Forward-Backward Generative Adversarial Networks for Anomaly Detection

Youngnam Kim YOUNGNK@MEDI-WHALE.COM
Medi Whale Incorporated, Seoul, Republic of Korea

Seungjin Choi SEUNGJIN.CHOI.MLG@GMAIL.COM
BARO, Korea

Abstract

Generative adversarial network (GAN) has established itself as a promising model for density estimation, with its wide applications to various problems. Of particular interest in this paper is the problem of *anomaly detection* which involves identifying events that do not conform to expected patterns in data. Recent application of GANs to the task of anomaly detection, resort to their ability for learning probability distributions of normal examples, so that abnormal examples or outliers are detected when they reside in very low-probability regimes. Existing GAN methods often suffer from the bad *cycle-consistency* problem, which yields the large reconstruction error so that the anomaly detection performance is degraded. In order to alleviate this, we present a model that consists of a forward GAN and backward GAN, each of which has an individual discriminator, that are coupled by enforcing feature matching in two discriminators. We show that our forward-backward GANs (FBGANs) better captures the data distribution so that the anomaly detection performance is improved over existing GAN-based methods. Experiments on MNIST an KDD99 datasets demonstrate that our method, FBGANs, outperforms existing state-of-the-art anomaly detection methods, in terms of the area under precision recall curve (AUPR) and F_1-score.

Keywords: Generative Adversarial Networks, Anomaly Detection

1. Introduction

Anomaly detection is a problem of finding outliers that are largely different from inlier samples. In perspective of density estimation, samples that have significantly low likelihood can be regarded as outliers. Anomaly detection have many real-world applications such as cybersecurity Tan et al. (2011), medical diagnosis Salem et al. (2013); Schlegl et al. (2017) and surveillance video Sultani et al. (2018).

Traditional methods for anomaly detection are One-Class Support Vector Machine (OC-SVM) Schölkopf et al. (2001) and Kernel Density Estimation (KDE) Parzen (1962) which have difficulties in high-dimensional data. Recent methods Zhai et al. (2016); Zong et al. (2018) have been proposed based on deep neural networks LeCun et al. (2015) that have a strong representational power on high-dimensional data. In particular, some approaches based on Generative Adversarial Networks (GANs) Goodfellow et al. (2014) shows promising anomaly detection performance on some dataset Schlegl et al. (2017); Zenati et al. (2018). Especially, Bidiretional GANs (BiGAN) Donahue et al. (2017); Dumoulin et al. (2017) shows state-of-the-art results on some dataset in anomaly detection Zenati et al. (2018).

BiGAN has encoder and decoder which can generate points in data and latent space bidirectionally through adversarial training with a discriminator on inlier dataset. So they can reconstruct samples like autoencoder. To identify anomalousness of a sample, BiGAN uses reconstruction error as anomaly score. The optimal behaviour of BiGAN is to reproduce inlier samples itself faithfully, while forcing outliers to be reproduced in inlier distributiuon, which means giving poor reconstructions on outliers. Although BiGAN is good at the latter, they also show unsatisfactory reproduction qualities on inlier samples, which gives high reconstruction errors on lnliers, which can degrade anomaly detection performance significantly Donahue et al. (2017); Dumoulin et al. (2017). This problem that a model cannot reproduce sample itself faithfully and give large reconstruction error is called bad *cycle consistency* Zhu et al. (2017).

In this paper, we proposed an alternative model called Forward-Backward GAN (FBGAN) to mitigate bad cycle consistency of BiGAN. First, we introduce some background GAN-based anomaly detection methods (Section 2). And then, we describe bad cycle consistency in BiGAN (Section 3.1). try to figure out what causes bad cycle consistency in BiGAN (Section 3.2) and demonstrate that this can arise from ambiguity in discriminator's objective (Section 5.5). To remove the ambiguity, we use separate training signals for marginal distribution matching and coupling sample pairs between latent and data space respectively (Section 3). Consequently, FBGANs show significant improvements on anomaly detection performance compared to the previous methods (Section 5). FBGANs achieved state-of-the-art area under precision recall curve (AUPR) and F_1-score on MNIST and KDD99 dataset respectively.

2. Background

2.1. Generative Adversarial Networks

Generative adversarial networks Goodfellow et al. (2014) are a kind of implicit models to approximate a true data distribution by generating samples as if these samples are drawn from the distribution. GAN generally consists of two parts: a generator and a discriminator. Let $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{N}$ be a dataset where $\mathbf{x} \in \mathbb{R}^d$. The generator G synthesizes samples $G(\mathbf{z}) \in \mathbb{R}^d$ where $\mathbf{z} \in \mathbb{R}^m$ is a random variable of some prior distribution. Popular choices are an isotropic gaussian $\mathcal{N}(\mathbf{0}, \mathbf{I})$ or an uniform distribution $\mathcal{U}(0,1)$. The discriminator D learns to distinguish real samples from fakes, while the generator learns to generate fakes undistinguishable from reals. In this way, the discriminator and generator are trained adversarially to approximate the true data distribution. The original GAN formulation is defined as the following minimax game.

$$\min_G \max_D \mathbb{E}\left[\log D(\mathbf{x}) + \log\left(1 - D(G(\mathbf{z}))\right)\right] \tag{1}$$

2.2. AnoGAN

Anomaly detection GAN (AnoGAN) Schlegl et al. (2017) tried to exploit GAN's ability to capture a data distribution. Let's assume that GAN's training is completed over a dataset consisting of only inlier samples, then the generator will be more likely to generate samples close to the inliers than outliers. In other words, if a sample $\mathbf{x}$ is an inlier, then a point $\mathbf{z}$

is likely to exist, from which a generated sample $G(\mathbf{z})$ is almost identical to the original $\mathbf{x}$: $G(\mathbf{z}) \approx \mathbf{x}$. In contrast, if a sample $\mathbf{x}$ is outlier, such point $\mathbf{z}$ is less likely to exist. AnoGAN used this difference between inlier and outlier samples for anomaly detection. To identify anomalousness of sample $\mathbf{x}$, an arbitrary point $\mathbf{z}_0$ is drawn from the latent prior $p(\mathbf{z})$ and minimization of residual error $\|\mathbf{x}-G(\mathbf{z}_k)\|_2^2$ with respect to $\mathbf{z}_k$ is performed iteratively using backpropagation as follows.

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \eta \nabla_{\mathbf{z}_k} \|\mathbf{x} - G(\mathbf{z}_k)\|_2^2 \tag{2}$$

where $\mathbf{z}_k$ is k-th update of $\mathbf{z}$ and η is a learning rate. Updates go on up to predefined terminal step T and an anomaly score function $\mathcal{A}(\mathbf{x})$ is defined as follows.

$$\mathcal{A}(\mathbf{x}) = (1-\lambda)S_R(\mathbf{x}) + \lambda S_D(\mathbf{x}) \tag{3}$$

where $S_R(\mathbf{x})$ is a residual score, $S_D(\mathbf{x})$ is a discrimination score and λ is an score weight hyperparameter that adjusts the proportion of each score; $0 \leq \lambda \leq 1$. A residual score $S_R(\mathbf{x})$ is defined as the Euclidean distance between a sample $\mathbf{x}$ and a generated sample $G(\mathbf{z}_T)$: $\|\mathbf{x} - G(\mathbf{z}_T)\|_2$. A discrimination score $S_D(\mathbf{x})$ can be defined in 2 ways: (i) a negative log probability that $\mathbf{x}$ is a real sample: $-\log(D(\mathbf{x}))$, (ii) a discriminative feature matching error: $\|D^h(\mathbf{x}) - D^h(G(\mathbf{z}_T))\|_2$ where D^h is the last intermediate layer of the discriminator. These discrimination scores are beneficial to detect anomalous samples in practice Schlegl et al. (2017); Zenati et al. (2018).

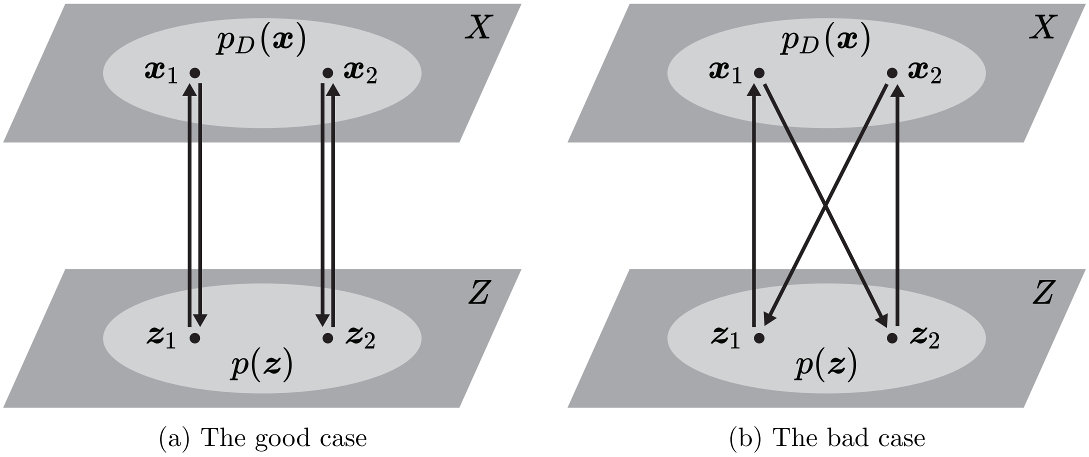
2.3. Efficient GAN-based Anomaly Detection

Although AnoGAN is a competitive method, cumbersome backpropagation steps like (2) are required to evaluate each sample's anomalousness. Efficient GAN-based Anomaly Detection (EGBAD) Zenati et al. (2018) solved this problem by introducing Bidirectional GAN (BiGAN) Donahue et al. (2017); Dumoulin et al. (2017) for anomaly detection.

In general, GANs learn a mapping from a latent space to a data space. In addition to learning this forward mapping, BiGAN also learns the mapping from the data space to the latent space bidirectionally through adversarial training. BiGAN consists of 3 parts: a discriminator D, a generator G, and an encoder E. The discriminator's input is a pair of a sample $\mathbf{x}$ in a data space and a point $\mathbf{z}$ in a latent space: $(\mathbf{x}, \mathbf{z})$. The discriminator learns to tell a pair synthesized by the generator: $(G(\mathbf{z}), \mathbf{z})$ from a pair synthesized by the encoder: $(\mathbf{x}, E(\mathbf{x}))$. The generator learns to make $(G(\mathbf{z}), \mathbf{z})$ undistinguishable from $(\mathbf{x}, E(\mathbf{x}))$. Likewise, the encoder learns to make $(\mathbf{x}, E(\mathbf{x}))$ undistinguishable form $(G(\mathbf{z}), \mathbf{z})$. This minimax game is defined as follows.

$$\min_{G,E} \max_{D} \mathbb{E}\big[\log D(\mathbf{x}, E(\mathbf{x})) + \log(1 - D(G(\mathbf{z}), \mathbf{z}))\big] \tag{4}$$

If the encoder and generator are deterministic, when BiGAN's objective (4) have reached its optimal point, the encoder is equal to the inverse mapping of the generator: $E \approx G^{-1}$, and vice versa: $G \approx E^{-1}$ Donahue et al. (2017); Dumoulin et al. (2017). For this reason, BiGAN can be used as like autoencoders and can be applied directly to evaluate anomalousness of samples. The anomaly score function based on BiGAN is defined as the same form as (3), but a residual score $S_R(\mathbf{x})$ and a discrimination score $S_D(\mathbf{x})$ is



(a) The good case

(b) The bad case

Figure 1: Conceptual images of cycle consistency. In each image, X is a data space and Z is a latent space. The circle area in the data space is an inlier data distribution $p_D(\mathbf{x})$, and the circle area in the latent space is a predefined latent distribution $p(\mathbf{z})$ (a) The model with good cycle consistency can reproduce original samples consistently (b) On the contrary, in the model with bad cycle consistency, a point $\mathbf{x}_1$ maps to $\mathbf{z}_2$ and reproduce $\mathbf{x}_2$, not the original point $\mathbf{x}_1$.

slightly different from AnoGAN. Let $\hat{\mathbf{x}}$ be a reconstructed sample of $\mathbf{x}$: $\hat{\mathbf{x}} = G(E(\mathbf{x}))$. The residual score $S_R(\mathbf{x})$ is defined as $\|\mathbf{x} - \hat{\mathbf{x}}\|_2$. The discrimination score $S_D(\mathbf{x})$ can be defined as: (i) a negative log probability over a pair from a encoder: $-\log D(\mathbf{x}, E(\mathbf{x}))$, (ii) a discriminative feature matching error between a original sample and a reconstructed sample: $\|D^h(\mathbf{x}, E(\mathbf{x})) - D^h(\hat{\mathbf{x}}, E(\mathbf{x}))\|_2$ where D^h is an intermediate layer of the discriminator in BiGAN. Consequently, an efficient evaluation can be achieved by using BiGAN, because the residual score $S_R(\mathbf{x})$ and the discrimination score $S_D(\mathbf{x})$ can be computed efficiently without costly backpropagation steps of (2) Zenati et al. (2018).

FBGANs also use an encoder for an efficient evaluation, but FBGANs is different from BiGAN in that FBGANs couples an encoder and a generator by discriminative feature matching, not adversarial training. The experimental results (Section 5.5) shows that using a discriminative feature matching loss for coupling the encoder and generator is helpful for a bad cycle consistency problem of BiGAN which will be described in the next section.

3. Proposed Model

3.1. Bad Cycle Consistency Ploblem of BiGAN

The BiGAN based anomaly detection method resort to a reconstruction ability, and the basic assumption in these methods is that the models should reproduce a sample itself faithfully from given inlier samples and force outlier samples to be reproduced within an inlier distribution.

Although BiGAN forces outlier samples to be reproduced within an inlier distribution, they both suffer from a poor reconstruction problem on inlier samples Donahue et al. (2017); Dumoulin et al. (2017). This problem makes it difficult for the GAN-based method to detect outlier samples precisely. We can describe this poor reconstruction problem by using a concept of cycle-consistency Zhu et al. (2017). Cycle-consistency is the very desirable property that all auto-encoding models must have. Cycle consistency allows a model to encode a sample and reproduce itself over and over consistently. For example, let $\mathbf{z}_1$ and $\mathbf{z}_2$ be latent points drawn from a predefined latent distribution $p(\mathbf{z})$, and $\mathbf{x}_1$ and $\mathbf{x}_2$ be data points from a true data distribution $p_D(\mathbf{x})$, where $\mathbf{z}_1 \neq \mathbf{z}_2$ and $\mathbf{x}_1 \neq \mathbf{x}_2$. In the model with cycle consistency, when $\mathbf{x}_1$ is fed into an encoder E and the encoder produces $\mathbf{z}_1$: $E(\mathbf{x}_1) = \mathbf{z}_1$, a generator produces $\mathbf{x}_1$ from $\mathbf{z}_1$: $G(\mathbf{z}_1) = \mathbf{x}_1$. Also, when $E(\mathbf{x}_2) = \mathbf{z}_2$, $G(\mathbf{z}_2) = \mathbf{x}_2$ (Fig. 1 (a)). In the model with bad cycle consistency, when $\mathbf{z}_1$ or $\mathbf{z}_2$ is fed into the generator, the generator can produce a sample from $p_D(\mathbf{x})$. Likewise, when $\mathbf{x}_1$ or $\mathbf{x}_2$ is fed into the encoder, the encoder can produce a sample from $p(\mathbf{z})$, but the generator and encoder have difficulties coupling the samples. For example, when the encoder produces $\mathbf{z}_1$ from $\mathbf{x}_1$, the generator produces $\mathbf{x}_2$ from $\mathbf{z}_1$: $E(\mathbf{x}_1) = \mathbf{z}_1$ and $G(\mathbf{z}_1) = \mathbf{x}_2$. In other words, the model cannot reproduce a sample itself consistently (Fig. 1 (b)).

This bad cycle consistency of BiGAN degrades anomaly detection performance, because the poor reconstruction ability increases anomaly scores of inlier samples.

3.2. On Training Signals from Discriminator in BiGAN

Theoretically, in BiGAN, bad cycle consistency cannot occur at the optimal point over (4), but in practice BiGAN fails to reach the optima and shows a poor reconstruction ability. Before we investigate what causes the cycle consistency problem in BiGAN, it is essential to know how the discriminator works for the adversarial training. As aforementioned, a discriminator of BiGAN learns to predict where a pair $(\mathbf{x}, \mathbf{z})$ comes from: an encoder or a generator. To achieve this objective the discriminator should take care of the following three missions: (i) the discriminator should be able to judge whether $\mathbf{x}$ is real or fake, (ii) the discriminator should be able to judge whether $\mathbf{z}$ is real or fake, (iii) the discriminator should consider the relation between $\mathbf{x}$ and $\mathbf{z}$. Note that (i) and (ii) are important for the model to generate realistic points in both data and latent spaces, and (iii) is important for the cycle consistency. In BiGAN, the encoder and generator take training signals only through the discriminator. For these reasons, the role of the discriminator is critical to the entire training of BiGAN. In this respect, bad cycle consistency can be caused by the discriminator, because in practice the discriminator can achieve the original objective by concentrating just one of the 3 missions. More specifically, given a pair $(\mathbf{x}, \mathbf{z})$, the discriminator can decide the identity of the pair by inspecting only $\mathbf{x}$ without taking into consideration (ii) and (iii). Note that if the discriminator do not use the relation between $\mathbf{x}$ and $\mathbf{z}$ for its training goal, then the generator and encoder would not be coupled properly. This can cause bad cycle consistency which can be called a poor reconstruction problem. In section 5.5, our experiment shows unreliability of a learned discriminator in BiGAN empirically to support our assumption.

3.3. Forward & Backward GANs

We assumed that bad cycle consistency comes from training signals of the discriminator, thus we distribute the three missions of the BiGAN discriminators to each component of FBGANs to give explicit training signals for each misssion to the generator and encoder.

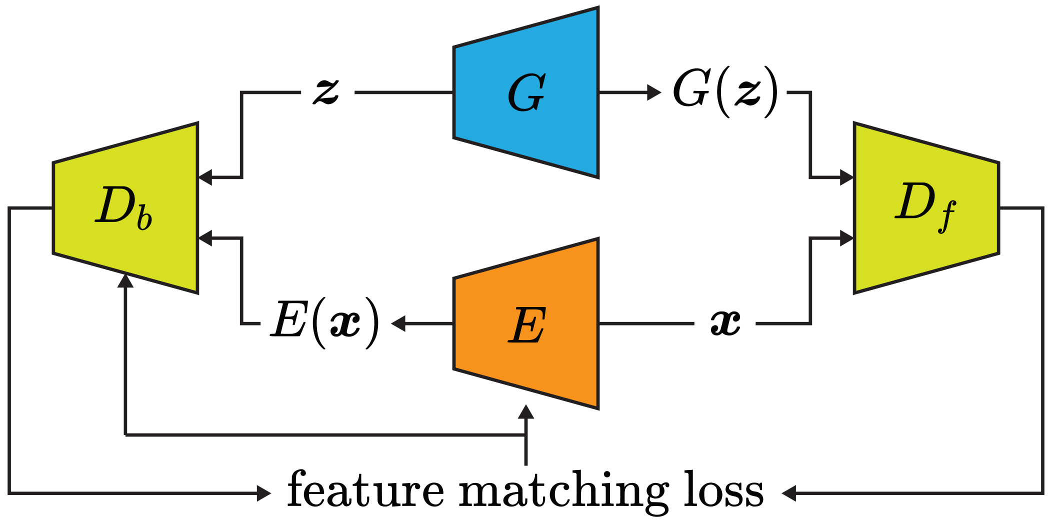


Figure 2: Forward-Backward GANs for Anomaly Detection. A forward GAN (G and D_f) learns to approximate a true data distribution $p_D(\mathbf{x})$. In contrast, A backward GAN (E and D_b) learns to approximate a predefined latent prior $p(\mathbf{z})$. FBGANs uses a feature matching loss between D_f and D_b to couple the forward and backward GAN.

FBGANs consists of forward and backward GANs. The forward GAN learns the mapping from a latent space to a data space, and the backward GAN learns the mapping from the data space to the latent space. The objective function of the forward GAN is exactly the same as that of standard GANs.

$$\mathcal{V}_f(D_f, G) = \mathbb{E}\left[\log D_f(\mathbf{x}) + \log(1 - D_f(G(\mathbf{z})))\right] \tag{5}$$

where D_f is a discriminator and G is a generator of the forward GAN. In contrast, the backward GAN learns to produce realistic latent points from real data points.

$$\mathcal{V}_b(D_b, E) = \mathbb{E}\left[\log D_b(\mathbf{z}) + \log(1 - D_b(E(\mathbf{x})))\right] \tag{6}$$

where $D_b(\mathbf{x})$ is a discriminator and E is a generator of the backward GAN. The foward and backward GANs force the generated samples into a true data distribution $p_D(\mathbf{x})$ and a latent prior $p(\mathbf{z})$ respectively. Consequently, not only inlier samples but also outlier samples are forced to be reproduced within a true data distribution $p_D(\mathbf{x})$. Note that FBGANs distributes mission (i, ii) of BiGAN's discriminator to a forward discriminator D_f and a backward discriminator D_b. This distiribution of missions replace embiguous training signals from BiGAN discriminator and can help the generator and encoder to capture the forward and backward mapping respectively. Note that the forward and backward GANs also need to be coupled. The coupling loss will be described in the next section.

3.4. Coupling Forward and Backward GANs

We do not use an adversarial loss for coupling the generator and encoder as in BiGAN, because the logistic sigmoid score from the discriminator is an ambiguous training signal for tight coupling and results in a bad cycle consistency problem (Section 5.5). Thus, FBGANs distribute mission (iii) of BiGAN discriminator to a discriminative feature matching loss for coupling the forward and backward GANs tightly. The idea is very simple. A latent point $\mathbf{z}$ and a generated sample $G(\mathbf{z})$ must have the same discrimination score both in forward and backward discriminators: $D_b(\mathbf{z}) = D_f(G(\mathbf{z}))$. Likewise, a data sample $\mathbf{x}$ and an encoded sample $E(\mathbf{x})$ must have the same discrimination score both in forward and backward discriminator: $D_b(E(\mathbf{x})) = D_f(\mathbf{x})$. However, the discrimination scores do not give enough information for coupling the forward and backward GANs properly. Thus, we define two discriminative feature matching losses as follows.

$$\mathcal{L}_{gen} = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \| D_f^h(G(\mathbf{z})) - D_b^h(\mathbf{z}) \|_2^2 \tag{7}$$

$$\mathcal{L}_{enc} = \mathbb{E}_{\mathbf{x}\sim p_D(\mathbf{x})} \| D_f^h(\mathbf{x}) - D_b^h(E(\mathbf{x})) \|_2^2 \tag{8}$$

where D_f^h is the last hidden layer of the discriminator of the forward GAN, and D_b^h is that of the backward GAN. D_f^h and D_b^h should have the same dimensionality. $\mathcal{L}_{gen}$ is an expected Euclidean distance between discriminative features of a latent point $\mathbf{z}$ and a generated sample $G(\mathbf{z})$. $\mathcal{L}_{enc}$ is that of a data point $\mathbf{x}$ and an encoded sample $E(\mathbf{x})$. For latent samples, the backward discriminator D_b learns to match itself with the forward discriminator D_f by minimizing $\mathcal{L}_{gen}$. For the pairs from the encoder, the encoder learns a reverse mapping of the generator by minimizing $\mathcal{L}_{enc}$. Total optimization of FBGANs is as follows.

$$D_f^* = \arg\max_{D_f} \mathcal{V}_f(D_f, G) \tag{9}$$

$$G^* = \arg\min_{G} \mathcal{V}_f(D_f, G) \tag{10}$$

$$D_b^* = \arg\min_{D_b} (-\mathcal{V}_b(D_b, E) + w_1 \mathcal{L}_{gen}) \tag{11}$$

$$E^* = \arg\min_{E} (\mathcal{V}_b(D_b, E) + w_2 \mathcal{L}_{enc}) \tag{12}$$

where w_1 and w_2 are feature matching weight hyperparameters. When the feature matching losses $\mathcal{L}_{gen}$ and $\mathcal{L}_{enc}$ reach a global minimum, the generator G can reproduce a sample itself from a latent point that are produced by the encoder E (Fig. 2). Using more explicit training signals than sigmoid output from BiGAN's discriminator results in mitigation of the bad cycle consistency problem significantly (Section 5.5).

4. Related Works

DAGMM Zong et al. (2018) proposed an end-to-end learning method for density estimation with Gaussian mixture model and autoencoder. DAGMM uses negative log likelihood as anomaly score function. DADGT Golan and El-Yaniv (2018) introduced self-supervised learning for anomaly detection. DADGT detects outliers depending on how well the model identify geometric transformations applied to a sample. Deep SVDD Ruff et al. (2018) learns to encode inlier samples into a latent space and to minimize volume of the hypersphere including encodings of those inliers. Deep SVDD uses the distance from center of the hypersphere to the encoding of a sample to detect outliers. AnoGAN Schlegl et al. (2017) shows that GANs can be one of the most promising way for anomaly detection. EGBAD Zenati et al. (2018) significantly improves AnoGAN by both detection ability and speed using BiGAN. In this paper, we focused on drawbacks in existing GAN-based methods and proposed novel way to mitigate these drawbacks.

5. Experiments

5.1. Datasets

To evaluate anomaly detection capability of FBGANs, we executed experiments on two benchmark datasets: MNIST and KDD99 10 percent dataset.

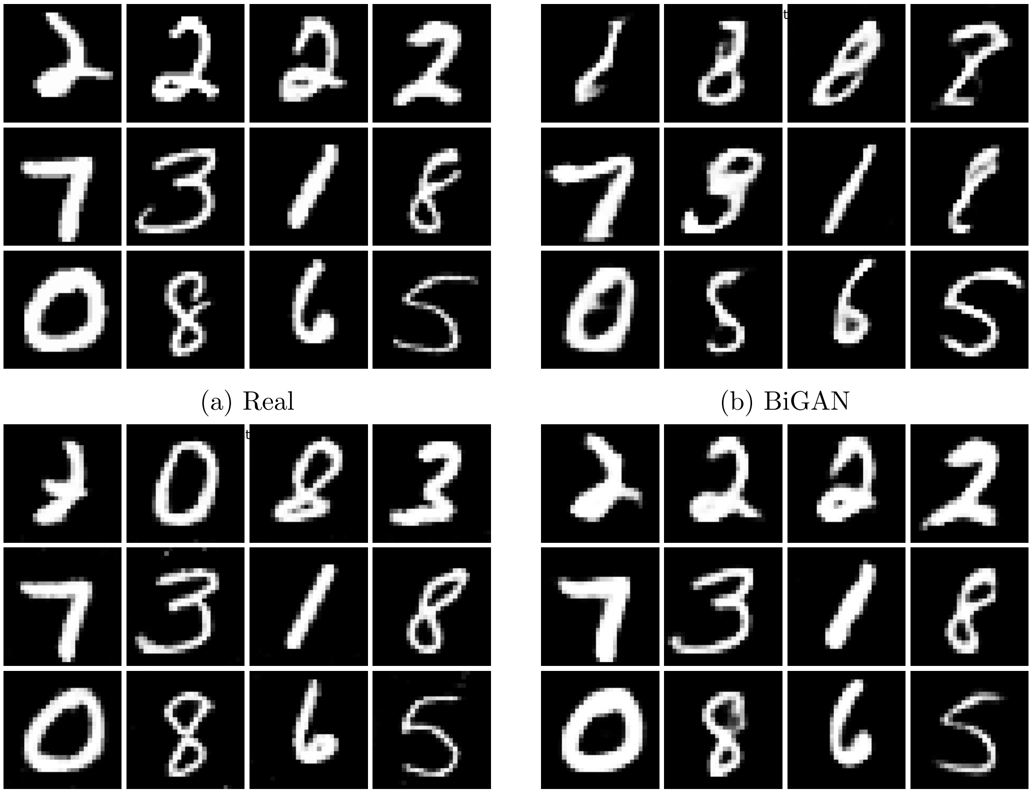
MNIST This is a hand-written digit dataset that consists of 10 kind of digits from 0 to 9. We synthesized 10 datasets from original MNIST for anomaly detection. To build each dataset, we selected one digit class as an outlier class, and samples of the other classes are regarded as an inlier samples. The 80% samples of all inlier samples is used as a train set, and the other 20% samples and all samples of the outlier class compose a test set. In ohter words, a train set consists of only inlier samples and a test set includes both inlier and outlier samples.

KDD99 10 percent The KDD99 10 percent dataset is a representative benchmark dataset for anomaly detection Dheeru and Karra Taniskidou (2017). In this paper, we call this dataset KDD99 for brevity. KDD99 consists of samples that have 41 dimensions: 34 of them are continuous and 7 is categorical. We preprocessed the categorical attributes by using one-hot encoding so that samples have 120 dimensions. KDD99 has two classes: the one is “normal” and the other is “attack”. We defined “normal” as an outlier class because “attack” class is a majority group. The whole dataset is randomly divided into 2 datasets at the same size. The one dataset in which outlier samples are removed is used as a train set, and the other dataset is used as a test set. As a result, a train set only includes inlier samples and a test set consists of both inlier and outlier samples.

5.2. Baseline

In this paper, we mainly target to present an improved GAN-based anomaly detection method, so our baselines include AnoGAN and BiGAN. In addition to the previous GAN-based methods, comparing FBGANs to other anomaly detection methods is necessary. In this section, we briefly introduce other baselines in our experiments.

OC-SVM One-class support vector machine Schölkopf et al. (2001) is a traditional kernel-based methods for anomaly detection. Residual basis kernel is used.



(a) Real

(b) BiGAN

(c) FBGANs

(d) Autoencoder

Figure 3: Reconstructions in MNIST dataset when an anomaly digit is "2". (a) Real samples in test set. (b) Although BiGAN gives poor reconstructions on anomalous samples (1st row), BiGAN gives reconstructions which are similar but largely different from the original inlier samples in detail (2nd and 3rd rows). (c) FBGANs reproduces outlier samples to be inlier samples (1st row) and gives almost identical reconstructions from the orignal inlier samples (2nd and 3rd rows). (d) In contrast, autoencoder gives good reconstructions on both inlier and outlier samples, which is harmful to anomaly detection performance.

DSEBM A deep structured energy based model for anomaly detection has been proposed by Zhai et al. (2016). DSEBM-r uses reconstruction error as an anomaly score and DSEBM-e uses sample energy as an anomaly score.

DAGMM Deep auto-encoding Gaussian mixture model Zong et al. (2018) learns both dimensionality reduction and density estimation in an end-to-end manner by using an autoencoder and a Gaussian mixture model.

For AnoGAN and BiGAN, we implemented these models based on a public implementation given by Zenati et al. (2018) and followed the same evaluation protocols proposed in Zenati et al. (2018). For OC-SVM, DSEBM-r, DSEBM-e and DAGMM, we use results presented by Zong et al. (2018).

5.3. Training

Architecture In MNIST, our forward GAN follows in split of Deep Convolutional GANs (DCGANs) Radford and Metz (2016). We used strided convolution instead of pooling layer for the discriminator, and used strided transposed convolution for the generator. In the backward GAN, the encoder also used strided convolutions and the discriminator is a Multilayer Perceptron (MLP). In KDD99, all component are MLPs. In all experiments, FBGANs used some regularization techniques: batch normalization Ioffe and Szegedy (2015) and dropout Srivastava et al. (2014). For fair comparison, we used the same architecture for all generators and encoders in BiGAN and FBGANs and minimized architectural differences between discriminators of each model as possible as we can.

Hyperparameter Setting We used ADAM optimizer Kingma and Ba (2015) for training FBGANs and all baseline models. In FBGANs, learning rates are selected as 10^{-3} and 10^{-5} and latent dimension sizes are selected as 35 and 32 for MNIST and KDD99 respectively. In MNIST, learning rates and latent dimension sizes were searched in $[10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}]$ and in $[20, 35, 50, 100, 200]$ respectively. After performing this hyperparameter search, mean AUPRs of AnoGAN and BiGAN were improved by 10% and 42% respectively compared to those reported in Zenati et al. (2018). In KDD, we adopted the same learning rate and latent dimension size in Zenati et al. (2018). Trainings were performed during 100 epochs with batch size of 100 and 50 for MNIST and KDD99 respectively. In FBGANs, the feature matching weight w_1 and w_2 are not sensitive, so we set them as 1 for all experiments.

5.4. Evaluation

Anomaly Score We defined an anomaly score as like (3). We defined a residual score as a reconstruction error: $S_R(\mathbf{x}) = \|\mathbf{x} - G(E(\mathbf{x}))\|_2$ and a feature matching error as a discrimination score: $S_D(\mathbf{x}) = \|D_f^h(\mathbf{x}) - D_f^h(G(E(\mathbf{x})))\|_2$. We set the score weight hyperparameter λ as 0 for both MNIST and KDD99. We found that using a discrimination score is not helpful for anomaly detection performance of FBGANs. Using a discrimination score ($\lambda \neq 0$) degrades anomaly detection performance in all dataset. This can be interpreted such that it's because FBGANs alleviates a bad cycle consistency problem significantly.

Evaluation Metric We evaluated our model for MNIST by using Area Under Precision Recall curve (AUPR) because the proportion of abnormal samples in a test set is slightly different depending on an anomaly digit class. In KDD99, anomaly ratio ρ in test set is fixed as a constant: $\rho = 0.2$. In other words, test samples having top 20% anomaly score are classified as anomalies. We evaluated our model by using precision, recall and F_1 score.

5.5. Results

Anomaly Detection In qualitative results on MNIST, both FBGANs and BiGAN give poor reconstructions on outlier samples, but FBGANs gives better reconstructions on inlier images compared to BiGAN (Fig. 3). In quantitative results on MNIST, FBGANs shows much higher AUPR over all digits except '9' than previous approaches (Table 1). In KDD, FBGANs shows significantly improved performance over precision, recall and F_1-scores compared to previous state-of-the-art methods (Table 2).

Table 1: Mean AUPR on MNIST (3 seeds)

Anomaly digit	**AnoGAN**	**AUPR** **BiGAN**	**FBGANs**
0	0.7548	0.8709	**0.9425**
1	0.3209	0.3359	**0.4408**
2	0.7529	0.9040	**0.9415**
3	0.5662	0.6375	**0.7955**
4	0.5502	0.7575	**0.8032**
5	0.5961	0.7437	**0.7975**
6	0.7353	0.7468	**0.8429**
7	0.5419	0.6508	**0.6490**
8	0.5379	0.7266	**0.7979**
9	0.3704	**0.4624**	0.4529
mean	0.5723	0.6791	**0.7464**

Table 2: Mean precision and recall on KDD99 (10 seeds)

Models	**Precision**	**Recall**	**F1-score**
OC-SVM	0.7457	0.8523	0.7954
DSEBM-r	0.8521	0.6472	0.7328
DSEBM-e	0.8619	0.6446	0.7399
DAGMM	0.9297	0.9442	0.9369
AnoGAN	0.8786	0.8297	0.8865
BiGAN	0.9200	0.9582	0.9372
FBGANs	**0.9539**	**0.9691**	**0.9614**

Unreliable Discriminator in BiGAN We executed an additional experiment to show reliability of learned discriminators in BiGAN. As aforementioned, we assumed that a discriminator in BiGAN does not need to consider relation between a data point and a latent point to achieve its training goal. The discriminator can achieve its goal by just looking at a data point and making a decision whether this point is real or fake. Consequently, the discriminator cannot give informative training signals for the encoder and generator to be coupled tightly, which can result in a bad cycle consistency problem in BiGAN. To support this assumption empirically, we compared mean sigmoid outputs of learned discriminator from some different inputs (Fig. 4). The results show that the outputs of the learned discriminators are highly determined by a data point. Let S_{gen} be a mean output of pairs from generator: $S_{gen} = \frac{1}{N}\sum_{i=1}^{N} D(G(\mathbf{z}_i), \mathbf{z}_i)$ and S_{enc} be a mean output of pairs from encoder: $S_{enc} = \frac{1}{N}\sum_{i=1}^{N} D(\mathbf{x}_i, E(\mathbf{x}_i))$ and S_{rand} be a mean output of random pairs: $S_{rand} = \frac{1}{N}\sum_{i=1}^{N} D(\mathbf{x}_i, \mathbf{z}_i)$ where $\mathbf{x}_i$ is a data sample in test set and $\mathbf{z}_i$ is a sample randomly drawn from a predefined latent distribution. N is the number of test samples. $|S_{rand} - S_{gen}|$ is an absolute difference after $G(\mathbf{z}_i)$ in S_{gen} is replaced by a random data sample $\mathbf{x}_i$, which

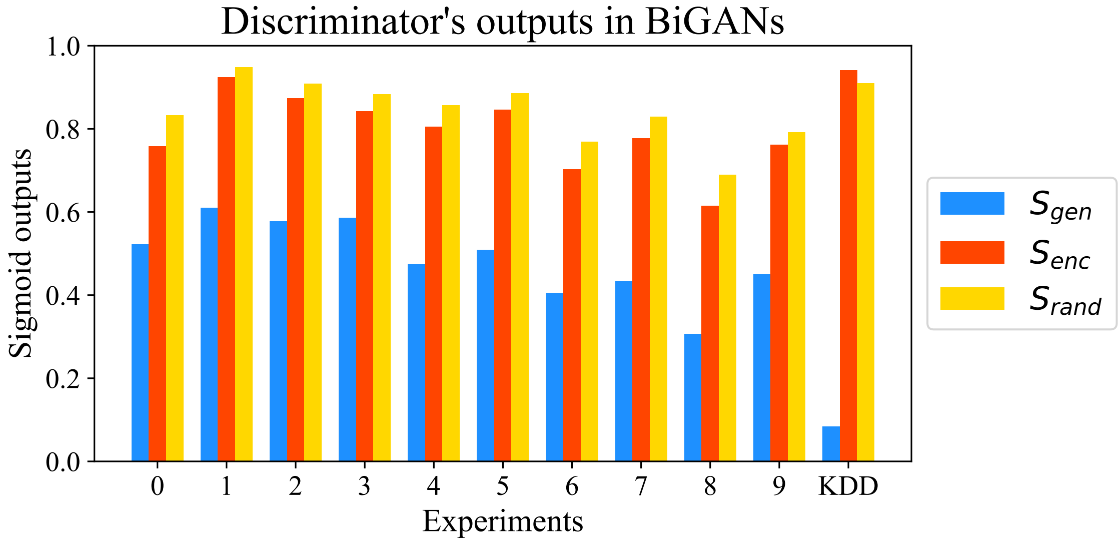


Figure 4: Comparison of mean sigmoid outputs from learned discriminators in BiGAN. X-axis indicates experiments performed in this paper and each digit indicates the anomaly class used in each MNIST experiment. Y-axis indicates mean sigmoid outputs of BiGAN's discriminators after training. The sigmoid output is a discriminator's prediction to the probability that a given pair is from an encoder. S_{gen} are mean sigmoid outputs of pairs from a generator and S_{enc} are the values of pairs from an encoder. S_{rand} are the values from pairs sampled randomly from true data distributions and predefined latent priors independently. Over all experiments, $|S_{rand} - S_{gen}|$ are 4.2 $\sim$ 26.5× greater than $|S_{rand} - S_{enc}|$, which means that the output of a leanred discriminator in BiGAN is highly determined by a data point

means how much effect changes in a data point have on the learned discriminator. In contrast, $|S_{rand} - S_{enc}|$ is an absolute difference after $E(\mathbf{x}_i)$ in S_{enc} is replaced by a random latent point $\mathbf{z}_i$, which means the effect on discriminator's output caused by changes in a latent point. In our experiments, $|S_{rand} - S_{gen}|$ are 4.2 $\sim$ 26.5× greater than $|S_{rand} - S_{enc}|$. In other words, discriminator's output is highly determined on a data point. In addition, we also found that S_{rand} is greater than S_{enc} in all MNIST experiments, which means BiGAN discriminators believe that random pairs comes from an encoder more confidently than even actual samples from the encoder. These results empirically show how unreliable a discriminator and training signals it gives in BiGAN. FBGANs gives more informative training signals, a discriminative feature matching loss, to an encoder and a generator and mitigate a bad cycle consistency problem which is frequently observed in BiGAN.

6. Conclusion

We proposed FBGANs to mitigate bad cycle consistency problem that BiGAN suffer from. By replacing an ambiguous training signal from BiGAN discriminator with an explict feature matching loss, FBGANs alleviates the bad cycle consistency problem while forcing outliers to be reproduced within an inlier distribution. Superior anomaly detection performance of FBGANs over the previous state-of-the-art methods is also demonstrated in experiments.

References

Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

J. Donahue, T. Darrell, and P. Krähenbühl. Adversarial feature learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

I. Golan and R. El-Yaniv. Deep anomaly detection using geometric transformations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

I. J. 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 (NeurIPS)*, volume 27, 2014.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning (ICML)*, Lille, France, 2015.

D. P. Kingma and J. L. Ba. ADAM: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–444, 2015.

E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

A. Radford and L. Metz. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016.

L. Ruff, R. A.Vandermeulen, N. Görnitz, L. Deecke, S. A.Siddiqui, A. Binder, E/ Müller, and M. Kloft. Deep one-class classification. In *Proceedings of the International Conference on Machine Learning (ICML)*, Stockholm, Sweden, 2018.

O. Salem, A. Guerassimov, A. Mehaoua, A. Marcus, and B. Furh. Sensor fault and patient anomaly detection and classification in medical wireless sensor networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, 2013.

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 *Proceedings of the International Conference on Information Processing in Medical Imaging (IPMI)*, 2017.

B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471, 2001.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

W. Sultani, C. Chen, and M. Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, Utah, USA, 2018.

S. C. Tan, K. M. Ting, and T. F. Liu. Fast anomaly detection for streaming data. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2011.

H. Zenati, C.-S. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar. Efficient GAN-based anomaly detection. In *ICLR Workshop Track*, 2018.

S. Zhai, Y. Cheng, W. Lu, and Z. Zhang. Deep structured energy based models for anomaly detection. In *Proceedings of the International Conference on Machine Learning (ICML)*, New York, NY, USA, 2016.

J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017.

B. Zong, Q. Song, M. R. Min†, W. Cheng, C. Lumezanu, D. Cho, and H. Chen. Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.