Latent Diffusion Based Multi-class Anomaly Detection

Chenxing Wang(✉) and Alireza Tavakkoli

University of Nevada, Reno, Reno, NV 89512, USA
{chenxingw,tavakkol}@unr.edu

Abstract. The unsupervised anomaly detection problem holds great importance but remains challenging to address due to the myriad of data possibilities in our daily lives. Currently, distinct models are trained for different scenarios. In this work, we introduce a reconstruction-based anomaly detection structure built on the Latent Space Denoising Diffusion Probabilistic Model (LDM). This structure effectively detects anomalies in multi-class situations. When normal data comprises multiple object categories, existing reconstruction models often learn identical patterns. This leads to the successful reconstruction of both normal and anomalous data based on these patterns, resulting in the inability to distinguish anomalous data. To address this limitation, we implemented the LDM model. Its process of adding noise effectively disrupts identical patterns. Additionally, this advanced image generation model can generate images that deviate from the input. We have further proposed a classification model that compares the input with the reconstruction results, tapping into the generative power of the LDM model. Our structure has been tested on the MNIST and CIFAR-10 datasets, where it surpassed the performance of state-of-the-art reconstruction-based anomaly detection models.

Keywords: Anomaly Detection · Diffusion Model

1 Introduction

The field of anomaly detection [2,17] has gained substantial popularity in recent years, as techniques in this domain are increasingly applied across various sectors. The advent of deep learning has significantly enhanced our capacity to represent complex data. This advancement facilitates improved feature representation for high-dimensional, graph, or spatial data in anomaly detection. Currently, we observe the utilization of anomaly detection in areas such as medical data analysis, risk management, and AI safety, to name a few. In most real-world applications, access to anomalous data is not feasible, and normal data often comprises various types of objects. For instance, in invasive species detection, access to anomaly data is limited, and the normal dataset includes different local animal species.

G. Bebis et al. (Eds.): ISVC 2023, LNCS 14361, pp. 487–498, 2023.
https://doi.org/10.1007/978-3-031-47969-4_38

Many contemporary anomaly detection algorithms are designed for one-class anomaly detection. In this approach, the model is trained on samples from a particular class. The model learns a probability density function that captures behavior for that specific class. Samples from other classes are considered anomalies, regardless of whether they belong to normal data or not. For multi-class anomaly detection, a model should learn a probability density function for all classes to delineate the boundaries of all normal data.

In this study, we aim to construct an unsupervised anomaly detection model capable of identifying anomalies across various normal object classes. Specifically, the training data consists of normal samples from several different object categories. During both training and inference processes, we do not have access to the category labels of any samples in the training data.

A commonly adopted methodology in anomaly detection involves the use of image or feature reconstruction. This approach assumes that a well-tuned model can consistently generate normal samples, even in the presence of potential anomalies in the input data. However, many widely used reconstruction networks often fail to meet the stringent requirements of this task. This failure is evidenced by an observed "identity shortcut" pattern. This shortcut leads to the direct replication of the input, potentially allowing for the accurate replication of even anomalous samples and, consequently, hampering their detection.

This challenge becomes more pronounced in contexts where the normal data distribution is inherently complex. When attempting to construct a unified model capable of reconstructing a broad range of objects, the model must endeavor to understand the joint distribution. Resorting to an "identity shortcut" might be a simpler path, but it compromises the model's effectiveness in anomaly detection.

The ever-increasing volume of digital data necessitates the development of sophisticated probabilistic models to handle inherent noise and distortion. Diffusion Denoising Probabilistic Models (DDPM), a unique category within these models, provide an innovative approach to the information bottleneck problem. Trained to systematically de-noise corrupted inputs, these models reshape the strategy for noise management. Unlike traditional models where the bottleneck is an intrinsic property, DDPM views the bottleneck as an externally adjustable feature during model inference. While previous studies have explored DDPMs as autoencoders with externally adjustable bottlenecks, none have harnessed this property for reconstruction-based anomaly detection. This paper aims to fill this void, delving into novel insights and methodologies to leverage DDPMs for enhanced anomaly detection.

In this work, we proposed a multi-class anomaly detection structure based on the LDM model. We examined the use of latent space within the DDPM framework and developed a classification model that utilizes the generative capabilities of the diffusion model. This is to determine whether the input and reconstructed image belong to the same category.

2 Related Work

2.1 Anomaly Detection

The anomaly detection problem has a long research history. Classic anomaly detection techniques include the Gaussian Distribution estimation method [22,26], Mahalanobis distance [13], mixture distribution [7] and nonparametric density estimation [5]. With the advancement of deep learning and machine learning techniques, recent anomaly detection models are now based on deep learning structures. For example, one-class classification-based methods such as v-SVC [21], and Support Vector Data Description (SVDD) [23] train a deep neural network to map a set of data instances into a sphere of minimal volume, assessing whether test samples conform to the training data. Knowledge distillation methods [28,29] assist in understanding the frequency of various underlying patterns in the data.

2.2 Reconstruction Models

The basic idea of reconstruction methods is to learn a model which optimized to reconstruct all normal data instance and detection the anomalous data instance by high reconstruction error.

The objective function of reconstruction models can be shown as $\phi(\theta) : X \rightarrow X$ which is a feature mapping from data to itself. It includes two steps, the encoding step

$$z = \phi_e\left(x; \theta_e\right) \tag{1}$$

and the decoding step

$$\hat{x} = \phi_d\left(z; \theta_d\right) \tag{2}$$

Autoencoder networks are the most frequently used algorithms in reconstruction models. They utilize various types of neural networks to encode the input data and then decode it for recovery. Originally, autoencoders were used for dimension reduction [11,24]. However, nowadays, they have become the most popular algorithms employed in anomaly detection [3,4,32]. The reconstruction loss function is used to learn the parameters of both networks. To represent the low-dimensional feature space, a bottleneck network is typically used.

To minimize the reconstruction error and detect anomalous data, features extracted in the latent space should be highly relevant to normal data instances. Only in this manner will the reconstruction error of an anomalous data instance be significantly higher than the reconstruction error of normal data. Consequently, the reconstruction error can be used as an anomaly score.

Several innovative types of autoencoders have been developed to enhance feature representations. The denoising autoencoder [27] s designed to train on corrupted data instances, making it robust against minor variations. The sparse autoencoder [16] aims to increase sparsity in the hidden layer, retaining only the top K most active units. The contractive autoencoder [18] is proposed to be robust against minor variations around its neighbors.

2.3 GAN Based Models

AnoGAN [20] is an example of the first usage of GAN in anomaly detection. Similar with the autoencoder, the training process is only focus on normal data instance. The main idea is that given input data instances x, try to find out z in the latent feature space of the generative network G in order to make $G(z)$ and x as similar as possible. The training of GAN in only normal data will let the generator learn the underlying distribution of normal data. Once an anomalous image is encoded, the reconstruction result will be a normal image generated by G. The difference between input image and reconstruction image will show the anomalous area.

Many other GAN based anomaly detection methods have been introduced based on different GAN architectures. For example, EBGAN [31] first introduced BiGAN architecture in anomaly detection based on the idea from [10], f-AnoGAN [20] which uses Wasserstein GAN [1] to replace the standard GAN in anomaly detection.

3 Method

3.1 Diffusion Models

The basic design of diffusion models are based on two Markov chains. Given any data $x_0 \sim q(x_0)$, the first Markov chain is called the forward chain, which transfer the data into noise. Standard Gaussian noise is typical choice when using the diffusion model because of its unique properties. The forward Markov chain uses T steps, with Gaussian noise added into the data for each step.

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I\right) \tag{3}$$

where $t = 1, 2, ..., T$ and $\beta \in [0, 1]$ denotes the noise variance schedule. From the equation above, given data x_0 and step t, we can get the distribution of a noise image

$$q(x_t \mid x_0) = \mathcal{N}\left(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t) I\right) \tag{4}$$

where here we use $\bar{\alpha}_t$ represent $\prod_{s=1}^{t}(1-\beta_s)$

The other Markov chain represents the reverse process, which begins from the standard Gaussian noise image and keeps adding small amount of noise in order to recover the input data. This process begins at the point

$$p(x_T) = \mathcal{N}(x_T; 0, I) \tag{5}$$

And small amount of Gaussian noise will be added onto the image step by step.

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{6}$$

where μ_θ and Σ_θ are the mean value and standard variation of the Gaussian noise added in each step. In order to reverse the forward process, we set $\Sigma_\theta(x_t, t) = \beta_t I$ and μ_θ should estimate $\frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon\right)$, thus we can set

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right) \tag{7}$$

In order to estimate $\epsilon_\theta(x_t, t)$, a U-net is built to minimize the objective function

$$L = E_{t\sim[1-T], x_0\sim q(x_0), \epsilon\sim N(0,I)}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right] \tag{8}$$

where $\epsilon \sim \mathcal{N}(0, I)$. From equation above, the U-net model is trained so that, given any input x_t, the output of the U-net model should be equal to $\mathcal{N}(0, I)$ In the inference process we can get

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right) + \beta_t z \tag{9}$$

3.2 Architecture with Latent Diffusion Models

As depicted in Fig. 1 our multi-class anomaly detection model comprises three components: a compression model, a diffusion network, and a classification network. The compression model compresses the image into a lower-dimensional space. The diffusion network reconstructs the latent space of normal data, while the classification network determines whether the input and output of the compression model belong to the same class. An anomaly class is detected if the input and output are classified into different classes.

Compression Model. The input of compression model is the original image x, the compression procedure can be expressed as $z = E(x)$, and the decode procedure can be denoted as $x' = D(z)$. The architecture of our compression model, based on work [8], trains an autoencoder considering both perceptual loss and adversarial objectives. Therefore, during the image compression, it accounts for not only pixel-wise information but also the composition of image parts from a codebook constructed by the image.

Utilizing a compression model before the diffusion model in anomaly detection provides several benefits:

i The computational complexity during the training of the diffusion model is reduced since this model operates in the latent space.
ii The latent space prevents the DDPM reconstruction structure from encountering the "identity shortcut" issue, which arises when the network consistently produces a copy of the input data.
iii A compression model allows for flexibility in choosing an appropriate latent space for the DDPM process. Typically, both the input and output of the autoencoder should depict the original image. In this study, however, we've

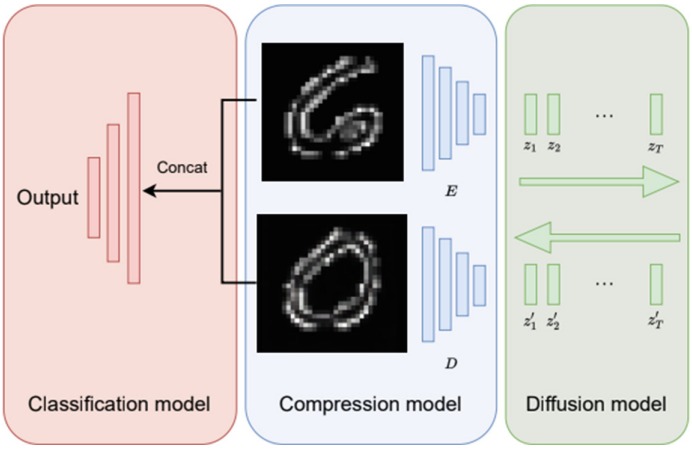

Fig. 1. An overview of our framework, which comprises a compression model, a diffusion model, and a classification model. The compression model constructs an encoder and decoder to create a latent space. The diffusion model continuously adds noise during the forward process and estimates the input latent data in the reverse process. The classification model determines whether the input image and the reversed input image belong to the same class.

also experimented with transforming the autoencoder's output into an edge label, a change that can reduce the propensity for the structure to fall into the "identity shortcut" issue.

Diffusion Model. The input of the diffusion model is the latent space vector z from compression model. Then following the forward process of the DDPM from Eq. 4. Given any time $t \in [0, T]$, the latent space z_t can be calculate by

$$z_t = z_0\sqrt{\bar{\alpha}_t} + \epsilon_t\sqrt{1 - \bar{\alpha}_t} \tag{10}$$

where $\epsilon_t \sim \mathcal{N}(0, I)$.

With the t becomes larger and larger, more and more Gaussian noise is added into the image and the latent vector z_t loose its original spatial structure and looks near the Gaussian noise. In the reverse process, we can follow the Eq. 9. We need to train the U-net model $\epsilon(x, t)$ in order to let it predict the noise ϵ.

In our anomaly task, the result of DDPM reconstruction should be the same with input latent vector z if it is a latent representation from normal data. In practice, the reconstruction could keep the similarity of input if the reverse process begins from time t. As the choice of t becomes larger, the output would become more random and lose the ability to keep the input similarity even it comes from a normal data instance.

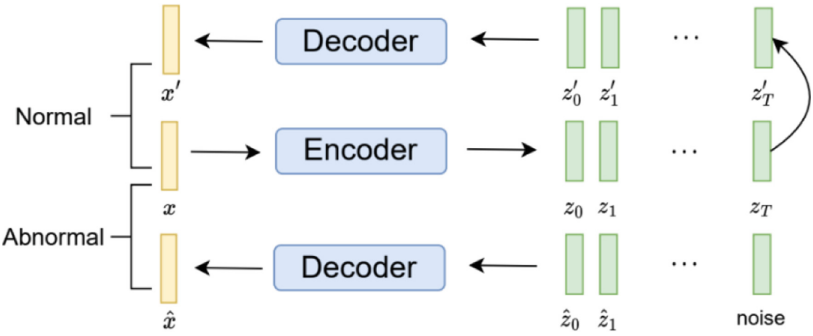

Fig. 2. Illustration of the training process within the classification model. If the reversed input equals the forward diffusion process, the input image and reversed image are considered to belong to the same class. If the reversed input equals random noise, the input image and reversed image are considered to belong to different classes.

Classification Model. As illustrated in Fig. 2 the classification network determines whether the input image and the reconstructed image belong to the same category. The classification network takes in a channel-wise concatenation of the input image x_0 and the reconstruction estimation $\hat{x}_0$. A CNN-based architecture is employed for this classification network. One challenge in this classification is that we only have access to normal data instances, providing us with only positive labels. To obtain negative labels, for each training data input x_0 we reconstruct x'_0 by reversing the DDPM process from a random noise latent space.

4 Experiment

4.1 Datasets and Metrics

Datasets. This research primarily employs two datasets: MNIST [14] and CIFAR-10 [12]. MNIST is an extensive database of handwritten digits, ranging from 0 to 9, with images sized at 28×28 pixels. CIFAR-10 is a widely-recognized image classification dataset containing ten distinct object categories, each image being $32 \times 32 \times 3$ in size.

For anomaly detection studies associated with both datasets, the prevalent approach is the one-versus-rest scenario. In this, one object category is treated as normal data, while the others are deemed anomalies. Notably, prior literature hasn't explored the MNIST dataset in a many-versus-one scenario. In this setting, models are trained on nine categories as normal data, with the remaining category considered anomalous. For the CIFAR-10 dataset, Semantic AD [6] has tackled the many-versus-one scenario using transfer learning. Meanwhile, UniAD [30] employed an embedding method in a many-versus-many context.

In our study, we explore the many-versus-one setting for the MNIST dataset and delve into the many-versus-many scenario for the CIFAR-10 dataset, employing a fundamentally distinct approach.

Metrics. In this paper, all the experiments are using the Area Under the Receiver Operating Curve (AUROC) as the evaluation metric. AUROC scored is defined based on False Positive Rate (FPR) and True Positive Rate (TPR).

$$FPR = \frac{FP}{FP + TN} \tag{11}$$

$$TPR = \frac{TP}{TP + FN} \tag{12}$$

where FP represents false positive, TN represents true negative, TP represents true positive and FN represents false negative.

4.2 Reconstruction Selection

Reconstruction-based anomaly detection algorithms are one of the most researched topics in anomaly detection. Numerous studies [15,20,27] have been developed in recent years. The primary assumption behind using a reconstruction model is that the reconstruction distribution should closely match the normal distribution. This assumption rarely fails under the one-versus-rest setting because learning the distribution of one category is typically straightforward. However, in a many-versus-one setting or many-versus-many setting, normal data includes different object categories, making the distribution challenging to describe. Often, the reconstruction-based model falls victim to the "identity shortcut" issue, where the output always attempts to replicate the input, regardless of the context.

Diffusion models show immense potential in image generation. Because the forward process of diffusion involves adding noise to the image, the reverse

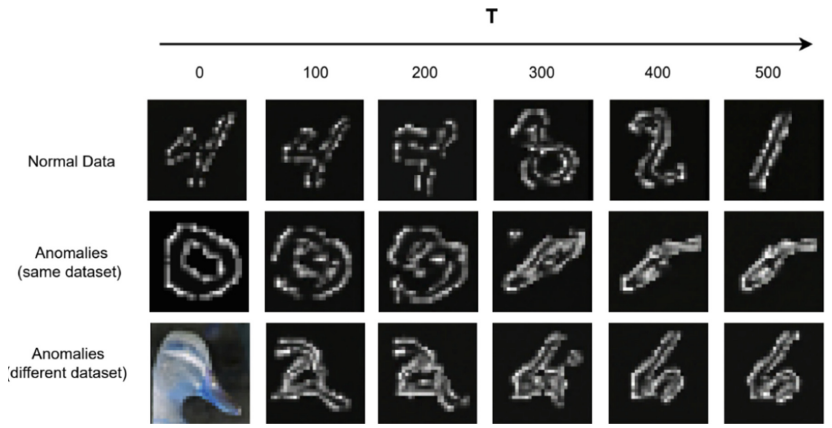

Fig. 3. Reconstructions using our model trained on the MNIST dataset, excluding all instances of the digit '0'. The figure depicts reconstruction results for normal data, anomalies from the same dataset, and anomalies from a different dataset.

process becomes unstable. This instability can be beneficial, as it can prevent the model from taking the "identity shortcut" when evaluating an anomalous instance. However, it can also cause the reconstructed version of normal data to differ from the input. As seen in Fig. 3 the reconstruction results change from timestamps 0 to 500. The stability of the reconstruction of normal data starts deteriorating after the diffusion timestamp 200. Yet, the reconstruction results for anomalies begin to deviate from the input even before the diffusion timestamp 200. Therefore, we have chosen diffusion timestamp 200 in this study to effectively detect anomalous data.

4.3 Anomaly Detection on MNIST

For our MNIST experiments, we adopted a many-versus-one setting. In each iteration, one digit was designated as anomalous data while our model was trained using images of the remaining nine digits. The architecture of the compression and diffusion models is grounded on the Latent Diffusion Model [19]. For the compression model, we employed a 3-layer autoencoder with channel sizes of [64, 128, 256]. This model compresses the image from a size of 32×32 down to a $8 \times 8 \times 3$ latent space, and it also incorporates a VQ-regularization [25] term. Subsequently, the diffusion training is facilitated by a 3-layer U-net model with channel sizes [224, 448, 672]. For classification, we deployed the ResNet-18 model. The input to this classifier is a concatenation of the original and the reconstructed image. As observed from Table 1 when our model is compared to three other reconstruction-based anomaly detection methodologies, our method consistently outperforms the others. Specifically, across all ten experiments, our model ranked as the most effective in nine out of the ten anomaly detection tests.

Table 1. AUROC score of anomaly detection on MNIST dataset

Anomaly digit	0	1	2	3	4	5	6	7	8	9
Autoencoder	53.1	60.2	62.2	57.8	55.2	56.9	56.3	50.3	63.1	51.2
AnnoDDPM	57.0	54.6	57.3	51.0	54.8	57.3	60.9	53.1	58.9	52.1
DDPM [9]	65.0	61.4	67.5	65.8	59.9	65.5	61.5	51.2	61.5	52.1
Our Method	64.9	73.2	72.6	69.7	69.7	68.7	68.0	72.6	71.5	56.3

Table 2. AUROC score of anomaly detection on CIFAR-10 dataset

Anomaly classes	{01234}	{23456}	{45678}	{67890}
Autoencoder	50.4	51.4	60.8	51.2
AnnoDDPM	52.3	56.4	54.7	56.2
DDPM	57.6	51.8	54.6	53.3
Our Method	64.5	60.1	54.0	57.4

4.4 Anomaly Detection on CIFAR-10

For the CIFAR-10 dataset, our experimental approach was grounded in the many-versus-many setting. In each iteration, we designated five distinct classes as the 'normal' dataset and the remaining five as 'anomalous' datasets. To clarify, in Table 2, the numerals 0 through 9 respectively symbolize the classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck.

The architectural foundation of our model for the CIFAR-10 dataset remains consistent with that employed for the MNIST dataset. However, our results on the CIFAR-10 were not as promising as those on the MNIST. Even though our model still surpassed other existing reconstruction-based algorithms, the performance decrement can primarily be attributed to the less stable reconstruction results on the CIFAR-10 dataset.

This instability might arise due to CIFAR-10 images being more complex and diverse in content than MNIST's handwritten digits. Thus, while our model demonstrates superiority over other reconstruction-based approaches, there remains a potential for refining and optimizing it further, especially when tackling complex datasets like CIFAR-10.

5 Conclusion

Tackling multi-class anomaly detection is a formidable challenge, given the intricate distribution characterizing normal data. Our approach, anchored in the latent diffusion model, underscores the promise and efficacy of this method for addressing such anomaly detection challenges. Notably, our model presents a remedy to the identity-shortcut predicament that frequently plagues conventional reconstruction-based anomaly detection mechanisms. A promising frontier for ensuing research in this domain is delving deeper into methodologies that can further stabilize the reverse process in diffusion during anomaly detection tasks.

Acknowledgements. Research reported in this publication was supported in part by the National Science Foundation under grant numbers [OAC-2201599] and the National Institute of General Medical Sciences of the National Institutes of Health under grant numbers [P30 GM145646].

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: International Conference on Machine Learning, pp. 214–223. PMLR (2017)
2. Boukerche, A., Zheng, L., Alfandi, O.: Outlier detection: methods, models, and classification. ACM Comput. Surv. (CSUR) **53**(3), 1–37 (2020)
3. Chalapathy, R., Menon, A.K., Chawla, S.: Robust, deep and inductive anomaly detection. In: Ceci, M., Hollmén, J., Todorovski, L., Vens, C., Džeroski, S. (eds.) ECML PKDD 2017. LNCS (LNAI), vol. 10534, pp. 36–51. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71249-9_3

4. Chen, J., Sathe, S., Aggarwal, C., Turaga, D.: Outlier detection with autoencoder ensembles. In: Proceedings of the 2017 SIAM International Conference on Data Mining, pp. 90–98. SIAM (2017)
5. Dasgupta, D., Nino, F.: A comparison of negative and positive selection algorithms in novel pattern detection. In: SMC 2000 Conference Proceedings. 2000 IEEE International Conference on Systems, Man and Cybernetics. Cybernetics Evolving to Systems, Humans, Organizations, and their Complex Interactions cat. no. 0, vol. 1, pp. 125–130. IEEE (2000)
6. Deecke, L., Ruff, L., Vandermeulen, R.A., Bilen, H.: Transfer-based semantic anomaly detection. In: International Conference on Machine Learning, pp. 2546–2558. PMLR (2021)
7. Eskin, E.: Anomaly detection over noisy data using learned probability distributions (2000)
8. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12873–12883 (2021)
9. Graham, M.S., Pinaya, W.H., Tudosiu, P.D., Nachev, P., Ourselin, S., Cardoso, J.: Denoising diffusion models for out-of-distribution detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2947–2956 (2023)
10. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of Wasserstein GANs. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
11. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
12. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
13. Laurikkala, J., Juhola, M., Kentala, E., Lavrac, N., Miksch, S., Kavsek, B.: Informal identification of outliers in medical data. In: Fifth International Workshop on Intelligent Data Analysis in Medicine and Pharmacology, vol. 1, pp. 20–24. Citeseer (2000)
14. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
15. Lu, W., et al.: Unsupervised sequential outlier detection with deep architectures. IEEE Trans. Image Process. **26**(9), 4321–4330 (2017)
16. Makhzani, A., Frey, B.: K-sparse autoencoders. arXiv preprint arXiv:1312.5663 (2013)
17. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep learning for anomaly detection: a review. ACM Comput. Surv. (CSUR) **54**(2), 1–38 (2021)
18. Rifai, S., Vincent, P., Muller, X., Glorot, X., Bengio, Y.: Contractive auto-encoders: explicit invariance during feature extraction. In: ICML (2011)
19. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10684–10695 (2022)
20. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Niethammer, M., et al. (eds.) IPMI 2017. LNCS, vol. 10265, pp. 146–157. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59050-9_12
21. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**(7), 1443–1471 (2001)

22. Shewhart, W.A.: Economic Control of Quality of Manufactured Product. Macmillan And Co Ltd, London (1931)
23. Tax, D.M., Duin, R.P.: Support vector data description. Mach. Learn. **54**(1), 45–66 (2004)
24. Theis, L., Shi, W., Cunningham, A., Huszár, F.: Lossy image compression with compressive autoencoders. arXiv preprint arXiv:1703.00395 (2017)
25. Van Den Oord, A., Vinyals, O., et al.: Neural discrete representation learning. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
26. Velleman, P.F., Hoaglin, D.C.: Applications, Basics, and Computing of Exploratory Data Analysis. Duxbury Press, New York (1981)
27. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A., Bottou, L.: Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res. **11**(12) (2010)
28. Wang, S., Wu, L., Cui, L., Shen, Y.: Glancing at the patch: anomaly localization with global and local feature comparison. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 254–263 (2021)
29. Xia, Y., Zhang, Y., Liu, F., Shen, W., Yuille, A.L.: Synthesize then compare: detecting failures and anomalies for semantic segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 145–161. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_9
30. You, Z., et al.: A unified model for multi-class anomaly detection. Adv. Neural. Inf. Process. Syst. **35**, 4571–4584 (2022)
31. Zenati, H., Foo, C.S., Lecouat, B., Manek, G., Chandrasekhar, V.R.: Efficient GAN-based anomaly detection. arXiv preprint arXiv:1802.06222 (2018)
32. Zhou, C., Paffenroth, R.C.: Anomaly detection with robust deep autoencoders. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 665–674 (2017)