History-based anomaly detector: an adversarial approach to anomaly detection

Pierrick Chatillon[1] and Coloma Ballester[2]

[1] École normale supérieure Paris-Saclay, France,
pierrick.chatillon@ens-paris-saclay.fr
[2] Universitat Pompeu Fabra, Spain,
coloma.ballester@upf.edu

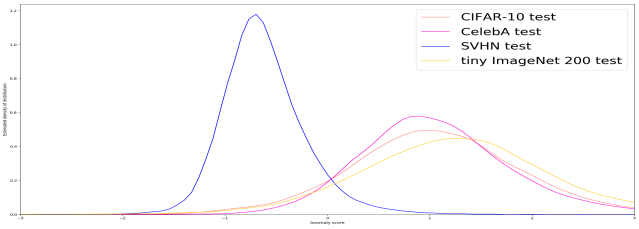
Abstract. Anomaly detection is a difficult problem in many areas and has recently been subject to a lot of attention. Classifying unseen data as anomalous is a challenging matter. Latest proposed methods rely on Generative Adversarial Networks (GANs) to estimate the normal data distribution, and produce an anomaly score prediction for any given data. In this article, we propose a simple yet new adversarial method to tackle this problem, denoted as History-based anomaly detector (HistoryAD). It consists of a self-supervised model, trained to recognize 'normal' samples by comparing them to samples based on the training history of a previously trained GAN. Quantitative and qualitative results are presented evaluating its performance. We also present a comparison to several state-of-the-art methods for anomaly detection showing that our proposal achieves top-tier results on several datasets.

Keywords: Anomaly detection, Generative Adversarial Networks, Wasserstein and Total Variation distances.

1 Introduction

Anomaly detection usually refers to the identification of unusual patterns that do not conform to expected behaviour of data, be it visual data such as images and videos, or other modalities such as acoustics or natural language. Its applications are numerous and include the detection of anomalies in medical or biological imaging such as failure of neurocognitive functions in damaged brains [1–3], real-life image forgery resulting in fake news or even fraud [4–7], anomaly detection in image or video for autonomous navigation, driver assistance systems or surveillance systems for, *e.g.*, violence alerting or evidence investigation [8–13], or for detection of violation and foul in sports analysis, detection of defective samples in manufacturing industry [14–16], sea mines in side-scan sonar images [17] or extrange aerial objects in aerial images that may produce collisions [18], anomalies from multi-modal data including visual data, audio data or natural language [19], to name but a few of its applications.

The precise definition of anomalous data is inherently difficult as, in practice, an unexpected anomaly can be detected only against the ground of a pattern



Approximate density of anomaly score distribution for each dataset.

test split	CIFAR-10	CelebA	Tiny ImageNet
AUPRC	0.941	0.976	0.949

Table 1. AUPRC for SVHN compared to other datasets.

Fig. 1. Method trained on SVHN and evaluated on several datasets.

regularity. This is one of the reasons that anomaly detection is frequently approached as out-of-distribution or outlier detection. A detailed account of the many existing methods to approach this problem can be found in [20–23].

This paper proposes a new method for anomaly detection in the context of image processing that is based on the unsupervised learning of the underlying probability distribution of normal data through appropriate GANs and the proposal of a new anomaly score for the detection of abnormal images. Our anomaly detector leverages a recorded history of the normal data generator to fully discriminate regions where true data points are more dense and use this learning to successfully detect anomalies. It results in a general anomaly detector that is free of assumptions on the data and thus it can be applied in any context and data modality. Fig. 1 illustrates an example of the performance of our anomaly detector on structurally different datasets. In this experiment, the distribution of the Street View House Numbers (SVHN) dataset [24] is first learned (details in Section 3) and considered as normal data. Then, our anomaly score is computed on samples of it and also on samples from the CIFAR-10 [25], CelebA [26] and ImageNet [27] datasets. The approximated density of the anomaly score distribution for each dataset is shown at the top of Fig. 1. Let us notice that, for the normal data, its anomaly values are around -1 while for all the 'anomalous' datasets, the anomaly scores are around $+1$. On the other hand, Table 1 in Fig. 1 bottom shows the area under the precision-recall curve (AUPRC).

The outline of this paper is as follows. Section 2 reviews related research. Section 3 details our proposal. In Section 4, the model architecture and implementation details are provided while the experimental results are presented in Section 5. Finally, the paper is concluded in Section 6.

2 Related Work

The automatic identification of abnormal or manipulated data is crucial in many contexts [2, 28, 4, 6, 7, 10, 16]. Anomaly detection, has been a topic in statistics

for centuries (see [20–23] and references therein). The authors of [22] classify the methods in the literature by the structural assumption made on the normality. Other works challenge anomaly detection with unsupervised or self-supervised learning strategies by taking advantage of the huge amount of data frequently at our disposal [29, 30]. Some of them use generative models to learn the (normal) data distribution. Generative models are methods that produce novel samples from high-dimensional data distributions, such as images or videos. Currently the most prominent approaches include autoregressive models [31], variational autoencoders (VAE) [32], and generative adversarial networks (GANs) [33]. GANs are often credited for producing less burry outputs when used for image generation. In the anomaly detection context, several approaches tackle it using autoencoders [34] or GANs [30, 35–41] (we refer to [42] for a summary of those GAN-based anomaly detection methods). Some works focus on the implicit inversion of the generator in order to detect anomalous data that do not fall in the learned model [43, 38], while others directly infer likelihoods with, for instance, normalizing flows [44–48]. On the other hand, the recent paper [34] uses a memory-augmented autoencoder which learns and records a fixed number of prototypical normal encoded vectors. Given an input sample, it is encoded and the memory is then accessed with an attention-based module to express this encoding by a sparse combination of the stored normal prototypes that used to reconstruct the input data via a decoder. The l_2 distance between the input and its reconstruction is used as anomaly score. Very differently, our anomaly detector leverages the recorded history of the normal data generator to fully discriminate regions where true data points are more dense and use this learning to successfully detect anomalies. The idea of producing an anomaly score prediction for any given data has also been investigated [38, 30, 39]. Our proposal fits in the class of self-supervised approaches and it is trained only on normal (non-anomalous) samples. The proposed method is general, efficient and simple as it uses the rich information of the training process in the construction the anomaly detector.

3 Proposed method

We will attribute an anomaly score to any image. Inspired by some ideas in [40] and [41], this score consists of the output of a network.

More precisely, let $\mathbb{P}_{\text{data}}$ be the probability distribution of a given 'normal images' dataset. Our proposal is grounded on, first, the learning of the probability distribution $\mathbb{P}_{\text{data}}$ using a GAN learning strategy while simultaneously keeping track of the states of the associated generator and discriminator during training. Secondly, we create a probability distribution (denoted as $\mathbb{P}_{G_{\text{hist}}}$) that combines different states of the previous generator's history. We finally train our anomaly detector by computing the Total Variation distance between the real data distribution $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$. Fig. 2 displays an outline of the whole method, and in the following Sections 3.1, 3.2, and 3.3, these steps of our approach are detailed and justified.

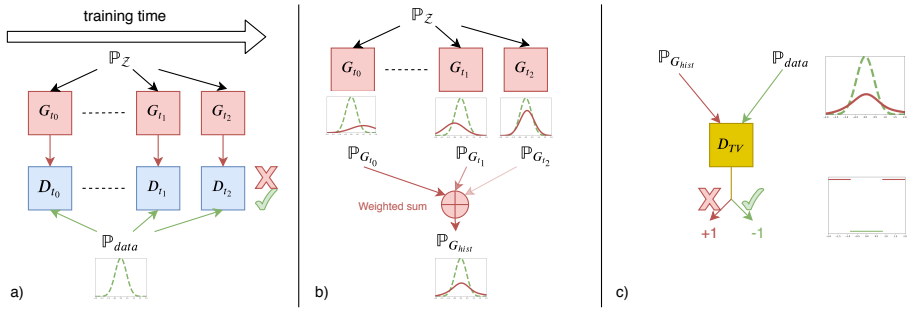


Fig. 2. Outline of the proposed method: a) Some states of the generator are saved during GAN training (G_{t_i} and D_{t_i} represent the states of of the generator and discriminator at training time t_i). b) these networks are used to form a new distribution $\mathbb{P}_{G_{\text{hist}}}$ rich in 'anomalous' samples. c) We use this distribution as negative class for a classifier. D_{TV}

3.1 Learning to generate training-like data

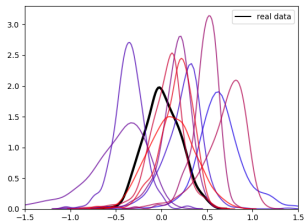
As mentioned, a GAN-based adversarial strategy is followed. Let us recall that the GAN strategy [33] is based on a game theory scenario between two networks, the generator and the discriminator, having adversarial objectives. The generator maps a noise vector (of density $\mathbb{P}_Z$) from the latent space to the image space trying to trick the discriminator, while the discriminator receives either a generated or a real image and must distinguish between both. This procedure leads the probability distribution of the generated data to be as close as possible, for some distance, to the one of the real data. For the Vanilla GAN [33], the minimized distance is the Jensen-Shannon Divergence, which has arguably bad properties (see Section 2 of [49] for details). The authors of [49] introduced the idea of minimizing the Wasserstein-1 distance (denoted as $\mathbb{W}_1$) instead. They proved several of its nice properties, including its continuity and differentiability almost everywhere (under certain hypotheses). The Wasserstein-1 distance can be computed with the Kantorovich duality property: if $\mathbb{P}_1$ and $\mathbb{P}_2$ are two probability distributions, then

$$\mathbb{W}_1(\mathbb{P}_1, \mathbb{P}_2) = \sup_{D \in \mathcal{D}} \mathbb{E}_{x \sim \mathbb{P}_1}[D(x)] - \mathbb{E}_{y \sim \mathbb{P}_2}[D(y)], \tag{1}$$

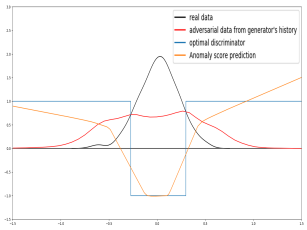
where $\mathcal{D}$ is set of 1-Lipschitz functions, i.e., in the notations of [50], the set of c-convex functions for the cost function $c(x, y) =\mid x - y \mid$. Let G and D be the generator and the discriminator learned by optimizing the adversarial Wasserstein GAN loss (WGAN),

$$\inf_G \sup_{D \in \mathcal{D}} \mathbb{E}_{x \sim \mathbb{P}_G}[D(x)] - \mathbb{E}_{x \sim \mathbb{P}_{\text{data}}}[D(x)]\,. \tag{2}$$

Notice that the optimal dual variable D^* obtained from the optimization of (2) will be negative on real data samples and positive on generated ones. In this



(a) Empirical generated distributions; color from blue to red indicates the progression of training.



(b) Comparison of $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$ and profile of optimal D^*_{TV} and trained D_{TV}.

Fig. 3. Method illustration on a toy one-dimensional dataset of points sampled from a normal law.

paper, we use the learning strategy of [51] which is based on approximating the class $\mathcal{D}$ by neural networks D subject to a gradient penalty (forcing the L^2 norm of the gradient of the discriminator with respect to its input to be close to 1). The choice of WGAN instead of other GAN losses favours nice properties such as avoiding vanishing gradients and mode collapse, and achieves more stable training.

3.2 Generator's history probability distribution

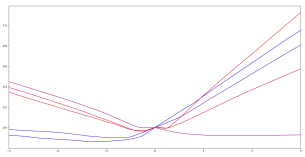
Let us start by presenting the underlying idea. During training (equation (2) with algorithm of [51]), the discriminator D will indicate regions that may contain real data, and G learns to produce samples in that zone. If these zones do not contain real data, then the discriminator will act as a critic and indicate it to the generator and point at other regions. This way, screenshots of the generator during training keep track of data points surrounding the real data manifold. In this paper, we merge the screenshots of the generator during training to form:

$$\mathbb{P}_{G_{\text{hist}}} \triangleq \int_{\alpha}^{n_{\text{epochs}}} c \cdot G_t(\mathbb{P}_Z) \cdot e^{-\beta t} dt, \tag{3}$$

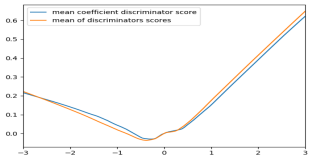
(see Fig. 2) where G_t denotes the state of the generator at training time t and $\mathbb{P}_Z$ is the latent space distribution (parameters α and β are discussed in Section 4). As a weighted mean of training generated distributions, we may assume by construction that $\mathbb{P}_{G_{\text{hist}}}$ covers $\mathbb{P}_{\text{data}}$. That is,

$$\textbf{Hypothesis}: \quad \text{supp}\left(\mathbb{P}_{\text{data}}\right) \subset \text{supp}\left(\mathbb{P}_{G_{\text{hist}}}\right). \tag{4}$$

To illustrate our hypothesis and our whole method, we present a proof of concept by creating a toy one-dimensional dataset of points sampled from the normal law. We then train the WGAN, with the generator initialized with an offset so that it does not match training data. As previously explained (details in Section 4) we save the states of the generator. Fig. 3(a) displays empirical generated distributions of some of these states.



(a) Discriminator score output during training (from blue to red).



(b) Average outputs of discriminators versus output of a average coefficient discriminator.

Fig. 4. Justification of D_{TV} initialization on the toy example.

In order to satisfy Hypothesis (4), we use momentum based optimizers, so that $\mathbb{P}_G$ oscillates around $\mathbb{P}_{\text{data}}$ (see Fig. 3(a)), making the support of $\mathbb{P}_{G_{\text{hist}}}$ cover the one of $\mathbb{P}_{\text{data}}$ better (see Fig. 3(b), where $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$ are, respectively, plotted in black and red).

This empirically confirms the hypothesis in this toy case.

3.3 Training our anomaly detector D_{TV}

As announced above, we will compute the Total Variation distance between $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$. Let us explain how and why. Firstly, we recall the Total Variation distance definition:

$$\delta(\mathbb{P}_1, \mathbb{P}_2) = \sup_{A \text{ Borel subset}} |\mathbb{P}_1(A) - \mathbb{P}_2(A)|\,, \tag{5}$$

which represents the choice $c(x, y) = 1_{x \neq y}$ in the optimal transport problem, as stated in [50] (where 1_A denotes the indicator function of a set A, as usual). As noticed by several authors (see, *e.g.*, [49]), the topology induced by the Total Variation distance is stronger that the one induced by the Wasserstein-1 distance. Let us remark that $\delta(\mathbb{P}_1, \mathbb{P}_2) = \frac{1}{2}\|\mathbb{P}_1 - \mathbb{P}_2\|_{TV}$, where $\|\cdot\|_{TV}$ denotes the Total Variation norm. The Kantorovich duality yields:

$$2\ \delta(\mathbb{P}_1, \mathbb{P}_2) = \sup_{-1 \leq D \leq 1} \left(\mathbb{E}_{x \sim \mathbb{P}_1}[D(x)] - \mathbb{E}_{y \sim \mathbb{P}_2}[D(y)]\right). \tag{6}$$

From this equation (6), we infer our ideal training objective:

$$\sup_{-1 \leq D \leq 1} \mathbb{E}_{x \sim \mathbb{P}_{G_{\text{hist}}}}[D(x)] - \mathbb{E}_{x \sim \mathbb{P}_{\text{data}}}[D(x)]\,. \tag{7}$$

Let us notice that the optimal state D^* in (6) is completely understood: Paraphrasing [49], take $\mu = \mathbb{P}_1 - \mathbb{P}_2$, which is a signed measure, and (P, N) its Hahn decomposition ($P = \{d\mathbb{P}_1 > d\mathbb{P}_2\}$). Then, we can define $D^* := 1_P - 1_N$, we have

$-1 \leq D^* \leq 1$, and

$$\begin{aligned} E_{x\sim\mathbb{P}_1}\left[D^*(x)\right] - \mathbb{E}_{x\sim\mathbb{P}_2}\left[D^*(x)\right] &= \int D^* \mathrm{d}\mu \\ &= \mu(P) - \mu(N) \\ &= \|\mu\|_{TV} \\ &= 2\ \delta\left(\mathbb{P}_1, \mathbb{P}_2\right) \end{aligned} \tag{8}$$

which closes the duality gap with the Kantorovitch optimal transport primal problem, hence the optimality of the dual variable D^*.

Now, we can approximate the Total Variation distance between $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$ by optimizing (7) over D_{TV}, our neural network approximation of the dual variable D.

Several authors (*e.g.*, [36]) have pointed out that the output of a discriminator obtained in the framework of adversarial training is not fitted for anomaly detection. Nevertheless, notice that our discriminator D_{TV} deals with two fixed distributions, $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$. Here, the purpose of computing Total Variation distance is only used to reach the optimal D^*_{TV} in this well-posed problem, assuming Hypothesis (4). D_{TV} should converge to $D^*_{TV} = 1_P - 1_N$ where (P, N) is the Hahn decomposition of $d\mathbb{P}_{G_{\text{hist}}} - d\mathbb{P}_{\text{data}}$ (see Fig. 3(b), blue curve). Importantly, we hope that thanks to the structure of the data ($\mathbb{P}_{G_{\text{hist}}}$ covering $\mathbb{P}_{\text{data}}$), D_{TV} will be able to generalize high anomaly scores on unseen data. Again, this seems to hold true in our simple case: The orange curve in Fig. 3 keeps increasing outside of $\text{supp}(\mathbb{P}_{G_{\text{hist}}})$.

To avoid vanishing gradient issues, we enforce the 'boundedness' condition on D_{TV} not by a *tanh* non-linearity (for instance), but by applying a smooth loss (weighted by $\lambda > 0$) to its output:

$$\lambda \cdot d(D_{TV}(x), [-1, 1])^2 \tag{9}$$

where $d(v, [-1, 1])$ denotes the distance of a real value $v \in \mathbb{R}$ to the set $[-1, 1]$.

Our final training loss, to be minimized, reads:

$$\begin{aligned} \mathcal{L}(D) =& \mathbb{E}_{x\sim\mathbb{P}_{\text{data}}}\left[D(x)\right] - \mathbb{E}_{x\sim\mathbb{P}_{G_{\text{hist}}}}\left[D(x)\right] \\ &+ \lambda\, \mathbb{E}_{x\sim\frac{\mathbb{P}_{\text{data}}+\mathbb{P}_{G_{\text{hist}}}}{2}}\left[d(D(x), [-1, 1])^2\right] \end{aligned} \tag{10}$$

As proved in the Appendix, the optimal D for this problem is $D^* = D^*_{TV} + \Delta^*$, with

$$\Delta^*(x) = \frac{d\mathbb{P}_{G_{\text{hist}}}(x) - d\mathbb{P}_{\text{data}}(x)}{\lambda(d\mathbb{P}_{G_{\text{hist}}}(x) + d\mathbb{P}_{\text{data}}(x))} \tag{11}$$

and the minimum loss is

$$-2 \cdot \delta(\mathbb{P}_{\text{data}}, \mathbb{P}_{G_{\text{hist}}}) - \frac{1}{2\lambda} \int \frac{(d\mathbb{P}_{G_{\text{hist}}}(x) - d\mathbb{P}_{\text{data}}(x))^2}{(d\mathbb{P}_{G_{\text{hist}}}(x) + d\mathbb{P}_{\text{data}}(x))} dx. \tag{12}$$

By letting $\lambda \longrightarrow \infty$, the second term in (10) becomes an infinite well regularization term which enforces $-1 \leq D \leq 1$, approaching the solution of (7). This

explains why the second term in (12) vanishes when $\lambda \longrightarrow \infty$. In practice, for better results, we allow a small trade-off between the two objectives with $\lambda = 10$.

From now on, we will use the output of D_{TV} as anomaly score. In a nutshell, our method should fully discriminate regions where data points are more dense than synthetic anomalous points from $\mathbb{P}_{G_{\text{hist}}}$. This yields a fast feed-forward anomaly detector that ideally assigns -1 to normal data and 1 to anomalous data. Fig. 1 shows an example.

Our method also has the advantage of staying true to the training objective, not modifying it as in [41]. Indeed, the authors of [41] implement a similar method but using a non-converged state of the generator as an anomaly generator. In order to achieve this, they add a term to the log loss that prevents the model from converging all the way.

4 Model architecture and implementation details

In this section, the architecture and implementation of each of the three steps detailed in Section 3 is described.

4.1 Architecture description.

G uses transpose convolution to upscale the random features, Leaky ReLU nonlinearities and BatchNorm layers. D is a classic Convolutionnal Neural Network for classification, that uses pooling downscaling, Leaky ReLU before passing the obtained features through fully connected layers. Both G and D roughly have 5M parameters. D_{TV} has the same architecture as D.

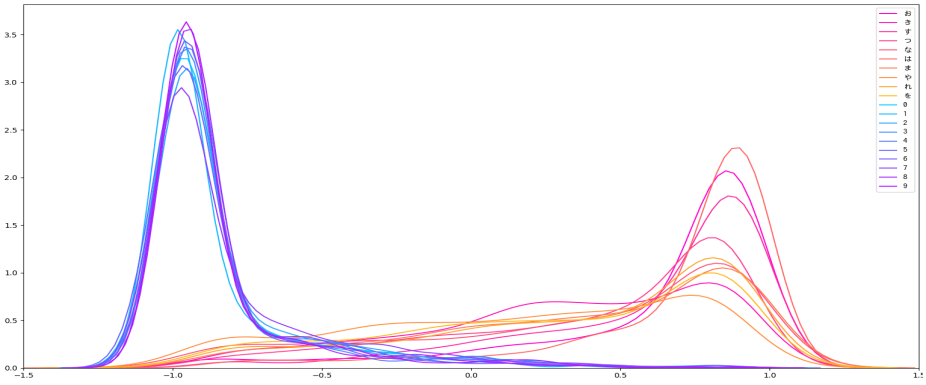
4.2 Learning to generate training-like data.

We first train until convergence of G and D according to the WGAN-GP objective of Section 3.1 for a total of n_{epochs} epochs, and save the network states at regular intervals (50 times per epoch). We optimize our objective loss using Adam optimizers, with decreasing learning rate initially equal to $5 \cdot 10^{-4}$.

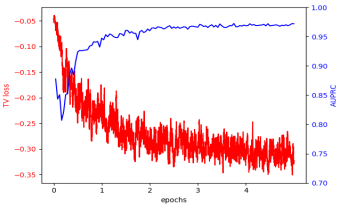
4.3 Generator's history probability distribution

As announced in the previous section, if the training process were to be continuous we would arbitrarily define $\mathbb{P}_{G_{\text{hist}}}$ by (3), that is, $\mathbb{P}_{G_{\text{hist}}} = \int_{\alpha}^{n_{\text{epochs}}} c \cdot G_t(\mathbb{P}_Z) \cdot e^{-\beta t} dt$, where c is a normalization constant that makes $\mathbb{P}_{G_{\text{hist}}}$ sum to 1. We avoid the first α epochs to avoid heavily biasing $\mathbb{P}_{G_{\text{hist}}}$ in favour of the initial random state of the generator. The exponential decay gives less importance to higher fidelity samples at the end of the training. In practice, we approximate $\mathbb{P}_{G_{\text{hist}}}$ by sampling data from $\mathbb{P}_{G_t}$ where t is a random variable of density of probability:

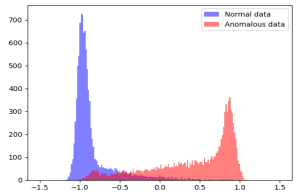
$$c \cdot 1_{[\alpha, n_{\text{epochs}}]} \cdot e^{-\beta t} \tag{13}$$



Approximate density of score distribution among each class of datasets (blue shades: MNIST, red shades: KMNIST).



Red curve: Total Variation distance minimisation. Blue curve: AUPRC of the model during training.



Histograms of anomaly scores for the test splits of MNIST and KMNIST.

One sample from each class of MNIST and KMNIST.

Fig. 5. Method trained on MNIST (normal) and evaluated on KMNIST (anomalous).

4.4 Training our anomaly detector $\mathbf{D_{TV}}$.

D_{TV} is also optimized using Adam algorithm. Since it has the same architecture as the discriminator used in the previous WGAN training, its weights, denoted as $W_{D_{TV}}$, can be initialized as:

$$W_{D_{TV}} = \int_{\alpha}^{n_{\text{epochs}}} c \cdot W_{D_t} \cdot e^{-\beta t} dt, \tag{14}$$

where D_t is the state of the WGAN discriminator at training time t. Let us comment on the reason of such initialization. Fig. 4(a) seems to indicate that averaging the discriminators' outputs is a good initialization. The obtained average is indeed shaped as a 'v' centered on the real distribution in our toy

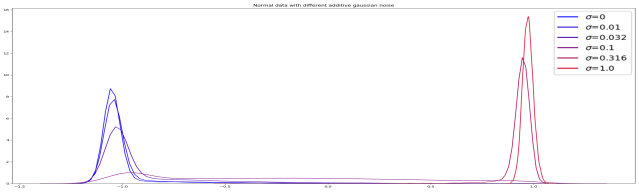


Fig. 6. Method trained on MNIST (normal) and evaluated on modified MNIST images for different levels σ of gaussian additive noise.

example. Fig. 4(b) empirically shows that the initialization of the discriminator with average coefficients is somewhat close to the average of said discriminators. As explored in [52], this Deep Network Interpolation is justified by the strong correlation of the different states of a network during training. To further discuss how good is exactly this initialization, Fig. 7 (blue error bars in the figure) shows a comparison of area under the precision-recall curve (AUPRC) with other methods on the MNIST dataset. The x-axis indicates the MNIST digit chosen as anomalous.

The following experimental cases are tested:

- **Experimental case 1:** The training explained above is implemented.
- **Experimental case 2:** The same process is applied, only modifying $\mathbb{P}_{G_{\text{hist}}}$, corrupting half generated images, by sampling the latent variable with a wider distribution $\mathbb{P}_{Z'}$:
  $\mathbb{P}_{G_{\text{hist}}} = \int_{\alpha}^{n_{\text{epochs}}} c \cdot \frac{G_t(\mathbb{P}_Z)+G_t(\mathbb{P}_{Z'})}{2} \cdot e^{-\beta t} dt.$
  The idea behind this is encouraging $\mathbb{P}_{G_{\text{hist}}}$ to spread its mass further away from $\mathbb{P}_{\text{data}}$.

Fig. 1 was computed on experimental case 1 with $n_{\text{epochs}} = 10$, $\alpha = 1$ and $\beta = 5$. Fig. 5, 6 and 7 were computed with $n_{\text{epochs}} = 5$, $\alpha = 1$, $\beta = 3$.

5 Experimental Results and Discussion

This section presents quantitative and qualitative experimental results.

Our model behaves as one would expect when presented normal images modified with increasing levels of noise, which is attributing an increasing anomaly score to them. This is illustrated in Fig. 6 where a clear correlation is seen between high values of the standard deviation of the added Gaussian noise and high density of high anomaly scores.

As a sanity check, we take the final state of the generator, trained on MNIST with (2) and [51] algorithm, and verify that our method is able to detect generated sample that do not belong to the normal MNIST distribution. In Figure 8, we randomly select two latent variable (z_1 and z_2) which are confidently classified as normal, then linearly interpolate all latent variables between them, given by $(1-t)z_1 + tz_2, \forall t \in [0,1]$. Finally, we evaluate the anomaly score of each generated image.

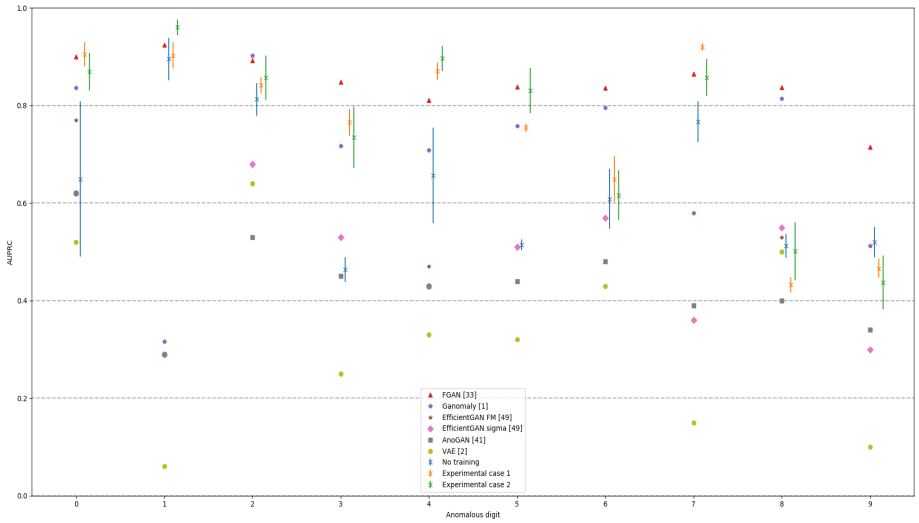


Fig. 7. Comparison of AUPRC with other methods (x-axis denotes the MNIST digit chosen as anomalous).

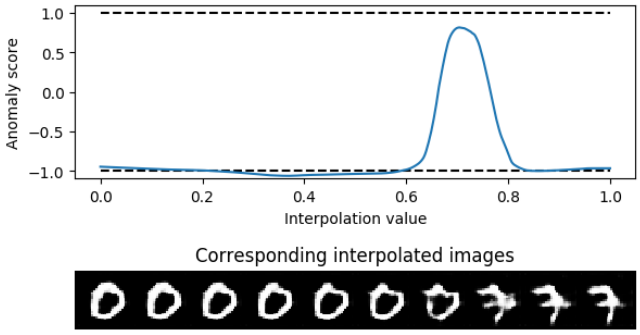


Fig. 8. Method trained on MNIST and evaluating scores on images generated from interpolated latent variables $(1-t)z_1 + tz_2$, for $t \in [0,1]$.

Finally we check the influence of the number of saves per epoch on the performance of the model. Figure 9 displays the AUPRC of normal data (MNIST) against KMNIST dataset for different values of saving frequency during WGAN training. For low values, the information carried by $\mathbb{P}_{G_{\text{hist}}}$ starts at a 'early stopping of GANs' ([40]) level, and gets richer as the number of saves per epoch increases; hence the increase in AUPRC. We do not have an explanation for the small decay in performance for big values.

Fig. 7 compares six state-of-the-art anomaly detection methods with the presented method with both experimental cases and with our D_{TV} initialization

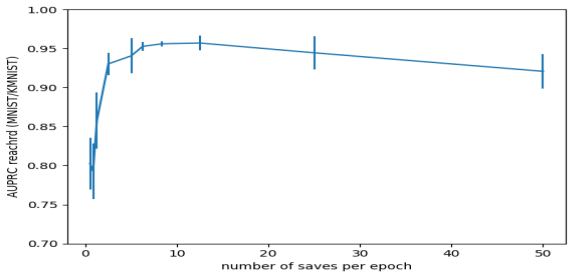


Fig. 9. Influence of the number of checkpoints per training epoch

(denoted as no training). Apart from a few digits, HistoryAD challenges state of the art anomaly detection methods. Fig. 5 shows the anomaly detection results when our method was trained on MNIST dataset and evaluated on KMNIST. Notice that most of the histogram mass of normal and anomalous data is located around -1 and 1, respectively. This figure empirically proves the robustness of the method to anomalous data structurally close to training data. On the other hand, Fig. 1 shows how well the method performs on structurally different data. Our method was trained on Street View House Numbers [24], and reached high AUPRC results. Both the approximate density and the AUPRC comparison show that the presented method is able to discriminate anomalous from normal data.

6 Conclusions and future work

In this paper, we presented our new anomaly detection approach, HistoryAD, and estimated its performance. Unlike many GAN-based methods, we do not try to invert the generator's mapping, but use the rich information of the whole training process, yielding an efficient and general anomaly detector. Further can be done in exploiting the training process of GANs, for instance, using multiple training histories to improve the adversarial complexity of $\mathbb{P}_{G_{\text{hist}}}$.

Acknowledgements

First author acknowledges support by ENS Paris-Saclay. Second author acknowledges partial support by MICINN/FEDER U project, reference PGC2018-098625-B-I00, and by H2020-MSCA-RISE-2017 project, reference 777826 NoMADS.

Appendix

The goal of this appendix is to obtain a solution of the minimization problem

$$\min_{D} \mathcal{L}(D) \tag{15}$$

where $\mathcal{L}(D)$ is given by (10). Assuming that the probability distributions $\mathbb{P}_{\text{data}}$ and $\mathbb{P}_{G_{\text{hist}}}$ admit densities $d\mathbb{P}_{\text{data}}(x)$ and $d\mathbb{P}_{G_{\text{hist}}}(x)$, respectively, the loss can be written as integral of the point-wise loss l defined below in (17):

$$\mathcal{L}(D) = \int l(D(x))dx \tag{16}$$

where

$$\begin{aligned} l(D(x)) =& (d\mathbb{P}_{\text{data}}(x) - d\mathbb{P}_{G_{\text{hist}}}(x))D(x) \\ &+ \lambda \frac{d\mathbb{P}_{\text{data}}(x) + d\mathbb{P}_{G_{\text{hist}}}(x)}{2} d(D(x), [-1,1])^2. \end{aligned} \tag{17}$$

Let us recall that $D^*_{TV} = 1_P - 1_N$ where (P, N) is the Hahn decomposition of $d\mathbb{P}_{G_{\text{hist}}} - d\mathbb{P}_{\text{data}}$ (therefore, $\text{sign}(D^*_{TV}) = D^*_{TV}$).

We notice that for all x and for all $\epsilon > 0$,

$$l[(1-\epsilon)D^*_{TV}(x)] \geq (d\mathbb{P}_{\text{data}} - d\mathbb{P}_{G_{\text{hist}}})(x)(1-\epsilon)D^*_{TV}(x) \tag{18}$$

$$> (d\mathbb{P}_{\text{data}} - d\mathbb{P}_{G_{\text{hist}}})(x)D^*_{TV}(x) \tag{19}$$

$$\text{i.e.} > l[D^*_{TV}(x)] \tag{20}$$

Indeed, inequality (18) comes from the positivity of the distance $d(\cdot, [-1,1])$. On the other hand, inequality (19) comes from the definition of D^*_{TV}. Indeed, if $d\mathbb{P}_{\text{data}}(x) - d\mathbb{P}_{G_{\text{hist}}}(x) < 0$, then $D^*_{TV}(x) = 1$; the other case $d\mathbb{P}_{\text{data}}(x) - d\mathbb{P}_{G_{\text{hist}}}(x) > 0$ gives $D^*_{TV}(x) = -1$. Either way, we obtain that $(d\mathbb{P}_{\text{data}}(x) - d\mathbb{P}_{G_{\text{hist}}}(x))(1-\epsilon)D^*_{TV}(x) > (d\mathbb{P}_{\text{data}}(x) - d\mathbb{P}_{G_{\text{hist}}}(x))D^*_{TV}(x)$. Finally, inequality (20) is obtained from $d(D^*_{TV}(x), [-1,1]) = 0$.

We can always write a real function D as $D = D^*_{TV} + \Delta$, where Δ is a certain function. We just proved that if $\text{sign}(\Delta(x)) = -\text{sign}(D^*_{TV}(x))$ on a non-negligible set, then D cannot minimize (10), since $D^*_{TV}(x)$ achieves lower value than $D(x)$ on this set.

Hence all minimizer D^* of (10) must be of the form $D^*(x) = (D^*_{TV} + \Delta)(x)$, where $\text{sign}(\Delta) = \text{sign}(D^*_{TV})$ almost everywhere. We can now re-write the point-wise loss formula (17) as

$$l(D(x)) = (d\mathbb{P}_{\text{data}}(x) - d\mathbb{P}_{G_{\text{hist}}}(x)) \cdot (D^*_{TV}(x) + \Delta(x)) \tag{21}$$

$$+ \lambda \frac{d\mathbb{P}_{\text{data}}(x) + d\mathbb{P}_{G_{\text{hist}}}(x)}{2} \Delta(x)^2(x) \tag{22}$$

$$= -2 \cdot \delta(\mathbb{P}_{\text{data}}, \mathbb{P}_{G_{\text{hist}}}) \tag{23}$$

$$+ \int (d\mathbb{P}_{\text{data}} - d\mathbb{P}_{G_{\text{hist}}}) \cdot \Delta + \lambda \frac{d\mathbb{P}_{\text{data}} + d\mathbb{P}_{G_{\text{hist}}}}{2} \Delta^2 \tag{24}$$

Minimizing this point-wise second order equation in Δ, we obtain

$$\Delta^*(x) = \frac{d\mathbb{P}_{G_{\text{hist}}}(x) - d\mathbb{P}_{\text{data}}(x)}{\lambda(d\mathbb{P}_{G_{\text{hist}}}(x) + d\mathbb{P}_{\text{data}}(x))}. \tag{25}$$

Finally, the minimum loss is

$$-2 \cdot \delta(\mathbb{P}_{\text{data}}, \mathbb{P}_{G_{\text{hist}}}) - \frac{1}{2\lambda} \int \frac{(d\mathbb{P}_{G_{\text{hist}}}(x) - d\mathbb{P}_{\text{data}}(x))^2}{(d\mathbb{P}_{G_{\text{hist}}}(x) + d\mathbb{P}_{\text{data}}(x))} dx. \tag{26}$$

References

1. Bénédicte Grosjean and Lionel Moisan. A-contrario detectability of spots in textured backgrounds. *Journal of Mathematical Imaging and Vision*, 33(3):313, 2009.
2. Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, pages 146–157. Springer, 2017.
3. Kristina Prokopetc and Adrien Bartoli. Slim (slit lamp image mosaicing): handling reflection artifacts. *International Journal of Computer Assisted Radiology and Surgery*, pages 1–10, 2017.
4. Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 101–117, 2018.
5. Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Learning rich features for image manipulation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1053–1061, 2018.
6. Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
7. T Nikoukhah, J Anger, T Ehret, M Colom, JM Morel, and R Grompone von Gioi. Jpeg grid detection based on the number of dct zeros and its application to automatic and localized forgery detection. In *CVPR*, pages 110–118, 2019.
8. Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 341–349, 2017.
9. Romero Morais, Vuong Le, Truyen Tran, Budhaditya Saha, Moussa Mansour, and Svetha Venkatesh. Learning regularity in skeleton trajectories for anomaly detection in videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
10. Jia-Xing Zhong, Nannan Li, Weijie Kong, Shan Liu, Thomas H. Li, and Ge Li. Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
11. Trong-Nguyen Nguyen and Jean Meunier. Anomaly detection in video sequence with appearance-motion correspondence. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.
12. Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
13. Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1801–1810, 2019.
14. Karim Tout, Florent Retraint, and Remi Cogranne. Automatic vision system for wheel surface inspection and monitoring. In *ASNT Annual Conference 2017*, pages 207–216, 2017.
15. Maria Zontak and Israel Cohen. Defect detection in patterned wafers using anisotropic kernels. *Machine Vision and Applications*, 21(2):129–141, 2010.

16. Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad – a comprehensive real-world dataset for unsupervised anomaly detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
17. Gal Mishne and Israel Cohen. Multiscale anomaly detection using diffusion maps. *IEEE Journal of selected topics in signal processing*, 7(1):111–123, 2013.
18. Andreas Nussberger, Helmut Grabner, and Luc Van Gool. Robust aerial object tracking from an airborne platform. *IEEE Aerospace and Electronic Systems Magazine*, 31(7):38–46, 2016.
19. Yuening Li, Ninghao Liu, Jundong Li, Mengnan Du, and Xia Hu. Deep structured cross-modal anomaly detection. *arXiv preprint arXiv:1908.03848*, 2019.
20. Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.
21. Marco AF Pimentel, David A Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 2014.
22. Thibaud Ehret, Axel Davy, Jean-Michel Morel, and Mauricio Delbracio. Image anomalies: A review and synthesis of detection methods. *Journal of Mathematical Imaging and Vision*, pages 1–34, 2019.
23. Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.
24. Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, 2011.
25. Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.
26. Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.
27. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
28. Susan M Schweizer and José MF Moura. Hyperspectral imagery: Clutter adaptation in anomaly detection. *IEEE Trans. on Information Theory*, 46(5):1855–1871, 2000.
29. Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1), 2015.
30. Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. *CoRR*, abs/1703.05921, 2017.
31. Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in neural information processing systems*, pages 4790–4798, 2016.
32. Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013.
33. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Adv in neural inf processing systems*, pages 2672–2680, 2014.
34. Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

35. Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*, 2018.
36. Lucas Deecke, Robert Vandermeulen, Lukas Ruff, Stephan Mandt, and Marius Kloft. Image anomaly detection with generative adversarial networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 3–17. Springer, 2018.
37. Mahdyar Ravanbakhsh, Moin Nabi, Enver Sangineto, Lucio Marcenaro, Carlo Regazzoni, and Nicu Sebe. Abnormal event detection in videos using generative adversarial nets. In *ICIP*, pages 1577–1581. IEEE, 2017.
38. Ilyass Haloui, Jayant Sen Gupta, and Vincent Feuillard. Anomaly detection with wasserstein gan. *arXiv preprint arXiv:1812.02463*, 2018.
39. Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian Conference on Computer Vision*, pages 622–637. Springer, 2018.
40. Volker Tresp Jindong Gu, Matthias Schubert. Semi-supervised outlier detection using a generative and adversary framework. *E& T Int Journal Computer and Information Engineering*, 12(10), 2018.
41. Cuong Phuc Ngo, Amadeus Aristo Winarto, Connie Khor Li Kou, Sojeong Park, Farhan Akram, and Hwee Kuan Lee. Fence GAN: towards better anomaly detection. *CoRR*, abs/1904.01209, 2019.
42. Federico Di Mattia, Paolo Galeone, Michele De Simoni, and Emanuele Ghelfi. A survey on gans for anomaly detection. *CoRR*, abs/1906.11632, 2019.
43. Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *CoRR*, abs/1605.09782, 2016.
44. Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Adv. Neural Information Processing Systems*, pages 10215–10224, 2018.
45. Hyunsun Choi, Eric Jang, and Alexander A Alemi. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.
46. Dan Hendrycks, Mantas Mazeika, and Thomas G Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.
47. Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? *arXiv preprint arXiv:1810.09136*, 2018.
48. Joan Serrà, David Álvarez, Vicenç Gómez, Olga Slizovskaia, José F Núñez, and Jordi Luque. Input complexity and out-of-distribution detection with likelihood-based generative models. *arXiv preprint arXiv:1909.11480*, 2019.
49. Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.
50. Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.
51. Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Adv Neural Inf Proces Sys*, pages 5769–5779, 2017.
52. Xintao Wang, Ke Yu, Chao Dong, Xiaoou Tang, and Chen Change Loy. Deep network interpolation for continuous imagery effect transition. *CoRR*, abs/1811.10515, 2018.