Synthesizing Iris Images using RaSGAN with Application in 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

In this work we design a new technique for generating synthetic iris images and demonstrate its potential for presentation attack detection (PAD). The proposed technique utilizes the generative capability of a Relativistic Average Standard Generative Adversarial Network (RaSGAN) to synthesize high quality images of the iris. Unlike traditional GANs, RaSGAN enhances the generative power of the network by introducing a "relativistic" discriminator (and generator), which aims to maximize the probability that the real input data is more realistic than the synthetic data (and vice-versa, respectively). The resultant generated images are observed to be very similar to real iris images. Furthermore, we demonstrate the viability of using these synthetic images to train a PAD system that can generalize well to "unseen" attacks, i.e., the PAD system is able to detect attacks that were not used during the training phase.

1. Introduction

The iris is the annular region of the eye surrounding the pupil. The rich texture of the iris, which is better discernible in the near-infrared spectrum, has been used as a biometric cue [6] in many recognition systems [14]. This has led to an increased interest in the texture and morphology of the iris. Consequently, researchers have strived to model the pattern of the iris. In this regard, a number of methods to generate synthetic digital irides have been developed. Cui et al. [4] used principal component analysis to select appropriate feature vector coefficients from real images, which were then used to generate synthetic irides. The quality of the generated data was improved using super-resolution. Zuo et al. [35] developed a model based on the morphology of the iris. Noise and light reflection were also added to the model to create more realistic looking samples. Shah and Ross [26] used Markov Random Field to model the stromal texture of the iris [22] and then added anatomical entities such as collarette, crypts, radial and concentric furrows. In [31], Venugopalan and Savvides generated synthetic iris images from IrisCodes. Other methods have also been proposed in the literature to generate good quality synthetic iris images [32, 9]. While these methods successfully generate digital iris images, they are still unable to truly model the distribution of real iris images [18].

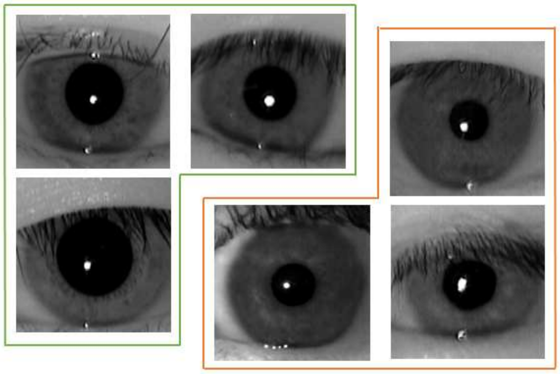


Figure 1: Samples of bonafide and RaSGAN-based synthetic iris images separated by green and red outline, respectively.

Recent advancements in deep learning techniques based on neural networks such as Convolutional Autoencoders (CAEs) [30, 28] and Generative Adversarial Networks (GANs) [11, 19] have paved the way for generating synthetic data, including iris, that look very realistic. Kohli et al. [18] proposed a deep learning based approach to synthesize iris images using a Deep Convolutional Generative Adversarial Network (iDCGAN). They used the inception score [25] and statistical characteristics of real iris images to define the realism of the generated data. Generated samples were shown to capture the texture distribution of a real iris image. However, unrealistic distortions and noise were observable near eyelids and lashes.

In this paper, we propose using Relativistic Average Standard Generative Adversarial Network (RaSGAN) [15] with Frechet Inception Distance (FID) [13] to synthesize good quality iris images. The traditional GANs such as Standard GAN (SGAN) [11] and iDCGAN consist of two

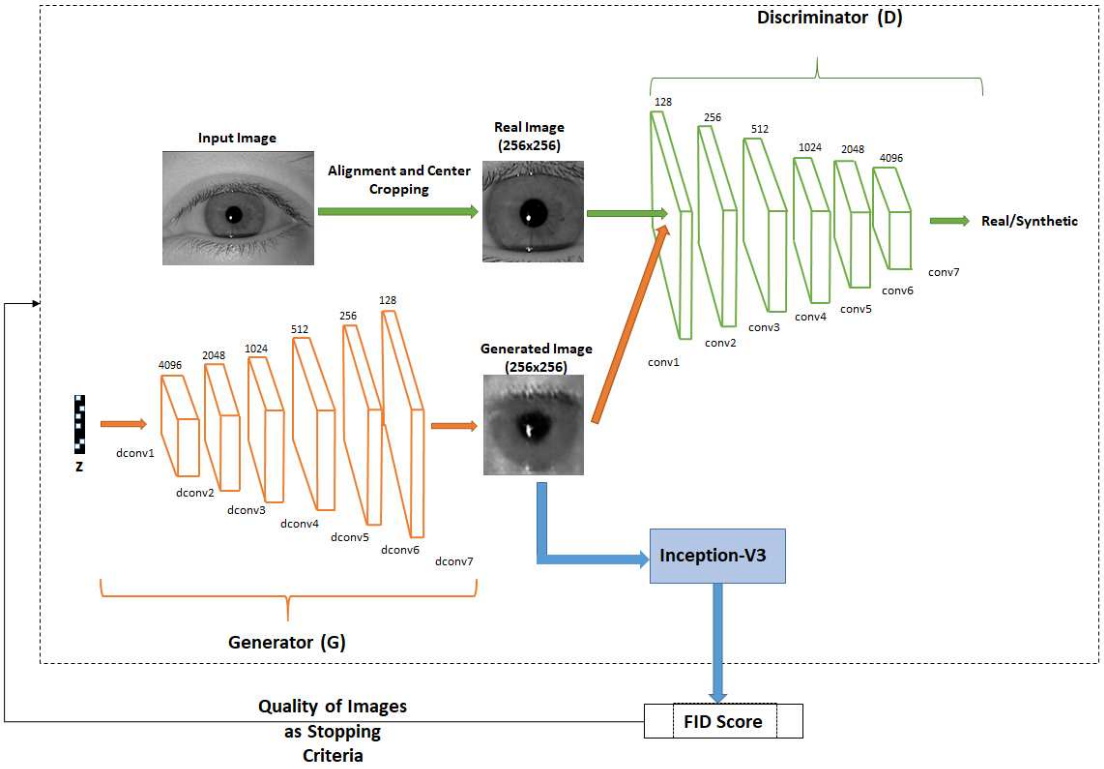


Figure 2: Schematic of the training process for the Relativistic Average Standard Generative Adversarial Network (RaSGAN) using real iris images. The training images for RaSGAN are first aligned and center-cropped using the pupil-iris center. Cropped images of size 256×256 are then sent to the discriminator for training. The discriminator tries to detect synthesized images while the generator competes with it to generate more realistic synthetic images by back-propagating the loss after each training iteration and updating the weights. For each generated image, a FID score is calculated to evaluate its quality. This process is repeated until images with lower (i.e., better) FID scores are generated.

critical modules - a *generator* that aims to increase the probability that the synthetic data is classified as real and a *discriminator* that aims to distinguish between real and synthetic data. Unlike iDCGAN, RaSGAN trains a generator that aims to maximize the probability that a randomly sampled set of synthetic samples are more realistic than a given set of real samples. In [15], Martineau showed that this property can be implemented in a Standard GAN using a "relativistic discriminator" that competes with the generator to maximize the probability that the real data is more realistic than the synthetic data. The author studied different cost functions and compared the quality of the generated samples using the FID score. He reported that RaSGAN obtained much lower (better) FID score on the CIFAR-10 dataset than SGAN, Least Squares GAN (LSGAN) [23] and Wasserstien GAN (WGAN) [1]. It was also observed that RaSGAN produces good quality images using fewer number of iterations even when other networks were not able to converge (especially for high resolution images).

In addition to synthesizing iris images using RaSGAN, we propose to utilize the synthetic data to improve the generalization capability of existing presentation attack detection (PAD) algorithms. Over the years, researchers have studied different kinds of presentation attacks (i.e., PAs) on iris recognition systems (such as the use of cosmetic contact lenses [33, 16] and printed eye images [12]) and have proposed methods to alleviate this vulnerability [17, 3, 5]. However, current PAD algorithms are not well designed to handle the problem of *unseen* attacks, i.e., using PAs that were not observed during the training stage of the algorithm. We demonstrate how synthetic iris images can be used to address this lapse.

The major contributions of this paper are summarized here:

- We use RaSGAN with FID score to generate synthetic iris images that can effectively model the distribution of real iris images.

- We investigate if state-of-the-art iris PAD algorithms can distinguish bonafide iris images as well as presentation attack images (e.g. cosmetic contact lens, printed iris and artificial eye images) from the generated synthetic images.

- We demonstrate the usefulness of the generated synthetic images for unseen presentation attack detection.

The rest of the paper is organized as follows. Section 2 introduces GANs. Section 3 discusses the principle behind RaSGAN and describes the specific method and architecture used to generate synthetic iris images. Section 4 introduces the datasets used in this work. Section 6 describes the experiments that were conducted and the results that were obtained. Section 7 summarizes the findings of this work.

2. Generative Adversarial Networks

Generative Adversarial Networks (GANs) [11] are neural networks that consist of two different components: a generator (G) that learns how to synthesize the data (e.g., images), and a discriminator (D) that aims to discriminate between real data and the synthesized data. These two networks are alternatively updated against each other in a min-max game where the objective of the generator is to maximally fool the discriminator while the objective of the discriminator is to not be fooled.

2.1. Background

Given a distribution of real data $\mathbb{P}_{data}$ and a multivariate normal noise distribution $\mathbb{P}_{\boldsymbol{z}}$, the discriminator in SGAN aims to output a high probability value for real data and a low probability value for synthetic data. This is achieved by maximizing objective function F as,

$$\begin{aligned}\max_{D} F(D) = \mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}_{data}}[log(D(\boldsymbol{x}))] \\ +\mathbb{E}_{\boldsymbol{z}\sim\mathbb{P}_{\boldsymbol{z}}}[log(1-D(G(\boldsymbol{z})))],\end{aligned} \tag{1}$$

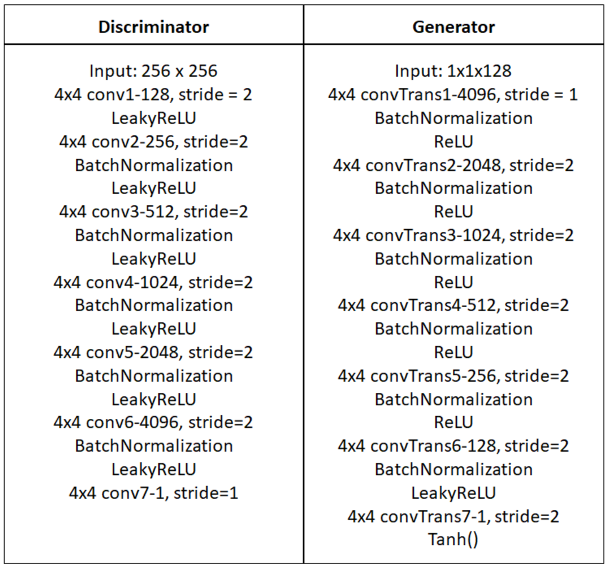
where, $\boldsymbol{x}$ represents the real data, $\boldsymbol{z}$ refers to a noisy input for G, $D(\boldsymbol{x}) = S(N(\boldsymbol{x}))$ and $N(\boldsymbol{x})$ is the non-transformed output of the discriminator. Here, S represents the standard logistic function that is used to classify sample $\boldsymbol{x}$ as real or synthetic. In SGAN, a negative value of $N(\boldsymbol{x})$ indicates that $\boldsymbol{x}$ is synthetic data while a positive value indicates that it is real data.

The generator, G, aims to generate images that maximize the value of $D(\boldsymbol{x})$ and this is achieved by minimizing F as,

$$\min_{G} F(G) = \mathbb{E}_{\boldsymbol{z}\sim\mathbb{P}_{\boldsymbol{z}}}[log(1-D(G(\boldsymbol{z})))]. \tag{2}$$

Therefore, in this min-max game between the discriminator and generator, the overall objective function of SGAN is defined as,

$$\begin{aligned}\min_{G}\max_{D} F(D,G) = \mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}_{data}}[log(D(\boldsymbol{x}))] \\ +\mathbb{E}_{\boldsymbol{z}\sim\mathbb{P}_{\boldsymbol{z}}}[log(1-D(G(\boldsymbol{z})))].\end{aligned} \tag{3}$$



Discriminator	Generator
Input: 256 x 256	Input: 1x1x128
4x4 conv1-128, stride = 2	4x4 convTrans1-4096, stride = 1
LeakyReLU	BatchNormalization
4x4 conv2-256, stride=2	ReLU
BatchNormalization	4x4 convTrans2-2048, stride=2
LeakyReLU	BatchNormalization
4x4 conv3-512, stride=2	ReLU
BatchNormalization	4x4 convTrans3-1024, stride=2
LeakyReLU	BatchNormalization
4x4 conv4-1024, stride=2	ReLU
BatchNormalization	4x4 convTrans4-512, stride=2
LeakyReLU	BatchNormalization
4x4 conv5-2048, stride=2	ReLU
BatchNormalization	4x4 convTrans5-256, stride=2
LeakyReLU	BatchNormalization
4x4 conv6-4096, stride=2	ReLU
BatchNormalization	4x4 convTrans6-128, stride=2
LeakyReLU	BatchNormalization
4x4 conv7-1, stride=1	LeakyReLU
	4x4 convTrans7-1, stride=2
	Tanh()

Figure 3: Details of the architecture of the RaSGAN used for generating synthetic iris images. In the discriminator, (4×4 conv1-128, stride=2) refers to a convolutional layer with kernel size 4×4, stride of 2 and 128 output filter maps. Similarly, in the generator, (4×4 convTrans1-4096, stride=1) describes a transposed convolutional layer with kernel size 4x4, stride of 1 and 4096 number of output filter maps.

3. Relativistic Average Standard Generative Adversarial Network (RaSGAN) for Synthetic Iris Generation

The discriminator in traditional GANs such as SGAN and DCGAN aims to maximize its capability to differentiate between real and synthesized samples. Such GANs have been shown to perform well in generating low-resolution images [11]. However, it is more challenging to synthesize high-resolution images due to instability in training and optimization [15]. This makes it difficult for the generator to produce high-resolution natural images while requiring the discriminator to accurately distinguish these generated samples from real samples. High-resolution images often contain more intricate details of the object being modeled than low-resolution images. Therefore, to obtain good quality synthetic digital irides that can capture the fine textural details of irides, a GAN model that can process information from high-resolution samples is desired.

3.1. Relativistic Standard Generative Adversarial Network (RSGAN)

In [15], Martineau determined that the min-max game of the SGAN does not always generate good quality data, especially for high-resolution inputs. Therefore, to enhance the generative power of GAN, he introduced the *relativis-*

tic discriminator, D_R, that can be represented using non-transformed layers $N(\boldsymbol{x_r})$ and $N(\boldsymbol{x_s})$ as,

$$D_R(\boldsymbol{x}) = S(N(\boldsymbol{x_r}) - N(\boldsymbol{x_s})). \tag{4}$$

Here, $\boldsymbol{x_r}$ and $\boldsymbol{x_s}$ represent real and synthetic data, respectively. Also, $N(.)$ represents the output of the last convolutional layer before the standard logistic function, S, is applied to it. The relativistic discriminator, D_R, maximizes the probability that the given real iris image is more realistic than the synthetic data. Similarly, $D_R^{rev} = S(N(\boldsymbol{x_s}) - N(\boldsymbol{x_r}))$ maximizes the probability that the given synthetic iris image is more realistic than the real data itself. Using this information, the loss function for the discriminator D_R and generator G_R can be updated as,

$$F^{RSGAN}(D_R) = -\mathbb{E}_{(\boldsymbol{x_r},\boldsymbol{x_s})\sim(\mathbb{P}_{data},\mathbb{Q})}[log(D_R(\boldsymbol{x}))], \tag{5}$$

$$F^{RSGAN}(G_R) = -\mathbb{E}_{(\boldsymbol{x_r},\boldsymbol{x_s})\sim(\mathbb{P}_{data},\mathbb{Q})}[log(D_R^{rev}(\boldsymbol{x}))], \tag{6}$$

where, $\mathbb{Q}$ represents the distribution of synthetic data $\boldsymbol{x_s}$. This ensures that unlike SGAN, gradients of D_R come from both real and synthetic data. This helps G_R to generate more realistic looking iris images.

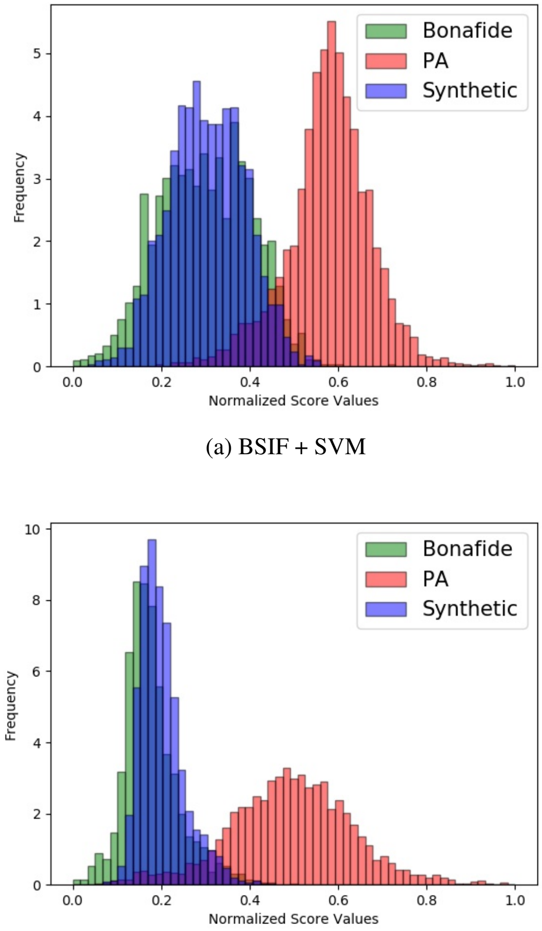
3.2. Relativistic Average Standard Generative Adversarial Network (RaSGAN)

As discussed in the previous section, the objective function of RSGAN compares a sample from the set of real images with some samples in $\mathbb{Q}$. This might not be a very effective approach as the loss functions for both D_R and G_R depend directly on $\boldsymbol{x_r}$ and $\boldsymbol{x_s}$. Therefore, to make the relativistic discriminator and generator "more global" over the dataset, the discriminator's loss function in Equation (5) is further updated to compare the input data (real/synthetic) with the average of samples from the opposite class (synthetic/real) [15]:

$$\begin{aligned} F^{RaSGAN}(D) = -\mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}_{data}}[log(\bar{D}(\boldsymbol{x_r}))] - \\ \mathbb{E}_{\boldsymbol{x_s}\sim\mathbb{Q}}[log(1-\bar{D}(\boldsymbol{x_s}))], \end{aligned} \tag{7}$$

$$\bar{D} = \begin{cases} S(N(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{x_s}\sim\mathbb{Q}}N(\boldsymbol{x_s})), \text{ if } \boldsymbol{x}\in\mathbb{P} \\ S(N(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{x_r}\sim\mathbb{P}}N(\boldsymbol{x_r})), \text{ if } \boldsymbol{x}\in\mathbb{Q}. \end{cases} \tag{8}$$

Using the updated loss functions, the relativistic *average* discriminator, $\bar{D}$, and generator compete with each other to learn a realistic representation for iris images. To maintain the quality of the generated images, the Frechet Inception Distance (FID) [2] score is calculated at each training epoch. FID compares the distribution of the generated synthetic iris images with the real samples to compute a score that helps determine the quality of RaSGAN-based synthetic iris images (see Section 5).



(a) BSIF + SVM

(b) Fine-tuned VGG-16

Figure 4: Normalized PA score distribution of RaSGAN-based synthetic iris images for Experiment-1 when tested on two PAD algorithms: (a) BSIF+SVM [7] and (b) Fine-tuned VGG-16 [10]. These histograms emphasize the similarity between bonafide samples and the generated dataset.

3.3. Model Architecture and Implementation Details

The RaSGAN used to generate synthetic iris images in this work is trained on 2,778 bonafide samples from a publicly available dataset named Berc-iris-fake [20, 21].

The input to the discriminator are real iris images aligned using the center of the pupil-iris obtained using the VeriEye iris-segmenter,[1] and center-cropped to size 256×256 (as shown in Figure 1). The RaSGAN model has been implemented in Python using PyTorch libraries[2] where both the

[1] www.neurotechnology.com/verieye.html

[2] https://github.com/alexiajm/relativisticgan

Table 1: Performance (in %) of PAD algorithms in Experiment-0 that is used as baseline for analysis and comparison with other experiments.

	BSIF	**VGG-16**	**DESIST**	**Iris-TLPAD**
EER	5.44	5.85	19.37	2.78
TDR(@1%)	71.70	86.20	21.62	96.31
TDR(@5%)	93.79	93.17	48.80	97.15

generator and discriminator are built using convolutional neural networks. The generator consists of seven transposed convolutional layers with kernel size of 4×4, and a stride of 1 for the first convolutional layer and 2 for the remaining six layers. Each convolutional layer is followed by batch normalization and rectified linear units. The discriminator is built using seven convolutional layers with kernel size 4×4 and a stride of 2 for all layers except for the last convolution layer. Similar to the generator, each convolutional layer is accompanied by batch normalization and leaky rectified linear units (see Figure 3)

4. Datasets Used

In this paper, we utilized image samples from multiple iris datasets, viz., Berc-iris-fake [20, 21], Casia-iris-fake [29], LivDet2015 [34], NDCLD15 [7] and a self collected dataset named MSU-IrisPA-01, for training and testing under different experimental set-ups. Images in MSU-IrisPA-01 were collected using the IrisID 7000 scanner over multiple sessions. This dataset contains 1,343 bonafide samples, 1,938 printed iris images, 108 colored contact lens images, 352 artificial eyes and 125 Kindle replay-attack images. To the best of our knowledge, this is the first dataset that contains Kindle replay attacks. Replay attacks have been a topic of discussion in the biometric literature [24], but were less commonly used in the context of NIR based iris sensors. However, the E-ink display of some Kindle models reflects NIR illumination that can be imaged by iris sensors. The artificial eye images exhibit ten different colors pertaining to three different brands. Similarly, five different-colored contact lenses were used to collect images of cosmetic contact lenses. Printed iris images were acquired using six different laser printers on four kinds of paper (office glossy, office matt, professional glossy and professional matt). The print attacks are further categorized into two different types: printed iris and printed iris with pupil cut-out. Thus, this dataset exhibits diverse types of presentation attacks.

As mentioned earlier, RaSGAN is trained using 2,778 bonafide samples from the Berc-iris-fake dataset, while the generation capability of the trained network is tested using 6,277 bonafide samples from the Casia-iris-fake, LivDet2015, NDCLD15 and MSU-IrisPA-01 datasets. This generates 6,277 synthetic iris samples that are utilized in different experiments.

Table 2: Performance (in %) of PAD algorithms in Experiment-1 (top) and Experiment-2 (bottom) when RaSGAN-based synthetic iris images are used as bonafide samples.

	BSIF	**VGG-16**	**DESIST**	**Iris-TLPAD**
EER	7.42	6.74	16.07	3.19
TDR(@1%)	81.01	82.89	28.98	95.65
TDR(@5%)	90.45	92.33	56.25	97.25

	BSIF	**VGG-16**	**DESIST**	**Iris-TLPAD**
EER	10.38	14.11	18.53	23.85
TDR(@1%)	60.36	51.69	43.05	53.82
TDR(@5%)	84.27	68.36	54.50	62.93

5. Evaluating Image Quality: Frechet Inception Distance (FID)

In [25], Salimans et al. proposed to use a pre-trained inception-V3 network to generate images and then compare the marginal label distribution with the conditional label distribution to generate the inception score. With respect to large KL-Divergence between the distributions, higher the inception score, the better the quality of the generated data. The inception score provides a good metric for evaluating image quality but it does not include statistics that compare real data against synthetic data.

Instead of analyzing synthetic iris images in isolation, the Frechet Inception Distance [13] compares the statistics of the generated synthetic samples against the real samples:

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

where, $\boldsymbol{\mu_r}, \boldsymbol{\mu_s}, \boldsymbol{\Sigma_r}$ and $\boldsymbol{\Sigma_s}$ represent the statistics of the two distributions and Tr is the trace of the co-variance matrix $(\boldsymbol{\Sigma_r} + \boldsymbol{\Sigma_s} - 2\sqrt{\boldsymbol{\Sigma_r \Sigma_s}})$.

Since FID is measured as the distance between the distributions of real and generated data, the lower the FID score, the higher the similarity between real and generated data. As described in [27], this score can be as high as 400-600 (or even more with respect to the deviation of generated data from the original distribution), but a score this high would indicate that the quality of the generated dataset is unacceptable. We evaluated the quality of synthetic iris samples generated using the trained G_R component of RaSGAN (Equation (9)) and obtained an overall score of 39.17 that is comparable to FID scores obtained in [8]. **Hence, we conclude that the RaSGAN based synthetically generated iris samples closely resemble bonafide iris samples.**

6. Analysis of RaSGAN-based Iris Images

The generated images are analyzed and evaluated for their usefulness as both bonafide images and presentation attack images, using state-of-the-art PAD algorithms, viz., DESIST [17], BSIF+SVM [7], Iris-TLPAD [3] and pretrained VGG-16 [10]. Seven different experiments are conducted with 6,277 RaSGAN-based synthetically generated irides, 6,277 bonafide irides and 9,467 PA samples from Casia-iris-fake, NDCLD15, LivDet2015 and MSU-IrisPA-01 datasets. Further division of these datasets for training and testing the PAD algorithms is explained in the experimental protocols described below. In all cases, training and test sets were mutually disjoint.

6.1. Baseline on Current PAD Algorithms

Experiment-0: This experiment is used as a baseline to evaluate the performance of state-of-the-art PAD methods on traditional PAs such as cosmetic contact lenses, printed eyes, Kindle replay-attack and artificial eyes. The PAD algorithms are trained using 4,312 bonafide samples and 5,538 PA samples; the test set consists of 1,965 bonafide samples and 3,929 PA samples.

Results for this experiment are summarized in Table 1, where Iris-TLPAD achieves the best performance with an Equal Error Rate (EER) as low as 2.78% followed by BSIF and VGG-16 with 5.44% and 5.85%, respectively.

6.2. Synthetic Iris as Bonafide Sample

FID scores do not merely provide an estimate of the quality of the images, but also about the similarity between the distributions of the synthetic data and the real data. To further establish the "bonafide nature" of the generated synthetic images, we conducted two more experiments.

6.2.1 Experimental Protocol

The experiments below use the generated synthetic iris images as bonafide samples.

- **Experiment-1**: The PAD algorithms are trained using 4,312 bonafide and 5,538 PA samples including printed eye, cosmetic contact lens, artificial eye and Kindle images. The test set was created using 1,965 bonafide samples, 3,929 PA samples and 1,965 synthetically generated images (labeled as bonafide samples).
- **Experiment-2**: This experiment focuses on evaluating the capability of the generated synthetic data to replace the need for bonafide samples. Thus, the PAD algorithms are trained using 4,312 RaSGAN-based synthetic iris images and 5,538 PA samples from Experiment-1. Testing is done on 1,965 bonafide irides and 3,929 PA samples.

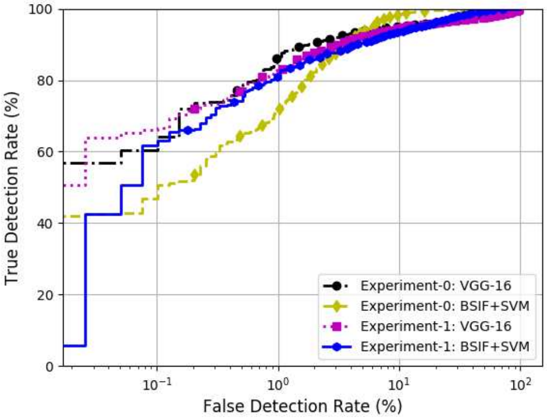


Figure 5: ROC curves illustrating the performance of PAD algorithms BSIF+SVM [7] and VGG-16 [10] when trained with bonafide images (and PAs) and tested using synthetic samples (Experiment-1), and comparing them with the corresponding baselines (Experiment-0).

6.2.2 Analysis

From Table 1, we observe that even in the presence of synthetic data (labeled as bonafide) during testing, the performance of PAD algorithms in Experiment-0 and Experiment-1 are comparable. There is an increase of only 1.98% in the EER of BSIF for Experiment-1. Congruent behavior is observed for other PAD algorithms implying that majority of RaSGAN based synthetic iris images are being classified as bonafide samples (see Figure 4 and 5).

However, when PAD algorithms are trained using synthetic iris images (instead of bonafide images) in Experiment-2, an increase in EER is observed (see Table 2). But some of the PAD algorithms still achieve a competitive True Detection Rate (TDR) of 84.27% at 5% False Detection Rate (FDR). This signifies that even though the generated iris images closely resemble bonafide samples, there are some fundamental differences between the two sets of images. This suggests the possibility of exploiting the synthetic images in a different way to enhance PAD algorithms, as will be shown later.

6.3. Synthetic Iris as Presentation Attack Sample

The synthetically generated dataset can be exploited by an adversary to impersonate someone else's identity. In the next two experiments, we study the impact of the synthetic data on PAD algorithms when used as a presentation attack.

6.3.1 Experimental Protocol

The experiments below use the generated synthetic iris images as presentation attack samples.

Table 3: Performance (in %) of PAD algorithms in Experiment-3 (top) and Experiment-4 (bottom) when RaSGAN-based synthetic iris images are used as presentation attack images.

	BSIF	VGG-16	DESIST	Iris-TLPAD
EER	16.03	10.25	10.37	2.47
TDR(@1%)	52.06	75.77	83.87	95.11
TDR(@5%)	74.96	85.14	86.06	95.17

	BSIF	VGG-16	DESIST	Iris-TLPAD
EER	50.64	30.94	57.45	25.14
TDR(@1%)	0.51	3.21	0.35	1.37
TDR(@5%)	3.10	7.43	2.23	9.16

- **Experiment-3**: In this experiment, we analyzed the performance of the PAD algorithms when the synthetic iris data is used as a “known” presentation attack. So, PAD algorithms are trained using 4,312 bonafide and 4,312 synthetic samples while testing is done using 1,965 samples from each class. Unlike Experiment-1 and 2, here synthetic images are labeled as PA.

- **Experiment-4**: In this experiment, we analyzed the performance of the PAD methods when the generated iris data are used as “unseen” presentation attacks. Here, the PAD algorithms are trained using 4,312 bonafide and 4,312 PA samples while testing is done using 1,965 bonafide and 1,965 synthetic samples.

6.3.2 Analysis

Comparing the results of Experiment-0 and Experiment-3, we observe a considerable increase in EER when RaSGAN-based synthetic iris images are used as PAs (except for Iris-TLPAD). A decrease in TDR is observed for all PAD algorithms (except for DESIST) that confirms the viability of using RaSGAN generated synthetic images as presentation attack vectors on current state-of-the-art methods. Also, when RaSGAN based synthetic data is used only in the test set as an unseen attack (Experiment-4), a very significant drop in the performance of PAD algorithms is observed. For example, in Table 3 (bottom), EER values for BSIF and DESIST are more than 50% with TDR at an FDR of 5% as low as 3.10% and 2.23%, respectively. Similar observation can be made for other PAD algorithms.

6.4. Synthetic Iris for Unseen Presentation Attack Detection

Experimental results from the previous section shows that RaSGAN-based synthetic images can be used as effective PAs as they closely resemble the bonafide samples. Thus, if existing PAD algorithms are trained using synthetic images as PAs, it is possible that these algorithms can learn a better representation of bonafide samples. In this section, we explore the idea of using synthetically generated data for enhancing the ability of PAD methods to detect *unseen* PAs, i.e., those PA types that were not used during training.

Table 4: Performance (in %) of PAD algorithms in Experiment-5 (top) and Experiment-6 (bottom), illustrating the benefits of training PAD methods using RaSGAN-based synthetic images for unseen PA detection.

	BSIF	VGG-16	DESIST	Iris-TLPAD
EER	28.57	31.37	42.01	25.18
TDR(@1%)	1.78	5.27	1.27	21.21
TDR(@5%)	13.84	27.44	5.29	38.65

	BSIF	VGG-16	DESIST	Iris-TLPAD
EER	22.79	26.99	39.54	18.52
TDR(@1%)	11.65	13.61	1.02	38.91
TDR(@5%)	48.14	30.93	5.44	56.59

6.4.1 Experimental Protocol

To examine the possibility of using synthetic iris images for unseen presentation attack detection, the type of presentation attacks on which the PAD algorithms are trained and tested are mutually exclusive of each other.

- **Experiment-5**: In this experiment, the PAD algorithms are trained on 4,312 bonafide iris images and 5,530 PA samples representing print-attack and artificial eyes, and testing is done on 1,965 bonafide iris images and 1,917 PA samples corresponding to cosmetic contact lens and Kindle images. This experiment evaluates the efficacy of current PAD algorithms on unseen presentation attacks.

- **Experiment-6**: In this experiment, unlike Experiment-5, training set consists of 4,312 bonafide iris images, the corresponding 4,312 synthetic images and 1,135 PA samples pertaining to print-attack and artificial eyes. The test set consists of 1,965 bonafide iris images and 1,917 PA samples pertaining to cosmetic contact lens and Kindle images. Here, we analyze the usefulness of synthetic iris images for unseen PA detection.

6.4.2 Analysis

As seen from the results in Table 4 and Figure 6, current state-of-the-art algorithms for presentation attack detection

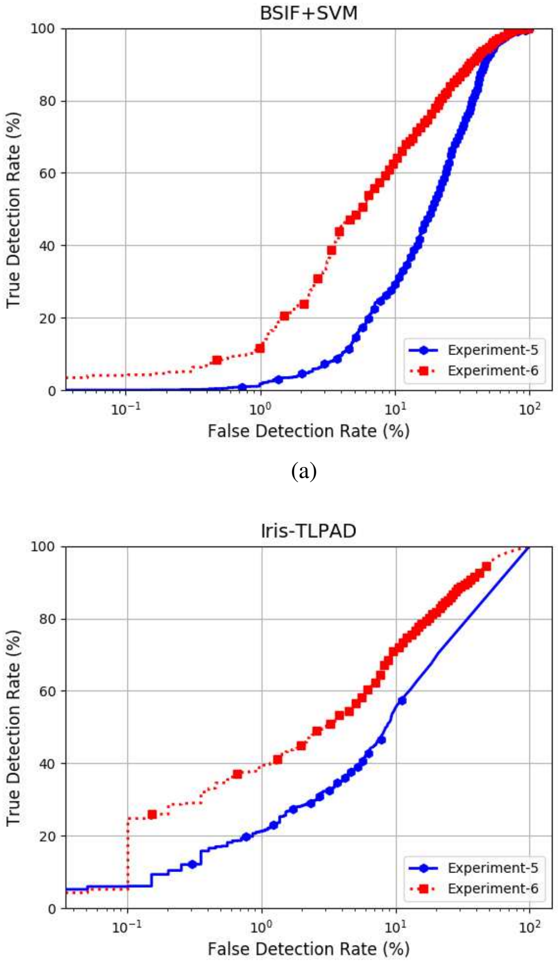


Figure 6: ROC curve illustrating the benefits of RaSGAN-based synthetic iris images for generalized unseen presentation attack detection.

are not well-equipped for handling unseen presentation attack with EER values in Experiment-5 as high as 28.57% and 25.18% for BSIF and Iris-TLPAD, respectively. On the other hand, EER reduces in Experiment-6 for most of the PAD algorithms when they are trained using synthetic images and a few PA samples (printed and artificial eyes). This indicates that RaSGAN-based synthetic iris images can be used by PAD algorithms to learn a better representation for bonafide iris images, which will help them detect unseen PA types.

7. Summary

In this work, we designed a new technique based on RaSGAN to generate synthetic irides. Results obtained in this paper suggest that there are multiple applications for synthetic iris images: (1) they can be used to imitate real iris images, which eliminates the hassle of large data collection, (2) they can efficiently model the bonafide samples (see Figure 4), making them potential presentation attack vectors, and (3) they can be used to train existing PAD algorithms for "unseen" presentation attack detection.

We plan to extend this work by judiciously using the RaSGAN-generated synthetic irides to train a new PAD algorithm that can better generalize over unseen attacks.

8. 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. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

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

[2] S. Barratt and R. Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018. 4

[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. 2, 6

[4] J. Cui, Y. Wang, J. Huang, T. Tan, and Z. Sun. An iris image synthesis method based on PCA and super-resolution. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, volume 4, pages 471–474, 2004. 1

[5] A. Czajka and K. W. Bowyer. Presentation attack detection for iris recognition: An assessment of the state-of-the-art. *ACM Comput. Surv.*, 51(4):86:1–86:35, 2018. 2

[6] J. Daugman. New methods in iris recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(5):1167–1175, 2007. 1

[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. 4, 5, 6

[8] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016. 5

[9] 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*, 117:1512–1525, 10 2013. 1

[10] 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. 4, 6

[11] 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. 1, 3

[12] 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. 2

[13] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter. GANs trained by a two time-scale update rule converge to a Nash equilibrium. *arXiv preprint arXiv:1706.08500*, 12(1), 2017. 1, 5

[14] A. K. Jain, K. Nandakumar, and A. Ross. 50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79:80–105, 2016. 1

[15] A. Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard GAN. *arXiv preprint arXiv:1807.00734*, 2018. 1, 2, 3, 4

[16] 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. 2

[17] N. Kohli, D. Yadav, M. Vatsa, R. Singh, and A. Noore. Detecting medley of iris spoofing attacks using desist. In *IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–6, 2016. 2, 6

[18] 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. 1

[19] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4681–4690, 2017. 1

[20] 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. 4, 5

[21] 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):127204, 2007. 4, 5

[22] S. Makthal and A. Ross. Synthesis of iris images using Markov Random Fields. In *IEEE 13th European Signal Processing Conference*, pages 1–4, 2005. 1

[23] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2794–2802, 2017. 2

[24] K. B. Raja, R. Raghavendra, and C. Busch. Video presentation attack detection in visible spectrum iris recognition using magnified phase information. *In IEEE Transactions on Information Forensics and Security (TIFS)*, 10(10):2048–2056, 2015. 5

[25] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2234–2242, 2016. 1, 5

[26] S. Shah and A. Ross. Generating synthetic irises by feature agglomeration. In *IEEE International Conference on Image Processing (ICIP)*, pages 317–320, 2006. 1

[27] J. Shelton, K. Roy, B. O'Connor, and G. V. Dozier. Mitigating iris-based replay attacks. *International Journal of Machine Learning and Computing (JMLC)*, 4(3):204, 2014. 5

[28] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2107–2116, 2017. 1

[29] 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 (TPAMI)*, 36(6):1120–1133, 2014. 5

[30] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al. Conditional image generation with CNN decoders. In *Advances in Neural Information Processing Systems (NIPS)*, pages 4790–4798, 2016. 1

[31] S. Venugopalan and M. Savvides. How to generate spoofed irises from an iris code template. *In IEEE Transactions on Information Forensics and Security (TIFS)*, 6(2):385–395, 2011. 1

[32] L. Wecker, F. Samavati, and M. Gavrilova. Iris synthesis: a reverse subdivision application. In *Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, pages 121–125. ACM, 2005. 1

[33] 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 (TIFS)*, 9:851–862, 2014. 2

[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. 5

[35] J. Zuo, N. A. Schmid, and X. Chen. On generation and analysis of synthetic iris images. *In IEEE Transactions on Information Forensics and Security (TIFS)*, 2(1):77–90, 2007. 1