Relativistic Discriminator: A One-Class Classifier for Generalized Iris Presentation Attack Detection

Shivangi Yadav
Michigan State University
yadavshi@msu.edu

Cunjian Chen
Michigan State University
cunjian@msu.edu

Arun Ross
Michigan State University
rossarun@msu.edu

Abstract

Iris based recognition systems are vulnerable to presentation attacks (PAs) where artifacts such as cosmetic contact lenses, artificial eyes and printed eyes can be used to fool the system. While many learning-based algorithms have been proposed to detect such attacks, very few are equipped to handle previously unseen or newly constructed PAs. In this research, we propose a presentation attack detection (PAD) method that utilizes a discriminator that is trained to distinguish between bonafide iris images and synthetically generated iris images. We hypothesize that such a discriminator will generate a tight boundary around the bonafide samples. This would allow the discriminator to better separate the bonafide samples from all types of PA samples. For generating synthetic irides, we train the Relativistic Average Standard Generative Adversarial Network (RaSGAN) that has been shown to generate higher resolution and better quality images than standard GANs. The relativistic discriminator (RD) component of the trained RaSGAN is then appropriated for PA detection and is referred to as RD-PAD. Experimental results convey the efficacy of the RD-PAD as a one-class anomaly detector.

1. Introduction

Iris based recognition systems are known for their reliability and are being widely used in many applications [25], but are vulnerable to challenges posed by presentation attacks (PAs) [28]. Researchers have developed numerous methods to generate artificial samples that can successfully circumvent iris biometric systems (Figure 1). For example, an adversary can present a printed image [13, 34] to an iris sensor to impersonate another subject, or use cosmetic contact lenses [31, 17] and artificial eyes [10] to either obfuscate their own identity or to create a virtual identity. Due to their serious impact on the security of a system, detecting such spoof or obfuscation attacks has become a key research topic in biometrics. Some of the commonly used iris presentation attack detection (PAD) algorithms are summarized below:

- Print Attack: Gupta et al. [13] used textual descriptors such as LBP, HOG and GIST to detect print attacks. Raghavendra and Busch [27] used multi-scale binarized statistical image features (BSIF) combined with cepstral features for print attack detection.
- Cosmetic Contact Lens: Kohli et al. [18] used a variant of LBP to obtain useful textural features for contact lens detection. Other approaches include weighted local binary pattern and deep features extracted from CNNs [24].
- Synthetic/Artificial Eye: This type of attack is less common than print and cosmetic contact lens but is gaining interest in recent times [19]. Some of the proposed methods for detection are based on multispectral imaging [5] and eye gaze tracking [20].
- Multiple Attacks: PAD algorithms can also be designed to address various types of PAs. In [14], Hoffman et al. designed a CNN that used patch information along with a segmentation mask from an unnormalized iris image to learn image characteristics that differentiate PA samples from bonafide samples. Menotti et al. [24] proposed a CNN based approach to detect spoofs in different modalities, viz., iris, face and fingerprint.

While the aforementioned methods exhibit reasonably good performance on *known* PAs,[1] most of them formulate presentation attack detection as a binary-class problem. This demands the availability of a large collection of both bonafide[2] and PA samples to train classifiers. Obtaining a large number of PA samples can be much more difficult than bonafide iris samples. Further, classifiers are usually trained and tested across similar PAs, but PAs encountered in operational systems can be diverse in nature and may not be

[1] We use the term "Known PAs" to refer to PAs that are used or observed during the development or training stage of the detector.

[2] In previous literature, the term "live" has been used in lieu of "bonafide". Both terms refer to biometric samples that are *not* PAs.

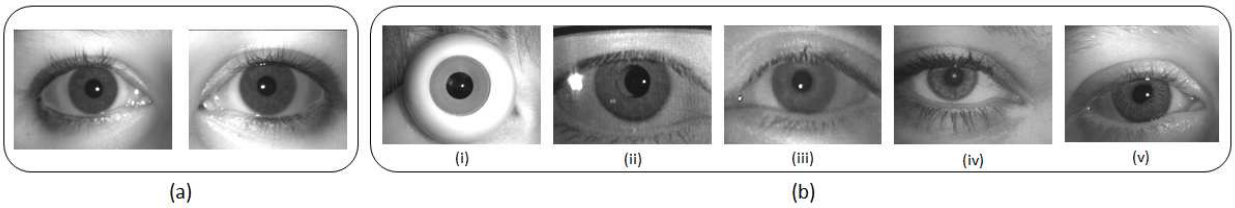


Figure 1: Samples of real and spoof iris images from MSU-Iris-PA01 [32]: (a) bonafide samples and (b) presentation attack samples: (i) artificial eye, (ii) - (iii) printed iris, (iv) Kindle display and (v) cosmetic contact lens

available during the training stage. Therefore, PAD algorithms based on binary classifiers might fail to generalize to *unseen* PAs.[3] In the literature, researchers have attempted to impart *generalizability* to PAD algorithms by adopting an anomaly detection approach also known as one-class classification. In this approach, the PA detector learns the distribution of bonafide samples only and uses this information to detect "outliers" that would presumably correspond to PAs. In [6], Ding and Ross proposed an ensemble of one-class classifiers, trained on hand-crafted features, to detect unseen fingerprint PAs. Nikisins et al. [26] used the gaussian mixture model as a one-class classifier trained on image quality measure (IQM) features [9] for generalized face PA detection. In [35, 29, 15, 8] researchers used deep architectures such as CNNs and generative adversarial networks (GANs) [12] for anomaly detection in general image classification problems, which suggest the efficacy of deep features in anomaly detection.

A GAN has two components - a generator that generates synthetic images, and a discriminator that distinguishes between real images and synthesized images. In GAN-based anomaly detection [29], the discriminator is trained to learn the distribution of the class-of-interest while competing with the generator. The better the quality of the generated samples, the more accurate is the distribution of the class-of-interest learned by the discriminator. Therefore, it is vital that the GAN architecture used for anomaly detection also possesses good *generative* capability. In [16], Jolicoeur-Martineau introduced the notion of the *relativistic* discriminator to enhance the generative capability of the generator, and showed that such a GAN produced better quality images than a Standard GAN (SGAN) [12], Least Squares GAN (LS-GAN) [23] and Wasserstien GAN (WGAN) [1]. The author postulated that the gradients of the discriminator in SGAN and other related architectures come only from the generated samples, thereby preventing the discriminator to learn from real images and depending entirely on generated images. Consequently, the generative capability of such networks is restricted and the synthesized samples may not look natural. On the other hand, the Relativistic Average Standard Generative Adversarial Network (RaSGAN) learns the distribution of both real and generated samples as its gradients come from real as well as synthesized data. This helps the generator to learn a distribution that is much more similar to that of the given bonafide samples. In [32], Yadav et al. used RaSGAN to generate high resolution synthetic irides, and then used the generated images to train a PA detector.

In this paper, we train a RaSGAN to generate high quality synthetic irides from bonafide irides, but for a very different purpose compared to [32]. The relativistic discriminator (RD) of the ensuing RaSGAN learns to separate bonafide irides from their synthetic counterparts. In the process, the RD fits a tight boundary (possibly non-contiguous) around the bonafide samples making it an effective one-class anomaly detector, which we refer to as RD-PAD (see Figure 4). **The proposed method, in principle, does not require any PA samples during training; only bonafide samples are needed during training.** Consequently, anything that lies outside the learned distribution on bonafide samples is classified as PA. The major contributions of this work are:

- We propose RD-PAD for unseen PA detection that utilizes the relativistic discriminator from a RaSGAN to discriminate bonafide samples from PAs. The proposed PAD algorithm requires only bonafide samples for training.
- We analyze the performance of state-of-the-art PAD algorithms on unseen PAs and compare them with the proposed method.
- We evaluate the performance of the proposed RD-PAD when it is fine-tuned using a few PA samples and tested on PAs that are not used during training.

2. Background

GANs [12] have been utilized to generate natural looking visual stimuli using two different components: discriminator (D) and generator (G) that compete with each other. G aims to generate good quality synthetic data that can fool D, while D challenges G by learning how to distinguish between real and synthetically generated data.

[3]The term "Unseen PAs" refers to PAs that were not used or observed during the training phase of the detector.

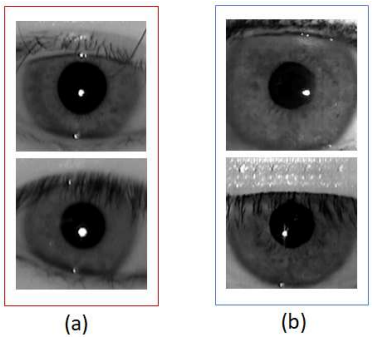


Figure 2: (a) Real bonafide iris images and (b) Synthetic iris images generated using RaSGAN.

2.1. Standard Generative Adversarial Networks (SGANs)

As mentioned previously, SGANs consists of two networks D and G that are wrapped in a min-max game to update their weights and compete against each other. This is achieved by alternatively minimizing and maximizing the objective function S as,

$$\min_G \max_D S(D,G) = \mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}}[log(D(\boldsymbol{x_r}))] + \mathbb{E}_{\boldsymbol{z}\sim\mathbb{M}}[log(1-D(G(\boldsymbol{z})))]. \quad (1)$$

Here, $\boldsymbol{x_r} \sim \mathbb{P}$ indicates that $\boldsymbol{x_r}$ is from the true data distribution $\mathbb{P}$. Also, $D(\boldsymbol{x})$ is the output obtained after applying the sigmoid function (sig), to the non-transformed layer $NT(\boldsymbol{x})$,

$$D(\boldsymbol{x}) = sig(NT(\boldsymbol{x})). \quad (2)$$

Here, $NT(\boldsymbol{x})$, refers to the output of the last convolutional layer before the application of logistic regression. Traditional GANs such as SGAN, WGAN and DCGAN design discriminators that optimize their ability to distinguish synthetically generated data from bonafide samples. While they have been reported to perform well [16] on low resolution datasets, unstable training and optimization have been observed when they are used with high-resolution data [16]. This instability can be explained in terms of the gradient of the traditional discriminator:

$$\nabla_\theta S_D = -\mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}}[(1-D(\boldsymbol{x_r}))\nabla_\theta NT(\boldsymbol{x_r})] + \mathbb{E}_{\boldsymbol{x_s}\sim\mathbb{Q}}[D(\boldsymbol{x_s})\nabla_w NT(\boldsymbol{x_s})]. \quad (3)$$

Here, $\boldsymbol{x_s} \sim \mathbb{Q}$ indicates that $\boldsymbol{x_s}$ is from the model distribution $\mathbb{Q}$, i.e., synthetically generated data. During training, when the discriminator is optimized, $1 - D(\boldsymbol{x_r})$ converges to 0 indicating that the gradient of D comes mostly from synthetically generated data. Consequently, the generator stops learning to generate natural looking images. This in turn restricts the ability of the discriminator to learn a good representation for bonafide irides. However, we would like to learn a stable model with a discriminator that has a better understanding of the distribution of bonafide irides.

2.2. Relativistic Standard Generative Adversarial Networks (RSGAN)

In [16], Jolicoeur-Martineau introduced the *relativistic* discriminator, D_R, which aims to maximize the probability that bonafide irides are more real than synthetically generated irides using the following objective function:

$$R(D_R) = -\mathbb{E}_{(\boldsymbol{x_r},\boldsymbol{x_s})\sim(\mathbb{P},\mathbb{Q})}[log(sig(NT(\boldsymbol{x_r}) - NT(\boldsymbol{x_s})))]. \quad (4)$$

In this case, the training of the discriminator depends on both bonafide and synthetic data. From Equation (4), we can see that its gradient depends on $\boldsymbol{x_r}$ as well as $\boldsymbol{x_s}$, which ensures that the generator G_R continues learning to synthesize real looking irides until convergence. In RSGAN, G_R aims to generate images that maximize the probability that they are more real than bonafide samples:

$$R(G_R) = -\mathbb{E}_{(\boldsymbol{x_r},\boldsymbol{x_s})\sim(\mathbb{P},\mathbb{Q})}[log(sig(NT(\boldsymbol{x_s}) - NT(\boldsymbol{x_r})))]. \quad (5)$$

Therefore, D_R and G_R compete with each other to generate good quality synthetic irides.

2.3. Relativistic Average Standard Generative Adversarial Network (RaSGANs)

In RSGAN, a sample in distribution $\mathbb{P}$ is compared with every sample in $\mathbb{Q}$ (and vice-versa), which might not be very efficient. Therefore, to make this adversarial network more efficient, Jolicoeur-Martineau [16] updated the objective function of D_R and G_R to compare a sample in distribution $\mathbb{P}$ with the *average* of samples from $\mathbb{Q}$ (and vice-versa):

$$R^{avg}(D_R) = -\mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}}[log(\hat{D}(\boldsymbol{x_r}))] - \mathbb{E}_{\boldsymbol{x_s}\sim\mathbb{Q}}[log(1-\hat{D}(\boldsymbol{x_s}))], \quad (6)$$

$$R^{avg}(G_R) = -\mathbb{E}_{\boldsymbol{x_s}\sim\mathbb{Q}}[log(\hat{D}(\boldsymbol{x_s}))] - \mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}}[log(1-\hat{D}(\boldsymbol{x_r}))], \quad (7)$$

$$\hat{D} = \begin{cases} sig(NT(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{x_s}\sim\mathbb{Q}}NT(\boldsymbol{x_s})), \text{ if } \boldsymbol{x} = \boldsymbol{x_r} \\ sig(NT(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}}NT(\boldsymbol{x_r})), \text{ if } \boldsymbol{x} = \boldsymbol{x_s}. \end{cases} \quad (8)$$

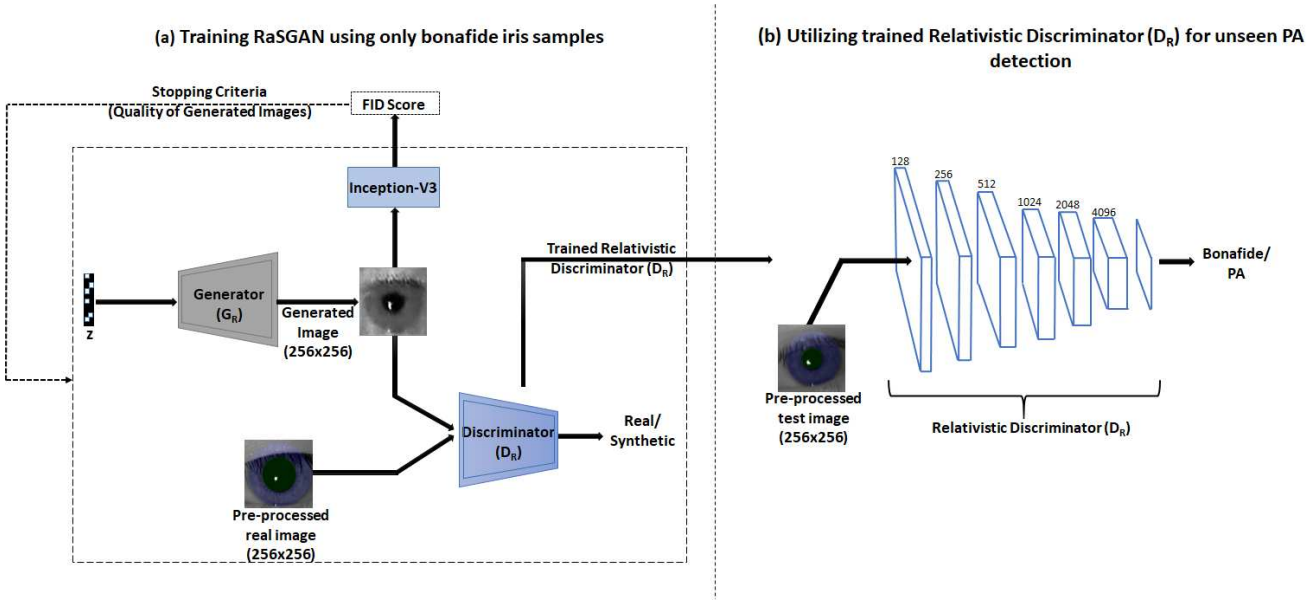


Figure 3: Schematic of the proposed presentation attack detector. The proposed RD-PAD (right) utilizes the relativistic discriminator, D_R, of a RaSGAN that is trained predominantly using real bonafide samples (left). D_R learns to distinguish between real bonafide samples and the corresponding high-quality synthetic samples generated by the generator, G_R. The trained discriminator is then leveraged and used to distinguish between bonafide samples and presentation attack samples.

This network is referred to as Relativistic Average Standard Generative Adversarial Network (RaSGAN). The generator and discriminator in RaSGAN use the *relative* information between bonafide and synthetic data to generate good quality irides. When the generator competes with the discriminator in this fashion, it gives the discriminator the opportunity to learn a more effective distribution for bonafide irides. *This is the key observation that we exploit in this work: the ability of the generator to synthesize more natural looking irides, and the ability of the discriminator to learn a more accurate distribution for bonafide irides (see Figure 4).* The quality of the generated synthetic samples are further analyzed and enhanced using Frechet Inception Distance (FID) score [2]. This involves comparing the distribution of synthesized data ($\mathbb{Q}$) with real data ($\mathbb{P}$) to generate a score that can be utilized to evaluate the quality of the generated samples:

$$FID = \|\boldsymbol{\mu_r} - \boldsymbol{\mu_s}\|^2 + Tr(\boldsymbol{\Sigma_r} + \boldsymbol{\Sigma_s} - 2\sqrt{\boldsymbol{\Sigma_r}\boldsymbol{\Sigma_s}}). \quad (9)$$

Here, $\boldsymbol{\mu_r}, \boldsymbol{\mu_s}, \boldsymbol{\Sigma_r}$ and $\boldsymbol{\Sigma_s}$ represent the statistics of the real and synthetically generated data samples and Tr is the trace of $(\boldsymbol{\Sigma_r} + \boldsymbol{\Sigma_s} - 2\sqrt{\boldsymbol{\Sigma_r}\boldsymbol{\Sigma_s}})$. Thus, the RaSGAN is trained until good quality images are obtained, i.e., the generated images have low FID scores [2].

3. Model Architecture

The RaSGAN architecture used in this work consists of two important components: relativistic discriminator and generator that are implemented using PyTorch libraries.[4] The network is trained using only bonafide samples (see Section 5) that are first pre-processed to align them using the center coordinates of the pupil and the iris. The coordinates themselves are obtained using VeriEye SDK.[5] The aligned images are further center-cropped and then resized to obtain images of size 256×256. The input to the relativistic discriminator (D_R) are pre-processed bonafide samples and synthetically generated irides from G_R. The input to the generator is a noise sample $\boldsymbol{z}$ of size 1×1×128, where $\boldsymbol{z}$ is sampled from a normal noise distribution. The architecture of the RaSGAN used in this work is summarized below.

- **Relativistic Discriminator:** The D_R in RaSGAN has been constructed using seven convolutional layers with kernel size 4×4 and stride=2 (apart from the last convolution layer where stride=1). The first convolutional layer is followed by leaky rectified units while the remaining layers (except for the last convolutional layer) are followed by both batch normalization and leaky rectified units.

- **Relativistic Generator:** The G_R aims to generate natural looking irides of size 256×256 from input $\boldsymbol{z}$, and has been implemented using seven transposed convolutional layer. Each layer has a kernel size of 4×4 and

[4] https://github.com/alexiajm/relativisticgan
[5] www.neurotechnology.com/verieye.html

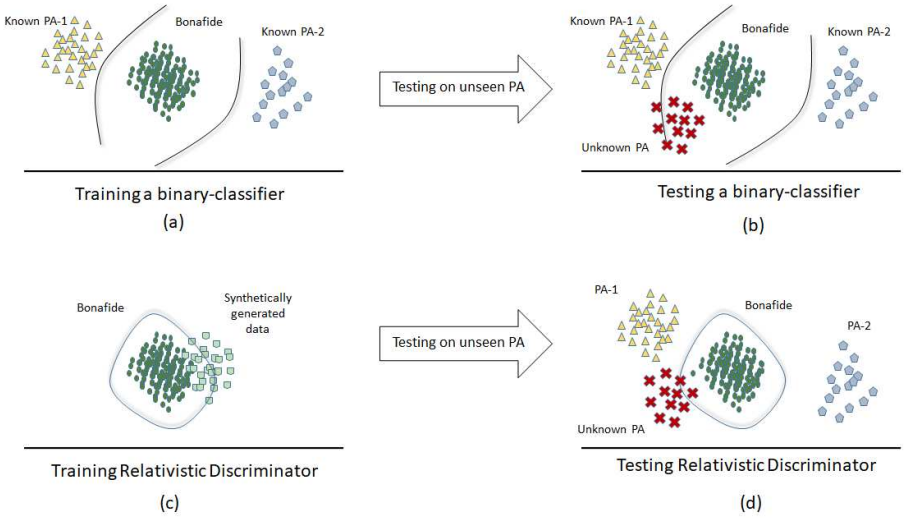


Figure 4: Illustration of why the relativistic discriminator of the RaSGAN is beneficial for generalized PA detection. (a) The decision boundary learned when training a typical binary-classifier depends on the samples from both the classes during training. (b) If a new PA type is encountered during testing, such classifiers are bound to fail dramatically. (c) The Relativistic Discriminator is trained using bonafide and the corresponding synthetically generated images, which help it to learn a tight boundary around the bonafide distribution. (d) Since training of the discriminator does not depend on PAs, the performance of this one-class classifier is less hindered in presence of newly encountered PAs.

stride=2, except for the first transposed layer that has stride=1. Batch normalization and rectified linear units are applied to the output of each transposed convolutional layer.

4. Relativistic Discriminator-PAD (RD-PAD)

4.1. Method-I: RD-PAD Trained with Bonafide Samples Only

Training a good discriminator is an important aspect of the proposed method. Therefore, as the first step, RaSGAN is trained using bonafide irides only. All the samples used during training are center aligned and cropped to size 256×256, as described earlier. The D_R obtained after RaSGAN training outputs the probability that a given input sample belongs to the bonafide distribution, i.e., an ideally trained D_R should satisfy $D_R(\boldsymbol{x}) \approx 1$, when $\boldsymbol{x}$ belongs to the bonafide iris category, and $D_R(\boldsymbol{x}) \approx 0$, when $\boldsymbol{x}$ represents some PA sample.

4.2. Method-II: RD-PAD Fine-tuned with Some PA Samples

The D_R in Method-I is familiar with the distribution of bonafide samples but has no knowledge of any PA distributions. So, it learns a tight boundary encompassing the bonafide class, which can lead to misclassification of some bonafide irides (especially in the cross-sensor scenario). In Method-II, we further expand the capabilities of the RD-PAD by fine-tuning D_R using bonafide samples *and* a few known PAs. This enables D_R to learn the *difference* between bonafide irides and some PAs albeit in a limited way.

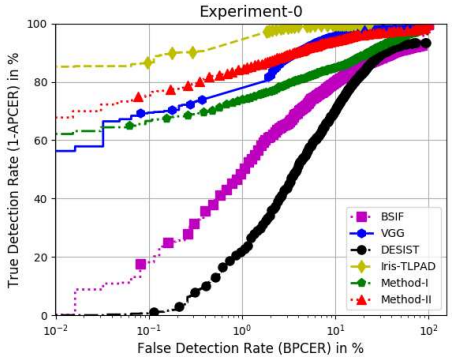


Figure 5: ROC curve demonstrating the performance of existing PAD algorithms and the proposed methods on known PAs (as described in Experiment-0).

Further, since this research focuses on unseen iris PA detection, the PA types used to fine-tune D_R are mutually disjoint with the PA types used in the test set (see Section 5 for more details).

5. Analysis of RD-PAD for Seen and Unseen Presentation Attack Detection

In this section, we evaluate the efficacy of RD-PAD in detecting unseen PAs using publicly available iris datasets

Table 1: The iris datasets used in this research. The gray cells represents PA types that are not present in some datasets. The iris images are first pre-processed to produce images of size 256×256. Images that could not be processed by VeriEye were removed from the training sets. On the other hand, all such images are labeled as PAs in the test sets. The datasets are further adjusted to balance the samples in the two classes (bonafide versus PA).

	Berc-Iris-Fake[22]		**Casia-Iris-Fake** [30]		**NDCLD15** [7]		**LivDet15** [34]		**LivDet17** [33]		**MSU-Iris-PA01** [32]	
	Total	**Used**	**Total**	**Used**	**Total**	**Used**	**Total**	**Used**	**Total**	**Used**	**Total**	**Used**
Bonafide	2,778	2,778	6,000	6,000	1,100	695	2,606	1,402	11,372	10,763	1,343	1,000
Printed eyes	1,200	1,200	640	640			4,473	4,259	12,099	7,336	1,938	1,830
Cosmetic contact lenses	140	140	740	740	1,100	1,100			5,287	5,287	108	108
Artificial eyes	80	80	400	400							352	352
Kindle display											125	125

Table 2: Attack Presentation Classification Error Rate (APCER) at 0.1%, 1% and 5% Bonafide Presentation Classification Error Rate (BPCER) of existing PAD algorithms and the proposed RD-PAD (Method-I and Method-II) on known PAs as described in Experiment-0. **Lower the APCER, better is the performance**.

	BSIF+SVM [7]	**Pre-trained VGG-16** [11]	**DESIST** [18]	**Iris-TLPAD** [3]	**Method-I**	**Method-II**
APCER(@0.1%)	81.83	30.55	99.04	**13.34**	33.40	24.65
APCER(@1%)	51.26	23.27	78.13	**6.64**	26.18	15.68
APCER(@5%)	27.54	8.09	43.64	**0.89**	18.35	8.43

summarized in Table 1. We also compute the performance of current state-of-the-art PAD algorithms, viz., BSIF+SVM [7], pre-trained VGG-16 [11], DESIST [18] and Iris-TLPAD [3, 4], and compare them against that of RD-PAD. A total of 22,638 bonafide irides and 23,597 PA samples are utilized to train and test these algorithms. PA samples used in this work consist of multiple types of attacks including cosmetic contact lenses, artificial eyes, Kindle display-attack and printed eyes.

5.1. Seen Presentation Attacks

This is a **baseline experiment**, Experiment-0, which demonstrates the performance of existing PAD algorithms and proposed Method-I and Method- II on known PAs. In this experiment, the PAD algorithms were trained using 12,875 bonafide irides and 12,326 PAs containing cosmetic contact lens, printed eye, artificial eye and Kindle display-attack. On the other hand, the proposed Method-I was trained using **only bonafide** samples. Method-II was trained using bonafide samples and only 800 randomly selected known PAs. All the trained algorithms were then tested on 6,207 bonafide and 6,529 PA samples consisting of cosmetic contact lenses, printed eyes, artificial eyes and Kindle-display attacks.

5.2. Unseen Attack: Cosmetic Contact Lenses and Kindle Display

In this section, we evaluate the performance of the PAD algorithms for generalized PA detection when training data does not include PAs such as cosmetic contact lens and Kindle display-attack.

- **Experiment-1**: Here, the other PAD algorithms are trained using 2,778 bonafide samples from Berc-iris-fake [21, 22], and 3,007 printed eyes and artificial eyes from the other datasets. Note that Method-I is trained using **only bonafide** samples and Method-II is first trained using only bonafide samples and then **fine tuned** using only 800 PA samples. All the trained algorithms are then tested using 3,913 bonafide samples (excluding Berc-Iris-Fake) and 3,279 PA samples corresponding to cosmetic contact lenses and Kindle display-attacks.

- **Experiment-2**: Here, the other PAD algorithms are trained using 6,000 bonafide samples from Casia-Iris-Fake [30], and 6,187 PA samples from the other datasets consisting of only printed eyes and artificial eyes. Similar to the previous experiment, Method-I is trained using **only bonafide** samples while Method-II is first trained using only bonafide samples and then **fine-tuned** using only 800 PA samples. These algorithms are tested using 5,634 bonafide samples from other datasets (excluding Casia-Iris-Fake) and 5,556 PAs consisting of cosmetic contact lenses and Kindle display-attacks.

5.3. Unseen Attack: Printed Eyes and Artificial Eyes

In this section, we evaluate the performance of the PAD algorithms when printed eyes and artificial eyes are used as unseen presentation attacks.

- **Experiment-3**: In this experiment, the other PAD algorithms are trained using 2,778 bonafide samples

Table 3: Attack Presentation Classification Error Rate (APCER) at 0.1%, 1% and 5% Bonafide Presentation Classification Error Rate (BPCER) of existing PAD algorithms and the proposed methods on unseen PAs as described in Experiment-1 and Experiment-2. **Lower the APCER, better is the performance**.

	BSIF+SVM [7]	**Pre-trained VGG-16** [11]	**DESIST** [18]	**Iris-TLPAD** [3]	**Method-I**	**Method-II**
APCER(@0.1%)	99.95	99.95	94.14	99.52	74.39	**50.26**
APCER(@1%)	97.83	95.88	79.29	95.25	44.35	**37.05**
APCER(@5%)	91.61	72.55	54.34	85.06	34.06	**21.79**

(a) Experiment-1

	BSIF+SVM [7]	**Pre-trained VGG-16** [11]	**DESIST** [18]	**Iris-TLPAD** [3]	**Method-I**	**Method-II**
APCER(@0.1%)	100	99.98	100	99.63	66.98	**61.80**
APCER(@1%)	98.31	82.04	96.99	96.33	53.69	**39.40**
APCER(@5%)	75.05	70.09	83.05	89.87	38.72	**26.49**

(b) Experiment-2

Table 4: Attack Presentation Classification Error Rate (APCER) at 0.1%, 1% and 5% Bonafide Presentation Classification Error Rate (BPCER) of existing PAD algorithms and the proposed RD-PAD (Method-I and Method-II) on unseen PAs as described in Experiment-3 and Experiment-4. **Lower the APCER, better is the performance**.

	BSIF+SVM [7]	**Pre-trained VGG-16** [11]	**DESIST** [18]	**Iris-TLPAD** [3]	**Method-I**	**Method-II**
APCER(@0.1%)	90.29	90.26	97.88	N/A	58.34	**37.13**
APCER(@1%)	90.29	80.79	93.55	27.49	45.72	**27.19**
APCER(@5%)	87.75	66.78	81.35	**17.06**	36.74	19.56

(a) Experiment-3

	BSIF+SVM [7]	**Pre-trained VGG-16** [11]	**DESIST** [18]	**Iris-TLPAD** [3]	**Method-I**	**Method-II**
APCER(@0.1%)	95.37	99.92	100	N/A	60.71	**32.49**
APCER(@1%)	90.74	94.19	98.90	34.86	38.30	**23.30**
APCER(@5%)	81.52	77.69	93.68	17.59	25.06	**17.58**

(b) Experiment-4

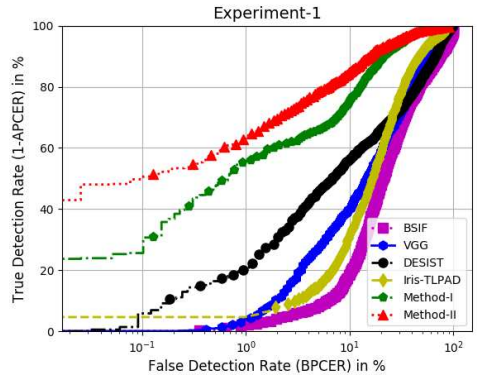
from Berc-Iris-Fake and 3,093 PA samples from the other datasets. The PA samples in training set consists of only cosmetic contact lenses and Kindle display-attacks. The proposed Method-I is trained using **only bonafide** samples and Method-II is first trained using only bonafide samples and then **fine-tuned** using only 500 PA samples from the training set. The test set consists of 3,450 bonafide samples and 3,347 PA samples corresponding to printed eyes and artificial eyes.

- **Experiment-4**: Here, the PAD algorithms are trained using 6,000 bonafide samples from Casia-Iris-Fake [30] and 5,681 PA samples from the other datasets corresponding to cosmetic lenses and Kindle display-attacks. Similar to the previous experiment, Method-I is trained using **only bonafide** samples while Method-II is first trained using only bonafide samples and then **fine-tuned** using only 500 PA samples. These algorithms are tested using 8,517 bonafide samples from other datasets (excluding Casia-Iris-Fake) and 8,865 PAs corresponding to printed eyes and artificial eyes.

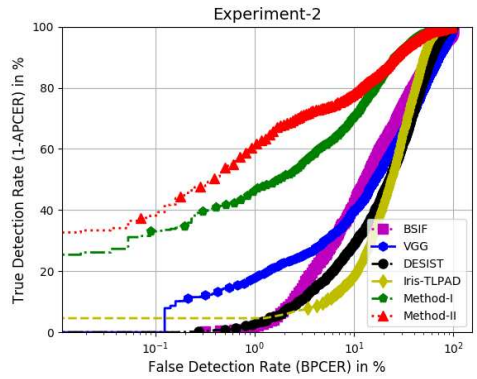
5.4. Analysis

The results in Table 2 show that deep networks such as VGG-16, Iris-TLPAD and the proposed methods achieve good (low) Attack Presentation Classification Error Rate (APCER) at 5% Bonafide Presentation Classification Error Rate (BPCER)[6] when trained and tested on the same type of PAs. However, Tables 3 and 4 show that current PAD algorithms do not perform well when tested on unseen PAs. In Experiment-1, APCERs of 34.06% and 21.79% are obtained at 5% BPCER for the proposed Method-I and Method-II, respectively. On the other hand, current PAD algorithms obtained a much higher APCER thereby highlighting the shortcomings of these algorithms for unseen PA detection. In Experiment-3 and Experiment-4, Method-II and TL-PAD obtained comparable performance at 5% BPCER for unseen printed and artificial eyes. However, TL-PAD failed to produce any valid output (NA) at 0.1% BPCER and has a higher APCER than Method-II at 1% BPCER. Also, TL-PAD performed poorly on un-

[6]APCER is equivalent to (1 - True Detection Rate (TDR)) while BPCER is equivalent to False Detection Rate (FDR).

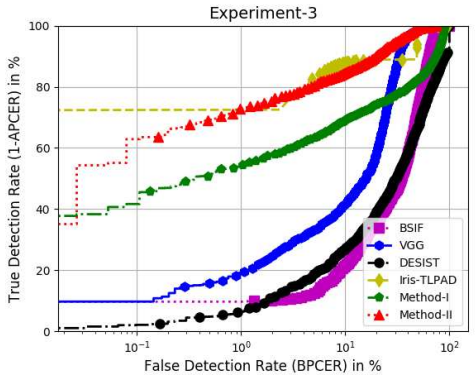


(a)

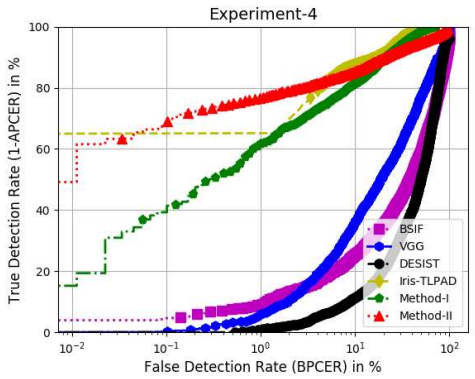


(b)

Figure 6: ROC curves demonstrating the performance of existing PAD algorithms and the proposed RD-PAD methods on unseen PAs, as described in Experiment-1 and Experiment-2.



(a)



(b)

Figure 7: ROC curves demonstrating the performance of existing PAD algorithms and the proposed methods on unseen PAs, as described in Experiment-3 and Experiment-4

seen cosmetic contact lenses and Kindle display-attacks in Experiment-1 and Experiment-2. This indicates the shortcoming of TL-PAD in handling unseen cosmetic contact lenses. Comparing all the results, we can conclude that the proposed algorithms have better generalizability over both seen and unseen attacks (see Figures 5, 6 and 7). Additionally, in [4, 3] TL-PAD was evaluated on a subset of the LivDet-Iris 2017 dataset, and achieved better performance than the three participating algorithms in the competition. Hence, this paper makes an indirect comparison against the other algorithms published in LivDet-Iris 2017.

6. Summary and Future Work

The goal of this work was to develop an iris presentation attack detector for unseen attacks. To facilitate this, we harness the relativistic discriminator of a RaSGAN that is trained to distinguish between bonafide iris samples and the corresponding synthetically generated iris samples. We hypothesize that such a discriminator more effectively learns the distribution of bonafide samples and will, therefore, reject PA samples that do not fall within this distribution. In this regard, the discriminator behaves as a one-class classifier since, in principle, it does not require data from PA samples during the training stage. Experimental results demonstrate the efficacy of the proposed method over current state-of-the-art PAD methods, especially on unseen attacks. The proposed approach can be further expanded to other biometric modalities such as face and fingerprint.

7. Acknowledgment

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA RD Contract No. 2017 - 17020200004. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning (ICML)*, pages 214–223, 2017.

[2] S. Barratt and R. Sharma. A note on the inception score. *International Conference on Machine Learning Workshops (ICMLW)*, 2018.

[3] C. Chen and A. Ross. A multi-task convolutional neural network for joint iris detection and presentation attack detection. In *IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pages 44–51, 2018.

[4] C. Chen and A. Ross. Exploring the use of iriscodes for presentation attack detection. In *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–9, 2018.

[5] R. Chen, X. Lin, and T. Ding. Liveness detection for iris recognition using multispectral images. *Pattern Recognition Letters*, 33(12):1513–1519, 2012.

[6] Y. Ding and A. Ross. An ensemble of one-class SVMs for fingerprint spoof detection across different fabrication materials. In *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2016.

[7] J. S. Doyle and K. W. Bowyer. Robust detection of textured contact lenses in iris recognition using bsif. *IEEE Access*, 3:1672–1683, 2015.

[8] J. J. Engelsma and A. K. Jain. Generalizing Fingerprint Spoof Detector: Learning a One-Class Classifier. *arXiv preprint arXiv:1901.03918*, 2019.

[9] J. Galbally, S. Marcel, and J. Fierrez. Image quality assessment for fake biometric detection: Application to iris, fingerprint, and face recognition. *IEEE Transactions on Image Processing*, 23(2):710–724, 2014.

[10] J. Galbally, A. Ross, M. Gomez-Barrero, J. Fierrez, and J. Ortega-Garcia. Iris image reconstruction from binary templates: An efficient probabilistic approach based on genetic algorithms. *Computer Vision and Image Understanding*, 117(10):1512–1525, 2013.

[11] L. Gatys, A. S. Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 262–270, 2015.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial Nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014.

[13] P. Gupta, S. Behera, M. Vatsa, and R. Singh. On iris spoofing using print attack. In *IEEE International Conference on Pattern Recognition (ICPR)*, pages 1681–1686, 2014.

[14] S. Hoffman, R. Sharma, and A. Ross. Convolutional neural networks for iris presentation attack detection: Toward cross-dataset and cross-sensor generalization. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1620–1628, 2018.

[15] S. Hoffman, R. Sharma, and A. Ross. Iris + Ocular: Generalized Iris Presentation Attack Detection Using Multiple Convolutional Neural Networks. In *IAPR International Conference on Biometrics (ICB)*, 2019.

[16] A. Jolicoeur-Martineau. The Relativistic Discriminator: a key element missing from standard GAN. In *International Conference on Learning Representations (ICLR)*, 2019.

[17] N. Kohli, D. Yadav, M. Vatsa, and R. Singh. Revisiting iris recognition with color cosmetic contact lenses. In *International Conference on Biometrics (ICB)*, pages 1–7, 2013.

[18] N. Kohli, D. Yadav, M. Vatsa, R. Singh, and A. Noore. Detecting medley of iris spoofing attacks using DESIST. In *IEEE International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–6, 2016.

[19] N. Kohli, D. Yadav, M. Vatsa, R. Singh, and A. Noore. Synthetic iris presentation attack using iDCGAN. In *IEEE International Joint Conference on Biometrics (IJCB)*, pages 674–680, 2017.

[20] E. C. Lee, Y. J. Ko, and K. R. Park. Fake iris detection method using purkinje images based on gaze position. *Optical Engineering*, 47(6):1 – 16, 2008.

[21] S. J. Lee, K. R. Park, and J. Kim. Robust fake iris detection based on variation of the reflectance ratio between the iris and the sclera. In *IEEE Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pages 1–6, 2006.

[22] S. J. Lee, K. R. Park, Y. J. Lee, K. Bae, and J. H. Kim. Multifeature-based fake iris detection method. *Optical Engineering*, 46(12):1–10, 2007.

[23] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley. Least squares generative adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2794–2802, 2017.

[24] D. Menotti, G. Chiachia, A. da Silva Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcao, and A. Rocha. Deep representations for iris, face, and fingerprint spoofing detection. *In IEEE Transactions on Information Forensics and Security*, 10:864–879, 2015.

[25] I. Nigam, M. Vatsa, and R. Singh. Ocular biometrics: A survey of modalities and fusion approaches. *Information Fusion*, 26:1–35, 2015.

[26] O. Nikisins, A. Mohammadi, A. Anjos, and S. Marcel. On effectiveness of anomaly detection approaches against unseen presentation attacks in face anti-spoofing. In *IAPR International Conference on Biometrics (ICB)*, 2018.

[27] R. Raghavendra and C. Busch. Presentation attack detection algorithm for face and iris biometrics. In *European Signal Processing Conference (EUSIPCO)*, pages 1387–1391, 2014.

[28] A. Ross, S. Banerjee, C. Chen, A. Chowdhury, V. Mirjalili, R. Sharma, T. Swearingen, and S. Yadav. Some Research Problems in Biometrics: The Future Beckons. In *IAPR International Conference on Biometrics (ICB)*, 2019.

[29] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging (IPMI)*, pages 146–157, 2017.

[30] Z. Sun, H. Zhang, T. Tan, and J. Wang. Iris image classification based on hierarchical visual codebook. *In IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1120–1133, 2014.

[31] 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. *In IEEE Transactions on Information Forensics and Security*, 9:851–862, 2014.

[32] S. Yadav, C. Chen, and A. Ross. Synthesizing Iris Images using RaSGAN with Application in Presentation Attack Detection. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019.

[33] D. Yambay, B. Becker, N. Kohli, D. Yadav, A. Czajka, K. W. Bowyer, S. Schuckers, R. Singh, M. Vatsa, A. Noore, et al. LivDet Iris 2017 - Iris Liveness Detection Competition. In *IEEE International Joint Conference on Biometrics (IJCB)*, pages 733–741, 2017.

[34] D. Yambay, B. Walczak, S. Schuckers, and A. Czajka. LivDet-Iris 2015 - iris liveness detection competition 2015. In *IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, pages 1–6, 2017.

[35] M. Zhang, J. Wu, H. Lin, P. Yuan, and Y. Song. The application of one-class classifier based on CNN in image defect detection. *Procedia Computer Science*, 114:341–348, 2017.