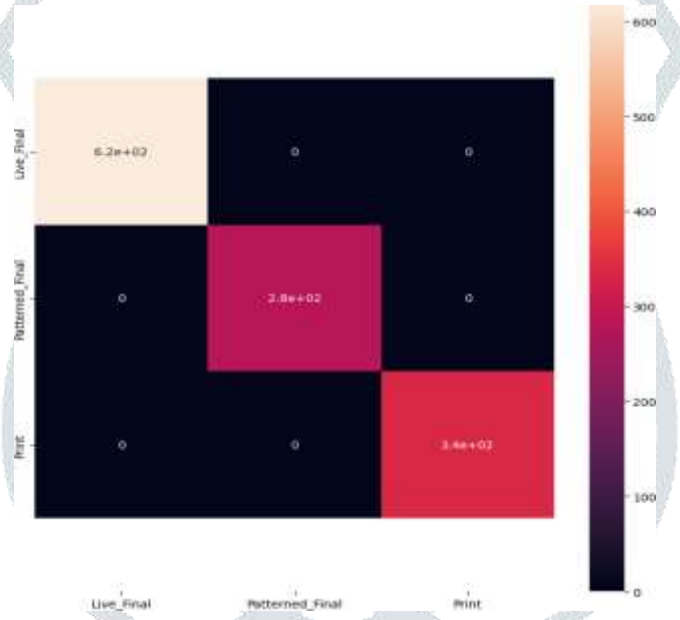


Figure 7. Confusion matrix for the three-class test in MobileNetV2 architecture

The below figure 8 shows the graph for MobileNetV3 architecture's model accuracy and model loss for the 30 epochs. And the figure 9 shows the confusion matrix of the MobileNetV3 architecture with the live iris image and PAI species images such as contact lens (patterned) and printed iris images from the LivDet-Iris 2017 dataset.

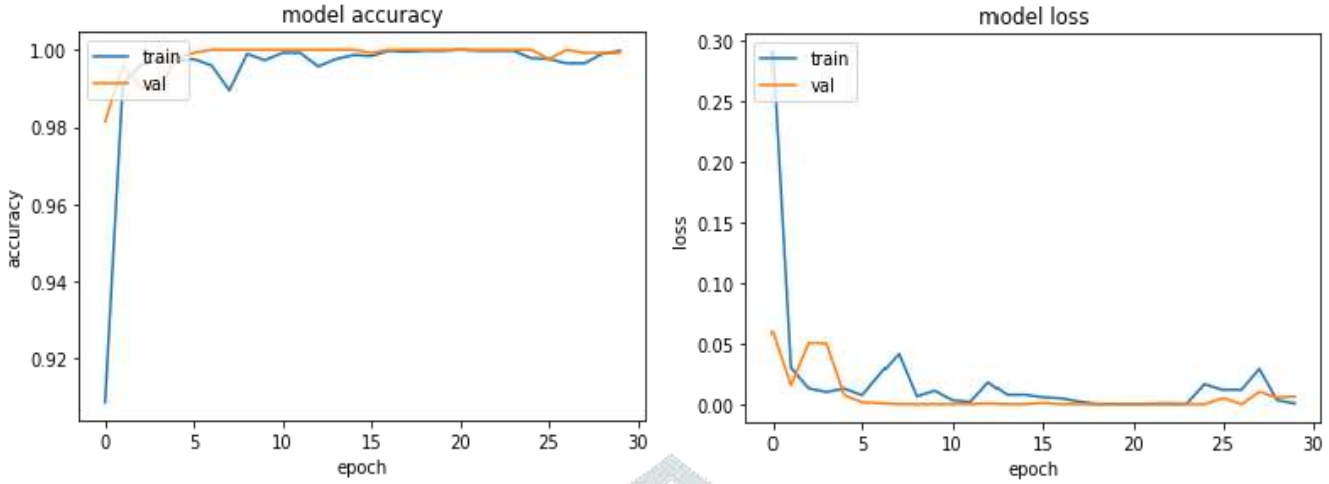


Figure 8. The Accuracy and Error in the MobileNetV3 architecture using the LivDet-Iris 2017 dataset is shown in the graph

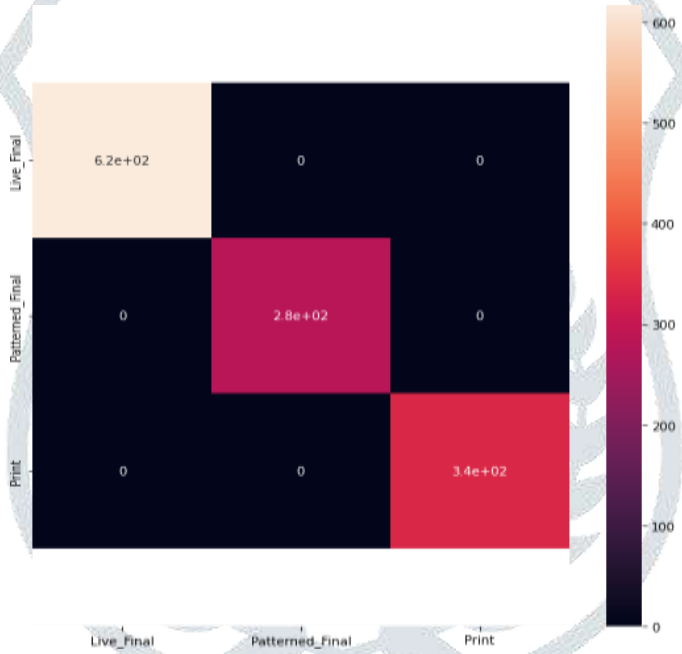


Figure 9. Confusion matrix for the three-class test in MobileNetV3 architecture

Figure 10 is about EfficientNet where the model accuracy and model loss of the 30 epochs are shown. Figure 11 is the confusion matrix of the EfficientNet architecture with the live iris image and PAI species images such as contact lens (patterned) and printed iris images from the LivDet-Iris 2017 dataset.

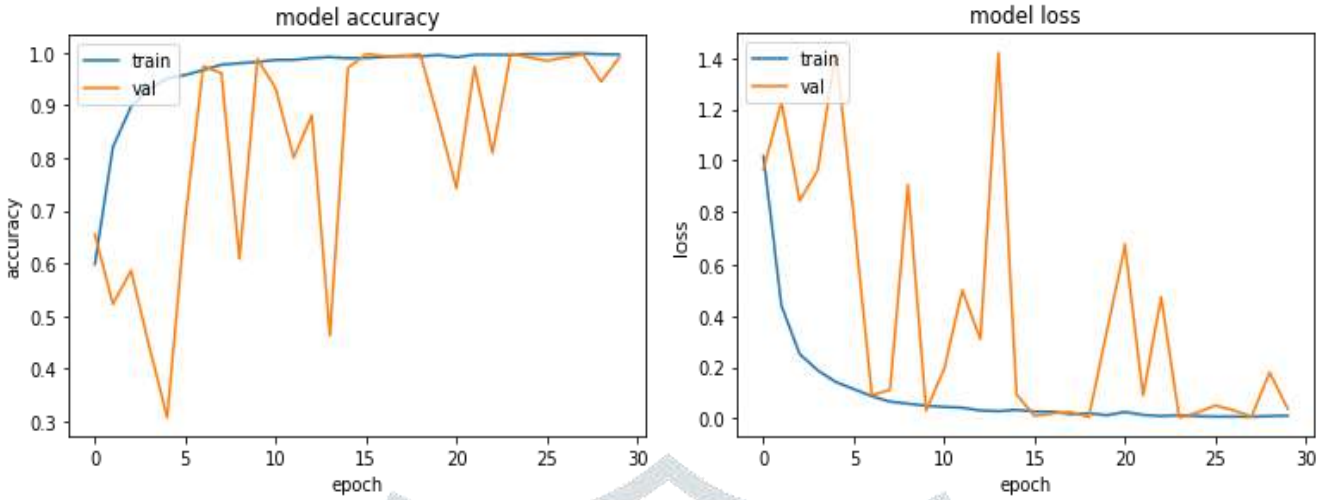


Figure 10. The Accuracy and Error in the EfficientNet architecture using the LivDet-Iris 2017 dataset is shown in the graph

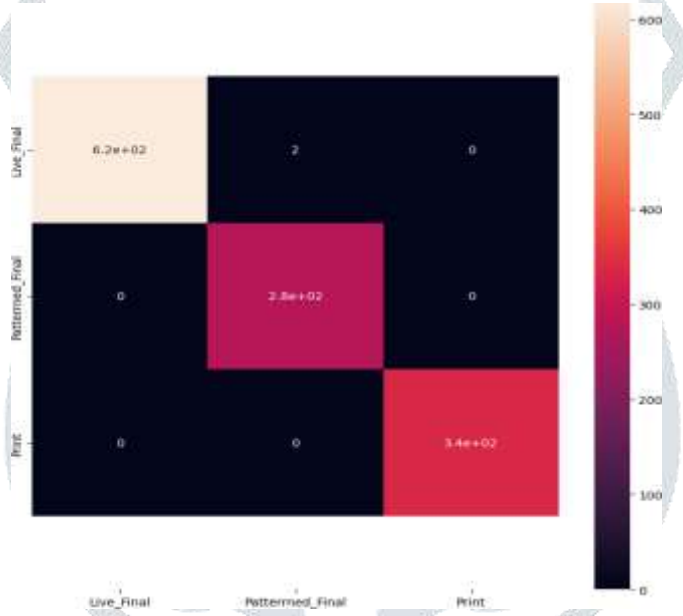


Figure 11. Confusion matrix for the three-class test in EfficientNet architecture

Table II shows the time taken by each model to run 30 epochs and Table III shows the APCER and BPCER values for each of the existing and proposed models for the LivDet-Iris 2017 Dataset. From the above figures and tables, it can be seen that there is a trivial difference in values between VGG-19, MobileNetV2, MobileNetV3, and EfficientNet architectures by using LivDet-Iris 2017 Dataset which consists of live iris, patterned, and printed iris images.

TABLE II. TIME TAKEN BY EACH MODEL.

Models	Time taken for each step (In milli seconds)	Time (In seconds)
VGG-19	85	5
MobileNetV2	54	3
MobileNetV3	19	1
EfficientNet	34	4

TABLE III. APCER AND BPCER FOR EACH MODEL.

Models	APCER (in %)	BPCER (in %)
VGG-19	0.92	0.36
MobileNetV2	0.83	0.16
MobileNetV3	0.24	0.12
EfficientNet	0.38	0.23

F. Comparative Analysis:

The below table gives the comparative analysis of the existing models and proposed models based on the algorithm used, their accuracy and error rates.

TABLE IV. COMPARISON TABLE

S. No.	Title and Ref. No	Algorithm/s	Models	Accuracy
1	Open-source presentation attack detection baseline for iris recognition [2]	Iris PAD algorithm	Uses many models (name not mentioned)	Correct classification rate of 84%.
2	Convolutional neural networks for iris presentation attack detection: Toward cross-dataset and cross-sensor generalization [15]	Presentation Attack Detection Algorithm	CNN Model	FDR of 0.2% on Casia dataset.
3	Iris Liveness Detection Using Fusion of Domain-Specific Multiple BSIF and DenseNet Features [17]	SVM classifier algorithm	Fusion of MBSIF and DCLNet framework	Accuracy of the proposed method ranks 1st in all the experiments (more than 97%).
4	Human off Angle Iris Liveness Detection Based on Fusion of DCT and Zernike Moments [18]	Extreme Learning Machine (ELM) algorithm	Fusion of discrete cosine transform and Zernike moment	EER is less than 0.1% for the proposed method.
5	Fusion of Handcrafted and Deep Learning Features for Large-scale Multiple Iris Presentation Attack Detection [20]	Multi-level Haralick and VGG Fusion (MHVF) algorithm	VGG model (pretrained)	Total error of 1.01%.
6	**Iris Liveness Detection using MobileNetV3 and EfficientNet using LivDet Dataset**	**Adam Optimization algorithm**	**MobileNetV3 and EfficientNet**	**APCER = 0.24%, BPCER = 0.12% (MobileNetV3) APCER = 0.38%, BPCER = 0.23% (EfficientNet).**

VI. CONCLUSION

The existing methods are primarily based on presentation attack detection, on a specific set of datasets, and they are mainly evaluated based on error rates such as APCER, and BPCER with a result of 0.83% and 0.16% respectively. The proposed models gives out 0.24% APCER and 0.12% BPCER for MobileNetV3 and 0.38% APCER and 0.23% BPCER for EfficientNet.

The networks implemented here such as VGG-19, MobileNetV2, MobileNetV3, and EfficientNet are trained from scratch, by using the LivDet-Iris 2017 dataset. The image input size is taken as 128 × 128, which is enough to classify bonafide images successfully. The results of image classification accuracy done using MobileNetV3 and EfficientNet are very similar but the computation time taken by EfficientNet is a bit more which means that MobileNetV3 is efficient than EfficientNet.

For future work, the other EfficientNet models such as B1 – B7 can be tested, and new PAI species should be included with a wider range of dataset, considering, for example, synthetic images, prosthetic iris images.

REFERENCES

[1] Yangyu Chen and Weigang Zhang., “Iris Liveness Detection: A Survey”, in IEEE Fourth International Conference on Multimedia Big Data, 2018.

[2] J. McGrath, K. W. Bowyer, and A. Czajka, “Open-source presentation attack detection baseline for iris recognition”, 2018, arXiv:1809.10172.

[3] A. Howard et al., “Searching for MobileNetV3,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 1314–1324.

[4] M. Tan and Q. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in Proc. Mach. Learn. Res., vol. 97, K. Chaudhuri and R. Salakhutdinov, Eds., Jun. 2019, pp. 6105–6114.

[5] A. Czajka, R. Singh, M. Vatsa, A. Noore, “LivDet-Iris 2017 – Iris Liveness Detection Competition 2017”, In IEEE International Joint Conference on Biometrics (IJCB), pp. 733–741, 2017.

[6] Budi, N., Anny, Y., “Performance of Root-Mean-Square Propagation and Adaptive Gradient Optimization Algorithms on Covid-19 Pneumonia Classification”, In 2022 IEEE 8th Information Technology International Seminar (ITIS), 2022.

[7] Zanlorensi, L. A., Luz, E., Laroca, R., Britto, A. S., Oliveira, L. S., & Menotti, D., “The Impact of Pre-processing on Deep Representations for Iris Recognition on Unconstrained Environments”, 31st SIBGRAPI

Conference a Graphics, Patterns and Images (SIBGRAPI), doi:10.1109/sibgrapi.2018.00044. 2018.

[8] Meenakshi Choudhary, Vivek Tiwari, Venkanna U, "An approach for iris contact lens detection and classification using ensemble of customized DenseNet and SVM". Future Generation Computer Systems, 101, 1259–1270. doi:10.1016/j.future.2019.07.003. 2019.

[9] Meiling F., Naser D., Fadi B., Florian K., Arjan K., "Cross-database and cross-attack Iris presentation attack detection using micro stripes analyses". Image and Vision Computing, 105, 104057. doi:10.1016/j.imavis.2019.104057. 2019.

[10] A. Agarwal, A. Noore, M. Vatsa, R. Singh, "Enhanced iris presentation attack detection via contraction-expansion CNN", Pattern Recognition Letters, 2022.

[11] Federico Pala, Bir Bhanu, "Iris Liveness Detection by Relative Distance Comparisons", in IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). doi:10.1109/cvprw.2017.95, 2017.

[12] Y. Hu, K. Sirlantzis, G. Howells, "Iris liveness detection using regional features",in Pattern Recognition Letters, 82, 242–250. doi: 10.1016/j.patrec.2015.10.010. 2016.

[13] M. Trokielewicz, A. Czajkab, P. Maciejewiczc, "Post-mortem iris recognition with deep-learning-based image segmentation", Image and Vision Computing, 94, 103866. doi: 10.1016/j.imavis.2019.103866. 2019.

[14] James S. Doyle, Kevin W. Bowyer, "Robust Detection of Textured Contact Lenses in Iris Recognition Using BSIF", IEEE Biometrics Compendium, 1672 - 1683. 10.1109/ACCESS.2015.2477470. 2015.

[15] S. Hoffman, R. Sharma, and A. Ross, "Convolutional neural networks for iris presentation attack detection: Toward cross-dataset and cross-sensor generalization," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2018, pp. 1620–1628.

[16] Mehak Gupta, Vishal Singh, Akshay Agarwal, Mayank Vatsa, and Richa Singh, "Generalized Iris Presentation Attack Detection Algorithm under Cross-Database Settings", In 2019 25th International Conference on Pattern Recognition (ICPR), pp. 1051-4651, 2019.

[17] Meenakshi Choudhary, Vivek Tiwari, Venkanna U, "Iris Liveness Detection Using Fusion of Domain-Specific Multiple BSIF and DenseNet Features", IEEE Transactions on Cybernetics, pp. 2370-2381, vol. 52(4), 2022.

[18] S.N. Dharwadkar, Dr. Y.H.Dandawate and Dr. A.S.Abhyankar, "Human off Angle Iris Liveness Detection Based on Fusion of DCT and Zernike Moments", In International Journal of Computing and Digital Systems, pp. 743-751, 2022.

[19] Adam Czajka, "Iris Liveness Detection by Modeling Dynamic Pupil Features", in Advances in Computer Vision and Pattern Recognition, 10.1007/978-1-4471-6784-6_19. 2016.

[20] Daksha Yadav, Naman Kohli, Akshay Agarwal, Mayank Vatsa, Richa Singh, Afzel Noore, "Fusion of Handcrafted and Deep Learning Features for Large-scale Multiple Iris Presentation Attack Detection", In CVPR, 2018.

[21] J. Deng, W. Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database", In IEEE Conference on Computer Vision and Pattern Recognition. doi:10.1109/cvprw.2009.5206848. 2009.

[22] Ioannis, M., Ioannis, R., Ioannis, G., George, K., Anastasios, D., Nikolaos, D., "Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem" MDPI, pp.1-22, 2018.

[23] Mark, S., Andrew, H., Menglong, Z., Andrey, Z., and Liang-Chieh, C., "MobileNetV2: Inverted residuals and linear bottlenecks". In Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., pp. 4510–4520, 2018.

[24] S. Ioffe, C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In Proc. of the 32nd Inter. Conf. on Machine Learning, volume 37, 2015.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, arXiv:1412.6980.

[26] Muhammad Mateen, Junhao Wen *, Nasrullah, Sun Song and Zhouping Huang, "Fundus Image Classification Using VGG-19 Architecture with PCA and SVD", Symmetry, 11(1), 1. 2018. doi:10.3390/sym11010001.