Investigating the Impact of Hyper Parameters on Intrusion Detection System Using Deep Learning Based Data Augmentation

Umar Iftikhar, Syed Abbas Ali
Department of Computer & Information System Engineering, NED University of Engineering & Technology, Karachi, Pakistan

***Abstract*—The effects of changing learning rates, data augmentation percentage and numbers of epochs on the performance of Wasserstein Generative Adversarial Networks with Gradient Penalties (WGAN-GP) are evaluated in this study. The purpose of this research is to find out how they affect the data augmentation to enhance stability during training. In this research, the degree of system performance is measured using the Classification Model Utility approach. For this reason, this study aims to determine the interaction between learning rate, augmentation percentage and epoch value when using WGAN-GP to generate synthetic data for the recognition of the system performance. The results will provide the indications on how some of the hyper parameters can be adjusted up or down for having positive or negative consequences on the generation process for further research and use of WGAN-GP. It also provides insights into how the generative model is trained, and how that affects stability and quality of the result in various settings such as image synthesis or other generative tasks.**

***Keywords*—Artificial intelligence; learning rate; cyber threat; network intrusion detection; deep learning; data augmentation; generative adversarial networks epochs**

I. INTRODUCTION

The emergence of Generative Adversarial Networks (GANs) marked a turning point in the area of deep learning due to its unparalleled capabilities in data synthesis. Despite their impressive skill set, these models face some limitations which are crucial in achieving reliable stability and performance.

These models have proven their strength in providing realistic and high-quality data in multiple areas such as degenerate image data, augmentation, and style transfer. Yet, in spite of the boom, industrial-scale applications still face limitations. The capability and regularity of the sheer raw force of conventional GANs remain undermined by flaws such as the mode collapse and training imbalance [1]. Conventional GANs, for instance, still encounter some critical limitations that restrain their efficacy. Collapse of modes – when a generator creates a fixed number of varieties – and instability during training, to mention a few. A remedy for these issues is proposed by the Wasserstein GAN with Gradient Penalty (WGAN-GP) which is considered more robust. This revision utilizes the Wasserstein distance between two probability distributions as the loss for the generator and adds a gradient penalty for Lipschitz constraints. Therefore, this model increases the stability of training and leads to reliable outcomes. WGAN-GP demonstrates better performing metrics with less hyper parameter tuning, it does have extreme sensitivity within certain parameters, particularly the learning rate and the number of training epochs. Setting parameter values too high or too low can greatly hinder the convergence behavior of the model, negatively impacting its overall efficiency, resulting in poor quality outputs. This research intends to explore the influence of learning rate and epoch numbers along with other performance metrics on generative models, especially WGAN-GP. With this study, the authors hope to shed light on the intricate web of hyper parameters and model metrics, enabling better optimization of GANs training processes. The goal of this study is to investigate the relationship of learning rates and epoch values on the performance metrics of machine learning models [2].

A. Research Objective

The objective of this study is to compare the various learning rate and epoch values in WGAN-GP models in combination with the augmentation percentage. To provide the best settings for training WGAN-GP models, this research aims to gain an understanding of how changing these hyper parameters affects the training process. This research systematically investigates the effect of significant hyper parameters on the performance of WGAN-GP, especially in terms of learning rate, data augmentation percentage, and epoch count, as summarized in Table I. The effect of different learning rates on the quality of generated data is analyzed to determine the optimal balance between training stability and convergence efficiency. A too high learning rate may lead to mode collapse or unstable training dynamics, and if the learning rate is low enough, convergence could be too long and model performance suboptimal. Moreover, the role played by the fluctuation in different percentages of data augmentation in terms of output variability of WGAN-GP was evaluated in quantifying the value added in generating diverse samples without overfitting. Moderate augmentation may augment the generalization, but if augmentation is excessive, it will add noise to samples, deteriorating their fidelity [13], [16]. In addition, the quality of the model's ability to generate, with varying epoch numbers, is analyzed to pinpoint where the continued training no longer improves the model qualitatively or results in overfitting. Under fitting and overfitting are traded off to optimize representation capacity for a model. A comparative analysis of multiple training configurations is conducted, identifying trends in stability, mode collapse, convergence rate, and overall data fidelity. These insights enable the formulation of a comprehensive understanding of the interplay between hyper parameters and model performance. Based on these findings, optimal hyper parameter choices for WGAN-GP training are recommended, emphasizing configurations that maximize output quality while maintaining training efficiency. The study highlights best practices for tuning WGAN-GP, ensuring robust and high-quality generative modeling, with a focus on mitigating instability and enhancing sample diversity. The resulting guidelines provide a structured approach to hyper parameter optimization, facilitating effective training of WGAN-GP across various data domains.

The present research study extends the earlier study [30] by optimizing the selection of hyper parameters with methodological consistency. In this study, 12 different learning rates were tested from 1.00E-03 to 100E-01, with a difference of 9.10E-03 between steps, in addition to two epoch values (100 and 150) and two augmentation percentages (30% and 50%), resulting in 48 experimental runs. Even with these adjustments, the experimental setup continues to be consistent with that of the earlier study, maintaining continuity while aiming for a more efficient and focused hyper parameter tuning. The augmentation percentage parameter is taken into account for the experimentation in contrast to the previous study being conducted, which further helps to analyze its behavior on the overall performance of the system. This work builds on the earlier study by modifying the learning rate interval and augmentation method, maximizing the experimental configuration for a more accurate performance assessment.

The aim of this study is to systematically evaluate how generative models, specifically WGAN-GP, are affected by learning rate, augmentation percentage and epoch values. WGANs with Gradient Penalty (WGAN-GP) fundamentals including effects of learning rate, epoch count on training dynamics, and the dataset used for experimentation were discussed in Section II. In this section, an explanation of the background concepts was also provided. The subsequent Section III marks the beginning of the results section, which provides the steps carried out in this study such as model building, setting the hyper parameters, and establishing the model benchmarking procedures. A comprehensive analysis is represented in Section IV, and Section V where results from the different experiments are highlighted through trends and parameter comparatives throughout the experiments. In the final Section VI of the study, a summary of primary lessons alongside outstanding research opportunities are incorporated within the conclusions.

II. Background

A. Background on WGAN-GP

GANs are generated by two deep neural networks, as shown in Fig. 1, such as the generator, which produces synthesized data and the discriminator, which evaluates the synthesized data generated by the generator. As for the discriminator, the authors of traditional GANs utilize binary cross-entropy loss to adequately separate real and fake data points, yet the implementation is problematic because of gradient vanishing. The GAN model named WGAN replaces the standard loss functions with the concept of the Wasserstein distance, giving a smooth gradient and hence better training ability. However, the original WGAN come along with some issues such as weight clipping that at sometimes is not efficient. To resolve this issue, WGAN-GP uses gradient penalty term to maintain the value of Lipshitz on the required range and thus model converges stability [3]. However, modern WGAN-GP model still suffer from the problem of hyper parameter sensitivity. Learning rate means the rate at which the model adjusts its parameters, which defines the stability and convergence of the model. If the learning rate is too high, the model may never or if the learning rate is too low, then the time taken to complete the modeling process is doubled. On the other hand, the number of epochs shows for how long duration model has been trained; if this number is less, the model may not learn much and it may be under-fit, and if otherwise, the model may over-fit or computer time may be too much [4].

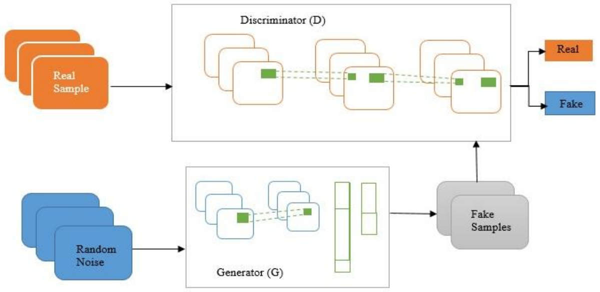

Fig. 1. Workflow of GAN.

B. Impact of Epoch on WGAN-GP

The number of training epochs in deep learning models, including WGAN-GP has a huge impact on the quality and stability of the produced outputs. An epoch means an iteration of the data set over the model's structure, which implies that the number of epochs determines how much the model learns the data. In decision-making during training, it is important to select the best number of epochs in order to get the best model without common problems, under fitting or overfitting. Learning with fewer epochs comes to the drawback as the generator is not fully trained to capture any realistic patterns in the data set [7]. This often results into production of poorly focuses, low quality or unrealistic output. The discriminator in WGAN-GP might also be weak, it does not offer the necessary feedback to enhance the generator. On the other hand, training continues for a number of epochs, degrades the general sample generation capability of the model as it overfits the training data. Overfitting causes the variation in the generated outputs to be low and might bring in artifacts that reduce the quality.

In particular, the number of epochs depends on such factors as the given data density, the structure of the generator and discriminator, and the available computing capabilities. Original experiments have revealed that for various types of GAN-based models including WGAN-GP the quality is highest at a particular number of epochs, and in fact degrades in specified epoch values. This happens due to the fact that the balance between the generator and discriminator is less optimal for successful training [8], [10]. This work provides a comprehensive analysis on the performance change of WGAN-GP based upon various epoch settings. In this case, the study seeks to pinpoint epoch range that effectively address the problem by comparing loss trends, sample quality, and training stability, while at the same time avoiding pitfalls such as mode collapse, overfitting, or inefficient training.

This research will seek to establish the effect that learning rates and epochs have on the performance of Wasserstein GAN with Gradient Penalty (WGAN-GP). In order to accomplish the research goals, systematic experiments involving a proper generative dataset and a strong model deployment system will be performed. The applied approach includes choosing the dataset, setting up of the WGAN-GP model, conducting of the experiments, and qualitative and quantitative assessment of the results as shown in Fig. 2.

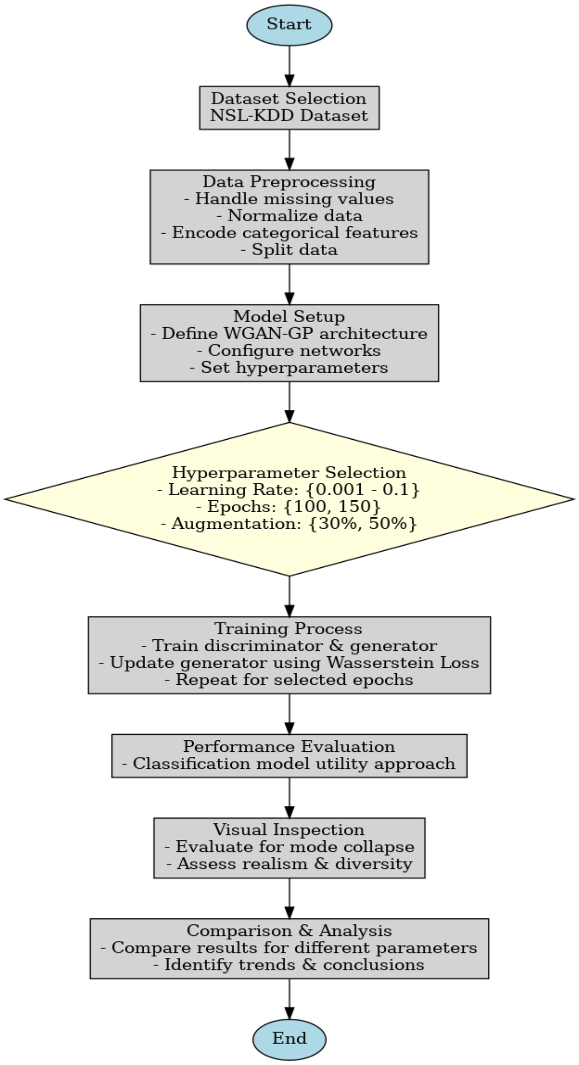
Start
Dataset Selection
NSL-KDD Dataset
Data Preprocessing
- Handle missing values
- Normalize data
- Encode categorical features
- Split data
Model Setup
- Define WGAN-GP architecture
- Configure networks
- Set hyperparameters
Hyperparameter Selection
- Learning Rate: {0.001 - 0.1}
- Epochs: {100, 150}
- Augmentation: {30%, 50%}
Training Process
- Train discriminator & generator
- Update generator using Wasserstein Loss
- Repeat for selected epochs
Performance Evaluation
- Classification model utility approach
Visual Inspection
- Evaluate for mode collapse
- Assess realism & diversity
Comparison & Analysis
- Compare results for different parameters
- Identify trends & conclusions
End

Fig. 2. Experimental process for evaluating WGAN-GP performance.

C. Impact of Learning rate in WGAN-GP

The learning rate is one of the key hyper parameters to deep learning models including the Wasserstein Generative Adversarial Networks with Gradient Penalty (WGAN-GP) [22]. Controls the size of weight update at back propagation and has a direct impact on speed of convergence, stability of the model and final performance. In order to obtain high quality generative results, and to ensure a stable training process, the choice of an adequate learning rate is crucial.

However, in WGAN-GP, an accurate learning rate helps the generator and discriminator learn well without introducing instability. A small learning rate means that it takes many epochs for the network to arrive at satisfactory performance. Although this improves the stability of the model, it also has some negative effects of high computational costs and thus time-consumptive [5], [19]. On the other hand, high learning rate will result in large weight updates, which results in very unstable learning, instability or even learning diverges. If the learning rate of the model is set to a wrong value, the generative model may not be able to learn valuable representations hence the generated outputs will be of low quality with some artifacts or mode collapse.

Several strategies have been proposed to address the issue of learning rate setting in deep learning. Some of the methods used include; learning rate scheduling, adaptive learning rates, and cyclical learning rates which enhances efficiency of the training progress. However, the optimal learning rate is still an issue of contention regarding the WGAN-GP due to its delicate training dynamics [6]. This research explores the performance of WGAN-GP in terms of convergence with some of its learning rates altered in order to evaluate the quality of the generated outputs. Thus, because of the systematic variation and examination of the learning rate in this study, it is expected to determine the adequate learning rate that leads to the improvement of the stability, speed and generative performance of the WGAN-GP.

D. Dataset Selection and Preprocessing

This study has chosen such datasets that are used the most in network anomaly detection, i.e. NSL-KDD [31]. This dataset is widely used for benchmarking the IDS, which is the reason for the use of this dataset. Also, it is suitable for academic research and to build a proof-of-concept models. That will be another advantage of using NSL-KDD, as it is compact and not complex to implement as compared to a fully-fledged model.

It does not include encrypted traffic or emerging attack types. The NSL-KDD dataset is an improved version of the KDD Cup 1999 dataset, which was developed due to the problems such as redundancy and class imbalance [9]. Thus, it is useful in improving and enhancing the standard for measuring the efficiency of the Network Intrusion Detection System (NIDS). The dataset provided here in KDD Cup 1999 contains a huge number of duplicate instances, which becomes too noisy during training of the machine learning model [12], [14], [26]. Because of this, NSL-KDD has an advantage, whereby most of the record duplication instances are considered as redundant and removed. The removal of such duplicated records leads to more optimized and refined data of the network traffic by increasing the capacity of evaluating the performance of NIDS.

There are 41 features in the NSL-KDD that were used as an ability to capture the characteristics of the network traffic. Some of these characteristics are certain connection details, timing details, the number of bytes that have been transferred and the actual payload content that has been transposed. These are in turn grouped in structural, content, time-based and host-based that aids in achieving enhanced statistical and machine-learning methods employed in intrusion detection. Therefore, the improvement over KDD-NSL concerning the reduction of loop redundancy and the balanced proportion of connections in different classes improves the ability to identify frequent and infrequent attacks in NIDS [11], [15]. This type of dataset has been widely used in the research to develop and evaluate new algorithms and, therefore, this kind of dataset is more appropriate for the application of network security and the detection of intrusion.

III. Methodology

A. Model Initialization

K-Nearest Neighbor (KNN) is a nonparametric, supervised learning approach commonly used for classification and regression purposes. This occurs through identifying the ‘k’ closest points referred to as neighbors in the featured space compared to an input point. Moreover, it offers predictions either on the majority class in the case of classification or the averages of their values in case of regression. These distance metrics include Euclidean distance metrics, Manhattan distance metrics, Minkowski distance metrics as well.

KNN is particularly useful in detecting an anomaly as it can be seen to be accustomed to identifying data points that are far from the nearest neighbors of the given point. On the contrary, abnormal observations seem to have averagely or significantly fewer or a significantly higher number of neighbors compared to normal observations. KNN’s are very handy, especially when it comes to handling multidimensional data and do not involve a lot of implementation complexities. That is; K= number of records in the training section analyzed to calculate the Euclidean distance for one record to all records in the training section and find a winner, K + 1= number of records stayed in the training section during the calculation of the distance between that record and all records in the training section and find the winner and member number of the record set of the training section.

$$\boldsymbol{d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}} \qquad (1)$$

where,

d is the distance between two points x and y in an n-dimensional space.

The formula for calculating the anomaly score is given as:

$$Anomaly\ Score(x) = d\left(x, x_{(k)}\right) \qquad (2)$$

where,

$x_{(k)}$is the kth nearest neighbor of (x).

B. Hyper parameter Configuration

Consequently, the aim of this study is to investigate the influence of learning rate and number of epochs on the accuracy of the WGAN-GP model. Therefore, the present research will feature plans to experiment with both of these two hyper parameters in an attempt to determine the impact of altering each of them on the required standards of the generated outputs. These are the learning rate that will also be altered to understand the effects it has when varied in its options, and number of epochs. The following is the summary of the hyper-parameters:

- Learning Rate: Determines change in weight per iteration and improves speed/stability of training.
- Num Epochs: Defines the number of times, for how many times the model is go to see the entire data while training.
- WGAN Data Augmentation Percentage: how much data is generated from the original data?

TABLE I. Hyper parameter Configuration

Total Instances	Learning Rate Range	Step Difference	Epoch Values	Augmentation Percentages
48	1.00E-03 to 1.00E-01	9.10E-03	100, 150	30%, 50%

C. Evaluation Metrics and Qualitative Assessments

Both quantitative and qualitative approaches will be adopted in measuring performance as a means of having a balanced research outlook in the bid to establish the extent to which the model can create realistic synthesized data. In the present research context, Wasserstein Loss is one of the measures taken into consideration. It is used to measure the generative data, and it is core to the assessment of convergence of WGAN-GP with the actual data distribution. Another benefit of Wasserstein Loss is that compared to cross entropy loss, Wasserstein Loss provides a continuous gradient that helps a model steadily improve and generate many nearly realistic samples. As a result, this loss should be monitored to determine whether the model is learning and converging over time. The experiment’s flow will be organized and conducted systematically in order to compare the effects of varying learning rate and epoch settings on WGAN-GP’s performance accurately. To start with, the datasets that are to be used for training, which are NSL-KDD, will be preprocessed. After that, the training process will be taken over by varying the values of learning rate and number of epochs, and the model will be trained for each setting of both hyper parameters. In each iteration, the WGAN-GP model will work in turn to update both the discriminator and the generator [17], [18]. Every epoch contains several steps that include a discriminator to recognize sampled real and generated data, and update the generator based on this discriminator feedback. The above-described procedure of training will go on until the model is trained up to the possible maximum epochs or for a fixed epoch. It will then be conducted for all the selected learning-rate and epoch values in order to establish the relationship between these parameters and model’s performances in producing realistic outputs.

The Classification Model Utility approach will be used to assess the model’s performance after every training run. This approach is where the model is trained on the original data set while the other model is trained on the original and synthetic data set and both models are tested on the same validation set. A higher performance on the validation dataset proves the synthetic data’s consistency and usefulness [29], [30]. This step will assist to detect flaws, for example, in the type of mode collapse, degenerate textures, or limited variation of the synthesized data produced. Adding these visual inspections to the quantitative measures will provide greater insight on the model’s behavior and execution. After all the training runs have been performed, one will be able to compare the learn rate and epochs thus getting the needful and expected outcomes. This comparison will involve comparing the effects of altering the hyper parameters in the performance of the adjusted model. The identification of trends in the quantitative metrics, coupled with the results of the visual inspections, will aid in determining the most appropriate learning rate and epochs to be used to prevent overfitting or insufficient training. This will be done numerically as well as in terms of the final generated samples to understand the impact of these hyper parameters when changing in WGAN-GP.

D. Performance Metrics

The parameters applied on WGAN-GP for examining the performance are as follows:

Accuracy (ACC): It is considered as the most important parameter in evaluating the model's performance. This metric evaluates the quantity of number of samples that are correctly predicted over the number of all samples. The formula for calculating this metric is:

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

Recall: This parameter refers to the capability of the machine-learning model for predicting positive samples. It is calculated by dividing the number of samples that are categorized as true positive over all positive samples. The formula for calculating this metric is:

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

Precision: In this parameter, true positive identified number of samples over number of samples that are predicted as positive. The equation for calculating this metric is:

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

IV. Experimental Results

In cyber threat detection the effectiveness of adaptive generative data augmentation is evaluated by a series of experiments, which were conducted on different augmentation series and training configurations. This study explores the impact on model performance occurs due to different settings of hyper parameters. Additionally, it focusses on generalization and optimization of the dynamics of model performance.

TABLE II. Performance Metrics of WGAN-GP Across Different Hyper parameter Configurations

Performance Metrics			**Hyper Parameter Configuration**		
Accuracy	**Precision**	**Recall**	**Learning Rate**	**EPOCH**	**Augmentation Percentage**
0.856822	0.870409	0.85682183	1.00E-02	100	50.00%
0.85297	0.855726	0.852969502	8.20E-02	150	50.00%
0.815516	0.840894	0.815516319	2.80E-02	100	50.00%
0.812413	0.812486	0.812413055	3.70E-02	150	50.00%
0.74646	0.768186	0.746459782	3.70E-02	150	30.00%
0.711932	0.79942	0.711931514	4.60E-02	100	50.00%

Table II depicts the better performance on the specific hyper parameter configurations than the original dataset. It shows that out of 48 instances, the mentioned combinations have showed a significant increase in performance metrics as compared to the model performance on original dataset before the data augmentation. The data shown in Table II indicates the highest learning rate of 1.00E-02 at 100 epochs and 50% data augmentation, suggesting an optimal balance for model generalization.

Graphical presentation of these data is shown below in Fig. 3, which shows that the 50% augmentation covers a band while 30% augmentation shows a significantly lower recall which reinforces that model performances can be enhanced if sufficient data augmentation (50%) is used.

Fig. 4 suggests that training without proper learning rate may leads to diminish returns. The trend for recall also suggests the same and indicates optimal performance in 100 epochs with well optimized learning rate. As shown in Fig. 4, it is found that the highest level of accuracy is achieved at epoch for both levels of data augmentation. Following this, accuracy begins to deteriorate slightly, this indicates overfitting or the training has been carried out to the extent that the model is memorizing the set data instead of just learning. This occurs since, with a lot of training, the model is tasked with memorizing the training data and not learn general patterns that may occur in other real data [21].

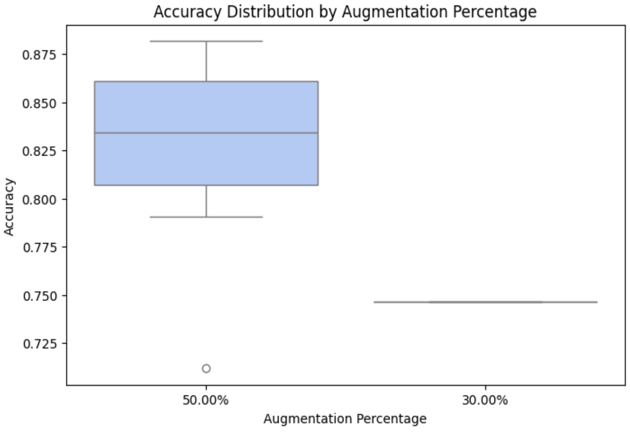

Fig. 3. Accuracy distribution by augmentation percentage.

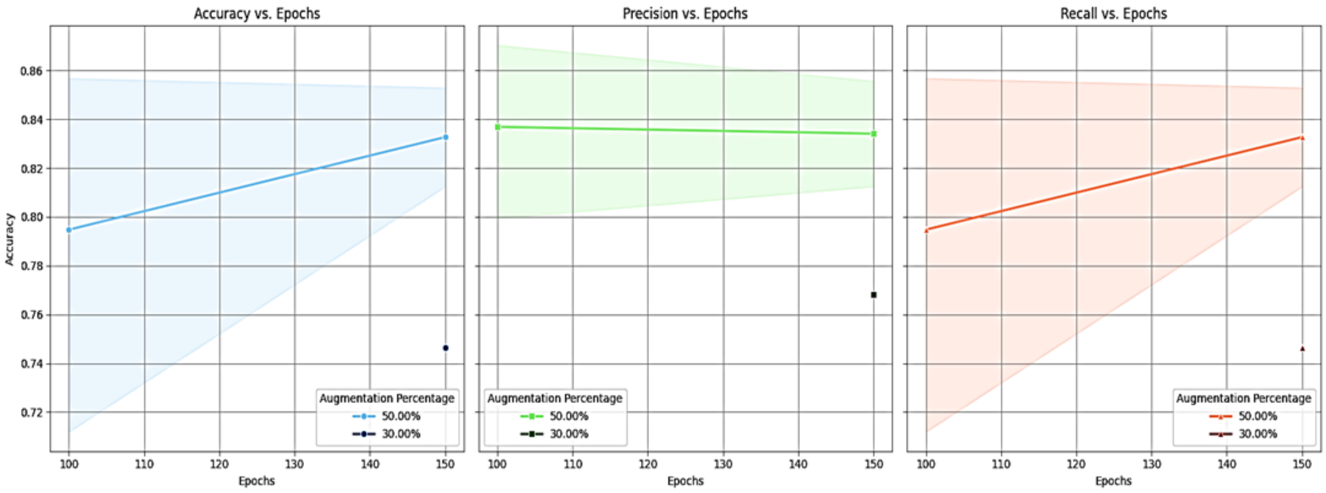

Fig. 4. Comparative view of accuracy, precision and recall with epochs.

As for configurations, the one of data augmentation equals to 50% provides better results in most cases in terms of accuracy compared to 30% augmentation. What this entails is that inputting a broader and comprehensive variety of augmented data aids in the model's performance in terms of generalization. Even when it comes to precision, the aim of having 50% of servings augmented is seen to be the best. As can be noticed, precision does not decrease with time, and therefore, it is stable in different epochs. This implies that a reduced 50% augmented model yields a fewer number of false positive as compared to the 30% augmented model. Fewer false positive means that the model is correctly segregating between relevant and irrelevant samples which is extremely important when the classification is done, and leads to certain results [20].

A higher level of dispersion in the accuracy values is noted as shown in Fig. 5 and Fig. 6, which suggests variability in the performance of models in the different configurations. The median has been increased compared to the 30 percent augmentation result which affirms the benefits of incrementing the augmentation percent. Nonetheless, there are a few points with the accuracy below the average, which indicates that the higher level of augmentation does not have as a positive effect on certain configurations. Nevertheless, a majority of the accuracy values have a higher stage, which provides evidence that 50 percent augmentation ratio still helps to enrich the model.

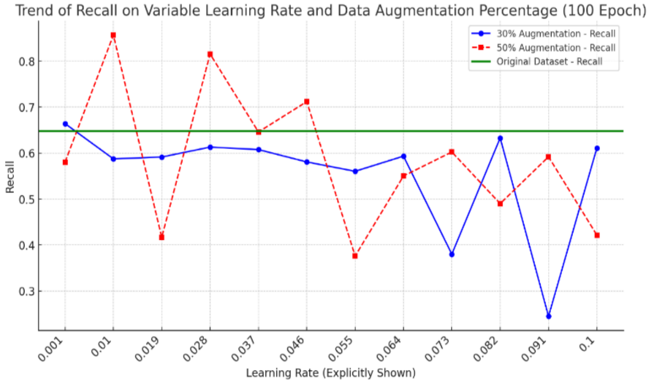

Fig. 5. Trends of Accuracy on variable learning rate and data augmentation with 100 epochs.

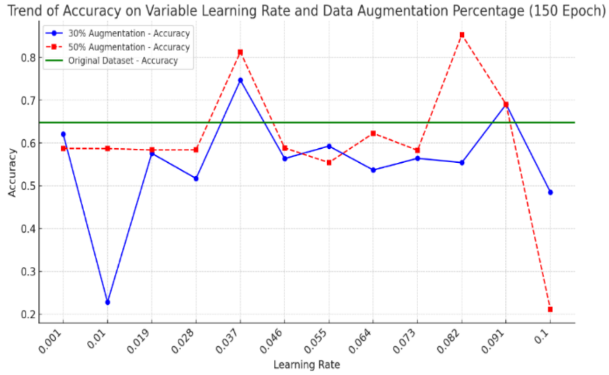

Fig. 6. Trend of accuracy on variable learning rate and data augmentation with 150 epoch.

From Fig. 5 and Fig. 6, the accuracy for 30% augmentation is lower compared to the previous results, which indicates less efficiency. Therefore, all the estimated accuracy values are less than in the 50% augmentation scenario. The box plot shows the minimal variation, proving that lower levels of augmentation only bring a marginal improvement in performance. In this case, the fact that there are few changes foretells that an augmentation by 30% does not lead to enhancing the model's ability to generalize or its robustness.

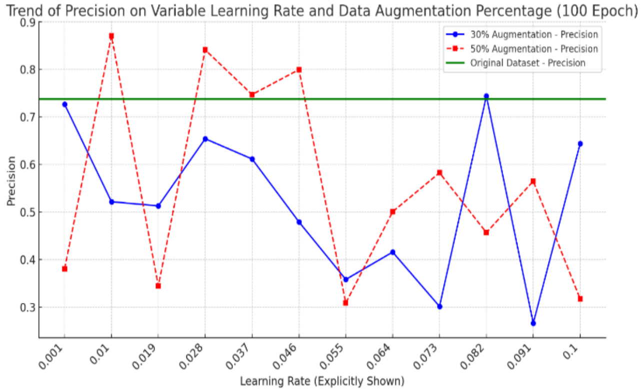

Fig. 7. Trend of precision in variable learning arte and data augmentation with 100 epoch.

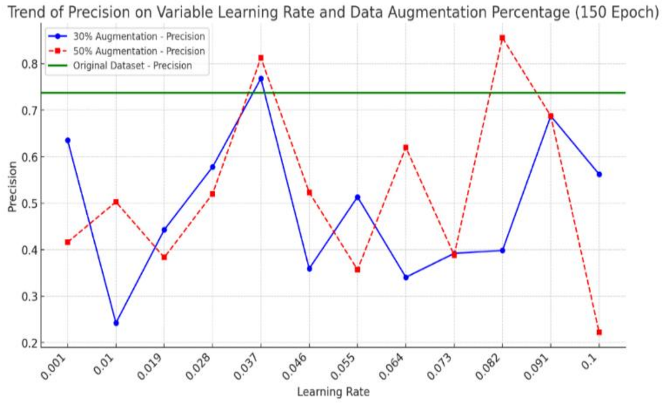

Fig. 8. Trend of Precision on variable learning rate and data augmentation with 150 epoch.

Fig. 7, and Fig. 8, present the trend of precision for various learning rates and augmentation levels of data (30% and 50%) with two specific epoch values. A drop at a learning rate of 0.053 indicates an unstable zone where performance drops in the model. The instability possibly stems from poor convergence or too much weight update, causing non-optimal learning. As the learning rate diverges from this unstable region, accuracy becomes stable, illustrating the significance of having a suitable learning rate.

In 100 Epochs, Fig. 9, recall for a 50% increase in most learning speeds, except for a noticeable peak of 0.082, is still relatively stable. This suggests that the medium learning speeds combined with high growth levels help the model maintain more positive examples. In contrast, a 30% increase shows more fluctuations, sharp falls of 0.01 and a gradual medieval recovery (0.046 to 0.082). Increased volatility at low growth levels indicates that insufficient data text affects the model's ability to normalize. With a high teaching rate (0.1), recall means both growth percentages, indicating the model's instability. In 150 Epochs, Fig, 10, the recall is improved and is stable compared to 100 epochs, reflecting the benefits of extended training. Extending the training period to 150 epochs, as depicted in Fig. 10, results not only in an enhancement in recall performance, but also significantly sharpens the consistency relative to the scenario with 100 epochs.

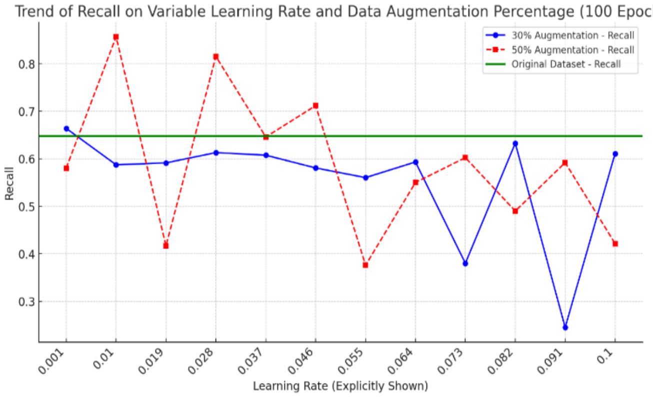

Fig. 9. Trend of recall on variable learning rate and data augmentation percentage with epoch 100.

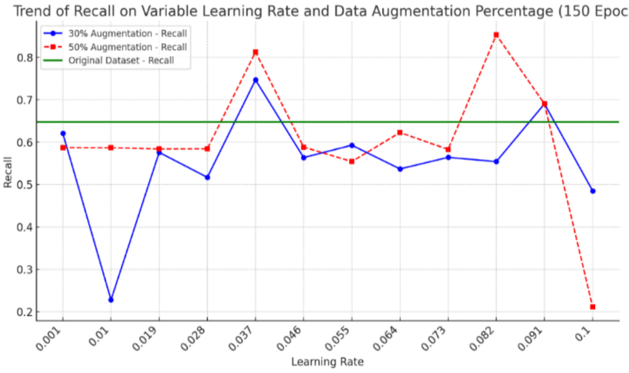

Fig. 10. Trend of recall on variable learning rate and data augmentation percentage with epoch 150.

V. DISCUSSION

The interaction among learning rates, epoch numbers, and levels of data augmentation is important to get a better understanding of the research objective. The results presented in the above section provided deeper insights into the behavior of the model under diverse training scenarios.

As shown in F II, some hyper parameter settings perform much better than the model trained on the original dataset alone without augmentation; of 48 instances tested, some combinations, especially those using more data and learning rates fine-tuned, showed significant improvement in performance. Specifically, the setup with a learning rate of 1.00E-02, 100 epochs of training, and 50% data augmentation produced the most beneficial outcomes, which suggests the best balance between training depth and model generalization. These findings further stress the need for well-chosen training parameters in improving the capability of the model to learn from and accommodate more varied data.

In cyber threat detection, an effective combination of moderate learning rate, controlled epochs and higher augmentation is crucial for optimization. In terms of accuracy and precision, the trend suggests that 100 epochs yield better performance than 150 epochs when paired with optimize learning rate.

The efficiency of the model is enhanced when trained for 150 epochs compared to 100 epochs to prove the model learnt sufficiently more epochs as training epochs increases. Further on in training, another important characteristic of the model develops in that the model is able to comprehend more complex patterns and representations, thus the accuracy, precision and recall will be improved. This means that the differences between 30% and 50% augmentation are as follows: The readers are as follows: But it was also observed that both augmentation percentages improve with an increase in training time, while 50% augmentation performs better than 30% augmentation in most cases, which ends the thought that higher synthetic data helps in improving the generalization of the model. Further, for all the evaluation criteria and epochs as well as augmentation level, it is found that all the Q-Learning rates in the mid-range categorized between 0.037 and 0.082 are the best performing one. These discoveries show that there is a factor of the training period and rate to be considered in order to achieve the best performance of the model. There is an improvement in all of the metrics, accuracy, precision and recall, if 50% is augmented instead of 30%. The fact that more augmentation allows for the creation of more various samples in the synthetic data, decreases overfitting and increases generalization of the model when it is trained on it. The improvement is most significant at a moderate learning rate, where the augmented data is valuable in providing the model with the right guiding patterns without the inclusion of noises. This means that in this particular dataset and for this particular task, a higher level of augmentation is beneficial on the model. As indicated, the learning rates between 0.037 and 0.082 are suitable for the high augmentation level and overall, provide the best results. These values determine the trade-off of fast convergence or stable convergence in a learning process. Too small learning rates such as 0.001 to 0.01 make the learning slow and ineffective, mostly due to slow convergence. Taking such small steps means that the model may take too long in order to learn meaningful features during the training phase which results in poor performance even when the number of iterations is increased. However, if the learning rate = 0.1, this causes instability and degradation of the performance. It stirs difficulty to reach nadir solution, as large update alters the weights radically, resulting in impulsive degradation of accuracy, precision and recall. This goes to support the argument that the learning rate should be properly selected and small to large values should be avoided, while small to large values could take a long time to converge and are somewhat unstable. The model tested for 150 time passes through the training set and, therefore, it demonstrates an increase in its performance that should be emphasized as a result of protracted training, particularly for large sets. More training iterations enable the model to perform very effectively in analyzing and identifying patterns in the data and thus increases the accuracy, precisions, and recall. However, as it has been demonstrated, more sophisticated training aids in the improvement of the learning rates but their tuning is still essential in order to avoid overfitting as well as instability. Selecting a bad learning rate, even if training is conducted for a long time, can have a negative impact on the performance. Consequently, it was identified that to obtain the best performance, the number of epochs must be at least 150 along with the best learning rate. Accuracy, precision, and recall exhibit similar trends across different learning rates, augmentation levels, and epoch counts. This ensures the accuracy of the analysis as it shows that the changes are not arbitrary but due to the model's response to various conditions. This fact implies that the model performs well across all three metrics because there is no extreme focus on one of the metrics while ignoring the other two. From the above analysis, it can be recommended that the following measures should be taken in order to enhance the performance of a model. First, a 50% of augmentation should be applied to decrease the model variance and subsequently improve generalization to all the metrics. Secondly the learning rate should be chosen in a certain range 0.037 to 0.082 in order to decrease speed of convergence and increase stability. Lastly, training should be carried out to the 150-epoch level in order to achieve the best results, especially where there is large data or complicated structures. That being said, the following recommendations will develop better performing workflow while keeping it stable and efficient to an extent.

Therefore, based on accuracy, precision, and recall, 50% augmentation is better than 30 % augmentation. The higher level of the data augmentation brings more diversified data samples that increase the ability of the model to generalize. This is very well illustrated in the convergence which shows that for all the epochs, 50% augmentation provides higher scores than models trained with other levels of augmentation. Also, it can be seen that standard deviations in 50% augmentation are distributed over a wider range showing more versatility that can be advantageous in practical applications across different settings. The other benefit that could easily be observed in the implementation of 50% augmentation is enhanced recall. As recall is about the model's ability to identify all required cases, the improvement indicates that a higher augmentation level includes more relevant changes in the training data [24]. Consequently, the model becomes better at recognizing positive cases as well as does not overlook any vital patterns. This will prove helpful in situations where the false negatives are debilitating, for example, in medical diagnosis or checks on fraudulent individuals. On the other hand, the increased ratio of just 30% does not seem to give the model a large enough variety of samples to learn adequately. Overall, the augmentation percentage is generally below what is obtained with a 50% augmentation number. Additionally, referring to entropy analysis and the trends in precision and recall, it can be concluded that the increase does not exceed 30% and thus cannot produce enough variation in order to achieve higher results. This shows that the model trained with 30% augmentation might have a difficulty in learning proper representations that can generalize well. However, the lower recall and precision observed in 30% augmentation suggest that the model is more error-prone. As shown from the results, the model entails lesser parameters hence, it fails to capture a large amount of data, which leads to higher percentage of false negatives, and it also has fewer capabilities of classifying relevant data from irrelevant data. Such issues indicate that 30% augmentation may not be the optimum solution for applications that require high reliability. Another noteworthy discovery revealed in this comparison is that precision is steady even as the epochs increase for 50% augmentation. It is clear from the above findings that the precision rate is still higher in models that have been trained with an augmentation of 50% as compared to that of 30%. This means that the model is more accurate in avoiding false positives which are very important in real-life scenarios due to the impact of wrong classification [25], [27].

In overall the result of every epoch shows that by increasing augmentation percentage, the precision rate is also more stabilized. The recall values also are exhibited in the same manner as accuracy. Thus, the performance scores for recall in both augmentation levels initially rises and after that reduces from the epoch of training. As it may be inferred, a higher augmentation percentage increases that kind of performance, but that extensive epochs diminish the ability to correctly classify positive samples. Hence, the results suggesting that recall of 50% was obtained with 50% augmentation supports the point about appropriate augmentation resulting in improved sampling of the whole population. Considering the accuracy, precision, and recall trends over time, it is clear that precision is a relatively stable metric where values fluctuate the least, although settings such as the '50% augmentation setting' show marginal downs and ups. Both accuracy and recall present some of the fluctuation though, whereas, the precision has ameliorated more or less in a general improved trend. This stability also shows the advantage of using a higher augmentation percentage because the models over emphasizing does not deteriorate such aspects [23], [28]. However, it is crucial to remember that after certain epoch level, the accuracy and recall become less efficient, which again reflects that there are more other steps such as early stopping to prevent overfitting. These three factors offer significant insight into the best setup for model training. The first important conclusion is that the increase of the number of samples by 50% also increases precision because it is the ratio of accurately identified positive cases. The study also notes that there is a tendency to overfit the training data when the epochs are trained beyond the threshold value. After this point, both recall and accuracy surface as having a decreased rate, which indicates that the oversimplification of training is detrimental. It is noted that this situation calls for either the use of early stopping techniques or the learning rate changes for the best results.

The evaluation based on Fig. 5 and Fig. 6, suggests that 50% augmentation is better in terms of median accuracy and variability as the method deploys fewer units at once but has a higher potential by maintaining precision across various configurations. The variability of accuracy seems to go up, and it means that higher levels of augmentation aid in enhancing the model's flexibility. On the other hand, a narrower range of accuracy in the 30% augmentation scenario also suggests that the augmentation is not very effective in enhancing the generality and stability of performance. Therefore, the use of 50% augmentation can also be seen to be more effective in improving the performance of the models than 30% augmentation.

The analysis of recall performance within various learning rates and data growth factors for different training durations offers important understanding into the characteristics and stability of the WGAN-GP model. Recall, for the most part, remains stable in the 50% data growth mark for most learning rates with a peak at 0.082. This is especially the case for intermediate learning rates, which appear to be more favorable when combined with higher data availability, as the model’s ability to recall and recognize salient examples improves. In comparison, the volatile recall performance at a 30% growth rate, with sharp subtractive spikes down 0.01 and gradual gains between 0.046 and 0.082, suggests that lower data growth levels might be inefficient, where insufficient training signals might hinder model generalizing and normalizing ability. Additionally, extreme learning rates, such as 0.1, invoke unstable recall at both growth percentages, highlighting the issue of structured adaptation under extreme learning conditions.

Higher levels of data augmentation also enhance generalization, minimizing overfitting and increasing robustness. Between the two augmentation approaches, 50% augmentation uniformly produces better accuracy at most learning rates and hence is the model of choice for stability and effectiveness of the model. From Fig. 9, and Fig. 10, the 50% increase improves an increase of 30% at all learning speeds continuously, strengthening the efficiency of improving the models' ability to capture positive examples. The best recall performance is seen in mid -range learning speeds (0.046 to 0.082), and corresponds to trends seen in accuracy and learning. However, recalling is still experiencing a sharp decline in the highest learning frequency (0.1) similar to other performance matrix. This further emphasizes that models prevent very high learning speeds instead of increasing performance.

VI. Conclusion

In order to identify the best hyper parameter configurations for reliable and effective training, this study methodically investigates the effects of learning rate, epoch count, and data augmentation percentage on the performance of WGAN-GP models. This study highlights that excessively low learning rates slow convergence and result in suboptimal model performance, while high learning rates can lead to mode collapse and unstable training dynamics. It does this by examining a range of learning rates and determining the crucial balance between training stability and convergence efficiency. The study also explores how data augmentation can improve sample diversity and generalization, showing that while excessive augmentation introduces noise and reduces the fidelity of generated data, moderate augmentation improves the model's capacity to generalize. In order to identify the ideal point at which further training stops improving sample quality or results in overfitting, the impact of epoch count on model performance is also evaluated. Maximizing the representation capacity of WGAN-GP models requires finding the ideal balance between under-fitting and overfitting. Deeper understanding of stability, mode collapse, convergence rates, and the general fidelity of generated data can be gained by comparing various training configurations in the future studies. This study also emphasizes the need for integrated hyper parameter tuning because learning rate, augmentation, and epoch count all affect training dynamics. The results provide specific suggestions for choosing the best hyper parameters, guaranteeing that WGAN-GP models produce output of higher quality with increased stability and effectiveness. In real-world use cases, including healthcare, finance, and autonomous systems, stability and adaptability are as important as high accuracy. This 50% augmentation level combined with tuned hyper parameters seems to be a suitable solution for optimized performance with any configuration. By establishing best practices for tuning WGAN-GP, this study provides a structured approach for optimization of hyper-parameters, offering a robust framework for training diverse data domains in generative models.

Even with the insightful findings of this research, there are a number of limitations that must be noted. The analysis was based on a limited set of learning rates, epochs, and augmentation levels, which might not completely represent the behavior of WGAN-GP models on different datasets or more sophisticated data domains. Moreover, the research was based on traditional augmentation methods, excluding more sophisticated options like synthetic sample creation or domain-specific transformations that might have had varying results. The hyper parameter tuning was similarly performed manually without using automated optimization methods like grid search or Bayesian optimization, which may provide more accurate configurations. These constraints points to the necessity of wider experimentation and more adaptive tuning approaches in subsequent research to further improve the generalizability and stability of WGAN-GP models.

Future studies should similarly compare more varying augmentation levels in order to find out if there is any improvement. Moreover, future explorations should focus on more sophisticated augmentation approaches like: generating synthetic samples, and geometric transformations, that ideally could improve both adaptability and stability of the models when trained on various datasets. The learning rate determines speed at which the model travels through the learning space and hence, correct setting of this factor is important so that the model does not oscillate uncontrollably. Advanced variations like grid search or Bayesian optimization should be employed in future studies to determine the suitable learning rate that would enable fast convergence while at the same time avoiding premature convergence.

REFERENCES

[1] X. Ouyang, Y. Chen, and G. Agam, “Accelerated WGAN update strategy with loss change rate balancing,” in Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis., 2021, pp. 2546–2555.

[2] Q. Zhou and B. Sun, “A Gaussian-based WGAN-GP oversampling approach for solving the class imbalance problem,” Int. J. Appl. Math. Comput. Sci., vol. 34, no. 2, 2024.

[3] Y. Lu, X. Tao, N. Zeng, J. Du, and R. Shang, “Enhanced CNN classification capability for small rice disease datasets using progressive WGAN-GP: Algorithms and applications,” Remote Sens., vol. 15, no. 7, p. 1789, 2023.

[4] D. Kavran, B. Žalik, and N. Lukač, “Comparing Beta-VAE to WGAN-GP for time series augmentation to improve classification performance,” in Int. Conf. Agents Artif. Intell., Cham: Springer Int. Publ., Feb. 2022, pp. 51–73.

[5] T. Zhang, Q. Liu, X. Wang, X. Ji, and Y. Du, “A 3D reconstruction method of porous media based on improved WGAN-GP,” Comput. Geosci., vol. 165, p. 105151, 2022.

[6] S. Hejazi, M. Packianather, and Y. Liu, “A novel approach using WGAN-GP and conditional WGAN-GP for generating artificial thermal images of induction motor faults,” Procedia Comput. Sci., vol. 225, pp. 3681–3691, 2023.

[7] J. Lee and H. Lee, “Improving SSH detection model using IPA time and WGAN-GP,” Comput. Secur., vol. 116, p. 102672, 2022.

[8] S. Westberg, Investigating the Learning Behavior of Generative Adversarial Networks, 2021.

[9] J. Hu and Y. Li, “Electrocardiograph based emotion recognition via WGAN-GP data enhancement and improved CNN,” in Int. Conf. Intell. Robot. Appl., Cham: Springer Int. Publ., Aug. 2022, pp. 155–164.

[10] L. Abou-Abbas, K. Henni, I. Jemal, and N. Mezghani, “Generative AI with WGAN-GP for boosting seizure detection accuracy,” Front. Artif. Intell., vol. 7, p. 1437315, 2024.

[11] J. Mi et al., “WGAN-CL: A Wasserstein GAN with confidence loss for small-sample augmentation,” Expert Syst. Appl., vol. 233, p. 120943, 2023.

[12] D. Srivastava, D. Sinha, and V. Kumar, “WCGAN-GP based synthetic attack data generation with GA based feature selection for IDS,” Comput. Secur., vol. 134, p. 103432, 2023.

[13] T. Jiang, C. Shen, P. Ding, and L. Luo, “Data augmentation based on the WGAN-GP with data block to enhance the prediction of genes associated with RNA methylation pathways,” Sci. Rep., vol. 14, no. 1, p. 26321, 2024.

[14] R. Bhat and R. Nanjundegowda, “A review on comparative analysis of generative adversarial networks’ architectures and applications,” J. Robot. Control (JRC), vol. 6, no. 1, pp. 53–64, 2025.

[15] S. Rana, S. Gerbino, and P. Carillo, “Comparative analysis of modified Wasserstein generative adversarial network with gradient penalty for synthesizing agricultural weed images,” 2024.

[16] M. Ryspayeva, “Generative adversarial network as data balance and augmentation tool in histopathology of breast cancer,” in Proc. IEEE Int. Conf. Smart Inf. Syst. Technol. (SIST), May 2023, pp. 99–104.

[17] S. Yean, W. Goh, B. S. Lee, and H. L. Oh, “extendGAN+: Transferable data augmentation framework using WGAN-GP for data-driven indoor localisation model,” Sensors, vol. 23, no. 9, p. 4402, 2023.

[18] Y. Zhang, Y. Xue, and F. Neri, “Multi-optimiser training for GANs based on evolutionary computation,” in Proc. IEEE Congr. Evol. Comput. (CEC), Jun. 2024, pp. 1–8.

[19] K. Li and D. K. Kang, “Enhanced generative adversarial networks with restart learning rate in discriminator,” Appl. Sci., vol. 12, no. 3, p. 1191, 2022.

[20] M. Anderson, M. Smith, and J. Doe, “2D-to-3D image translation of complex nanoporous volumes using generative networks,” Sci. Rep., vol. 11, no. 1, pp. 1–12, 2021.

[21] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and L. O. P. Courville, “Improved training of Wasserstein GANs,” arXiv preprint arXiv:1704.00028, 2017.

[22] Y. Li and Z. Kang, “Enhanced generative adversarial networks with restart learning rate in discriminator,” Appl. Sci., vol. 12, no. 3, pp. 1191, 2022.

[23] M. Tajmirriahi, A. M. K. Alavi, and R. M. K. Alavi, “A dual-discriminator Fourier acquisitive GAN for generating retinal optical coherence tomography images,” IEEE Trans. Instrum. Meas., vol. 71, pp. 1–10, 2022.

[24] F. Fajar, “Cyclical learning rate optimization on deep learning model for brain tumor segmentation,” IEEE Access, vol. 11, pp. 3326475–3326484, 2023.

[25] J. Hui, “GAN—What is generative adversarial networks (GAN)?,” Medium, Jun. 20, 2018. [Online]. Available: https://jonathan-hui.medium.com/gan-whats-generative-adversarial-networks-and-its-application-f39ed278ef09.

[26] D. Srivastava, D. Sinha, and V. Kumar, “WCGAN-GP based synthetic attack data generation with GA based feature selection for IDS,” Comput. Secur., vol. 134, p. 103432, 2023.

[27] X. Zhou, “Research on network intrusion detection model that integrates WGAN-GP algorithm and stacking learning module,” Int. J. Comput. Syst. Eng., vol. 8, no. 6, pp. 1–10, 2024.

[28] M. G. Constantin, D.-C. Stanciu, L.-D. Ştefan, M. Dogariu, D. Mihăilescu, G. Ciobanu, and M. Bergeron, “Exploring generative adversarial networks for augmenting network intrusion detection tasks,” ACM Trans. Multimedia Comput. Commun. Appl., vol. 21, no. 1, pp. 1–19, 2024.

[29] G. Abdelmoumin, J. Whitaker, D. B. Rawat, and A. Rahman, “A survey on data-driven learning for intelligent network intrusion detection systems,” Electronics, vol. 11, no. 2, p. 213, 2022.

[30] U. Iftikhar and S. A. Ali, “Enhanced cyber threat detection system leveraging machine learning using data augmentation,” Int. J. Adv. Comput. Sci. Appl., vol. 16, no. 2, 2025.

[31] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), Ottawa, ON, Canada, 2009, pp. 1–6.