IRIS PRESENTATION ATTACK DETECTION
MEDIA FORENSICS FINAL REPORT

Nicole ElChaar
nje222@lehigh.edu

Lehigh University
Computer Science & Engineering
113 Research Drive, Bethlehem, PA 18015

***Index Terms*—** Presentation Attack Detection, PAD, Biometrics, Authentication, Identity, Iris Detection, Forensics, Attack, Eye, DenseNet, Gabor Filter, Texture Features

1. INTRODUCTION

1.1. Motivation

Iris recognition is widely used in biometric authentication as it non-intrusive (it is contactless and does not cause any discomfort to the individual being identified), it can be used in a variety of lighting conditions with near-infrared imaging, and it is very accurate relative to other biometric techniques. Irises are robust as a human feature as they are less likely to degrade over time than fingerprints and are more unique than vein patterns and other facial features.[1] Though there are certain diseases such as cataracts, glaucoma, and iris atrophy that impact iris recognition, it remains resilient to human aging in the majority of cases. These properties make irises commonly chosen, often in conjunction with other biometric patterns, for the best accuracy in human identification.

At the same time, iris recognition is vulnerable to imposter and concealer attacks. Because these systems are crucial in identity verification for access control, financial transactions, border control, and many other applications, protecting iris recognition systems from attack is essential. There is already a wide variety of research into iris recognition and presentation attack detection (PAD), however, many of the techniques are not generalizable.[2]

1.2. Problem Statement

When an iris recognition system is presented with textured contact lenses, a printout, a prosthetic, synthetically-generated images, or other non-legitimate irises, we require a system that can identify and defend against these attacks.

2. BACKGROUND AND LITERATURE REVIEW

2.1. Attack Techniques

The lack of generalizability in iris PAD may stem from the wide variety of presentation attacks available.

In static images, textured contact lenses can be effective at hiding the identity of the wearer. These lenses can have patterns that mimic the crypts and furrows present in natural irises. It is possible to detect the extra pattern overlaid on the wearer's actual iris or inconsistent edges against the live iris, but this is much easier to do with videos of the eye than static images alone.[3]

Similarly, post-mortem irises have also been used in unauthorized access situations. There are advancements in liveness detection techniques (like identifying pupil constriction, blinking) and Gabor-based recognition methods that can identify these irises.[4] However, static images once again make it more difficult to determine the presence of a post-mortem iris.

Printed and synthetically generated eyes also should not fool iris detection systems. Using Near Infrared Light (NIR) when collecting iris images is helpful, but this is possible only when the same system gathers and classifies the images. If images are simply presented to the classifier, special imaging techniques cannot be effective.

2.2. Feature Extraction

There is some debate over whether feature extraction is necessary in iris PAD. While deep learning approaches can be effective without explicit feature extraction, combining feature extraction with the original image of the eye in multi-layer models tends to be more robust.[5] This is useful when specific artifacts indicate a bogus iris, such as eyelid retractors in post-mortem cases, as we combine contextual information with texture of the iris itself.

The traditional approaches of iris feature extraction include Gabor-based methods in which iris images are divided into different textured segments with a Gabor or Log-Gabor filter. After normalizing the iris image and shifting orientations and wavelengths, iris images are convolved with the filter to give a feature vector. These feature vectors can be compared with authorized irises to verify the identity of the individual posing or passed through a model to indicate the authenticity of the iris image.[6] Similarly, there are Discrete Cosine Transforms (DCT) and Discrete Wavelet Transforms (DWT) that mimic the patterns found in the Gabor-based methods with lower computational complexity and less required processing time. DCT is very similar to the Fourier Transform, but it uses only the real part of the cosine transformation and compresses this information into fewer coefficients.[7]

Principal Component Analysis (PCA), Gray-Level Co-occurence Matrices (GLCM), Local Binary Patterns (LBP), and Scale-Invariant Feature Transform Descriptors (SIFT) have also been used to extract iris features. PCA is thought to prevent overfitting to a particular dataset and aid in reducing the size of global features. LBP is best used in texture extraction and traditionally creates a binary comparison of each pixel to its X neighbors. SIFT descriptors are also used, but they do not tend to be as accurate as other feature extraction methods.[7]

2.3. Attack Detection

DenseNet architectures have been used across iris PAD, mostly leading to a binary result of either a bonafide or attack iris.[8][9][10] However, through 2020, many of these approaches were overfit to the training set and significantly underperformed on unseen sets. The exception is D-NetPAD[9], which uses a DenseNet121 architecture. This model performed better than VGG19 and ResNet101 models with both manipulated and true images and was more effective on combinations on separate training and testing datasets.

A recent 2022 paper suggested the use of expansion-contraction networks for iris PAD that is more generalizable.[2] This is in line with a stacked hourglass network from 2020 that improved segmentation for irises.[11] Rather than using edge detection, size-agnostic feature extraction gives better context surrounding the eye. This network is similar to a UNet but is more effective for irises with occlusion, scale variation, and off-angles. Because the available datasets are fairly pristine and iris biometric systems can require individuals to retake iris photos until they are, however, these advantages are not required in this exploration.

3. APPROACH

3.1. Data

NDIris3D, CASIA-Iris-SYN, CASIA-IrisV1, and LivDetIris-2023 sets were used in this analysis and are outlined below.

3.1.1. CASIA-Iris-SYN

CASIA-IrisV4, the latest release, contains six subsets of data: CASIA-Iris-Lamp, CASIA-Iris-Twins, CASIA-Iris-Interval, CASIA-Iris-SYN, CASIA-Iris-Distance, and CASIA-Iris-Thousand. There are 54,601 images from 1,800 genuine and 1,000 synthetic subjects.[12] For my exploration, I am using the CASIA-Iris-SYN (synthetic) dataset of the V4 release. The synthetic database was primarily generated to circumvent the legal and ethical consequences of sharing true iris images. It contains irises that are very difficult to distinguish from genuine irises, both for individuals viewing the irises and for machines attempting to find statistical differences between them. For this reason, they are interesting for my exploration. All images of CASIA-Iris-Syn are modified from the original CASIA-IrisV1 release. Iris textures were synthesized and embedded into the real iris images, making the artificial iris ring more realistic. Variations including deformation, blurring, and rotation were also introduced, making it more challenging to identify iris features.

3.1.2. CASIA-IrisV1

CASIA-IrisV1 is also included in this analysis as it is so closely related to the synthetic database. CASIA-IrisV1 contains 756 images of 108 irises. Lighting and imaging techniques are unique to this set, making them most comparable to the CASIA-Iris-SYN dataset.[12]

3.1.3. NDIris3D

NDIris3D is a dataset of 6,850 images of 89 subjects, with and without textured contact lenses from four manufacturers.[13] While there is no additional dataset in this selection to directly compare to, it provides an attack method that is common amongst other sets.

3.1.4. LivDetIris-2023

LivDetIris-2023 is the fifth competition in the LivDetIris series. As of this analysis, a training set of 60 labeled images has been released, including irises synthesized by modern Generative Adversarial Networks-based models (StyleGAN2 and StyleGAN3) and near-infrared pictures of various artifacts (unspecified) simulating physical iris presentation attacks.[14] The training images can be used for evaluation, but because the test images do not include ground truths, they are not relevant in this evaluation.

3.1.5. Considerations

While the original intention was to explore the generalizability of the network by training on many attack types and evaluating on untrained sets, I was unable to gain access to more than one dataset of scale offering the same attack type. Access to the WVU UnMIPA dataset[15] would remedy this, but the authors are no longer sharing the dataset with outside researchers. As a result, the available data restricts the model to a simple binary classification of live or spoof iris types. Explorations include (1) combining datasets to detect more than one presentation attack as spoof (though this does not address generalizability to unseen attack types) and (2) training on part of one set, comparing results to testing on the same dataset with testing on the remaining set (though this does not allow the network to better distinguish real images from the many possible attacks).

In addition, while there are many iris databases to address specific areas of concern, including textured contact lenses, lack of liveness in static images, iris detection at a distance, among others, most iris datasets contain subjects of Caucasian and Asian descent. CASIA has attempted to contribute towards the diversity of iris datasets with the CASIA-Iris-Africa set,[12] but without an additional set of attack irises derived from the same set, it is difficult to include. The lack of diversity in the available datasets is another notable flaw in this exploration.

3.2. Preprocessing

The full preprocessing pipeline is displayed in Figure 1, with the localized (cropped) iris image, unrolled iris image, and iris texture grid as outputs.

3.2.1. Iris Isolation

To preprocess the available datasets, first, the pupil is localized. The image is blurred with a median filter, and a Canny edge detector is used to expose prominent lines. Then, on the image of edges, Hough circle detection is used to find the most probable location of the pupil in each image. We expect lower average luminance in the pupil than in the surrounding pixels of the original image. Restricting pupil identification to a round shape makes it more sensitive to distorted images, but this restriction is justified for access identification systems that take their own images and expect the user to be looking directly into the camera. In addition, it is relevant in making the system more robust as we expect pristine images.

Once this circle is identified, I store the center and assume the iris expands outwards from the same center as the pupil. When the average luminance of the expected iris drops beyond a certain threshold, the detector stops expanding and stores the center and radii. We do this primarily in the horizontal direction because we expect the eye to be relatively upright in the image. Luminance should increase as we reach the whites of the eye which should be somewhat distinct from the iris' color. In addition, the iris is often occluded by the eyelids in the vertical direction, and we do not have a common heuristic for average iris color to average skin tone like we do in the horizontal direction. This assumption again is sensitive to distortion and rotation of the image, but these stronger assumptions are allowed in forensics applications. In this way, we locate all iris information present in the image.

In Figure 2, I have plotted a sample of a CASIA-Iris-SYN image. The center of the detected pupil, detected pupil, and detected iris are outlined in white.

While the ideal setup would auto-adjust the threshold to a particular iris image or dataset or use the keypoints of eyes to identify and extract each iris, these methods have not been as effective as the manual threshold. The existing setup is around 97.2% effective at choosing the correct pupil center and iris radius (28/1000 randomly sampled images incorrectly chosen).

3.2.2. Iris Extraction

Once the iris is localized, we sample in polar coordinates around its center to create an "unrolled", normalized version. I gather 60 samples between the pupil and iris at 360 evenly-spaced angles around the center (for each of 360 degrees). For all resolutions of iris images, this means that pixels closest to the pupil are sampled more often (or repeatedly) compared to pixels towards the outer edge of the iris. This is not ideal as textures close to the pupil are most likely to be impacted by pupil expansion, but my hope is that we can account for this in model iterations as is done in existing models. In Figure 3, we see the unrolled, normalized iris features of Figure 2. Note that we can see the residual eyelids as lighter crescent shapes at the bottom of this figure.

3.2.3. Gabor Filtering

Gabor filtering is then used in the preprocessing pipeline to extract feature vectors from the iris. I display two examples of output in Figure 4 and Figure 5. We can see the residual eyelid textures at the bottom of Figure 5 where horizontal textures are isolated. Stronger patterns in the vertical direction with the crests of the iris in shown in Figure 4. These images are then quantized to four levels and

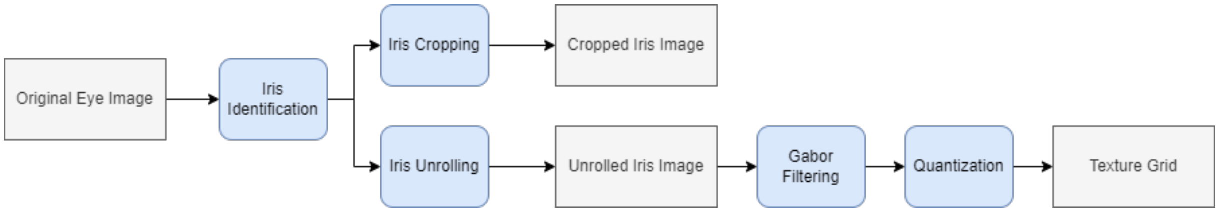

Fig. 1. Depiction of the preprocessing steps taken to extract the texture grid, unrolled iris image, and cropped iris image from the original eye. The cropped iris, unrolled iris, and texture grid are all stored as inputs.

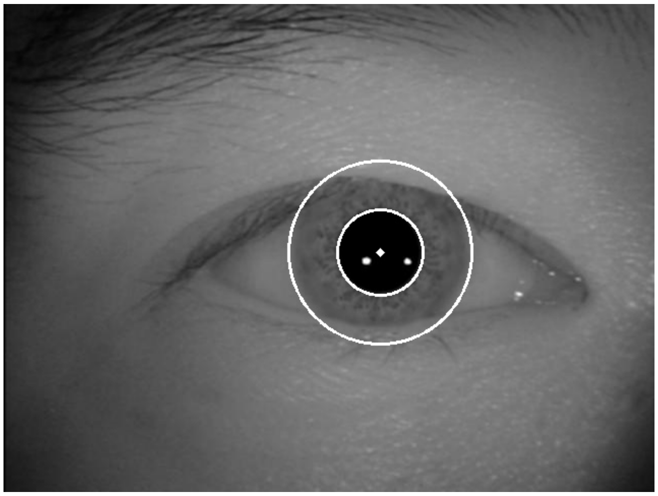

Fig. 2. CASIA-Iris-SYN image with detected center of pupil, pupil, and iris in white. Note that the white outline is not added to the original image, it is simply added for visualization here.

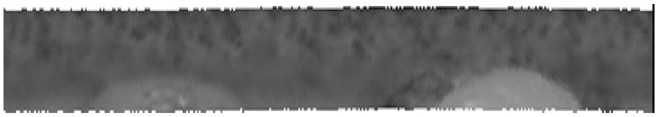

Fig. 3. Unrolled iris features from Figure 2. The noisy edges that appear towards the top and bottom are due to thresholding which is varied as a hyperparameter.

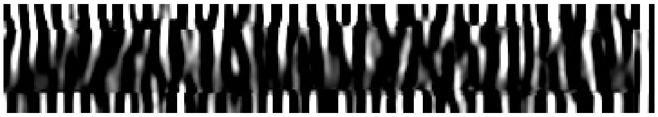

Fig. 4. Extraction of iris features in Figure 3 via a horizontal Gabor filter

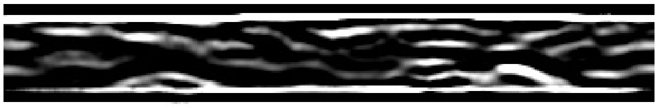

Fig. 5. Extraction of iris features in Figure 3 via a vertical Gabor filter

stored as a feature vector in addition to the other intermediate steps in Figure 1.

3.3. Model

The final architecture is shown in Figure 6 and combines D-NetPAD[9] with linear layers for texture pattern extraction. Feature vectors extracted from each part are concatenated and reduced to a single presentation attack (PA) score.

DenseNet is chosen because of its dense connection structure. Unlike traditional CNNs, where each layer is connected to only the previous layer, DenseNet connects each layer to every preceding layer. This architecture encourages feature reuse and improves gradient flow, making the network more efficient and accurate with fewer learnable parameters. It also allows DenseNet to learn discriminative features by fusing information from all layers, which is particularly useful in tasks where the iris images contain subtle differences.

Texture patterns are chosen to identify anoma-

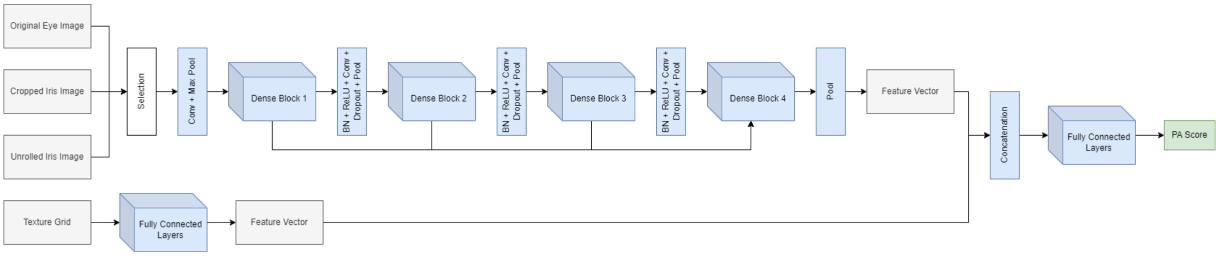

Fig. 6. The multi-modal approach for PA scoring. The upper pipeline depicts the image pathway for either the original eye image, the cropped iris image, or the unrolled iris image. The lower pipeline depicts the feature extraction from the texture grid extracted from the unrolled iris in Section 3.2

lies that may be associated with layered textures from textured contact lenses, unexpected frequency patterns from printouts, along with other texture-based anomalies. With linear layers, we efficiently reduce dimensionality of the incoming texture pattern and learn the weights of the most important sections of the pattern.

4. EXPERIMENTATION

4.1. Baseline

As a baseline, we test with D-NetPAD alone.

4.2. Hyperparameters

Hyperparameters varied include the input image type, texture pattern eyelash threshold, number of epochs, and learning rate. Input image types are either the full eye image, localized iris (crop), or unrolled iris. Texture pattern eyelash thresholds range from 50 to 250 in increments of 10. Either 50, 75, or 100 were used as the number of epochs. Learning rate was varied from 1^{-10} to 5^{-1}.

While the intention of this exploration was to explore the generalizability of the network, I also attempted at using multiple sets (multiple attack types) for training and testing. Using multiple sets for testing achieves a high accuracy of 85.1% across NDIris3D and CASIA sets as expected (as images from testing must come from the same set used in training). Because these results do not pertain to the ability of the network to detect unseen attack types or unseen datasets, they are irrelevant towards the goals of this paper and are excluded from Section 5.

5. RESULTS & DISCUSSION

5.1. Single-Set Training

For single-set training, we select either textured contact lenses (NDIris) or synthetic irises (CASIA-IrisV1 and CASIA-Iris-SYN) as the attack method for training, using the other set for testing. This is in line with the goal of testing the network on unseen sets and unseen attack types, though much more difficult and nearly impossible with only a single attack type in training.

The best performance comes from using NDIris3D as the training set, achieving 87.5% accuracy when validating on the remaining 20% of the set. The best hyperparameters use the unrolled iris in the DenseNet pipeline with 80 as the eyelash threshold for texture pattern extraction. We use 50 epochs and a learning rate of 1^{-5}. Compared to D-NetPAD which achieves 81.5% accuracy with the same parameters, we achieve 6.25% higher with the addition of texture feature patterns in this network. On the remaining sets, we find the results in Table 1.

The results from CASIA-IrisV1 indicate the ability of the network to generalize to unseen sets when classes are known. The accuracy of real images is only 2% lower for CASIA-IrisV1 than the validation set of NDIris3D.

At the same time, the network cannot distinguish the real images of CASIA-IrisV1 from the

Dataset	Accuracy	Per-Class Accuracy	Per-Class Precision	Per-Class Recall
LivDet-Iris-2023	64.29%	[61.29%, 66.67%]	[63.33%, 64.71%]	[61.29%, 66.67%]
CASIA-IrisV1	84.66%	[84.66%, nan]	[100%, nan]	[84.66%, nan]
CASIA-Iris-SYN	15.22%	[nan, 15.22%]	[0%, 100%]	[nan, 15.22%]

Table 1. By testing set, the per-class results from the best validated NDIris3D model.

Dataset	APCER	BPCER
LivDet-Iris-2023	33.33	39.71
CASIA-IrisV1	-	15.32
CASIA-Iris-SYN	84.78	-

Table 2. By testing set, the per-class results from the best validated NDIris3D model. APCER stands for Attack Presentation Classification Error Rate, and BPCER stands for Bonafide Presentation Classification Error Rate.

unseen attack type of synthetically generated images of CASIA-Iris-SYN. We can attribute some of this to the excellent work done by CASIA to generate realistic synthetic irises as CASIA-Iris-SYN recieves almost exactly the reverse accuracy as CASIA-IrisV1 within a tenth of a percent, but regardless, the network cannot pick up on synthetically-generated images.

For LivDetIris-2023, the many potential attack types with only 60 images leads to high anticipated error rates for attack images. In this case, however, we also see a high BPCER in Table 2. The low accuracy on real images is unexpected, especially considering that they are taken as NIR images in a similar way to the other sets.

Compared to the results claimed in D-NetPAD, these results are much worse. This may be due to the proprietary dataset of the D-NetPAD paper where all attack types that were tested were included in training.

The testing after using a training set combining CASIA-IrisV1 with CASIA-Iris-SYN achieved less than 40% accuracy on the LivDetIris-2023 and NDIris3D sets for both classes. For this reason, they are not discussed here.

6. FUTURE WORK

Iris PAD is a challenging task that requires research and development beyond what is presented here. One of the main considerations in this exploration is the lack of diversity in the available iris sets. Without additional sets of images containing textured contact lenses, synthetically generated irises, and additional attacks, a fair evaluation of this method is impossible. At the same time, this method seemed to be accurate in detecting attacks from the same set.

It should also be noted that the best hyperparameters selected in this exploration isolated the iris from contextual information around the eye. The model will not pick up on attacks that are best detected with contextual features or artifacts outside the iris, such as speculums in post-mortem cases. It is necessary to do further work in incorporating contextual features into an accurate iris PAD system so that it to be robust to different types of attacks.

It would also be beneficial to explore the use of additional networks or combinations of networks, beyond D-NetPAD and texture feature extractors, to further enhance performance. As biometric authentication systems become popular into the future, improving security, stability, and reliability is essential.

7. REFERENCES

[1] C. Raghavendra, A. Kumaravel, and S. Sivasubramanian, "Iris technology: A review on iris based biometric systems for unique human identification," in *2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET)*, 2017, pp. 1–6.

[2] Akshay Agarwal, Afzel Noore, Mayank Vatsa, and Richa Singh, "Enhanced iris presentation attack detection via contraction-expansion cnn," *Pattern Recognition Letters*, vol. 159, pp. 61–69, 2022.

[3] Aidan Boyd, Zhaoyuan Fang, Adam Czajka, and Kevin W. Bowyer, "Iris presentation attack detection: Where are we now?," *Pattern Recognition Letters*, vol. 138, pp. 483–489, 2020.

[4] Mateusz Trokielewicz, Adam Czajka, and Piotr Maciejewicz, "Post-mortem iris recognition with deep-learning-based image segmentation," *Image and Vision Computing*, vol. 94, pp. 103866, 2020.

[5] Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper, "Deep learning multi-layer fusion for an accurate iris presentation attack detection," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, 2020, pp. 1–8.

[6] J. Winston and Jude D, "A comprehensive review on iris image-based biometric system," *Soft Computing*, vol. 23, 10 2019.

[7] Hanaa Ahmed Salman and Mohammed Taha, "A brief survey on modern iris feature extraction methods," *International Journal of Engineering and Technology*, vol. 39, pp. 123–129, 01 2021.

[8] Daksha Yadav, Naman Kohli, Shivangi Yadav, Mayank Vatsa, Richa Singh, and Afzel Noore, "Iris presentation attack via textured contact lens in unconstrained environment," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, pp. 503–511.

[9] Renu Sharma and Arun Ross, "D-netpad: An explainable and interpretable iris presentation attack detector," in *2020 IEEE International Joint Conference on Biometrics (IJCB)*, 2020, pp. 1–10.

[10] Cunjian Chen and Arun Ross, "An explainable attention-guided iris presentation attack detector," in *2021 IEEE Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2021, pp. 97–106.

[11] Ranjeet Ranjan Jha, Gaurav Jaswal, Divij Gupta, Shreshth Saini, and Aditya Nigam, "Pixisegnet: pixel-level iris segmentation network using convolutional encoder–decoder with stacked hourglass bottleneck," *IET Biometrics*, vol. 9, no. 1, pp. 11–24, 2020.

[12] "Casia-irisv4 data set," 2020.

[13] "Ndiris3d," 2018.

[14] Adam Czajka, Patrick Tinsley, Mahsa Mitcheff, Patrick Flynn, and Kevin Bowyer, ," Mar 2023.

[15] Daksha Yadav, Naman Kohli, Mayank Vatsa, Richa Singh, and Afzel Noore, "Detecting

textured contact lens in uncontrolled environment using densepad," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019, pp. 2336–2344.