Enhancement of detection accuracy for preventing iris presentation attack

Priyanka Venkatesh, Gopal Krishna Shyam, Sumbul Alam
Department of Computer Science and Engineering, Presidency University, Bengaluru, India

Article Info	ABSTRACT
	A system that recognizes the iris is susceptible to presentation attacks (PAs), in which a malicious party shows artefacts such as printed eyeballs, patterned contact lenses, or cosmetics to obscure their personal identity or manipulate someone else's identity. In this study, we suggest the dual channel DenseNet presentation attack detection (DC-DenseNetPAD), an iris PA detector based on convolutional neural network architecture that is dependable and effective and is known as DenseNet. It displays generalizability across PA datasets, sensors, and artifacts. The efficiency of the suggested iris PA detection technique has been supported by tests performed on a popular dataset which is openly accessible (LivDet-2017 and LivDet-2015). The proposed technique outperforms state-of-the-art techniques with a true detection rate of 99.16% on LivDet-2017 and 98.40% on LivDet-2015. It is an improvement over the existing techniques using the LivDet-2017 dataset. We employ Grad-CAM as well as t-SNE plots to visualize intermediate feature distributions and fixation heatmaps in order to demonstrate how well DC-DenseNetPAD performs.

Corresponding Author:

Priyanka Venkatesh
Department of Computer Science and Engineering, Presidency University
Bengaluru, India
Email: priyankavranju@gmail.com, priyanka.v@presidencyuniversity.in

1. INTRODUCTION

A person is recognized by an iris biometric device based on the iris's textural pattern [1]. Due to their growing use and unattended operation, iris devices are susceptible to presentation attacks. a presentation attack (PA), refers to a deliberate act of presenting biometric data to the subsystem responsible for data capture, with the specific aim of causing interference or disruption in the normal operation of the biometric system. Presentation attack instruments (PAIs) are the tools or biometric characteristics used to begin a presentation attack [2]. With regard to the iris modality, some examples of PAIs are artificial eyes (patterned contact lens, glass, or doll eyes) [3], printed iris images [4], video displays of eye images [5], [6], cosmetic contacts [7], robotic eye models [8], cadaver eyes [6], and holographic eye images [9]. In Figure 1, a few examples of eye PAIs are shown. We must protect the safety of biometric iris systems against these well-known iris PAs and additional PAs that are neither recognized nor encountered during the training phase. Our goal in this effort is to create a PA detection of the iris that is both accurate and appropriate.

Hardware and software-based methods for eye PA mitigation are currently used in the literature. In order to assist in PA detection, hardware-based methods typically call for tangible devices as well as the common iris sensor. Examples include using a camera called IrisCUBE to capture pupil [3], stereo imaging to create a 3D model of an eye's structure [10], dual white LEDs on a CCD camera to initiate and capture the pupillary response [4], and eye tracking device EyeLink II to record the properties of oculomotor plants [11].

The hardware required for these methods adds to the expense. Additionally, using these techniques to acquire images usually consumes time and necessitates the user's explicit co-operation.
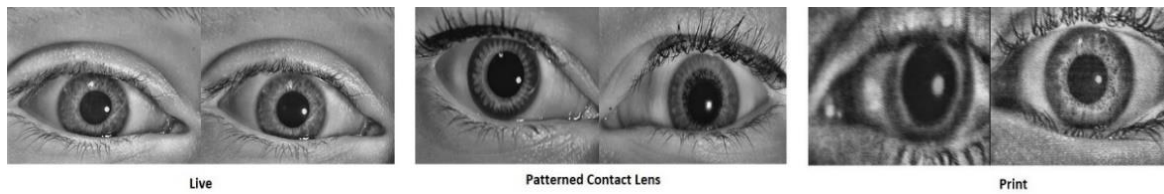

Figure 1. Few examples of eye live and presentation attacks (PAs) [12]

In contrast, software-based approaches extract key characteristics from an iris digital image to determine if it is genuine or fake. These characteristics can be produced manually or by applying deep learning methods. However, in recent times, several advanced techniques utilizing deep learning have been presented [13], [14]. Menotti *et al.* [13] introduce a deep architecture known as SpoofNet, which aims to detect presentation attacks (PA) in biometric systems. Triplet convolutional networks are the foundation of the deep framework developed by Pala and Bhanu [14]. Hoffman *et al.* [3] research focuses on identifying iris PAs using patch-batch convolutional neural networks (CNNs), which are effective in cross-dataset and cross-sensor situations. They build on their previous work [15] by examining how crucial it is to use the periocular area to find eye PAs. A multi-task CNN method was proposed by Chen and Ross [16] that involves an initial iris region detection step followed by the classification [16].

In order to find hidden or unidentified iris PAs, Yadav *et al.* [17] utilize a one-class predictor called a relativistic average standard generative adversarial network (RaSGAN). The LivDet-Iris competitions, conducted in 2013, 2015, and 2017 [18], [19], [12], offer a thorough and comparative analysis of various methods for iris presentation attack (PA) detection. The latest advancements in iris presentation attack detection (PAD) represent state-of-the-art methods and are also thoroughly reviewed by Yambay *et al.* [19]. The generalize ability between datasets, sensors, and PAs remains challenging, even though most of these approaches obtained very high PA detection rates [19]. Nowadays, iris presentation attack detection is becoming a very popular and hot topic in research, where hackers attack devices. To avoid such problems, generalized machine-learning techniques are used to detect and classify the presentation attack.

The aim of this research is to propose a solution to the current issue. By introducing a technique for iris PAD using a CNN framework, with a specific emphasis on the DenseNet architecture and to provide the generalized ability to identify unseen attacks [20]. In order to identify images of cosmetic contact PA taken by different mobile iris sensors, Yadav *et al.* [21] also make use of the DenseNet architecture. However, our research takes a significantly wider variety of iris PAs into account, including those recorded by several iris scanners for desktop and mobile devices. The distinctive feature of the DenseNet architecture is that each layer has a feed-forward connection to every other layer. The characteristics of different layers correlate to various resolutions. As the complicated iris stroma characteristics are visible in different resolutions, and iris patterns are stochastic by definition, the iris pattern is effectively described by the interaction of multiple-resolution features. Following are the work's major contributions: i) We suggest DC-DenseNetPAD, a dependable and effective iris PA detector constructed on the DenseNet design. We also demonstrate that the proposed detector is generalizable to various PAs, instruments, and datasets; and ii) We compare DC-DenseNetPAD with existing works results to evaluate the performance of the popular public datasets, namely LivDet-2017 [12] and LivDet-2015 [19].

In this paper, the suggested method is covered in section 2. The experimental setting and outcomes for both datasets are discussed in section 3. The paper is concluded in section 4.

2.   RESEARCH METHOD

In this section, we created a dual-channel DenseNet CNN framework for the classification and detection of bonafide and artifact iris. Our approach initially begins with data preprocessing, where the input image is split into two main segments and transmitted separately to each of DenseNet's two channels. Different phases of the proposed architecture are discussed in this section. Figure 2 shows the proposed work pipeline.
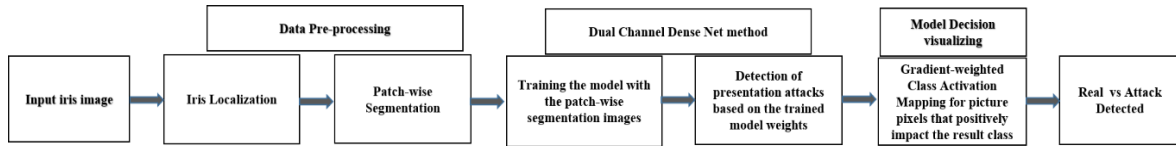
Figure 2. Proposed work pipeline

2.1. Data pre-processing

Images of ocular features other than the iris are frequently found in iris databases utilized in studies on biometrics. The iris region of some PAs, such as print, will only contain the data that the PA has provided; it is uncertain whether the artifact will be recognizable in the residual ocular area. As a result, we isolate and partition the area of the iris to lessen the influence of ocular information. Certain PAs, like cosmetic contacts, only have details regarding the PA in the iris region; the remainder of the ocular area is not likely to display any signs associated with the artifact. Because of this, we divide and partition the area of the iris to lessen the effect of ocular information. The resized picture size, an average of 256×256, was selected in a way that for the bulk of the photographs taken into consideration in this piece, upsampling occurs during resizing, as Loss of potentially important information occurs during downsampling. We next tessellate the divided and enlarged iris picture split into two overlapping parts of 128×128 pixels [22], as shown in Figure 3. The main purpose of a large number of iris PAD datasets, and this tessellation, is data augmentation. Inadequate samples of data to effectively train a neural network.

The proposed architecture consists of the following steps. First, the iris detection module receives an image of the eye that the iris sensor recorded and processed it. The VeriEye iris detector is used in this application to provide the centers of the iris and pupil, as well as their radiuses. Using this information, cropped from the eye picture is the area containing the iris. Next, the portion of the clipped iris is resized to a dimension of 224×224 and fed into a pre-trained DenseNet121 network. We then used the USIT segment tool [23] to iris image into two parts equally, as shown in Figure 3. Finally, we train the two patches in dual-channel DenseNet.
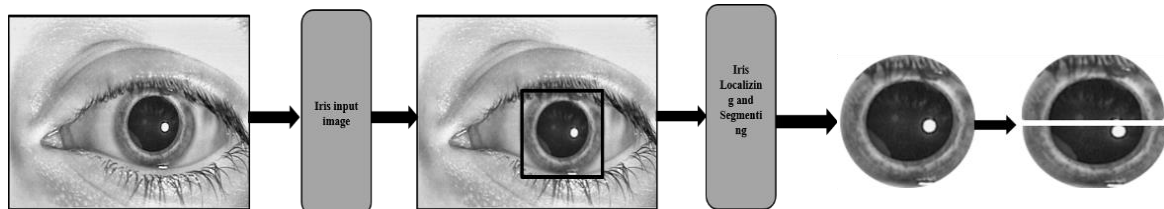

Figure 3. A description of preprocessing of data used in this work. Iris-cropped images are resized by segmenting them into two patches

Uniqueness compared to other dense-based schemes. We use a different preprocessing method than ResNet101, VGG19, and CNN. The authors in [13] attempt to recognize and train paper prints and feed CNN with the complete ocular image. Sharma and Ross [24] denseNet uses the complete ocular to identify the genuine and attack iris.

2.2. Dual channel DenseNet PAD design

In this study, using the DenseNet [20] architecture, we construct a dual-channel DNN architecture. The DenseNet derives the ResNet [25] deeper neural network architecture, which aids in extracting more significant information and reusing features to improve network performance. In the retrieved features, the dual-channel maintains both spectral and spatial information. The idea of DenseNet originated from existing networks like CNN [26] and ResNet. The DenseNet121 [20] serves as the foundation for dual channel dense network presentation attack detection (DC-DenseNetPAD). Using the two channels, we extract the features in detail to create a robust iPad. The design has 121 convolutional layers using 7×7 kernel sizes, totaling 121 levels. A layer does max-pooling, and then several dense blocks and transition layers come after it. There are two transition layers between each of the subsequent dense blocks. Two convolutional layers with 1×1 and 3×3 kernel sizes make up each dense block. An activation layer for non-linear ReLU comes after both

convolutional layers. The transition layer consists of one 1×1 convolutional layer and an average pooling layer. Reduced feature map sizes are those that are kept constant within a dense block. A fully connected layer makes up the final layer. The work in [21] takes advantage of the D-DenseNet design with three densely connected blocks at depth 2×2. The most notable property of DenseNet is the feed-forward connectivity between each layer and every other layer. Therefore, each layer receives feature maps from the preceding layer and sends them to the next layers. The input image is represented as y_0, the output of the ith layer as k_i, and each convolutional module is represented by a function P. Since all previous layer's outputs are used as the ith layer's input. DenseNet uses input for the Pfollowing layer using feature maps from all previous layer.

$$k_i = P_i([k_0, k_1, ..., k_{i-1}]) \qquad (1)$$

where $[k_0, k_1, ..., k_{i-1}])$ is the concatenation of all previous layer's outputs.

As opposed to the ResNet architecture's summation, concatenation is used to aggregate the information from earlier levels. The restriction of being the same in size throughout the feature maps is removed by concatenation. Instead of learning new feature maps every time, DenseNet stores them and uses them in the succeeding layers. Reusability with feature maps, particularly in the case of sparse training data, aids in reducing the overfitting issue. The proposed method uses the softmax classifier to classify the artifact and bonafide. Figure 4 displays the architecture of the dual-channel DenseNet presentation attack detector.
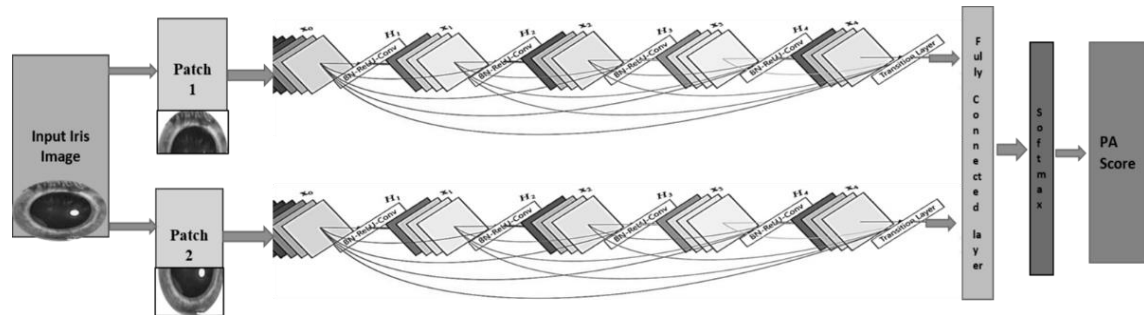

Figure 4. The dual channel DenseNet presentation attack detection architecture

The key aspects of our DenseNet model with dual channels can be summed up as follows: First, rather than utilizing the entire ocular or iris image, the model takes iris patches as input, which facilitates data augmentation during training. Second, patches used as input are extracted from the iris, ensuring an effective design of the iris presentation attack detector. Third, the cropped iris image patches are used to train the dual-channel DenseNet. The model focuses on detecting presentation attacks rather than learning location artifacts. These aspects contribute to the model's ability to accurately detect presentation attacks in iris recognition systems. A network generates a PA value that ranges from 0 to 1. When the score is 1, the input sample is considered to be a PA, while when the score is 0 the input sample is considered to be genuine. We established the threshold and the false detection rate for the final classification at 0.2%. The input sample is categorized as authentic below the set threshold; If not, it is a PA.

Dual channel DenseNet workflow:
− Step1: Input the iris image
− Step2: Iris image is segmented into two major patches
− Step3: Patches are entered into DC-DenseNet PAD
− Step4: Patch image passed into a convolutional, pooling followed by other layers.
− Step5: Finally, the decision-making is based on the softmax classifier.

2.3. Average technique to fusion a PA score

As discussed in section 2.1, the image iris consists of two distinct patches., and each patch generates its own score after being processed by the dual-channel DenseNet. However, a fusion is necessary to combine these two scores in order to reach a final decision. One potential fusion strategy involves taking the average of the two scores. This method consolidates the individual patch scores and provides a unified decision based on their combined information.

$$xy_{avg} = \frac{1}{n}\sum_{j=1}^{n} xy_j \tag{2}$$

The average score (xy_{avg}) is calculated by taking the sum of the scores of all the iris patches (xy_j) and dividing it by the total number of patches (j), which in this instance is 2. The average score ranges between 0 and 1, where a score of 0 indicates a live sample and 1 indicates a PA. This averaging method consolidates the individual patch scores and provides a decision based on the overall score of the iris image. The combined sample LivDet-2017 and LivDet-2015 dataset with PA score obtained by the model is shown in Figure 5.


```
C: > Users > Priyanka > Downloads > D-NetPAD-master > D-NetPAD-master > CroppedImages > 📊 Scores.csv
  1    .\CroppedImages\live1.jpg,0.06188474
  2    .\CroppedImages\live2.jpg,0.06268
  3    .\CroppedImages\live3.jpg,0.0570688
  4    .\CroppedImages\live4.jpg,0.095504
  5    .\CroppedImages\contactlen1.jpg,0.9728427
  6    .\CroppedImages\contactlen2.jpg,0.9023879
  7    .\CroppedImages\contactlen3.jpg,0.858923993
  8    .\CroppedImages\contactlen4.jpg,0.9954948
  9    .\CroppedImages\print1.jpg,1.0989922
 10    .\CroppedImages\print2.jpg,1.0563791
 11    .\CroppedImages\print3.jpg,1.151257
 12    .\CroppedImages\print4.jpg,1.15114056
```

Figure 5. The PA score obtained by the model on the sample LivDet-2017 and LivDet-2015 combined dataset

In training, we use a learning rate of 0.005, which decides how big the steps are when we adjust our model's settings. We process our data in groups of 20 samples, called batches. Our optimization method, called stochastic gradient descent with a momentum of 0.9, helps our model learn faster and stay steady. We go through our data 100 times during training, each called an epoch, so our model learns from it many times. Lastly, we measure how well our model is doing using a loss function called cross-entropy, which determines how closely our predictions match the correct responses. These parameters aid in the precise and effective learning of our model.

3. EVALUATION AND RESULTS

3.1. Experiment setup

In the experimental configuration, this model utilizes the Visual Studio integrated development environment (IDE) along with Python version 3.6.7. Additionally, the environment is set up to include essential packages such as Pip, Torch, NumPy, and SciPy. The code is implemented purely in Python. The model basically uses the popular data set available publicly with the 7×7 pixels, 6×6 pixels, and 8×8 pixels on the proprietary dataset and also uses the publicly available dataset.

3.2. Dataset

We designed the model with a publicly available dataset, namely LivDet-2015 [19] and LivDet-2017 [12]. LivDet description and results: LivDet-2015 [19] is a benchmark database that has been widely used in the evaluation of liveliness detection techniques for iris recognition systems. In the Clarkson LivDet 2015 dataset, the initial subset employs an IrisAccess EOU2200 camera from LG for iris data collection. This subset contains 1152 images with patterned contact lenses, 1,746 printed pictures of the iris and 828 live iris photos. Within the training dataset, there are 450 live iris images, 576 patterned contact lens images, and 846 printed iris images. The testing dataset consists of 576 photographs of patterned contact lenses, 900 printed iris images, and 378 live iris images. Significantly, the printed iris images in this portion are more varied due to Dalsa and LG cameras. A Dalsa camera, which captures images in the near-infrared spectrum, is employed in the second subset of the dataset. This subset consists of 1,746 images of the printed iris, 1078 images of the live iris, and 1,431 images taken while wearing contact lenses. There are 706 photos of live iris, 873 photographs of iris-patterned contact lenses, and 846 iris prints in the training dataset. There are 378 pictures of live irises, 558 pictures of printed iris shots, and 900 pictures of printed iris photos in the test dataset. The combination of LG and Dalsa trained and test datasets have been combined and shown in Table 1.

Table 1. Dataset used in the model are LivDet-2017 and LiveDet-2015

| Database | LivDet-2017 | | | | LivDet-2015 | |
	Train Clarkson	IIITD	Notre	Test Clarkson	Train Clarkson	Test
Bonafide	2469	2250	600	1486	1150	756
Print	1346	3000	-	909	1550	1800
Patterned contact lenses	1122	1000	600	765	1413	1134
Combined Print and Contact lenses	2468	4000	600	1674	2963	2934

The LivDet-2017 [12] dataset was another one that was evaluated. The Clarkson, Warsaw, Notre Dame, and IIITD-WVU databases were combined to create the LivDet-2017 dataset. The types of PAs that are found in Table 1 list the datasets and their quantity of iris images in test sets and training stages for every single one of the four datasets. A cross-PA testing situation is represented by the Clarkson dataset. In the LivDet-2017 dataset, we have data from three different sensors: Clarkson, IIITD, and Notre Dame. For the Clarkson sensor, there are 2,469 bonafide samples IITD-WVU consisting of 2,250 bonafide, Notre Dame consisting of 600 bonafide during the training set and 1,486 while testing set. Additionally, there are 1,346 print samples of Clarkson, 3,000 print samples of IIITD-WVU training set, and a test set of Clarkson 909. For patterned contact lenses, there are 1,122, 1,000, and 600, respectively, Clarkson, IITD-WVU, and Notre Dame training samples, respectively and 765 Clarkson testing samples. Table 1 shows the combined data set quantity.

− Pre-trained DC-DenseNetPAD: The model developed using the combined dataset is employed directly.
− Scratch DC-DenseNetPAD: The LivDet-2017 and LivDet-2015 train samples are employed to create the model from scratch.
− Fine-tuned DC-DenseNetPAD: The model, which was previously created using a merged dataset, is enhanced using the LivDet-2017 train sets. The result obtained by the fine-tuned model by combining bonafide and print in the single dataset. The results obtained from the LivDet-2017 and LivDet-2015 are depicted in the Table. 2. Figure 6 shows the confusion matrix for the suggested approach on a test data set, whereas Figure 6(a) shows a LivDet-2017 and LivDet-2015 in Figure 6(b).

Table 2. Metrics like TDR, accuracy, FDR, and misclassified rate are used to evaluate how well the suggested model performs in classification tasks

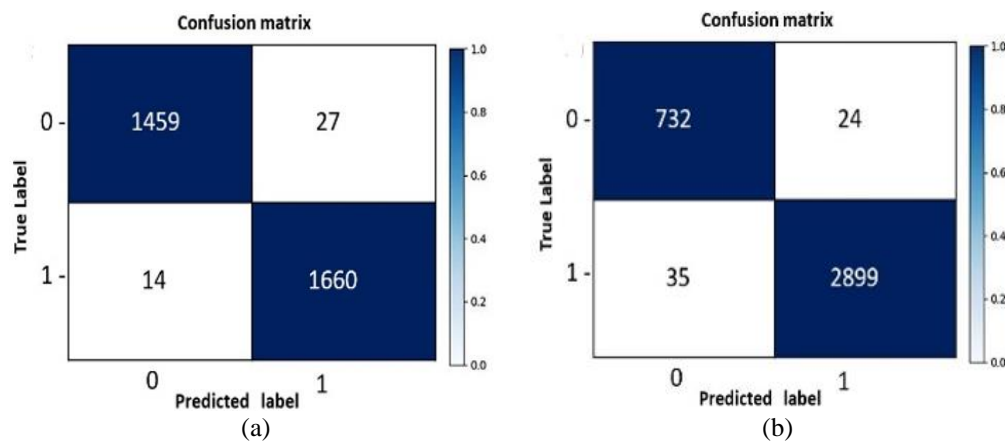| Dataset	Accuracy	TDR	FDR	Misclassified rate
LivDet-2017	98.70	99.16	0.9	1.31
LivDet-2015	98.80	98.40	4.6	1.6

Figure 6. Confusion matrix produced by the model on the (a) LivDet-2017 and (b) LivDet-2015 on the combined test dataset

The DC-DenseNetPAD is contrasted with the top three competitors in the LivDet-2017 contest. The pre-trained DC-DenseNetPAD performed poorly on the Clarkson test set 2015. The possibilities of cross-sensor and cross-PA are represented by the Clarkson dataset. Poor performance (30.6%) is caused by the visual differences between the photos taken with the IrisAccess EOU2200 and those taken with the

iCAM 7000 iris sensor. The outcome (sensor information) increases (94.05% and 95.51%) when the Clarkson 2017 train set is included in the training stage (from scratch or fine-tuned). Table 3 evaluates the performance of the recommended method against the five active algorithms. The TDR rate achieved on the Clarkson test dataset is 99.16% on LiveDet-2017 and 98.40% on LivDet-2015. The model will achieve better results in the fine-tuning process than the pre-trained one. Figure 7 displays the model's accuracy and loss graph for the training and validation data where Figure 7(a) and Figure 7(b) is LivDet-2017 and Figure 7(c) and Figure 7(d) is for the LivDet-2015 dataset. We observed that the proposed methodology outperformed the existing methods due to: i) the uniqueness of the data-preprocessing approach we used and ii) the training of the patches into the dual channel DenseNet rather than the complete set of images.

Table 3. Comparing the results between the existing methodologies and the proposed technique based on the LivDet-2017 and LivDet-2015

Algorithms	LivDet-2017 TDR (%)	Algorithms	LivDet-2015 TDR (%)
VGG19 [24]	96.26%	CNN [3]	97.69%
ResNet101 [24]	96.88%	Multiple CNN [15]	98.07%
D-NetPAD [24]	98.50%	Proposed dual channel DenseNet	98.40%
Proposed dual channel DenseNet	99.10%		

(a)   (b)

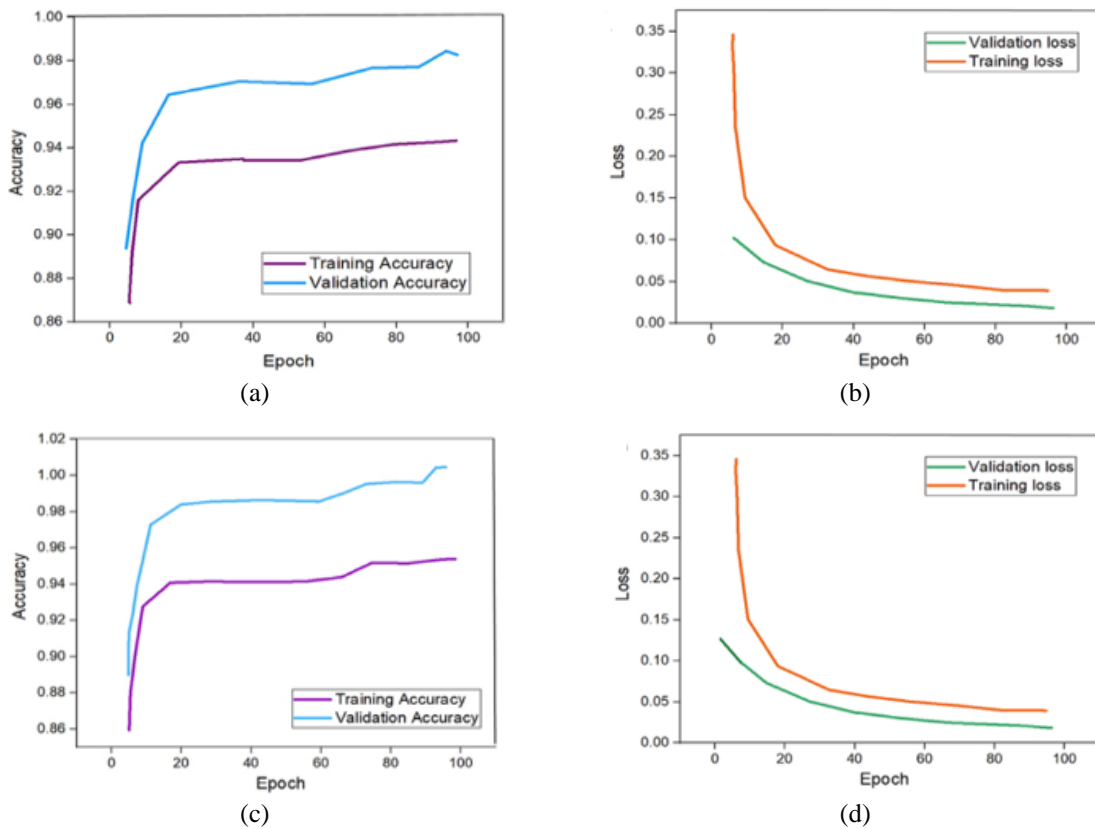(c)   (d)

Figure 7. Accuracy and loss graph of dual-channel DenseNet for training and validation data, a), b) represents LivDet-2017 and c), d) represents LivDet-2015

3.3. Visualization analysis

We use gradient-weighted class activation mapping (Grad-CAM) [27] heatmaps and t-Distributed stochastic neighbour embedding (t-SNE) [28] plots, as illustrated in Figure 8 by utilizing t-sne plots, we can capture and view the features at last of the Dense block. Our visualizations are based on the Dual channel DenseNetPAD training a model using the Combined dataset's training set. Samples from the LivDet-2017 [12] are used to test set for this purpose. By applying t-SNE, we reduce the dimensionality of the features extracted by Dual channel DenseNetPAD to a lower dimension, such as two, allowing us to create scatter plots that visually represent the extracted features.
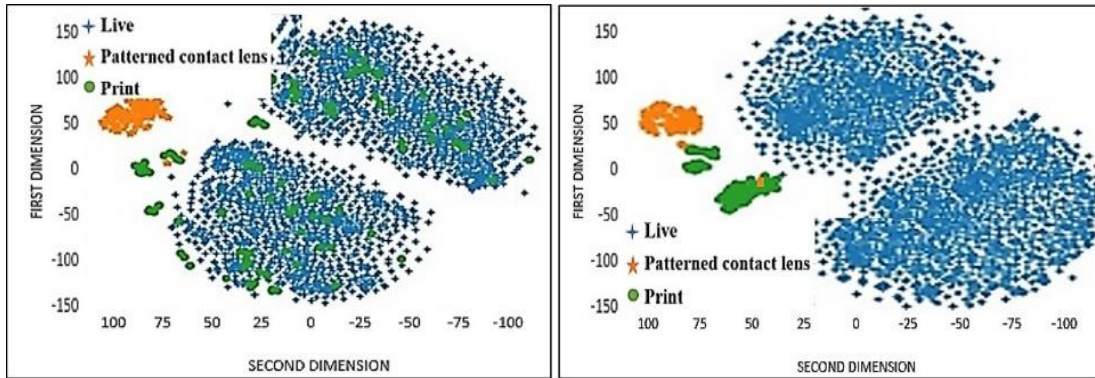
Figure 8. The first and final dense block visualized using t-SNE plots

We enhance the visualization of CNN activations using GradCAM [27] heatmaps. GradCAM generates a localization [29] map that highlights the salient regions within an image, indicating the regions that had the most significant impact on the network's performance were identified in the inference. These regions exhibit high activations in the neural network. The GradCAM [27] technique involves calculating a loss function's gradient and propagating it backwards, eventually applied to the input image through the convolutional layers. Figure 9 illustrates the CNN activation heatmaps displayed for live patterned contact lenses and print images obtained from the LivDet-2017 [12] test set. In the visualization, the first row depicts the patch-wise segmentation [23] of the visual image generated using GradCAM, and the second row presents the overall representation of the visual image obtained through GradCAM.

Visualization using grad-cam for the dual channel DenseNet PAD to keep track of features identified from the model of the given input. A technique used for visualizing and interpreting the decision-making process of convolutional neural networks is said to be the grad cam. In the figure, the first presents live, the second displays a contact lens image, and the final picture in the figure shows the print from LiveDet-2017. The red color represents the identification of the characteristics from the source image by the DC-DenseNetPAD. In the first stage, the model recognizes only the preliminary portion of the iris, and in the second stage, the model recognizes the complete iris.
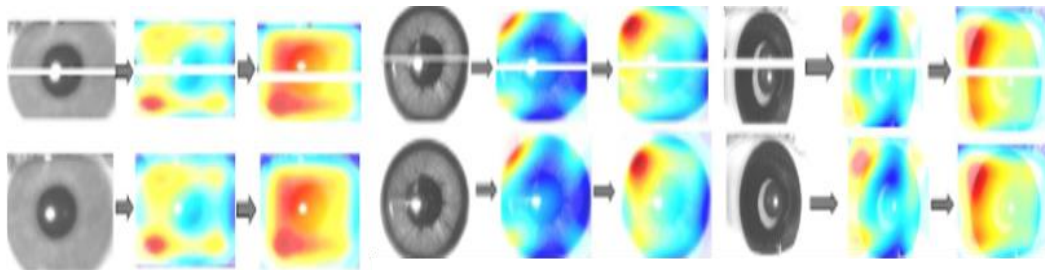

Figure 9. The live, patterned contact lens and print classes were visualized using Grad-CAM [27] heatmaps. The live class heatmap, the patterned contact lens class heatmap and the print class heatmap are shown

4.   CONCLUSION

We suggest DC-DenseNetPAD, a robust and reliable Iris PA detector with software. The DC-DenseNet PAD utilizes DenseNet121's architectural advantages. Two datasets are used for experiments to test the effectiveness of the system. These dataset's test sets match scenarios that test the DenseNetPAD's robustness across PA, datasets, and sensors. The proposed method achieves 99.16% and 98.40% on LivDet-2017 and LivDet-2015, respectively. We conduct a comparative analysis with other existing methodologies. Further, frequency analysis was utilized to analyze the performance of DC-DenseNetPAD in-depth, heat maps produced by Grad-CAM, and t-SNE plots. Future implementations of a new dataset that will be used in this technique that contains a brand-new PA kind: replay attacks on the Kindle to illustrate a wide range of PA variants to aid algorithms in finding broadly applicable. If the biometric sensor is used by the user, we may also gather ambient information to help with further generalizations.

ACKNOWLEDGEMENT

REFERENCES

[1] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148–1161, 1993, doi: 10.1109/34.244676.

[2] V. Priyanka and G. K. Shyam, "DC-CNNPAD to enhance the detection rate for iris presentation attack," in *Fourth Congress on Intelligent Systems*, Springer, Singapore, 2024, pp. 187–198.

[3] S. Hoffman, R. Sharma, and A. Ross, "Convolutional neural networks for Iris presentation attack detection: toward cross-dataset and cross-sensor generalization," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jun. 2018, pp. 1701–17018, doi: 10.1109/CVPRW.2018.00213.

[4] A. Czajka, "Database of iris printouts and its application: Development of liveness detection method for iris recognition," in *2013 18th International Conference on Methods and Models in Automation and Robotics (MMAR)*, Aug. 2013, pp. 28–33, doi: 10.1109/MMAR.2013.6669876.

[5] K. B. Raja, R. Raghavendra, and C. Busch, "Video presentation attack detection in visible spectrum iris recognition using magnified phase information," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2048–2056, Oct. 2015, doi: 10.1109/tifs.2015.2440188.

[6] A. Czajka and K. W. Bowyer, "Presentation attack detection for Iris recognition: an assessment of the state-of-the-art," *ACM Computing Surveys*, vol. 51, no. 4, pp. 1–35, Jul. 2018, doi: 10.1145/3232849.

[7] D. Yadav, N. Kohli, J. S. Doyle, R. Singh, M. Vatsa, and K. W. Bowyer, "Unraveling the effect of textured contact lenses on iris recognition," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 851–862, May 2014, doi: 10.1109/tifs.2014.2313025.

[8] O. V Komogortsev, A. Karpov, and C. D. Holland, "Attack of mechanical replicas: liveness detection with eye movements," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 716–725, Apr. 2015, doi: 10.1109/tifs.2015.2405345.

[9] A. Pacut and A. Czajka, "Aliveness detection for IRIS biometrics," in *Proceedings 40th Annual 2006 International Carnahan Conference on Security Technology*, Oct. 2006, pp. 122–129, doi: 10.1109/CCST.2006.313440.

[10] K. Hughes and K. W. Bowyer, "Detection of contact-lens-based iris biometric spoofs using stereo imaging," in *2013 46th Hawaii International Conference on System Sciences*, Jan. 2013, pp. 1763–1772, doi: 10.1109/HICSS.2013.172.

[11] O. V Komogortsev and A. Karpov, "Liveness detection via oculomotor plant characteristics: attack of mechanical replicas," in *2013 International Conference on Biometrics (ICB)*, Jun. 2013, pp. 1–8, doi: 10.1109/ICB.2013.6612984.

[12] D. Yambay *et al.*, "LivDet iris 2017 — Iris liveness detection competition 2017," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*, Oct. 2017, pp. 733–741, doi: 10.1109/BTAS.2017.8272763.

[13] D. Menotti *et al.*, "Deep representations for iris, face, and fingerprint spoofing detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 864–879, Apr. 2015, doi: 10.1109/tifs.2015.2398817.

[14] F. Pala and B. Bhanu, "Iris liveness detection by relative distance comparisons," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jul. 2017, pp. 664–671, doi: 10.1109/CVPRW.2017.95.

[15] S. Hoffman, R. Sharma, and A. Ross, "Iris + Ocular: generalized iris presentation attack detection using multiple convolutional neural networks," in *2019 International Conference on Biometrics (ICB)*, Jun. 2019, pp. 1–8, doi: 10.1109/ICB45273.2019.8987261.

[16] C. Chen and A. Ross, "A multi-task convolutional neural network for joint iris detection and presentation attack detection," in *2018 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, Mar. 2018, pp. 44–51, doi: 10.1109/WACVW.2018.00011.

[17] S. Yadav, C. Chen, and A. Ross, "Relativistic discriminator: a one-class classifier for generalized iris presentation attack detection," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2020, pp. 2624–2633, doi: 10.1109/WACV45572.2020.9093313.

[18] D. Yambay, J. S. Doyle, K. W. Bowyer, A. Czajka, and S. Schuckers, "LivDet-Iris 2013 - Iris liveness detection competition 2013," in *IEEE International Joint Conference on Biometrics*, Sep. 2014, pp. 1–8, doi: 10.1109/BTAS.2014.6996283.

[19] D. Yambay, B. Walczak, S. Schuckers, and A. Czajka, "LivDet-Iris 2015 - Iris liveness detection competition 2015," in *2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, Feb. 2017, pp. 1–6, doi: 10.1109/ISBA.2017.7947701.

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.

[21] D. Yadav, N. Kohli, M. Vatsa, R. Singh, and A. Noore, "Detecting textured contact lens in uncontrolled environment using DensePAD," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jun. 2019, pp. 2336–2344, doi: 10.1109/CVPRW.2019.00287.

[22] A. S. Akinfende, A. L. Imoize, and O. S. Ajose, "Investigation of iris segmentation techniques using active contours for non-cooperative iris recognition," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 19, no. 3, pp. 1275-1286, Sep. 2020, doi: 10.11591/ijeecs.v19.i3.pp1275-1286.

[23] H. Ohmaid, S. Eddarouich, A. Bourouhou, and M. Timouyas, "Iris segmentation using a new unsupervised neural approach," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 1, pp. 58–64, Mar. 2020, doi: 10.11591/ijai.v9.i1.pp58-64.

[24] R. Sharma and A. Ross, "D-NetPAD: an explainable and interpretable iris presentation attack detector," in *2020 IEEE International Joint Conference on Biometrics (IJCB)*, Sep. 2020, pp. 1–10, doi: 10.1109/IJCB48548.2020.9304880.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[26] H. S. Hamid, B. AlKindy, A. H. Abbas, and W. B. Al-Kendi, "An intelligent strabismus detection method based on convolution neural network," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 6, Dec. 2022, doi: 10.12928/telkomnika.v20i6.24232.

[27] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.
[28] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
[29] E. Hussein Ali, H. Abbas Jaber, and N. Naji Kadhim, "New algorithm for localization of iris recognition using deep learning neural networks," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 29, no. 1, pp. 110–119, Jan. 2022, doi: 10.11591/ijeecs.v29.i1.pp110-119.

BIOGRAPHIES OF AUTHORS

Priyanka Venkatesh 🆔 🔍 SC ▷ received the BCA and MCA degrees from Presidency College affiliated with Bengaluru University, Bengaluru, in 2018 and 2020, respectively. She is pursuing her Ph.D. in cybersecurity and deep learning at Presidency University, Bengaluru, India, and her research focuses on detecting and preventing presentation attacks using machine learning and deep learning. She is also interested in the area of biometric security and quantum blockchain. She has received many awards, such as Meritorious Student, Excellence in Academics, and a University Rank for her degrees. She has presented and published her research papers at national and international conferences. She also published her work in various books under springer's publisher. She can be contacted at email: priyanka.v@presidencyuniversity.in.

Gopal Krishna Shyam 🆔 🔍 SC ▷ received B.E., M.Tech. and Ph.D. degree in computer science engineering from Visveswaraya Technological University, Belagavi, India. He is currently working as a professor and HoD in School of Computer Engineering, Presidency University, Bengaluru, India. He has around 17 years of experience in teaching and research. He is involved in research in areas of cloud/grid computing, E-commerce, protocol engineering, and artificial intelligence applications. As per Google Scholar, he has around 1,200 citations. Some of his publications on resource management in cloud computing are among the top downloaded articles in Elsevier Journals of network and computer applications. He is the reviewer of several international/national Journals published by Elsevier, IEEE, ACM, and Springer. He has been a technical program committee member of more than 5 National/International Conferences. He has guided 32 B.E./M.Tech. Candidates, 2 PhD candidates and is currently guiding 4 PhD Scholars. He has given five invited talks at various engineering colleges. He can be contacted at email: gopalkirshna.shyam@presidencyuniversity.in.

Sumbul Alam 🆔 🔍 SC ▷ a bachelor's in computer science and engineering from Vinayaka Mission University in 2011, followed by a master's in computer science and engineering from Rajeev Gandhi Technical University in 2018. Currently pursuing a Ph.D., her research focuses on the detection and analysis of neurodevelopmental disorders using machine learning and deep learning, particularly through speech recognition, facial expression, and sensory responses. She has published research on image processing of MRI brain tumor images and has teaching experience in computer science, specializing in areas like object-oriented programming and data structures. As a lecturer at King Faisal University, she enjoys guiding students' learning and fostering critical thinking skills. She can be contacted at email: sumbulalam@presidencyuniversity.in.