Class Conditioned Data Augmentation on FashionMNIST Dataset using Conditional DCGAN

Omkar Anant Bare
220459749
Ammar Yasir Naich
MSc. Big Data Science, Queen Mary
University of London

***Abstract*— Data augmentation has been a proven strategy for enhancing the performance and generalization of machine learning models. In this project, we implemented a data augmentation approach on the FashionMNIST dataset to display its effectiveness. Our method utilizes Conditional Deep Convolutional Generative Adversarial Networks (DCGAN) conditioned on class labels to generate augmented images that retain the class-specific characteristics of images in the dataset. In this project, we demonstrate that it is possible to achieve a low Fréchet Inception Distance (FID) score (< 15), which serves as a widely-used metric for evaluating GANs. We also find a significant overlap between the two distributions by analyzing the 2D latent space features of both the original dataset and the newly produced dataset. These results demonstrate our GAN model's efficiency and reliability, not just in terms of quantitative metrics like FID but also in terms of its ability to generate images that are consistent with the underlying data distribution.**

Keywords—Deep Learning, Deep Convolutional Generative Adversarial Networks (DCGAN), Conditional Generative Adversarial Networks (cGAN), FashionMNIST Dataset.

I. Introduction

Data augmentation can improve the model's robustness and capacity to learn a variety of representations by producing more synthetic training samples[2]. Various data augmentation methods, such as rotation, scaling, and flipping, have been applied to artificially expand the dataset size and enhance the model's performance on classification[2]. Although these methods can undoubtedly produce new images, they might not fully represent the breadth of variability found in the original image dataset. As a result, the generated images may not accurately reflect the conditions under which the images are obtained, and may not be a suitable generalization to new unseen data.

Conditional Generative Adversarial Networks (cGANs) have the potential to serve as a powerful tool for augmenting image data. By integrating relevant conditioning information, such as class labels or attributes, conditional generative adversarial networks (cGANs) have the capability to produce new images that conform to the distinctive features of the original dataset [6]. Consequently, this approach can address the problem of generating new data that fails to accurately reflect the inherent qualities of the original dataset.

The 'FashionMNIST' dataset has grown in prominence as a benchmark dataset for a variety of computer vision tasks, particularly in the area of classification of single-channel images[1]. It consists of 70,000 images, each of which is classified into one of ten categories of accessories for fashion. Despite being an excellent resource for training machine learning models, the dataset's small size and lack of diversity can have an impact on how well and how broadly the models can function.

The purpose of this project is to build and evaluate a data augmentation method employing a Conditional Deep Generative Adversarial Network (DCGAN) for the generation of images on FashionMNIST dataset. The following are the primary objectives:

1. Develop a conditional GAN-based approach for data augmentation: Develop and implement a data augmentation strategy that makes use of Conditional GANs to produce images that preserve the FashionMNIST dataset's class-specific properties.

2. To evaluate the effectiveness of our proposed method: To comprehensively assess the efficacy of our proposed methodology, we employ a multifaceted approach that extends beyond visual inspection. This approach includes a variety of key metrics, each of which brings a unique viewpoint to the evaluation process. The Fréchet Inception Distance (FID) score[3], a Latent Space Comparison analysis utilising 2D features extracted by the t-SNE dimensionality reduction algorithm [4], and the evaluation of images through a classifier trained on original images are the intended metrics for assessing the calibre and similarity of generated images created by GAN.

II. Background and Related work

It has become clear that Generative Adversarial Networks (GANs) are a potent class of machine learning models that can create synthetic samples that precisely replicate actual data while capturing the underlying latent data distribution [21]. The adversarial training method, which involves competitively training the generator and discriminator neural networks, is the main concept of GANs. The discriminator network tries to determine the difference between real and fake data, while the generator network learns to produce data instances that cannot be distinguished from real examples. GANs reach a dynamic equilibrium, where the generator becomes skilled at producing images with statistical properties that closely resemble those of the training dataset, through an iterative process of training and refinement.

When adding to datasets, this method is quite useful. The created samples increase the dataset's size and diversity by essentially extending it. GANs provide new situations for machine learning models to learn from, hence enhancing the model's exposure to different data patterns. These additional instances that exhibit perceptual differences are introduced into the system. In order for the trained models to perform better on unknown data, this augmentation is crucial for enhancing their generalisation abilities.

GAN research has improved significantly in recent years, with a focus on conditional GANs. These variants allow for the inclusion of conditional information during the creation process, increasing control over generated samples. GANs can generate class-specific images with different traits and attributes matched with the chosen class by conditioning the generator on additional information such as class labels [6]. This means that the generated images not only have realistic qualities, but also transmit class-specific traits, which contribute to the overall fidelity and interpretability of the generated data.

The adversarial training framework was established as the forerunner of Generative Adversarial Networks (GANs) [5]. Later efforts included Conditional GANs, which allowed for more exact control over data production [6]. Furthermore, the progress continued with the advent of Self-Attention GANs, a revolutionary approach that used self-attention approaches to improve both visual coherence and overall image quality [7]. These cumulative efforts have had a substantial impact on the landscape of GAN research and their wide applications across multiple domains.

III. METHODOLOGY

A. Dataset Pre-processing

Each of the 70,000 photos in the FashionMNIST dataset has a size of 28x28 pixels. The dataset is normalized by 0.5 subtracted from each pixel value and then divided by 0.5 before being fed into the GAN model, to normalize values in range[-1, 1]. By ensuring that the pixel values are between -1 and 1, this normalisation step enables greater convergence during model training.

B. Model Architectures

Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) are a class of neural network models comprising two fundamental components, namely a generator model and a discriminator model. The primary objective of the generator is to generate data samples, such as images or text, that closely resemble real data by utilizing random noise as input. Conversely, the discriminator's role is to discern and differentiate between real data and the fake data generated by the generator. The two components undergo iterative training in a manner, whereby the generator's objective is to generate data that is indistinguishable from real data, while the discriminator tries to enhance its capacity to discern between real and generated data. Generative Adversarial Networks (GANs) have demonstrated exceptional efficacy in producing data of superior quality and coherence. These networks have found extensive utility in several domains, including image synthesis, style transfer, and data augmentation.

Conditional Generative Adversarial Networks (cGANs)

Conditional Generative Adversarial Networks (cGANs) expand upon the existing framework of Generative Adversarial Networks (GANs) by including supplementary information to facilitate and direct the generation process. Conditional Generative Adversarial Networks (cGANs) include additional conditional information, such as class labels which attribute descriptions, or additional information that is relevant to both the generator and discriminator components [6]. This feature enables modification and conditioning of the generated data based on predetermined attributes. In the context of image generation within our project, a conditional generative adversarial network (cGAN) is trained to produce images depicting various fashion items, utilizing the specified class labels as guidance.

1. Generator architecture

A sequential model with numerous convolutional transpose layers makes up the generator architecture. The first layer executes a convolutional transposition operation with a kernel size of (3, 3), a stride of (2, 2), and an output tensor of 256 channels. After each convolutional transpose layer, batch normalisation and ReLU activation are applied. Two further layers of this block are added, gradually bringing the number of channels down to 128 and then 64. To create the output image using a single channel, a convolutional transpose layer with a kernel size of (4, 4) and a stride of (2, 2) is employed. Fig. 1. illustrates the generator architecture.

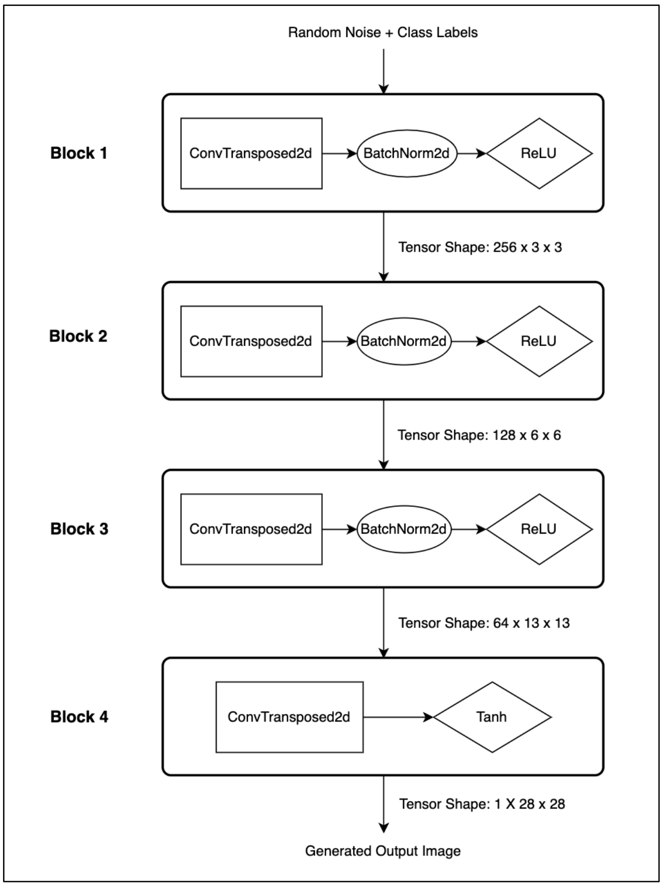

Fig.1. Generator Model Architecture

Transposed convolution

To up sample the spatial dimensions of feature maps, the generator architecture in our model employs convolutional transpose operations, often known as deconvolutional layers[8]. Transpose convolutions, as opposed to typical convolutional layers, boost input resolutions by inferring missing high-resolution information from low-resolution encodings. Fig. 2 shows how convolution operation differ from deconvolution operation[22]. A convolutional transpose layer, in particular, does the reversal of the convolution action by spreading out information rather than compacting it.

Transpose convolutions use the same fundamental algorithms as regular convolutions, but the forward and backward passes are reversed. The kernel is slid over the low resolution input on the forward pass to produce a larger output, with the weights shared across the spatial dimensions. The network can then learn enlarged versions of the input[9]. The stride and padding settings govern the enlargement.

We are able to gradually up sample low-resolution input noise vectors into high-resolution realistic images by incorporating numerous transpose convolutional layers in the generator design. During training, the network is able to learn useful up sampling kernels. Other approaches, such as nearest neighbour or bilinear up sampling, fail to infer realistic texture and tiny features when increasing photos. Convolutional transposition layers outperform other methods by learning non-linear up sampling functions that preserve the inherent statistics of the training data distribution [10].

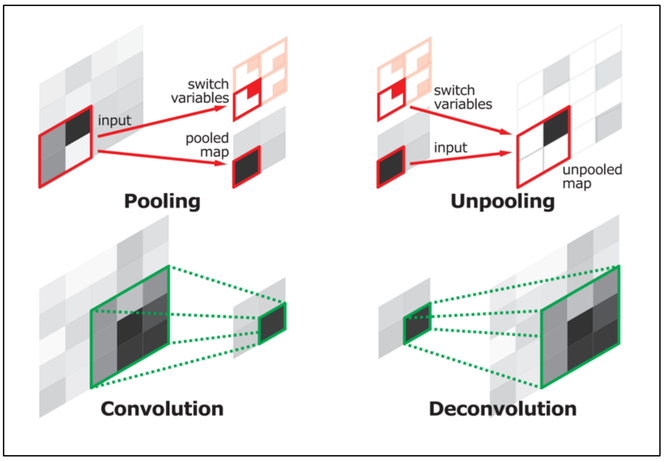

Fig. 2. Illustration of deconvolution and unpooling operations [22].

2. Discriminator Architecture

The sequential model of the discriminator architecture is made up of several convolutional layers. The first layer executes a convolution operation with a kernel size of (4, 4), a stride of (1, 1), and an output tensor of 64 channels. Each convolutional layer is followed by batch normalisation and LeakyReLU activation with a 0.2 negative slope. Two further layers of this blocks are added, eventually raising the number of channels to 128 and finally 256. The final output tensor with a single channel, which represents the discriminator's prediction, is created by using a convolution layer with a kernel size of (4, 4) and a stride of (2, 2) in the final step. Fig. 3. displays the discriminator architecture.

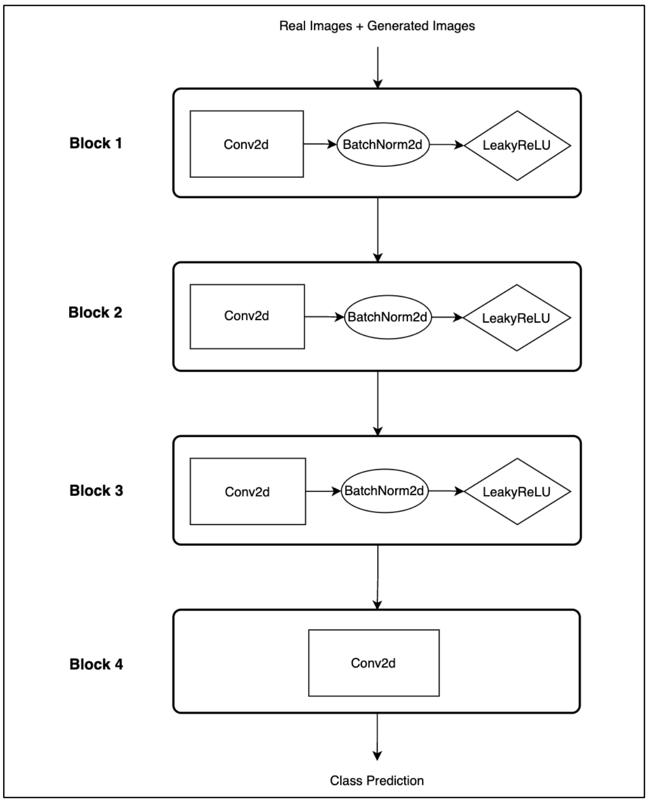

Fig. 3. Discriminator Network Architecture.

C. Training

Algorithm to train the generator and discriminator part of DCGAN conditioned on class labels:

For a specified number of epochs:
a. Produce random noise and sample labels for the classes and concatenate them to create the one-hot class vectors to feed as input in the generator network.

b. Use the generator network to produce augmented images by conditioning it on class labels as initialized above.

c. Use generated and actual images to train the discriminator network, supplying the corresponding class labels.

d. Train the generator network with adversarial loss, conditioning on the generated images and class labels for improved images.

The adversarial loss function used to train the generator and discriminator is the "Binary Cross Entropy Loss" function. When used in the context of a Conditional Generative Adversarial Network (cGAN) for image production, the Binary Cross Entropy (BCE) loss function is a critical tool for guiding the network's training process. In a binary classification scenario, this loss function assesses the dissimilarity between predicted probabilities and target labels [11].

For generator G and discriminator D, the BCE loss function is defined as [19]:

$$LBCE(G,D) = - E[y \log D(x|c) + (1-y) \log(1-D(x|c))]$$

Where x is the image, c is the condition (class label in our case), y is the real/fake label, and D(x|c) is the probability D which states that x is real under c. LBCE is computed for both G and D to optimize their objective functions. Minimizing their respective BCE losses enables effective adversarial training of conditional GANs for controlled image generation.

D. Evaluation

In this study, we evaluate the generated images from our GAN model using three quantitative metrics in addition to visual inspection:

1. Classification based on a CNN model

The effectiveness of the augmented dataset is evaluated from a practical application by training a CNN-based image classifier on the original dataset [15] and tested its effectiveness on the new augmented images. This provides a pragmatic assessment of how the augmented data impacts downstream classification performance and generalizability. We visually inspected random samples of the augmented images classified by the CNN classifier.

2. Fréchet Inception Distance (FID) Score

Inception Score (IS) and Fréchet Inception Distance (FID) are the two most often used metrics for GAN evaluation. They rely on an pre-built classifier (InceptionNet) that has been trained using ImageNet [12].

IS does not capture intra-class diversity, is insensitive to the prior distribution over labels[12] and is very sensitive to model parameters and implementations[13]. On the other hand, due to its consistency with human observation, FID has been widely adopted[12]. In our study, we have calculated the FID score for the generated images to evaluate our GAN model's capability.

In order to calculate the Fréchet Inception Distance (FID), we first pass the real and created images through the Inception-v3 convolutional neural network that has been trained on ImageNet. In a continuous latent space, this yields high-level feature representations for each image.

We can model these feature vectors for the real and generated images as continuous multivariate normal distributions with means and covariances given by (μx, Σx) and (μy, Σy) respectively. Here, μx and μy are the feature wise mean vectors for the real and generated features, while Σx and Σy are the covariance matrices capturing correlations between the features. Fréchet distance between two multivariate normal distributions X and Y is given by [14]:

$$(X, Y) = \|\mu_X - \mu_Y\|^2 + \mathrm{Tr}\left(\Sigma_X + \Sigma_Y - 2\sqrt{\Sigma_X \Sigma_Y}\right)$$

The quality and resemblance of generated images created by Generative Adversarial Networks (GANs) in comparison to actual images are evaluated quantitatively using the Fréchet Inception Distance (FID) score. A lower FID score indicates a closer resemblance between the feature distributions of actual and created images. This is done by comparing the feature distributions of real and generated images using a pre-trained Inception V3 model.

3. 2D Latent Space Comparison

Visualising the 2D latent representations created by using t-SNE algorithm on the latent vectors can reveal how effectively the GAN has trained to replicate the original dataset distribution [16]**.**

We can visually analyse the latent space learned by the GAN generator by reducing the dimensionality of the latent vectors to 2D using t-SNE. We would expect the t-SNE projected vectors of real and fake images to have comparable patterns and overlaps if the GAN has sufficiently learned the complex manifold of real images [16].

Specifically, we can look for two key characteristics:
1. Continuous latent space: The latent vectors should be spread out smoothly without large gaps, indicating the ability to generate diverse images.
2. Overlap between real and fake images: The generated images latent vectors should significantly overlap with real image vectors, suggesting similarity in distributions.

Therefore, t-SNE projected visualizations provide a qualitative way to assess if the generator has learned a robust latent space that captures the variation in real images. Complemented with quantitative metrics like FID, this evaluation approach delivers insights into the GAN's generative modelling capabilities.

IV. Results and Discussions

A. GAN Network Training

After training the model for 200 epochs, the generator loss was found to be 5.23 and the discriminator loss to be 0.07. The generator network may be successfully learning to create images that resemble the target dataset if the generator loss is decreasing. Similar to this, a decreasing discriminator loss shows that the discriminator network is getting better at telling actual images apart from produced ones. Accordingly, as seen in Fig. 4. our conditional DCGAN appears to be doing well in terms of convergence and fulfilling the required objectives based on these metrics to generate images similar to real data.

The final loss values and their convergence patterns show that both networks were successful in adversarial learning when viewed as a whole. While the discriminator effectively adapts to distinguish between real and fake images, the generator consistently generates more realistic images. We can infer that because both networks are successfully minimising their loss functions as intended, our conditional DCGAN implementation is correctly trained and achieves its intended goals of conditioned picture production and discrimination. The training dynamics and performance of the GAN model can be better understood by keeping an eye on the losses.

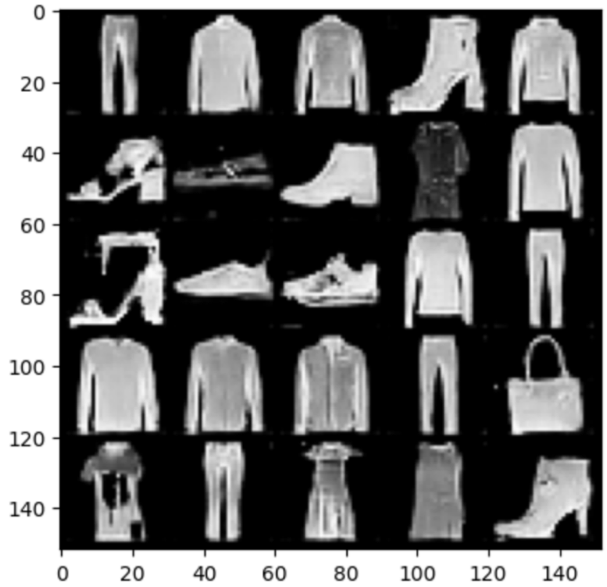

Fig. 4. Output from Generator network after training.

B. CNN-based classification of generated images

The CNN model achieved an accuracy of 94.52% on training data and 91.32% on test data, as is seen from the accuracy curves of Fig. 5. This indicates that the model has learned the patterns in the training data well, without much overfitting as evidenced by the small gap between training and test accuracy. The high accuracy on the test set suggests good generalization performance, meaning the model is able to correctly classify new unseen examples that it was not trained on originally. In order to accurately predict labels for new test data, the model must have successfully captured important dataset characteristics during training.

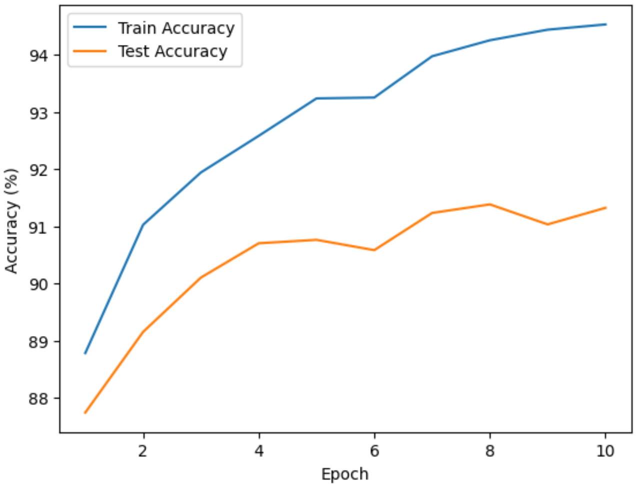

Fig. 5. Training set and test set accuracy of the CNN classifier.

However, there is still an opportunity for improvement in accuracy, particularly on the test set. This is probably because the model's capacity to generalise to a wider variety of data might be improved by adopting methods like hyperparameter tuning and selecting different CNN architectures or by employing transfer learning.

This trained CNN model was used to make predictions on the images generated by the conditional DCGAN model. Fig. 6. shows that trained CNN classifier can accurately distinguish between different class labels in the generated images.

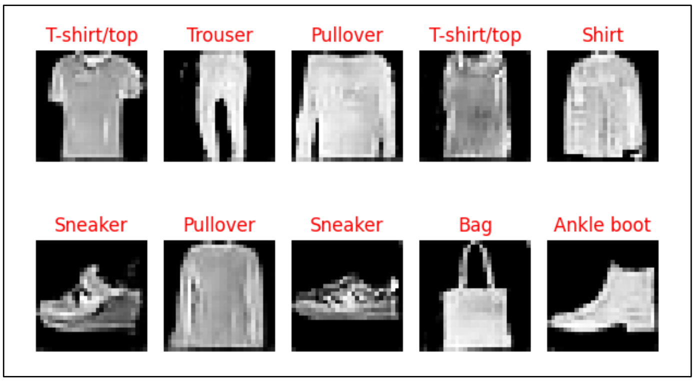

Fig. 6. CNN classifier's predictions on generated images.

C. Fréchet Inception Distance (FID) Score

A Fréchet Inception Distance (FID) score of 12.65 was achieved, indicating the high capability of our Generative Adversarial Network (GAN) model in generating images that closely adhere to the distribution of original data. The FID metric functions as a reliable measure of the resemblance between the distribution of generated images and the distribution of real data. It is worth noting that a FID score lower than 20 often signifies a high level of image quality and variety in the synthesized outputs[17].

The low FID score we have attained in this particular circumstance demonstrates the generator's proficiency in extracting important information from the original image distribution. Consequently, this capability enables the generator to produce a wide range of exceedingly realistic samples that contain the complexities inherent in the original distribution of data.

However, there is still a potential for more improvement. The potential for additional improvement in our FID score may be achieved with the use of advanced GAN architectures, specifically Conditional Orthogonal Ensemble Generative Adversarial Networks (COEGAN)[18]. This proposition is supported by the comparative research done on the MNIST dataset, in which COEGAN demonstrated better performance in comparison to the existing Deep Convolutional GAN (DCGAN) architecture [18]. The utilization of COEGAN has the potential to enhance the quality and variety of the images we generate, further advancing our progress in creating data that closely resembles the depth and complexity seen in original data distributions.

Even though there is still potential for improvement to further lower the FID score, the current number is a promising outcome that supports our GAN training approach. According to this quantitative parameter, our model has been successful in producing images that are noticeably similar to real images, as indicated by the overall FID score of 12.65.

D. 2D Latent Space Comparison

The t-SNE projections of the real and generated image latent vectors in Fig. 7. provide valuable insights into the latent space learned by our GAN model. Firstly, we can observe that the latent vectors are spread out smoothly across the 2D space without major gaps or isolated clusters. This continuous distribution indicates that the generator has learned to map the input noise vectors to a rich latent space supporting diverse image synthesis capabilities.

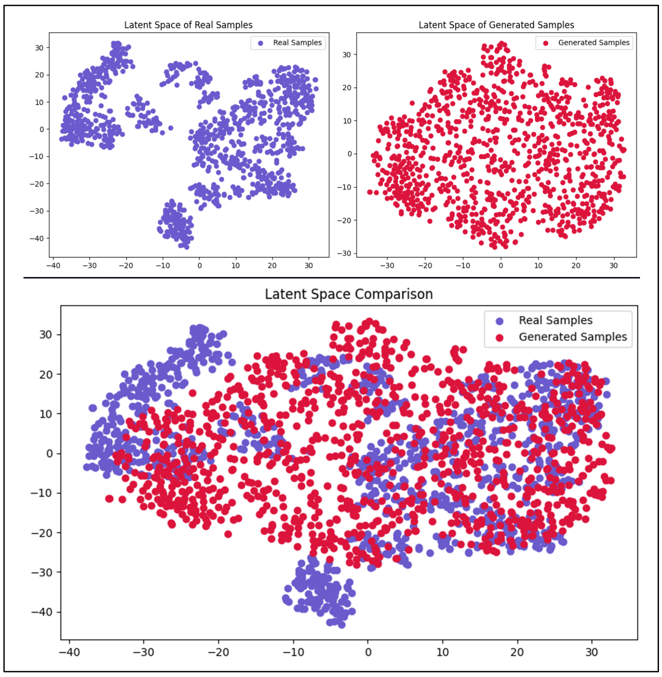

Fig. 7. Latent space comparison of 2D feature vectors obtained after t-SNE dimensionality reduction.

Most importantly, there is significant overlap between the distributions of the real and generated latent vectors as observed by the intermingling of data points across the two groups. This suggests that the generator has successfully learned a latent representation that captures the essential variations present in the real image dataset. The considerable similarity between the two distributions implies strong modelling of the real image manifold.

The generator's ability to create a variety of images that span the space occupied by real images is also demonstrated by the lack of isolated islands or sparse regions in the fake latent vector distribution. When considered collectively, these qualitative findings from contrasting the t-SNE projected real and produced latent vectors offer compelling visual proof that our GAN has trained an efficient latent representation for carefully orchestrated and diverse picture synthesis that closely mimics the real image manifold.

These qualitative results from the t-SNE projected visualisations provide convincing evidence that our GAN has uncovered a significant and rich latent space representation for creating new images with variety, realism, and control when taken as a whole. The figures in Fig. 7 show that the generator can map noise to a complex latent embedding that closely reflects the distribution of the real images.

V. FUTURE WORK

There are a number of directions that can be taken to improve research on conditional GANs for the FashionMNIST dataset. First, research into more potent and sophisticated GAN structures may aid in raising the quality and resolution of generated images. Examples include the use of attention processes by Self-Attention GANs (SAGANs) to model the global context and distant dependencies in images [7]. Similarly, by expanding networks and batches, BigGANs demonstrate astounding abilities to synthesise high-fidelity images [20]. Investigating such sophisticated models designed for fashion data may push the limits of image generation.

Another possibility is to try different adversarial loss functions for better training stability and optimisation. Mode collapse and vanishing gradient issues are mitigated by the Wasserstein GAN (WGAN) loss employing Earth Mover's distance [23]. To stabilise GAN training, The Consistency Term (CT-GAN) adds a regularisation term. Convergence in training and diversity could be improved by evaluating these losses.

To better modelling of class-specific aspects, conditional batch normalisation and self-attention could be added to the generator. Training may be made more consistent with the use of auxiliary classification in the discriminator. In conclusion, there are numerous potential to expand conditional GAN research on FashionMNIST by architectural advancements, different loss functions, and regularisation methods.

REFERENCES

[1] Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. arXiv preprint arXiv:1708.07747.

[2] Mikołajczyk-Grochowski, M. (2019). Data augmentation for improving deep learning in image classification problem. arXiv preprint arXiv:1905.04186. [3] Yu, Y., Wang, Z., Zhao, R., Shen, X., & Schwing, A. (2021). Frechet Inception Distance (FID) for Evaluating GANs. arXiv preprint arXiv:2109.02030.

[3] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S. (2017) GANs Trained by a Two Time-Scale Update Rule Converge to a

Local Nash Equilibrium, arXiv.org. Available at: https://arxiv.org/abs/1706.08500v6.

[4] Maaten, L. van der, & Hinton, G. (2008). Visualizing data using t-SNE. Journal of Machine Learning Research, 9(Nov), 2579-2605.

[5] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672-2680).

[6] Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.

[7] Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., & Metaxas, D. N. (2018). Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318.

[8] Dumoulin, V., & Visin, F. (2016). A guide to convolution arithmetic for deep learning. arXiv preprint arXiv:1603.07285.

[9] Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. (2010). Deconvolutional networks. In 2010 IEEE Computer Society Conference on computer vision and pattern recognition (pp. 2528-2535). IEEE.

[10] Noh, H., Hong, S., & Han, B. (2015). Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE international conference on computer vision (pp. 1520-1528).

[11] Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets. arXiv preprint arXiv:1411.1784.

[12] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016, June). Improved techniques for training gans. arXiv preprint arXiv:1606.03498.

[13] Barratt, S., & Sharma, R. (2018). A note on the inception score. arXiv preprint arXiv:1801.01973.

[14] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I. and Abbeel, P. (2016) InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets, arXiv.org. Available at: https://arxiv.org/abs/1606.03657v1.

[15] Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical report, University of Toronto.

[16] Costa, V., Lourenco, N., Correia, J., & Machado, P. (2021). Demonstrating the evolution of GANs through t-SNE. arXiv preprint arXiv:2102.00524.

[17] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S. (2017) GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, arXiv.org. Available at: https://arxiv.org/abs/1706.08500v6.

[18] Costa, V., Lourenço, N., Correia, J., & Machado, P. (2019). COEGAN: Evaluating the coevolution effect in generative adversarial networks. arXiv preprint arXiv:1912.06180.

[19] Zhang, F., Wang, Z., & Liu, Y. (2019). Binary cross entropy loss for image classification. arXiv preprint arXiv:1901.07883.

[20] Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096.

[21] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

[22] Noh, H., Hong, S. and Han, B. (2015). Learning Deconvolution Network for Semantic Segmentation, arXiv.org. Available at: https://arxiv.org/abs/1505.04366v1.

[23] Drakopoulos, V., Gidel, G., Simon, S., & Oudot, S. (2020). Consistency Regularization for Generative Adversarial Networks. In Advances in Neural Information Processing Systems (NeurIPS).