# Old is $\mathcal{G}^{\text{old}}$: Redefining the Adversarially Learned One-Class Classifier Training Paradigm

Muhammad Zaigham Zaheer[1,2], Jin-ha Lee[1,2], Marcella Astrid[1,2], Seung-Ik Lee[1,2]
[1]University of Science and Technology, [2]Electronics and Telecommunications Research Institute, Daejeon, South Korea
{mzz, jhlee, marcella.astrid}@ust.ac.kr, the_silee@etri.re.kr

## Abstract

*A popular method for anomaly detection is to use the generator of an adversarial network to formulate anomaly score over reconstruction loss of input. Due to the rare occurrence of anomalies, optimizing such networks can be a cumbersome task. Another possible approach is to use both generator and discriminator for anomaly detection. However, attributed to the involvement of adversarial training, this model is often unstable in a way that the performance fluctuates drastically with each training step. In this study, we propose a framework that effectively generates stable results across a wide range of training steps and allows us to use both the generator and the discriminator of an adversarial model for efficient and robust anomaly detection. Our approach transforms the fundamental role of a discriminator from identifying real and fake data to distinguishing between good and bad quality reconstructions. To this end, we prepare training examples for the good quality reconstruction by employing the current generator, whereas poor quality examples are obtained by utilizing an old state of the same generator. This way, the discriminator learns to detect subtle distortions that often appear in reconstructions of the anomaly inputs. Extensive experiments performed on Caltech-256 and MNIST image datasets for novelty detection show superior results. Furthermore, on UCSD Ped2 video dataset for anomaly detection, our model achieves a frame-level AUC of 98.1%, surpassing recent state-of-the-art methods.*

## 1. Introduction

Due to rare occurrence of anomalous scenes, the anomaly detection problem is usually seen as one-class classification (OCC) in which only normal data is used to learn a novelty detection model [22, 57, 25, 51, 11, 45, 40, 10, 44, 34, 35]. One of the recent trends to learn one-class data is by using an encoder-decoder architecture such as denoising auto-encoder [41, 52, 53, 49]. Generally, in this scheme, training is carried out until the model starts to produce good quality reconstructions [41, 43]. During the test time, it is expected to show high reconstruction loss for abnormal data which corresponds to a high anomaly score. With the recent developments in Generative Adversarial Networks (GANs) [8], some researchers also explored the possibility of improving the generative results using adversarial training [43, 32]. Such training fashion substantially enhances the data regeneration quality [30, 8, 43]. At the test time, the trained generator $\mathcal{G}$ is then decoupled from the discriminator $\mathcal{D}$ to be used as a reconstruction model. As reported in [41, 35, 36], a wide difference in reconstruction loss between the normal and abnormal data can be achieved due to adversarial training, which results in a better anomaly detection system. However, relying only on the reconstruction capability of a generator does not oftentimes work well because the usual encoder-decoder style generators may unexpectedly well-reconstruct the unseen data which drastically degrades the anomaly detection performance.

A natural drift in this domain is towards the idea of using $\mathcal{D}$ along with the conventional utilization of $\mathcal{G}$ for anomaly detection. The intuition is to gain maximum benefits of the one-class adversarial training by utilizing both $\mathcal{G}$ and $\mathcal{D}$ instead of only $\mathcal{G}$. However, this also brings along the problems commonly associated with such architectures. For example, defining a criteria to stop the training is still a challenging problem [8, 30]. As discussed in Sabokrou *et al*. [41], the performance of such adversarially learnt one-class classification architecture is highly dependent on the criteria of when to halt the training. In the case of stopping prematurely, $\mathcal{G}$ will be undertrained and in the case of overtraining, $\mathcal{D}$ may get confused because of the real-looking fake data. Our experiments show that a $\mathcal{G}$+$\mathcal{D}$ trained as a collective model for anomaly detection (referred to as a baseline) will not ensure higher convergence at any arbitrary training step over its predecessor. Figure 1 shows frame-level area under the curve (AUC) performance of the baseline over several epochs of training on UCSD Ped2 dataset [4].

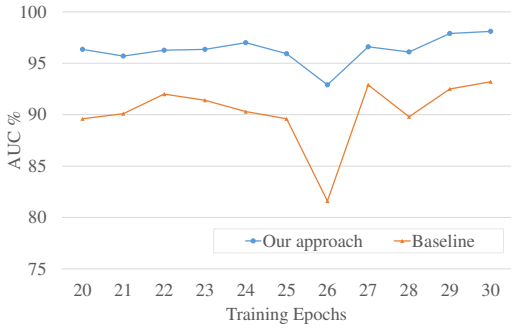



Figure 1: Dynamics of AUC performance over training epochs: The baseline shows high fluctuations while our approach not only shows stability across various epochs but also yields higher AUC.

Although we get high performance peaks at times, it can be seen that the performance fluctuates substantially even between two arbitrary consecutive epochs. Based on these findings, it can be argued that a $\mathcal{D}$ as we know it, may not be a suitable choice in a one-class classification problem, such as anomaly detection.

Following this intuition, we devise an approach for training of an adversarial network towards anomaly detection by transforming the basic role of $\mathcal{D}$ from distinguishing between real and fake to identifying good and bad quality reconstructions. This property of $\mathcal{D}$ is highly desirable in anomaly detection because a trained $\mathcal{G}$ would not produce as good reconstruction for abnormal data as it would for the normal data conforming to the learned representations. To this end we propose a two-stage training process. Phase one is identical to the common practice of training an adversarial denoising auto-encoder [30, 53, 49]. Once $\mathcal{G}$ achieves a reasonably trained state (i.e. showing low reconstruction losses), we begin phase two in which $\mathcal{D}$ is optimized by training on various good quality and bad quality reconstruction examples. Good quality reconstruction examples come from real data as well as the data regenerated by $\mathcal{G}$, whereas bad reconstruction examples are obtained by utilizing an old state of the generator ($\mathcal{G}^{old}$) as well as by using our proposed pseudo-anomaly module. Shown in Figure 3, this pseudo-anomaly module makes use of the training data to create *anomaly-like* examples. With this two-phase training process, we expect $\mathcal{D}$ to be trained in such a way that it can robustly discriminate reconstructions coming from normal and abnormal data. As shown in Figure 1, our model not only provides superior performance but also shows stability across several training epochs.

In summary, the contributions of our paper are as follows: 1) this work is among the first few to employ $\mathcal{D}$ along with $\mathcal{G}$ at test time for anomaly detection. Moreover, to the best of our knowledge, it is the first one to extensively report the impacts of using the conventional $\mathcal{G}+\mathcal{D}$ formulation and the consequent instability. 2) Our approach of transforming the role of a discriminator towards anomaly detection problem by utilizing an old state $\mathcal{G}^{old}$ of the generator along with the proposed pseudo-anomaly module, substantially improves stability of the system. Detailed analysis provided in this paper shows that our model is independent of a hard stopping criteria and achieves consistent results over a wide range of training epochs. 3) Our method outperforms state-of-the-art [41, 13, 37, 27, 7, 48, 25, 28, 11, 23, 35, 24, 34, 46, 10, 22, 52, 57, 58] in the experiments conducted on MNIST [18] and Caltech-256 [9] datasets for novelty detection as well as on UCSD Ped2 [4] video dataset for anomaly detection. Moreover, on the latter dataset, our approach provides a substantial absolute gain of 5.2% over the baseline method achieving frame level AUC of 98.1%.

## 2. Related Work

Anomaly detection is often seen as a novelty detection problem [22, 57, 25, 11, 51, 45, 40, 2, 10, 44, 34, 34, 35] in which a model is trained based on the known normal class to ultimately detect unknown outliers as abnormal. To simplify the task, some works proposed to use object tracking [50, 1, 26, 31, 56] or motion [16, 12, 5]. Handpicking features in such a way can often deteriorate the performance significantly. With the increased popularity of deep learning, some researchers [44, 35] also proposed to use pre-trained convolution network based features to train one-class classifiers. Success of such methods is highly dependent on the base model which is often trained on some unrelated datasets.

A relatively new addition to the field, image regeneration based works [7, 37, 52, 13, 27, 28, 53, 40] are the ones that make use of a generative network to learn features in an unsupervised way. Ionescu *et al*. in [13] proposed to use convolutional auto-encoders on top of object detection to learn motion and appearance representations. Xu *et al*. [52, 53] used a one-class SVM learned using features from stacked auto-encoders. Ravanbakhsh *et al*. [35] used generator as a reconstructor to detect abnormal events assuming that a generator is unable to reconstruct the inputs that do not conform the normal training data. In [27, 28], the authors suggested to use a cascaded decoder to learn motion as well as appearance from normal videos. However, in all these schemes, only a generator is employed to perform detection. Pathak *et al*. [30] proposed adversarial training to enhance the quality of regeneration. However, they also discard the discriminator once the training is finished. A unified generator and discriminator model for anomaly detection is proposed in Sabokrou *et al*. [41]. The model shows promising results, however it is often not stable and the performance relies heavily on the criteria to stop training. Recently, Shama *et al*. [43] proposed an idea of utilizing output

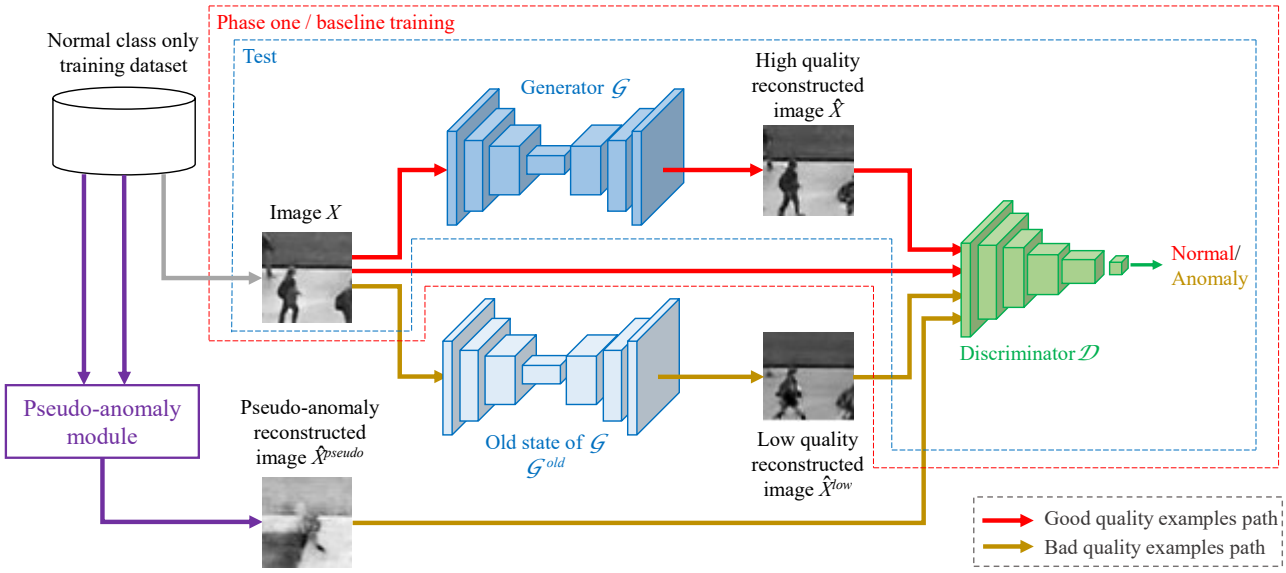



Figure 2: Our proposed OGNet framework. Phase one is the baseline training, carried out to obtain a reasonably trained state of $\mathcal{G}$ and $\mathcal{D}$. A frozen low epoch state ($\mathcal{G}^{old}$) of the generator is stored during this training. In phase two, only $\mathcal{D}$ is updated to distinguish between good and bad quality reconstructions. Good quality examples correspond to real training images as well as the images reconstructed using $\mathcal{G}$ while bad quality examples are obtained using $\mathcal{G}^{old}$ as well as the proposed pseudo-anomaly module. This module assists $\mathcal{D}$ to learn the underlying patterns of anomalous input reconstructions. During test, inferences are carried out through $\mathcal{G}$ and $\mathcal{D}$ only and the output of $\mathcal{D}$ is considered as anomaly score. Best viewed in color.

of an adversarial discriminator to increase the image quality of generated images. Although not related to anomaly detection, it provides an interesting intuition to make use of both adversarial components for an enhanced performance.

Our work, although built on top of an unsupervised generative network, is different from the approaches in [7, 13, 27, 28, 53, 40] as we explore to utilize the unified generator and discriminator model for anomaly detection. The most similar work to ours is by Sabokrou *et al*. [41] and Lee *et al*. [19] as they also explore the possibility of using discriminator, along with the conventional usage of generator, for anomaly detection. However, our approach is substantially different from these. In [41], a conventional adversarial network is trained based on a criteria to stop the training whereas, in [19], an LSTM based approach is utilized for training. In contrast, we utilize a pseudo-anomaly module along with an old state of the generator, to modify the ultimate role of a discriminator from distinguishing between real and fake to detecting between and bad quality reconstructions. This way, our overall framework, although trained adversarially in the beginning, finally aligns both the generator and the discriminator to complement each other towards anomaly detection.

## 3. Method

In this section, we present our OGNet framework. As described in Section 1, most of the existing GANs based anomaly detection approaches completely discard discriminator at test time and use generator only. Furthermore, even if both models are used, the unavailability of a criteria to stop the training coupled with the instability over training epochs caused by adversary makes the convergence uncertain. We aim to change that by redefining the role of a discriminator to make it more suitable for anomaly detection problems. Our solution is generic, hence it can be integrated with any existing one-class adversarial networks.

### 3.1. Architecture Overview

In order to maintain consistency and to have a fair comparison, we kept our baseline architecture similar to the one proposed by Sabokrou *et al*. [41]. The generator $\mathcal{G}$, a typical denoising auto-encoder, is coupled with the discriminator $\mathcal{D}$ to learn one class data in an unsupervised adversarial fashion. The goal of this model is to play a min-max game to optimize the following objective function:

$$\min_{\mathcal{G}} \max_{\mathcal{D}} \Big( \mathbb{E}_{X \sim p_t}[\log(1 - \mathcal{D}(X))] + \mathbb{E}_{\tilde{X} \sim p_t + \mathcal{N}_\sigma}[\log(\mathcal{D}(\mathcal{G}(\tilde{X})))] \Big), \tag{1}$$

where $\tilde{X}$ is the input image $X$ with added noise $\mathcal{N}_\sigma$ as in a typical denoising auto-encoder. Our model, built on top of the baseline, makes use of an old frozen generator ($\mathcal{G}^{old}$) to create low quality reconstruction examples. We also pro-

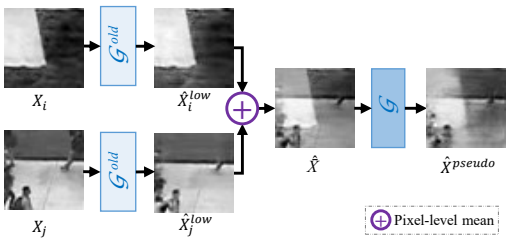



Figure 3: Our proposed pseudo-anomaly module. A pseudo-anomaly $\hat{\tilde{X}}$ is created by regenerating two arbitrary training images through $\mathcal{G}^{old}$ followed by a pixel-level mean. Finally, $\hat{X}^{pseudo}$ is created as $\mathcal{G}(\hat{\tilde{X}})$ to mimic the regeneration behavior of $\mathcal{G}$ for anomalous inputs.

pose a pseudo-anomaly module to assist $\mathcal{D}$ in learning the behavior of $\mathcal{G}$ in the case of unusual or anomalous input, which we found out to be very useful towards the robustness of our approach (Table 4). The overall purpose of our proposed framework is to alter the learning paradigm of $\mathcal{D}$ from distinguishing between real and fake to differentiating between good and bad reconstructions. This way, the discriminator gets aligned with the conventional philosophy of generative one-class learning models in which the reconstruction quality of the data from a known class is better than the data from unknown or anomaly classes.

### 3.2. Training

The training of our model is carried out in two phases (see Figure 2). Phase one is similar to the common practices in training an adversarial one-class classifier [41, 17, 42, 36]. $\mathcal{G}$ tries to regenerate real-looking fake data which is then fed into $\mathcal{D}$ along with real data. The $\mathcal{D}$ learns to discriminate between real and fake data, success or failure of which then becomes a supervision signal for $\mathcal{G}$. This training is carried out until $\mathcal{G}$ starts to create real looking images with a reasonably low reconstruction loss. Overall, phase one minimizes the following loss function:

$$\mathcal{L} = \mathcal{L}_{\mathcal{G}+\mathcal{D}} + \lambda \mathcal{L}_R, \tag{2}$$

where $\mathcal{L}_{\mathcal{G}+\mathcal{D}}$ is the loss function of our joint training objective defined in Equation 1, $\mathcal{L}_R = ||X - \mathcal{G}(\tilde{X})||^2$ is the reconstruction loss, and $\lambda$ is a weighing hyperparameter. Additionally, as phase one progresses, we save a low-epoch generator model ($\mathcal{G}^{old}$) for later use in phase two of the training. Deciding which low epoch to be used can be an intuitive selection based on the quality of regeneration. Obviously, we want $\mathcal{G}^{old}$ to generate low quality images compared to a trained $\mathcal{G}$. However, it is not necessary to select any specific epoch number for this generator. We will prove this empirically in Section 4 by showing that the final convergence of our model is not dependent on a strict selection of the epoch numbers and that various generic settings are possible to obtain a $\mathcal{G}^{old}$.

Phase two of the training is where we make use of the frozen models $\mathcal{G}^{old}$ and $\mathcal{G}$ to update $\mathcal{D}$. This way $\mathcal{D}$ starts learning to discriminate between good and bad quality reconstructions, hence becoming suitable for one-class classification problems such as anomaly detection.

Details of the phase two training are discussed next:

**Goal.** The essence of phase two training is to provide examples of good quality and bad quality reconstructions to $\mathcal{D}$, with a purpose of making it learn about the kind of output that $\mathcal{G}$ would produce in the case of an unusual input. The training is performed for just a few iterations since the already trained $\mathcal{D}$ converges quickly. A detailed study on this is added in Section 4.

**Good quality examples.** $\mathcal{D}$ is provided with real data ($X$), which is the best possible case of reconstruction, and the actual high quality reconstructed data ($\hat{X} = \mathcal{G}(X)$) produced by the trained $\mathcal{G}$ as an example of good quality examples.

**Bad quality examples.** Examples of low quality reconstruction ($\hat{X}^{low}$) are generated using $\mathcal{G}^{old}$. In addition, a pseudo-anomaly module, shown in Figure 3, is formulated with a combination of $\mathcal{G}^{old}$ and the trained $\mathcal{G}$, which simulates examples of reconstructed pseudo-anomalies ($\hat{X}^{pseudo}$).

**Pseudo anomaly creation.** Given two arbitrary images $X_i$ and $X_j$ from the training dataset, a pseudo anomaly image $\hat{\tilde{X}}$ is generated as:

$$\hat{\tilde{X}} = \frac{\mathcal{G}^{old}(X_i) + \mathcal{G}^{old}(X_j)}{2} = \frac{\hat{X}_i^{low} + \hat{X}_j^{low}}{2}, \text{where } i \neq j. \tag{3}$$

This way, the resultant image can contain diverse variations such as shadows and unusual shapes, which are completely unknown to both $\mathcal{G}$ and $\mathcal{D}$ models. Finally, as the last step in our pseudo-anomaly module, in order to mimic the behavior of $\mathcal{G}$ when it gets unusual data as input, $\hat{\tilde{X}}$ is then reconstructed using $\mathcal{G}$ to obtain $\hat{X}^{pseudo}$:

$$\hat{X}^{pseudo} = \mathcal{G}(\hat{\tilde{X}}). \tag{4}$$

Example images at each intermediate step can be seen in Figures 3 and 4.

**Tweaking the objective function.** The model in phase two of the training takes the form:

$$\max_{\mathcal{D}} \Big( \alpha \mathbb{E}_X[\log(1 - \mathcal{D}(X))] + (1-\alpha)\mathbb{E}_{\hat{X}}[\log(1 - \mathcal{D}(\hat{X}))] + \beta \mathbb{E}_{\hat{X}^{low}}[\log(\mathcal{D}(\hat{X}^{low}))] + (1-\beta)\mathbb{E}_{\hat{X}^{pseudo}}[\log(\mathcal{D}(\hat{X}^{pseudo}))] \Big), \tag{5}$$

where $\alpha$ and $\beta$ are the trade-off hyperparameters.

Quasi ground truth for the discriminator in phase one training is defined as:

$$GT_{phase\_one} = \begin{cases} 0 & \text{if input is } X, \\ 1 & \text{if input is } \hat{X}. \end{cases} \quad (6)$$

However, for phase two training, it takes the form:

$$GT_{phase\_two} = \begin{cases} 0 & \text{if input is } X \text{ or } \hat{X}, \\ 1 & \text{if input is } \hat{X}^{low} \text{ or } \hat{X}^{pseudo}. \end{cases} \quad (7)$$

### 3.3. Testing

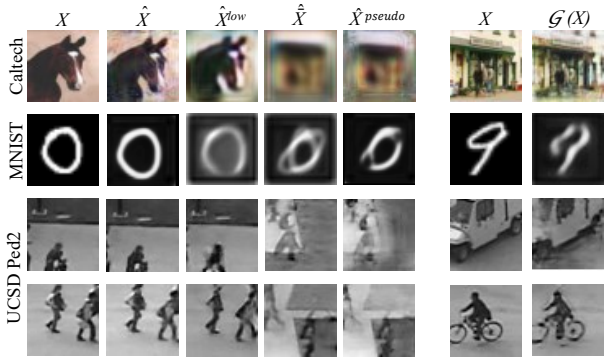
At test time, as shown in Figure 2, only $\mathcal{G}$ and $\mathcal{D}$ are utilized for one-class classification (OCC). Final classification decision for an input image $X$ is given as:

$$OCC = \begin{cases} \text{normal class} & \text{if } \mathcal{D}(\mathcal{G}(X)) < \tau, \\ \text{anomaly class} & \text{otherwise.} \end{cases} \quad (8)$$

where $\tau$ is a predefined threshold.

## 4. Experiments

The evaluation of our OGNet framework on three different datasets is reported in this section. Detailed analysis of the performance and its comparison with the state-of-the-art methodologies is also reported. In addition, we provide extensive discussion and ablation studies to show the stability as well as the significance of our proposed scheme. In order to keep the experimental setup consistent with the existing works [22, 57, 25, 51, 11, 45, 40, 10, 44, 34, 35, 41, 13, 7, 28, 27], we tested our method for the detection of outlier images as well as video anomalies.




(a) Training images (b) Test images

Figure 4: Example images from different stages of our framework. (a) Left to right: Original image ($X$), high quality reconstructed ($\hat{X}$), low quality reconstructed ($\hat{X}^{low}$), pseudo anomaly ($\tilde{\hat{X}}$), pseudo anomaly reconstructed ($\hat{X}^{pseudo}$). (b) Left column shows outlier / anomaly examples whereas right column shows respective regenerated outputs $\mathcal{G}(X)$.

**Evaluation criteria.** Most of our results are formulated based on area under the curve (AUC) computed at frame level due to its popularity in related works [48, 25, 28, 11, 23, 35, 24, 7, 34, 46, 10, 22, 52, 27, 57, 13, 58, 41]. Nevertheless, following the evaluation methods adopted in [47, 20, 54, 33, 21, 55, 41, 38, 35, 36, 52, 39, 40] we also report $F_1$ score and Equal Error Rate (EER) of our approach.

**Parameters and implementation details.** Our implementation is done in PyTorch [29] and the source code is provided at *https://github.com/xaggi/OGNet*. Phase one of the training in our reports is performed from 20 to 30 epochs. These numbers are chosen because the baseline shows high performance peaks within this range (Figure 1). We train on Adam [15] with the learning rate of generator and discriminator in all these epochs set to $10^{-3}$ and $10^{-4}$, respectively. Phase two of the training is done for 75 iterations with the learning rate of the discriminator reduced to half. $\lambda, \alpha,$ and $\beta$ are set to 0.2, 0.1, 0.001, respectively. Until stated otherwise, default settings of our experiments are set to the aforementioned values. However, for the detailed evaluation provided in a later part of this section, we also conducted experiments and reported results on a range of epochs and iterations for both phases of the training, respectively. Furthermore, until specified otherwise, we pick the generator after $1^{st}$ epoch and freeze it as $\mathcal{G}^{old}$. This selection is arbitrary and solely based on the intuition explained in Section 3. Additionally, in a later part of this section, we also present a robust and generic method to formulate $\mathcal{G}^{old}$ without any need of handpicking an epoch.

### 4.1. Datasets

**Caltech-256.** This dataset [9] contains a total of 30,607 images belong to 256 object classes and one 'clutter' class. Each category has different number of images, as low as 80 and as high as 827. In order to perform our experiments, we used the same setup as described in previous works [47, 20, 54, 33, 21, 55, 41]. In a series of three experiments, at most 150 images belong to 1, 3, and 5 randomly chosen classes are defined as training (inlier) data. Outlier images for test are taken from the 'clutter' class in such a way that each experiment has exactly 50% ratio of outliers and inliers.

**MNIST.** This dataset [18] consists of 60,000 handwritten digits from 0 to 9. The setup to evaluate our method on this dataset is also kept consistent with the previous works [51, 3, 41]. In a series of experiments, each category of digits is individually taken as inliers. Whereas, randomly sampled images of the other categories with a proportion of 10% to 50% are taken as outliers.

**USCD Ped2.** This dataset [4] comprises of 2,550 frames in 16 training and 2,010 frames in 12 test videos. Each frame is of $240 \times 360$ pixels resolution. Pedestrians dominate most

| | DPCP[47] | REAPER[20] | OutlierPersuit[54] | CoP[33] | LRR[21] | R-graph[55] | ALOCC[41] | Ours |
|---|---|---|---|---|---|---|---|---|
| AUC | 78.3% | 81.6% | 83.7% | 90.5% | 90.7% | 94.8% | 94.2% | **98.2%** |
| $F_1$ | 78.5% | 80.8% | 82.3% | 88.0% | 89.3% | 91.4% | 92.8% | **95.1%** |
| AUC | 79.8% | 79.6% | 78.8% | 67.6% | 47.9% | 92.9% | 93.8% | **97.7%** |
| $F_1$ | 77.7% | 78.4% | 77.9% | 71.8% | 67.1% | 88.0% | 91.3% | **91.5%** |
| AUC | 67.6% | 65.7% | 62.9% | 48.7% | 33.7% | 91.3% | 92.3% | **98.1%** |
| $F_1$ | 71.5% | 71.6% | 71.1% | 67.2% | 66.7% | 85.8% | 90.5% | **92.8%** |

Table 1: AUC and $F_1$ score performance comparison of our framework on Caltech-256 [9] with the other state of the art methods. Following the existing work [55], each subgroup of rows from top to bottom shows evaluation scores on inliers coming from 1, 3, and 5 different random classes respectively (best performance as bold and second best as underlined).

of the frames whereas anomalies include skateboards, vehicles, bicycles, etc. Similar to [48, 25, 28, 11, 23, 35, 24, 7, 34, 46, 10, 22, 52, 27, 57, 13, 58, 41], frame-level AUC and EER metrics are adopted to evaluate performance on this dataset.

### 4.2. Outlier Detection in Images

One of the significant applications of a one-class learning algorithm is outlier detection. In this problem, objects belonging to known classes are treated as inliers based on which the model is trained. Other objects that do not belong to these classes are treated as outliers, which the model is supposed to detect based on its training. Results of the experiments conducted using Caltech-256 [9] and MNIST [18] datasets are reported and comparisons with state-of-the-art outlier detection models [14, 55, 51, 41, 47, 20, 54, 33, 21] are provided.

**Results on Caltech-256.** Figure 4b shows outlier examples reconstructed using $\mathcal{G}$. It is interesting to observe that although the generated images are of reasonably good quality, our model still depicts superior results in terms of $F_1$ score and area under the curve (AUC), as listed in Table 1, which demonstrates that our model is robust to the *over-training* of $\mathcal{G}$.

**Results on MNIST.** As it is a well-studied dataset, various outlier detection related works use MNIST as a stepping-stone to evaluate their approaches. Following [51, 3, 41], we also report $F_1$ score as an evaluation metric of our method on this dataset. A comparison provided in Figure 5 shows that our approach performs robustly to detect outliers even when the percentage of outliers is increased. An insight of the performance improvement by our approach is shown in Figure 6. It can be observed that as the phase two training continues, score distribution of inliers and outliers output by our network smoothly distributes to a wider range.

### 4.3. Anomaly Detection in Videos

One-class classifiers are finding their best applications in the domain of anomaly detection for surveillance purposes [45, 48, 6, 57, 36, 35]. However, this task is more complicated than the outlier detection because of the involvement of moving objects, which cause variations in appearance.

**Experimental setup.** Each frame $I$ of the Ped2 dataset is divided into grayscale patches $X_I = \{X_1, X_2, ..., X_n\}$ of size $45 \times 45$ pixels. Normal videos, which only contain

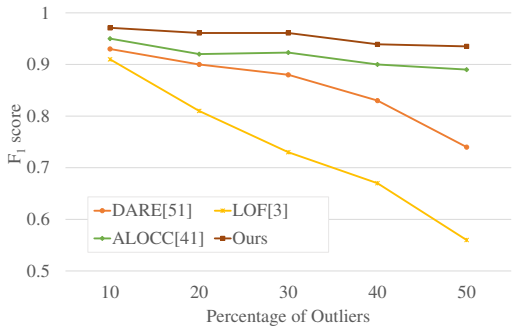



Figure 5: $F_1$ score results on MNIST dataset. Compared to state-of-the-art, our method retains superior performance even with an increased percentage of outliers at test time.

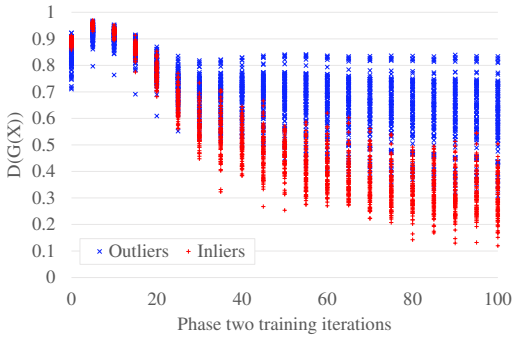



Figure 6: Anomaly score distribution on MNIST dataset over various training iterations of our framework. Divisibility of inliers and outliers is improved significantly as phase two of the training proceeds.

| RE[38] | AbnormalGAN[35] | Ravanbakhsh[36] | Dan Xu[52] |
|---|---|---|---|
| 15% | 13% | 14% | 17% |
| Sabokrou[39] | Deep-cascade[40] | ALOCC[41] | Ours |
| 19% | <u>9%</u> | 13% | **7%** |

Table 2: EER results comparison with existing works on UCSD Ped2 dataset. Lower numbers mean better results.

scenes of walking pedestrians, are used to extract training patches. Test patches are extracted from abnormal videos which contain abnormal as well as normal scenes. In order to remove unnecessary inference of the patches, a motion detection criteria based on frame difference is set to discard patches without motion. A maximum of all patch-level anomaly scores is declared as the frame-level anomaly score of that particular frame as:

$$A_I = \max_X \mathcal{D}(\mathcal{G}(X)), \text{where } X \in X_I \tag{9}$$

**Performance evaluation.** Frame-level AUC and EER are the two evaluation metrics used to compare our approach with a series of existing works [48, 25, 28, 11, 23, 35, 24, 7, 34, 46, 10, 22, 52, 27, 57, 13, 58, 41] published within last 5 years. The corresponding results provided in Table 2 and Table 3 show that our method outperforms recent state-of-the-art methodologies in the task of anomaly detection. Comparing with the baseline, our approach achieves an absolute gain of 5.2% in terms of AUC. Examples of the reconstructed patches are provided in Figure 4. As shown in Figure 4b, although $\mathcal{G}$ generates noticeably good reconstructions of anomalous inputs, due to the presence of our proposed pseudo-anomaly module, $\mathcal{D}$ gets to learn the underlying patterns of reconstructed anomalous images. This is why, in contrast to the baseline, our framework provides consistent performance across a wide range of training epochs (Figure 1).

| Method | AUC | Method | AUC |
|---|---|---|---|
| Unmasking[48] | 82.2% | TSC[25] | 92.2% |
| HybridDN[28] | 84.3% | FRCN action[11] | 92.2% |
| Liu et al[23] | 87.5% | AbnormalGAN[35] | 93.5% |
| ConvLSTM-AE[24] | 88.1% | MemAE[7] | 94.1% |
| Ravanbakhsh et al[34] | 88.4% | GrowingGas[46] | 94.1% |
| ConvAE[10] | 90% | FFP[22] | 95.4% |
| AMDN[52] | 90.8% | ConvAE+UNet[27] | 96.2% |
| Hashing Filters[57] | 91% | STAN[19] | 96.5% |
| AE_Conv3D[58] | 91.2% | Object-centric[13] | <u>97.8%</u> |
| Baseline | 92.9% | Ours | **98.1%** |

Table 3: Frame-level AUC comparison on UCSD Ped2 dataset with state-of-the-art works published in last 5 years. Best and second best performances are highlighted as bold and underlined, respectively.

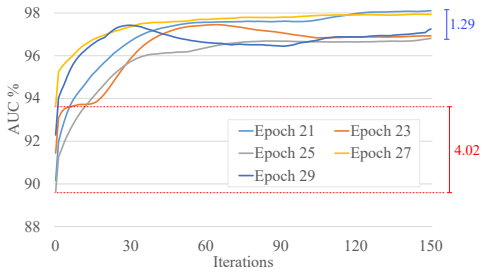


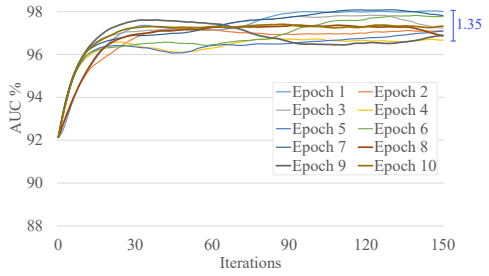

Figure 7: The plot shows frame level AUC performance of our phase two training, starting after different epochs of phase one (baseline) training. The model after phase two training shows significantly less variance than baseline/phase one.

### 4.4. Discussion

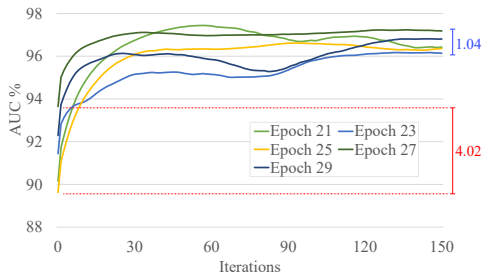
**When to stop phase one training?** The convergence of our framework is not strictly dependent on phase one training. Figure 7 shows the AUC performance of phase two training applied after various epochs of phase one on Ped2 dataset [4]. Values plotted at iterations = 0, representing the performance of the baseline, show a high variance. Interestingly, it can be seen that after few iterations into phase two training of our proposed approach, the model starts to converge better. Irrespective of the initial epoch in phase one training, models converged successfully showing consistent AUC performances.

**When to stop phase two training?** As seen in Figure 7 and Figure 8, it can be observed that once a specific model is converged, further iterations do not deteriorate its performance. Hence, a model can be trained for any number of iterations as deemed necessary.

**Which low epoch generator is better?** For the selection of $\mathcal{G}^{old}$, as mentioned earlier, the generator after the $1^{st}$ epoch of training was arbitrarily chosen in our experiments. This selection is intuitive and mostly based on the fact that the generator has seen all dataset once. In addition, we visually observed that after first epoch, although the generator was capable of reconstructing its input, the quality was not 'good enough', which is a suitable property for $\mathcal{G}^{old}$ in our model. However, this way of selection is not a generalized solution across various datasets. Hence, to investigate the matter further, we evaluate a range of low epoch numbers as candidates for $\mathcal{G}^{old}$. The baseline epoch of $\mathcal{G}$ is kept fixed throughout this experiment. Results in Figure 8a show that irrespective of the low epoch number chosen as $\mathcal{G}^{old}$, the model converges and achieves state-of-the-art or comparable AUC. In pursuit of another more systematic way to obtain $\mathcal{G}^{old}$, we also explored the possibility of using average parameters of all previous $\mathcal{G}$ models. Hence, for each given epoch of the baseline that we pick as $\mathcal{G}$, a $\mathcal{G}^{old}$ is formu-




(a) Experiments with the $\mathcal{G}^{old}$ taken across first 10 epochs, where $\mathcal{G}$ is kept fixed.




(b) Experiments with the $\mathcal{G}^{old}$ obtained at various arbitrary epochs by averaging the parameters of generators from all previous epochs.

Figure 8: Results from a series of experiments on UCSD Ped2 dataset show that our framework is not dependent on a strict choice of epoch number for $\mathcal{G}^{old}$. In (a), various $\mathcal{G}^{old}$ selected at a varied range of epochs are experimented with a fixed $\mathcal{G}$. In (b), an average of parameters from all past generators is taken as $\mathcal{G}^{old}$.

lated by taking an average of all previous $\mathcal{G}$ models until that point. The results plotted in Figure 8b show that such $\mathcal{G}^{old}$ also depicts comparable performances. Note that this formulation completely eradicates the need of handpicking a specific epoch number for $\mathcal{G}^{old}$, thus making our formulation generic towards the size of a training dataset.

### 4.5. Ablation

Ablation results of our framework on UCSD Ped2 dataset [4] are summarized in Table 4. As shown, while each input component of our training model (i.e. real images $X$, high quality reconstructions $\hat{X}$, low quality reconstructions $\hat{X}^{low}$, and pseudo anomaly reconstructions $\hat{X}^{pseudo}$) contributes towards a robust training, removing any of these at a time still shows better performance than the baseline. One interesting observation can be seen in the fourth column of the phase two training results. In this case, the performance is measured after we remove the last step of pseudo-anomaly module, which is responsible for providing regenerated pseudo-anomaly ($\hat{\hat{X}}^{pseudo}$) through $\mathcal{G}$, as in Equation 4. Hence, by removing this part, the

| | Phase one | Phase two | | | | |
|---|---|---|---|---|---|---|
| $X$ | ✓ | - | ✓ | ✓ | ✓ | ✓ |
| $\hat{X}$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\hat{X}^{low}$ | - | ✓ | ✓ | - | ✓ | ✓ |
| $\hat{X}^{pseudo}$ | - | - | - | ✓ | - | ✓ |
| $\hat{\hat{X}}$ as $\hat{X}^{pseudo}$ | - | - | - | - | ✓ | - |
| AUC | 92.9% | 94.4% | 95.1% | 95.9% | 88.5% | **98.1%** |

Table 4: Frame-level AUC performance ablation of our framework on UCSD Ped2 dataset.

fake anomalies ($\hat{\hat{X}}$) obtained using Equation 3 are channeled directly to the discriminator as one of the two sets of bad reconstruction examples. With this configuration, the performance deteriorates significantly (i.e. 9.6% drop in the AUC). The model shows even worse performance than the baseline after phase one training. This shows the significance of our proposed pseudo-anomaly module. Once pseudo-anomalies are created within the module, it is necessary to obtain a regeneration result of these by inferring $\mathcal{G}$. This helps $\mathcal{D}$ to learn the underlying patterns of reconstructed anomalous images, which results in a more robust anomaly detection model.

## 5. Conclusion

This paper presents an adversarially learned approach in which both the generator ($\mathcal{G}$) and the discriminator ($\mathcal{D}$) are utilized to perform a stable and robust anomaly detection. A unified $\mathcal{G}$ and $\mathcal{D}$ model employed towards such problems often produces unstable results due to the adversary. However, we attempted to tweak the basic role of the discriminator from distinguishing between real and fake to discriminating between good and bad quality reconstructions, a formulation that aligns well with the philosophy of conventional anomaly detection using generative networks. We also propose a pseudo-anomaly module which is employed to create fake anomaly examples from normal training data. These fake anomaly examples help $\mathcal{D}$ to learn about the behavior of $\mathcal{G}$ in the case of unusual input data.

Our extensive experimentation shows that the approach not only generates stable results across a wide range of training epochs but also outperforms a series of state-of-the-art methods [48, 25, 28, 11, 23, 35, 24, 7, 34, 46, 10, 22, 52, 27, 57, 13, 58, 41] for outliers and anomaly detection.

## 6. Acknowledgment

This work was supported by the ICT R&D program of MSIP/IITP. [2017-0-00306, Development of Multimodal Sensor-based Intelligent Systems for Outdoor Surveillance Robots]. Also, we thank HoChul Shin, Ki-In Na, Hamza Saleem, Ayesha Zaheer, Arif Mahmood, and Shah Nawaz for the discussions and support in improving our work.

## References


[1] Arslan Basharat, Alexei Gritai, and Mubarak Shah. Learning object motion patterns for anomaly detection and improved object detection. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[2] Francesco Bergadano. Keyed learning: An adversarial learning framework—formalization, challenges, and anomaly detection applications. *ETRI Journal*, 41(5):608–618, 2019.

[3] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM, 2000.

[4] Antoni Chan and Nuno Vasconcelos. Ucsd pedestrian dataset. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(5):909–926, 2008.

[5] Xinyi Cui, Qingshan Liu, Mingchen Gao, and Dimitris N Metaxas. Abnormal detection using interaction energy potentials. In *CVPR 2011*, pages 3161–3167. IEEE, 2011.

[6] Jayanta Kumar Dutta and Bonny Banerjee. Online detection of abnormal events using incremental coding length. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[7] 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.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[9] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

[10] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742, 2016.

[11] Ryota Hinami, Tao Mei, and Shin'ichi Satoh. Joint detection and recounting of abnormal events by learning deep generic knowledge. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3619–3627, 2017.

[12] Rui Hou, Chen Chen, and Mubarak Shah. Tube convolutional neural network (t-cnn) for action detection in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5822–5831, 2017.

[13] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2019.

[14] Jaechul Kim and Kristen Grauman. Observe locally, infer globally: a space-time mrf for detecting abnormal activities with incremental updates. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2928. IEEE, 2009.

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] Louis Kratz and Ko Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1446–1453. IEEE, 2009.

[17] Wallace Lawson, Esube Bekele, and Keith Sullivan. Finding anomalies with generative adversarial networks for a patrolbot. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 12–13, 2017.

[18] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. Mnist handwritten digit database. *ATT Labs [Online].Available: http://yann. lecun. com/exdb/mnist*, 2010.

[19] Sangmin Lee, Hak Gu Kim, and Yong Man Ro. Stan: Spatio-temporal adversarial networks for abnormal event detection. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1323–1327. IEEE, 2018.

[20] Gilad Lerman, Michael B McCoy, Joel A Tropp, and Teng Zhang. Robust computation of linear models by convex relaxation. *Foundations of Computational Mathematics*, 15(2):363–410, 2015.

[21] Guangcan Liu, Zhouchen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. In *ICML*, volume 1, page 8, 2010.

[22] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection–a new baseline. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6536–6545, 2018.

[23] Yusha Liu, Chun-Liang Li, and Barnabás Póczos. Classifier two sample test for video anomaly detections. In *BMVC*, page 71, 2018.

[24] Weixin Luo, Wen Liu, and Shenghua Gao. Remembering history with convolutional lstm for anomaly detection. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 439–444. IEEE, 2017.

[25] 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.

[26] Gérard Medioni, Isaac Cohen, François Brémond, Somboon Hongeng, and Ramakant Nevatia. Event detection and analysis from video streams. *IEEE Transactions on pattern analysis and machine intelligence*, 23(8):873–889, 2001.

[27] 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.

[28] Trong Nguyen Nguyen and Jean Meunier. Hybrid deep network for anomaly detection. *arXiv preprint arXiv:1908.06347*, 2019.

[29] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[30] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[31] Claudio Piciarelli, Christian Micheloni, and Gian Luca Foresti. Trajectory-based anomalous event detection. *IEEE Transactions on Circuits and Systems for video Technology*, 18(11):1544–1554, 2008.

[32] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[33] Mostafa Rahmani and George K Atia. Coherence pursuit: Fast, simple, and robust principal component analysis. *IEEE Transactions on Signal Processing*, 65(23):6260–6275, 2017.

[34] Mahdyar Ravanbakhsh, Moin Nabi, Hossein Mousavi, Enver Sangineto, and Nicu Sebe. Plug-and-play cnn for crowd motion analysis: An application in abnormal event detection. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1689–1698. IEEE, 2018.

[35] Mahdyar Ravanbakhsh, Moin Nabi, Enver Sangineto, Lucio Marcenaro, Carlo Regazzoni, and Nicu Sebe. Abnormal event detection in videos using generative adversarial nets. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 1577–1581. IEEE, 2017.

[36] Mahdyar Ravanbakhsh, Enver Sangineto, Moin Nabi, and Nicu Sebe. Training adversarial discriminators for cross-channel abnormal event detection in crowds. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1896–1904. IEEE, 2019.

[37] Huamin Ren, Weifeng Liu, Søren Ingvor Olsen, Sergio Escalera, and Thomas B Moeslund. Unsupervised behavior-specific dictionary learning for abnormal event detection. In *BMVC*, pages 28–1, 2015.

[38] Mohammad Sabokrou, Mahmood Fathy, and Mojtaba Hoseini. Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder. *Electronics Letters*, 52(13):1122–1124, 2016.

[39] Mohammad Sabokrou, Mahmood Fathy, Mojtaba Hoseini, and Reinhard Klette. Real-time anomaly detection and localization in crowded scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 56–62, 2015.

[40] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, and Reinhard Klette. Deep-cascade: Cascading 3d deep neural networks for fast anomaly detection and localization in crowded scenes. *IEEE Transactions on Image Processing*, 26(4):1992–2004, 2017.

[41] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3379–3388, 2018.

[42] 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.

[43] Firas Shama, Roey Mechrez, Alon Shoshan, and Lihi Zelnik-Manor. Adversarial feedback loop. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[44] Sorina Smeureanu, Radu Tudor Ionescu, Marius Popescu, and Bogdan Alexe. Deep appearance features for abnormal behavior detection in video. In *International Conference on Image Analysis and Processing*, pages 779–789. Springer, 2017.

[45] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6479–6488, 2018.

[46] Qianru Sun, Hong Liu, and Tatsuya Harada. Online growing neural gas for anomaly detection in changing surveillance scenes. *Pattern Recognition*, 64:187–201, 2017.

[47] Manolis C Tsakiris and René Vidal. Dual principal component pursuit. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 10–18, 2015.

[48] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. Unmasking the abnormal events in video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2895–2903, 2017.

[49] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

[50] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014.

[51] Yan Xia, Xudong Cao, Fang Wen, Gang Hua, and Jian Sun. Learning discriminative reconstructions for unsupervised outlier removal. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1511–1519, 2015.

[52] Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. Learning deep representations of appearance and motion for anomalous event detection. *arXiv preprint arXiv:1510.01553*, 2015.

[53] Dan Xu, Yan Yan, Elisa Ricci, and Nicu Sebe. Detecting anomalous events in videos by learning deep representations of appearance and motion. *Computer Vision and Image Understanding*, 156:117–127, 2017.

[54] Huan Xu, Constantine Caramanis, and Sujay Sanghavi. Robust pca via outlier pursuit. In *Advances in Neural Information Processing Systems*, pages 2496–2504, 2010.

[55] Chong You, Daniel P Robinson, and René Vidal. Provable self-representation based outlier detection in a union of subspaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3395–3404, 2017.

[56] Tianzhu Zhang, Hanqing Lu, and Stan Z Li. Learning semantic scene models by object classification and trajectory clustering. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1940–1947. IEEE, 2009.

[57] Ying Zhang, Huchuan Lu, Lihe Zhang, Xiang Ruan, and Shun Sakai. Video anomaly detection based on locality sensitive hashing filters. *Pattern Recognition*, 59:302–311, 2016.

[58] Yiru Zhao, Bing Deng, Chen Shen, Yao Liu, Hongtao Lu, and Xian-Sheng Hua. Spatio-temporal autoencoder for video anomaly detection. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1933–1941. ACM, 2017.