Quest Journals
Journal of Software Engineering and Simulation
Volume 10 ~ Issue 3 (2024) pp: 24-32
ISSN(Online) :2321-3795 ISSN (Print):2321-3809
www.questjournals.org

Research Paper

Unsupervised Multi-View Anomaly Detection with GAN

Fangyuan Zhang[1]
[1]*School of Control and Computer Engineering, North China Electric Power University, Baoding, 071051, China*
Corresponding Author: Fangyuan Zhang

ABSTRACT: *In the process of detecting anomalies in high-dimensional data such as images, unsupervised learning faces significant challenges and has become a research focus within the fields of machine learning and data mining. By modeling the distribution of standard samples, those that deviate from this distribution are identified as anomalies. Generative Adversarial Networks (GANs) have been proven to effectively model the complex high-dimensional distribution of normal samples, making them a suitable technique for addressing this issue. Based on this, we propose an unsupervised multi-view anomaly detection method, MVAD-GAN, which can utilize datasets containing anomalous samples and train deep neural networks without the need for labels, thus effectively identifying anomalies in the data. The method combines a multi-input GAN and an Attribute-Class Encoder (ACEncoder), with the former capturing features and their relationships from different views, and the latter being capable of training even in unsupervised environments facing data quality issues. Additionally, we introduce an anomaly scoring strategy based on the ACEncoder network to determine whether a sample is abnormal and assign anomaly categories by scoring samples. The experimental results on two large-scale image datasets, MNIST and Fashion-MNIST, validate the effectiveness of the MVAD-GAN method, demonstrating its potential in the field of high-dimensional data anomaly detection.*

KEYWORDS: *Deep Neural Network; Unsupervised Learning; Generative Adversarial Networks; Anomaly Detection; Multi-view*

R*eceived 02 Mar., 2024;* ***R****evised 11 Mar., 2024;* ***A****ccepted 13 Mar., 2024*

I. INTRODUCTION

In today's data-driven era, anomaly detection, as a crucial technology, has been widely applied in multiple fields, including but not limited to cybersecurity, medical diagnosis[1], financial monitoring[2] and industrial manufacturing[3]. Anomaly detection refers to the identification of data points that significantly differ from the majority of data, often indicating some form of anomalous or erroneous behavior. These anomalies may represent potential network intrusions, early signs of diseases, defective products on the production line, or financial fraud activities. Therefore, accurate anomaly detection is essential for preventing losses and taking timely measures.

Traditional anomaly detection methodologies can be broadly categorized into approaches based on statistics[4], density[5], clustering[6], graph theory[7], or Support Vector Machines (SVM)[8][9]. While these techniques can yield favorable outcomes in specific contexts, they often rely on manually set thresholds or presuppositions, which may prove inadequate in handling high-dimensional data or complex data distributions. With the continuous escalation in data volume and complexity, the limitations of traditional methods in terms of accuracy and efficiency have become increasingly apparent. Moreover, the scarcity of labeled data in many scenarios exacerbates the difficulty of model training, constraining the applicability of supervised learning approaches. In recent years, the rapid advancement of deep learning technologies[10][11] has offered novel perspectives for addressing these challenges. Especially noteworthy are the unsupervised deep learning approaches[12][13][14], which are capable of learning the normal distribution of data in the absence of labeled data, thereby identifying anomalies that deviate from this distribution.

In the current information era, multi-view data has become increasingly common. Multi-view data provides rich and complementary information for the same instance from different perspectives, collectively offering a comprehensive understanding of the instance. For example, in intelligent transportation systems, the same traffic scenario might be captured by three different data sources: ground cameras, drone perspectives, and vehicle-mounted sensors. Faced with such multi-view data, relying solely on single-view anomaly detection methods is no longer sufficient. This presents two challenges: on one hand, the data distribution within each independent view is extremely complex; on the other hand, there often exists inconsistency in features across different views. As introduced in [15], there could be three types of anomalies in multi-view data as shown in Diagram1: (i) class-anomaly: refers to that the pattern of an instance in some or even all views is inconsistent, (ii) attribute-anomaly: refers to the pattern of an instance that significantly deviates from the center of any cluster in every view, and (iii) class-attribute-anomaly: represents a mixture of class and attribute anomalies, showing inconsistent pattern of an instance in some views, while in other views they deviate significantly from the cluster center. Therefore, in recent years, anomaly detection techniques[15][16][17][18] for multi-view data have attracted widespread interest among researchers. However, these methods often face several challenges: (1) They tend to focus on anomaly detection in pairwise views, making it difficult to extend to multiple views; (2) They rely on limited labeled data, which restricts their effectiveness in practical applications; (3) And when attempting to address these issues through subspace clustering techniques, they encounter the problem of subspaces not being independent, which may decrease detection performance. Therefore, although multi-view anomaly detection is theoretically attractive, achieving efficient and accurate anomaly detection in practice still faces significant challenges.

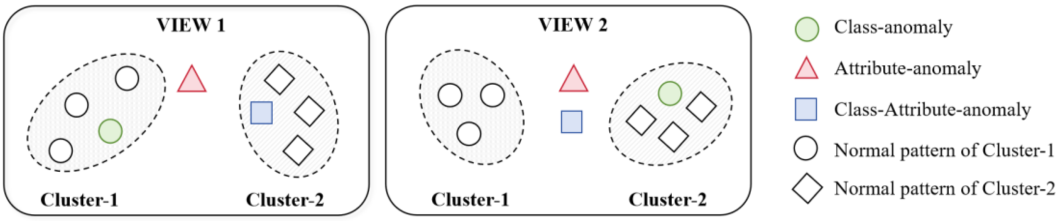

Diagram 1

In this context, we propose an unsupervised multi-view anomaly detection method named MVAD-GAN (Multi-View Anomaly Detection with ClusterGAN[33]). Additionally, our method also considers the need for novelty detection[19][20]—identifying new patterns in test data that did not appear in the training data. Since novelty detection and anomaly detection share a similar theoretical foundation, MVAD-GAN is also suitable for tasks of novelty detection. MVAD-GAN is committed to effectively integrating and analyzing large-scale image data from multiple-views, allowing deep learning networks to be trained without any label information, thereby achieving precise identification of anomalies and novelty in multi-view data. Specifically, during the model training phase, our method introduces a multi-input Generative Adversarial Network (GAN), with both the generator and discriminator designed to process and analyze inputs from multiple-views. This enables the capture of unique features within each view and their interrelations, enhancing the model's ability to identify anomalies. Furthermore, to improve the efficiency and effectiveness of unsupervised learning, we have integrated an Attribute-Class Encoder (ACEncoder). This encoder can be trained in conjunction with the GAN, and its design allows the model to be trained even when data quality is compromised (e.g., by noise), achieving true unsupervised learning. Through optimizing ACEncoder, we can extract meaningful representations in both the attribute and category domains of each sample, which is crucial for a deep understanding of data structure and content. In the anomaly detection phase, we utilize the hidden space representations of samples in the attribute and category domains obtained from the ACEncoder module and propose an anomaly scoring mechanism based on these representations. This mechanism determines whether a sample is an anomaly and to which category of anomaly it belongs by calculating its anomaly score. After experimental validation on MNIST and Fashion-MNIST, two widely used large-scale image datasets, MVAD-GAN has demonstrated its efficiency and accuracy in detecting anomalies in multi-view samples.

II. RELATED WORK

2.1 Traditional Anomaly Detection Methods

Early anomaly detection methods primarily focused on global outliers within a single view, i.e., data points that significantly differ from the majority of samples. With the rise of multi-view learning, research shifted towards integrating information from multiple views for anomaly detection, prompting the development of multi-view anomaly detection methods. A cluster-based approach (Horizontal Outlier Detection, HOAD)[15] was among the first to attempt multi-view anomaly detection. This method constructs a similarity matrix across sets and utilizes spectral embedding techniques to extract feature representations. Following HOAD, AP (Affinity Propagation)[21], MLRA (Multi-view Low-Rank)[22], and CL (Collective Learning)[23] were subsequently proposed to address Class-anomaly detection. To simultaneously identify attribute outliers and class outliers, L2 and L1 regularization techniques have been applied to low-rank subspace learning[22] and the K-Means clustering algorithm[24]. Nonetheless, these methods depend on class label information, which is often missing in real-world datasets, limiting the applicability of the aforementioned anomaly detection approaches. Moreover, a common drawback of these methods is that they are only suitable for identifying outliers in a dual-view context, leading to significant complications when dealing with three or more views. Subsequently, another approach called LDSR[15] based on low-rank subspace learning was proposed, suggesting that the goal of multi-view anomaly detection is not only to identify instances deviating from normal patterns (i.e., Attribute-anomalies and Class-Attribute-anomalies) but also to recognize instances with inconsistent behaviors across multiple-views (i.e., Class-anomalies). Although this method can overcome the pairwise limitation by separating the common representation from the residual representation, it still has a drawback: the algorithm essentially employs multi-view subspace clustering techniques across all views and calculates the deviations between views without generating a complete space. After that, there are some new research topics for multi-view anomaly detection[25], including method called Multi-view Bayesian Outlier Detector[26], and a Self-Representation method with Local Similarity Preserving[27].

2.2 Abnormality DetectionMethods Based on Deep Learning

GANs were initially introduced in the form of AnoGAN[1]. They utilized the L2-norm and the loss calculation between test images and their closest reconstructed images as an anomaly score. Building on this concept, Deecke et al. proposed ADGAN[28], which slightly improved the results. Unlike AnoGAN, ADGAN initiates a search for the closest matches in the latent space by initializing multiple closest matches. Schlegl et al. introduced f-AnoGAN[29], which enhances their AnoGAN method by substituting the deep convolutional GAN (DCGAN) with the Wasserstein GAN (WGAN-GP) and also introduced an encoder that can be trained separately for mapping images to the latent space. Recently, CGAEs[30] introduced a multi-view autoencoder model for detecting Attribute-anomalies and Class-anomalies. Three deep learning methods capable of detecting all three types of anomalies simultaneously include FMOD (Fast Multi-view Outlier Detection)[27], MODDIS (Multi-view Outlier Detection in Deep Intact Space)[31], and NCMOD (Neighborhood Consensus networks based Multi-view Outlier Detection)[32].

III. METHODOLOGY

We trained our MVAD-GAN model on training samples containing anomalous instances, resulting in the generation of a generator G, a discriminator D, and an encoder E that maps samples to a latent space representation. The training framework is illustrated in Diagram 2.

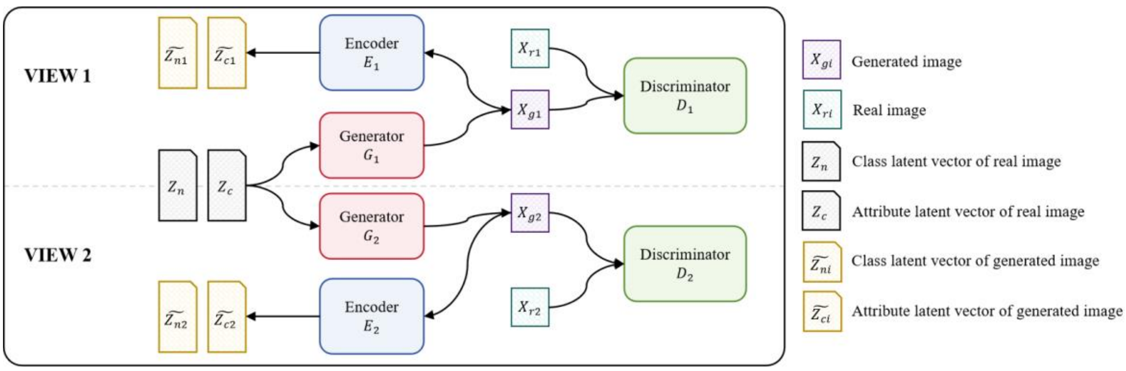

Diagram 2

The input is divided into two parts: the first part is the attribute latent vector Z_n, where this latent vector initializes the sample's attribute information, and the second part is the class latent vector Z_c, where this latent vector initializes the sample's class information. Generator G represents the generator in the GAN, Encoder E is the ACEncoder, and Discriminator D is the discriminator in the GAN. During the G+E training phase, the generated image X_g by the generator serves as the input to the encoder. Encoder strives to make the output similar to the input Z_n, Z_c. In the G+D training phase, similar to the GAN training process, the generated image X_g is presented to the discriminator D, aiming to deceive it into treating X_g as a real image, while the discriminator endeavors to correctly identify X_g as a fake image. Meanwhile, X_r represents real images in this context.

In more detail, the training phase comprises two steps: (1) jointly training the generator and the encoder. During this period, the parameters of the generator and encoder are optimized, while the parameters of the discriminator remain fixed. Subsequently, (2) we jointly train the generator and discriminator. In this phase, the parameters of the generator and discriminator are optimized, while the encoder's parameters remain fixed.

During the inference phase, the process involves three steps: (1) inputting a test sample, where the encoder maps the image to the latent space. (2) Inputting a normal image, and the encoder maps it from the image space to the latent space. Then, (3) by assessing the deviation in latent space between multiple views, the distances between latent spaces of normal samples should be very close, while the distance of latent space for anomalous samples should be comparatively distant from that of other samples. The degree of deviation for each sample is utilized for anomaly scoring.

3.1 Training Phase: Collaborative Training of the Generator and Encoder

Taking inspiration from ClusterGAN, we reverse the order of the encoder and decoder in the standard AutoEncoder (AE), constructing a Z-I-Z (Latent space-Image space-Latent space) structure. During this phase, the parameters of the discriminator are fixed. Our aim is to utilize the generator to generate fictitious samples in the image space from the latent space. Simultaneously, the encoder is trained to map these virtual samples back to the latent space. Therefore, for the training of the encoder, real image data is not required.

During training, we partition the input vector into two parts. The first part is the Z_n attribute vector, which we want to be similar to the reconstructed $E(G(Z_n))$. To achieve this, a mean squared error is applied as a constraint. The second part is the class vector Z_c, which we want to be similar to the reconstructed $E(G(Z_c))$. Given that Z_c is a one-hot type of discrete vector, we use a cross-entropy loss to impose constraints. Consequently, the loss function for the collaborative training of the generator and encoder is defined as follows:

$$\mathrm{L_{G+E}} = \beta_n \left\| z_n - E\big(G(z_n)\big) \right\|^2 + \beta_c H\left(z_c, E\big(G(z_c)\big)\right) \rightarrow (1)$$

where Z_c is the attribute vector, Z_n is the class vector, H represents the cross-entropy loss, and β_n and β_c are two hyperparameters. These hyperparameters are used to adjust the weights of the losses in the attribute domain and class domain for the encoder and generator.

3.2 Training Phase: Collaborative Training of the Generator and Discriminator

The purpose of this process is to enhance the performance of the generator G and discriminator D. During training, the loss function of the encoder E is frozen. Therefore, the loss function for the collaborative training of the generator and discriminator is:

$$L_{G+D} = D(x) + (1 - D\big(G(x)\big) \rightarrow (2)$$

where $D\big(G(x)\big)$ represents the loss for the generated images through the discriminator D, and $D(x)$ represents the loss for the real images x through discriminator D. Through the adversarial process, it becomes possible for the generator G to generate images that increasingly resemble real images, while the discriminator D becomes increasingly adept at accurately distinguishing between real and generated images.

Combining the two components, our overall objective function is:

$$\min_{\theta G, \theta E} \max_{\theta D} D(x) + (1 - D\big(G(x)\big) + \beta_n \left\| z_n - E\big(G(z_n)\big) \right\|^2 + \beta_c H\left(z_c, E\big(G(z_c)\big)\right) \rightarrow (3)$$

where $\theta G, \theta E$ and θD represent the parameters of the generator, encoder, and discriminator, respectively.

3.3 Inference Phase: Anomaly Detection

The trained encoder E has acquired the ability to represent the latent space. The next task is to perform anomaly detection by measuring the outlier score for each sample. In MVAD-GAN, we designed an anomaly score based on the model structure, comprising two components: attribute anomaly and class anomaly. The following will provide an introduction to these components.

The objective of attribute anomaly is to assess whether a sample is significantly distant from other samples within a single view. Here, the attribute outlier score is calculated using the distance to the k-th nearest neighbor. For the i-th sample, the attribute anomaly score in the v-th view is the distance to the k-th nearest neighbor when mapped from the v-th view to the set of other samples:

$$S_v^a(i) = \mathrm{knn\_dist}(z_{niv} | Z_{nv}) \rightarrow (4)$$

where Z_{nv} is the set of attribute latent spaces for all samples included in the v-th view, and $\mathrm{knn\_dist}(\cdot \mid \cdot)$ returns the distance to the k-th nearest neighbor of z_{niv} in Z_{nv}. By considering all V views for sample i, the attribute outlier score for the i-th sample is as follows.

$$S^a(i) = \sum_{i=1}^{V} S_v^a(i) \rightarrow (5)$$

The objective of the class outlier score is to measure the extent of differences in the neighborhood structure of a sample across different views. Given that a sample has V representations in the latent space, the class anomaly score for a sample can be assessed by measuring the mutual distances between its representations in different views of the latent space. Due to the structure of MVAD-GAN, we can determine whether the i-th sample is an anomalous sample by comparing its class labels across various views:

$$S^{c}(i) = \mathrm{e}(z_{civ}|Z_{ciV}) \rightarrow (6)$$

where z_{civ} is the class latent vector for each view of the i-th sample, Z_{ciV} is the class latent vector for all views of this sample, and $\mathrm{e}(\cdot \mid \cdot)$ is a decision function. If there are differences in class labels across views for this sample, indicating it is a class anomaly, $\mathrm{e}(\cdot \mid \cdot)$ returns a large value as its class anomaly score. If the class labels are identical across all views, suggesting it is not a class anomaly, $\mathrm{e}(\cdot \mid \cdot)$ returns 0.

During the training phase, the framework aims to learn sample representations by minimizing the distance between features mapped from different views. Such steps help bring the representations of normal samples closer in the latent space. However, if noticeable differences persist despite efforts to minimize distances during training, it may suggest an irreconcilable contradiction between different views. This opposition could indicate an anomaly, where the sample possesses an anomalous class attribute. Therefore, the overall anomaly score for the i-th sample is:

$$S(i) = S^{a}(i) + S^{c}(i) \rightarrow (7)$$

The larger the anomaly score for a sample, the more likely it is to be an anomalous point.

IV. EXPERIMENTAL RESULT

4.1 Experimental Setup

Dataset. We conducted experiments using two public datasets: MNIST and FashionMNIST. MNIST contains 70,000 images of handwritten digits from 0 to 9, while FashionMNIST encompasses images of 10 types of clothing items. To standardize dimensions and enhance comparability across datasets, we mapped the pixel values of these datasets from the range [0,255] to [0,1] and normalized the data using MinMaxScaler.

Constructing multi-view data. We utilize GLCM and LBP to extract features, creating three views including the original features. Specifically, GLCM is used to analyze the co-occurrence matrix of pixels in grayscale images and compute its feature values to reflect the texture characteristics of the images, involving parameters such as direction, step distance, and matrix order. The LBP algorithm compares the grayscale value of the center pixel with those of its surrounding pixels to generate an LBP value that reflects local texture information.

Generating Anomaly Data. To generate three types of anomalies, we first randomly create an outlier index list. Then, using this list, we replace the normal data in the dataset. Class anomalies are created by swapping features of a particular view in pairs of samples from different categories, while keeping the features of other views unchanged. Attribute anomalies are created by converting the features of all views in a sample into Gaussian noise. Class-attribute anomalies combine these two approaches: first swapping the features of a particular view in pairs of samples from different categories, then converting those features into Gaussian noise.

Baseline and Evaluation Metrics. To validate the effectiveness of the MVAD-GAN method, we compared it with three multi-view outlier detection methods: the Affinity Propagation anomaly detection method AP, Multi-View Low-Rank Analysis MLRA, and Latent Discriminative Subspace Representation LDSR. Experiments were conducted on 10 sets of outlier sample indexes generated repeatedly, with each experiment repeated 10 times and the results averaged to reduce the impact of fluctuations in data construction. We used the Area Under the ROC Curve (AUC) as the evaluation metric and explored several sets of outlier ratios under different anomaly categories to analyze MVAD-GAN's performance across various types of anomalies.

4.2 Detection Effectiveness

First, we examined scenarios where all three types of anomaly rates were 5%, followed by an analysis of scenarios with only 5% class anomalies. All results were obtained after the model underwent 200 training epochs. At the same outlier ratio, we assessed the AUC values for both datasets in dual-view and triple-view setups, with the best results highlighted in bold and '#' indicating that the algorithm does not support that scenario. The total anomaly data ratio was 15%, including class-anomaly, attribute-anomaly, and class-attribute anomaly, each accounting for 5%. Each experiment was repeated 10 times, and the results were averaged.

Table 1 shows that MVAD-GAN outperformed all baseline methods in handling large-scale datasets. Particularly in the triple-view dataset, the model demonstrated significant advantages, overcoming problems encountered by previous methods in dealing with multi-view data, such as the high computational resource consumption of pairwise comparisons and the limitation of the MLRA method in not supporting multi-view anomaly detection.

Table 1: All three abnormal ratios are 5%

Dataset	View	AP	MLRA	LDSR	OUR
MNIST	2	0.67	0.65	0.89	**0.90**
	3	0.49	#	0.35	**0.85**
Fashion MNIST	2	0.64	0.8	0.83	**0.85**
	3	0.52	#	0.56	**0.83**

Table 2: Contains 5% of all class-anomalies

Dataset	View	AP	MLRA	LDSR	OUR
MNIST	2	0.86	0.58	**0.91**	0.88
	3	0.51	#	0.76	**0.88**
Fashion MNIST	2	0.87	0.71	**0.94**	0.85
	3	0.53	#	0.80	**0.84**

Due to the characteristics of our proposed MVAD-GAN model, the type of anomaly can be directly determined by labels, rather than calculating the distance between neighbors in the hidden space as most methods do. Therefore, we designed this experiment for all class-anomalies, and the experimental results are shown in Table 2. The results indicate that MVAD-GAN performs similarly to the previously proposed method in two views, and our method still demonstrates significant advantages in three views. Our proposed MVAD-GAN model outperforms the baseline method in three views.

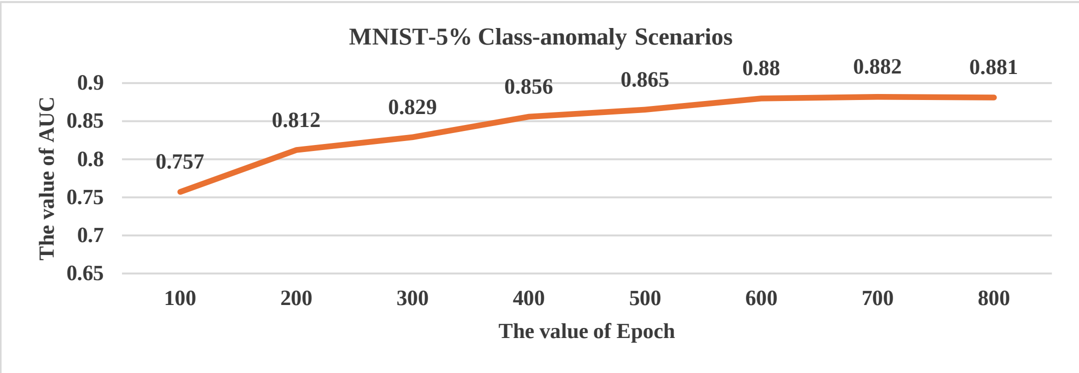

Figure 1: Model Performance with Changes in Training Epochs: AUC Values

4.3 Parameter Influence Analysis

The impact of training epochs. We trained GAN on the MNIST in 5% Class-anomaly scenarios. In the experiment, the model was trained for up to 800 epochs to fully observe the changes in model performance with training progress. From the experimental results in Figure 1, it can be observed that the performance of the model changes significantly with the increase of training epochs. During the first 700 epochs of training, the model performance showed a stable improvement trend with the increase of training epochs. This indicates that at this stage, the model gradually learns more features from the data and its performance continues to improve. However, starting from the 700th epoch, the performance of the model has decreased, which may be due to overfitting caused by over training. After 700 epochs, although the training continues, overfitting of the model to the training set may lead to a decrease in its generalization ability, resulting in a decrease in performance on the test set.

The impact of hyperparameters. During the model training process, we configured hyperparameters that can influence the weights of the attribute domain and class domain losses. The larger the weight, the faster the network updates during gradient descent. We designed multiple ratios, and the following experiments investigate the influence of these two parameters on the overall model performance.

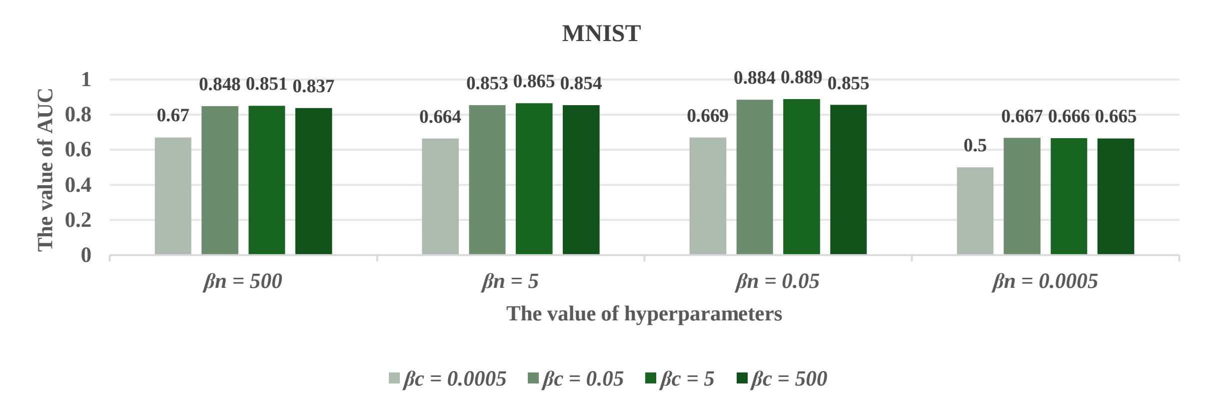

Figure 2: Effect of Changing Hyperparameters β_n and β_c on Performance in the MNIST Dataset

As shown in Figure 2, on the MNIST dataset, when β_n and β_c take similar values, the results are relatively high. The possible reason is that the β_n and β_c in our model are sensitive to the dataset's structure. When there is a significant difference in their proportions, it introduces substantial perturbations to the model training, leading to potential inconsistency in training on the two latent spaces. When their values are either too small or too large, this introduces dimensional differences in the overall model loss, causing discrepancies in the update rates of the ACEncoder module compared to the generator and discriminator. This results in a decline in the overall model performance.

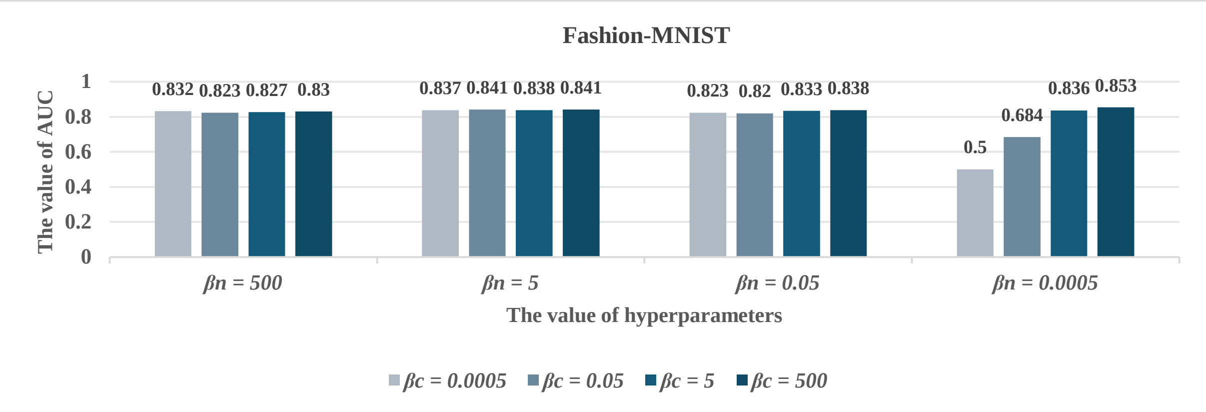

Figure 3: Effect of Changing Hyperparameters β_n and β_c on Performance in the Fashion-MNIST Dataset

As shown in Figure 3, on the Fashion-MNIST dataset, the results are relatively high when β_n and β_c take similar values. The reason is similar to the model's behavior on the MNIST dataset. However, due to the complexity of the Fashion-MNIST dataset, experimental results demonstrate that when the weight of β_n is relatively small, the results increase as the weight of β_c rises. This suggests that class anomalies are more challenging to converge during the overall model optimization process compared to attribute anomalies. This observation aligns with the fact that the Fashion-MNIST dataset is more challenging than the MNIST dataset in classification tasks.

V. CONCLUSION

We present MVAD-GAN, an unsupervised multi-view anomaly detection method that utilizes Generative Adversarial Networks (GAN) and Class-Attribute Encoder (ACEncoder). Our approach involves training with a large-scale dataset comprising mixed anomaly data, leveraging the capabilities of GAN and ACEncoder for efficient model training. By mapping images to the hidden space through ACEncoder, we achieve rapid inference and precise anomaly detection, with the added capability of accurately categorizing anomalies. Through joint training of the generator and ACEncoder, we establish similarity between the attribute domain hidden space and the class domain hidden space. This enhances the efficiency of hierarchical analysis in the hidden space, fortifying the model's robustness to anomalies within the training data. By assessing distances and labels in the hidden space, we effectively determine the anomaly category of a given sample. MVAD-GAN outperforms traditional methods by significantly accelerating the inference process, making it well-suited for large-scale datasets, and demonstrating commendable accuracy in anomaly categorization. Future endeavors will prioritize the analysis of hidden space structures, the investigation of different encoder losses' impact on them, and an exploration of optimal weight selection. Future work will focus on analyzing the structure of hidden spaces, studying the impact of different encoder losses on them, and delving into the issue of weight selection.

REFERENCES

[1]. Schlegl, Thomas, et al. "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery." International conference on information processing in medical imaging. Springer, Cham, 2017.

[2]. Awoyemi, John O., Adebayo O. Adetunmbi, and Samuel A. Oluwadare. "Credit card fraud detection using machine learning techniques: A comparative analysis." 2017 international conference on computing networking and informatics (ICCNI). IEEE, 2017.

[3]. Tandiya, Nistha, et al. "Deep predictive coding neural network for RF anomaly detection in wireless networks." 2018 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE, 2018.

[4]. Nguyen, Luong Ha, and James- A. Goulet. "Anomaly detection with the switching kalman filter for structural health monitoring." Structural Control and Health Monitoring 25.4 (2018): e2136.

[5]. Zhang, Liangwei, Jing Lin, and Ramin Karim. "Adaptive kernel density-based anomaly detection for nonlinear systems." Knowledge-Based Systems 139 (2018): 50-63.

[6]. Alguliyev, Rasim, Ramiz Aliguliyev, and Lyudmila Sukhostat. "Anomaly detection in Big data based on clustering." Statistics, Optimization & Information Computing 5.4 (2017): 325-340.

[7]. Akoglu, Leman, Hanghang Tong, and Danai Koutra. "Graph based anomaly detection and description: a survey." Data mining and knowledge discovery 29.3 (2015): 626-688.

[8]. Erfani, Sarah M., et al. "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning." Pattern Recognition 58 (2016): 121-134.

[9]. Seeböck, Philipp, et al. "Unsupervised identification of disease marker candidates in retinal OCT imaging data." IEEE transactions on medical imaging 38.4 (2018): 1037-1047.

[10]. J. Cao, G. Yang, and X. Yang, ‘‘A pixel-level segmentation convolutional neural network based on deep feature fusion for surface defect detection,’’ IEEE Trans. Instrum. Meas., vol. 70, pp. 1–12, 2021.

[11]. G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, ‘‘Deep learning for anomaly detection: A review,’’ ACM Comput. Surv., vol. 54, no. 2, pp. 1–38, Mar. 2021.

[12]. F.V. Massoli, F. Falchi, A. Kantarci, S. Akti, H.K. Ekenel, G. Amato, MOCCA: multilayer one-class classification for anomaly detection, IEEE Trans. Neural Netw. Learn. Syst. 33 (6) (2022) 2313–2323.

[13]. C. Yang, H. Wen, B. Hooi, Y. Wu, L. Zhou, A multi-scale reconstruction method for the anomaly detection in stochastic dynamic networks, Neurocomputing 518 (2023) 482–495.

[14]. Y. Yao, J. Ma, Y. Ye, Regularizing autoencoders with wavelet transform for sequence anomaly detection, Pattern Recognit. 134 (2023) 109084.

[15]. Kai Li, Sheng Li, Zhengming Ding, Weidong Zhang, and Yun Fu. 2018. Latent Discriminant Subspace Representations for Multi-View Outlier Detection. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018, New Orleans, Louisiana, USA, February 2-7, 2018. AAAI Press, 3522–3529.

[16]. Jing Gao, Wei Fan, Deepak S. Turaga, Srinivasan Parthasarathy, and Jiawei Han. 2011. A Spectral Framework for Detecting Inconsistency across Multi-source Object Relationships. In Proceedings of 11th IEEE International Conference on Data Mining, ICDM 2011, Vancouver, BC, Canada, December 11-14, 2011. IEEE Computer Society, 1050–1055.

[17]. Handong Zhao and Yun Fu. 2015. Dual-Regularized Multi-View Outlier Detection. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015. AAAI Press,4077 – 4083.

[18]. Hao Wang, Zhi-Qi Cheng, Jingdong Sun, Xin Yang, Xiao Wu, Hongyang Chen, and Yan Yang. 2023. Debunking Free Fusion Myth: Online Multi-view Anomaly Detection with Disentangled Product-of-Experts Modeling. In Proceedings of the 31st ACM International Conference on Multimedia (MM '23). Association for Computing Machinery, New York, NY, USA, 3277–3286. https://doi.org/10.1145/3581783.3612487

[19]. Sabokrou, Mohammad, et al. "Adversarially learned one-class classifier for novelty detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[20]. Pimentel, Marco AF, et al. "A review of novelty detection." Signal processing 99 (2014): 215-249.

[21]. Alejandro Marcos Alvarez, Makoto Yamada, Akisato Kimura, and Tomoharu Iwata. 2013. Clustering-based Anomaly Detection in Multi-view Data. In Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, CIKM 2013, San Francisco, CA, USA, October 27 - November 1, 2013. ACM, 1545 – 1548.

[22]. Sheng Li, Ming Shao, and Yun Fu. 2015. Multi-View Low-Rank Analysis for Outlier Detection. In Proceedings of the 2015 SIAM International Conference on Data Mining, SDM 2015, Vancouver, BC, Canada, April 30 - May 2, 2015. SIAM, 748 – 756.

[23]. Jun Guo and Wenwu Zhu. 2018. Partial Multi-View Outlier Detection Based on Collective Learning. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018, New Orleans, Louisiana, USA, February 2-7, 2018. AAAI Press, 298 – 305. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17166.

[24]. Zhao, Handong, et al. "Consensus regularized multi-view outlier detection." IEEE Transactions on Image Processing 27.1 (2017): 236-248.

[25]. Siqi Wang, Jiyuan Liu, Guang Yu, Xinwang Liu, Sihang Zhou, En Zhu, Yuexiang Yang, Jianping Yin, and Wenjing Yang. 2022. Multiview Deep Anomaly Detection: A Systematic Exploration. IEEE Transactions on Neural Networks and Learning Systems (2022), 1–15. https://doi.org/10.1109/TNNLS.2022.3184723.

[26]. Zhen Wang, Ji Zhang, Yizheng Chen, Chenhao Lu, Jerry Chun-Wei Lin, Jing Xiao, and Rage Uday Kiran. 2021. Learning Probabilistic Latent Structure for Outlier Detection from Multi-view Data. In Proceedings of the 25th Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD 2021, Virtual Event, May 11-14, 2021, Vol. 12712. Springer, 53–65.

[27]. Yu Wang, Chuan Chen, Jinrong Lai, Lele Fu, Yuren Zhou, and Zibin Zheng. 2023. A Self-Representation Method with Local Similarity Preserving for Fast MultiView Outlier Detection. ACM Trans. Knowl. Discov. Data 17, 1 (2023), 2:1–2:20.

[28]. Lucas Deecke, Robert Vandermeulen, Lukas Ruff, Stephan Mandt, and Marius Kloft, ‘Image Anomaly Detection with Generative Adversarial Networks’, in Machine Learning and Knowledge Discovery in Databases, pp. 3—-17. Springer International Publishing, (2018).

[29]. Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Georg Langs, Ursula Schmidt-Erfurth, f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks, Medical Image Analysis, Volume 54, 2019, Pages 30-44, ISSN 1361-8415, https://doi.org/10.1016/j.media.2019.01.010.

[30]. Shaoshen Wang, Yanbin Liu, Ling Chen, and Chengqi Zhang. 2021. Cross-aligned and Gumbel-refactored Autoencoders for Multi-view Anomaly Detection. In Proceedings of the 33rd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2021, Washington, DC, USA, November 1-3, 2021. IEEE, 1368– 1375. https://doi.org/10.1109/ICTAI52525.2021.00218

[31]. Yu-Xuan Ji, Ling Huang, Heng-Ping He, Chang-Dong Wang, Guangqiang Xie, Wei Shi, and Kun-Yu Lin. 2019. Multi-view Outlier Detection in Deep Intact Space. In Proceedings of the 2019 IEEE International Conference on Data Mining, ICDM 2019, Beijing, China, November 8-11, 2019. IEEE, 1132–1137. https://doi.org/10.1109/ICDM.2019.00136

[32]. Li Cheng, Yijie Wang, and Xinwang Liu. 2021. Neighborhood Consensus Networks for Unsupervised Multi-view Outlier Detection. In Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Virtual Event, February 2-9, 2021. AAAI Press, 7099–7106. https://ojs.aaai.org/index.php/AAAI/article/view/16873

[33]. Mukherjee, Sudipto, et al. "Clustergan: Latent space clustering in generative adversarial networks." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.