Fake It 'Til You Make It:
How Synthetic Images Boost Real Classifiers

Azfal Peermohammed, Prateek Gautam, Aziz Malouche

June 27, 2025

The Real-World Data Dilemma

Training deep learning models sounds straightforward—feed in a bunch of data, crank some GPUs, and voilà! But real-world datasets rarely cooperate. In practice, we often face two frustrating problems: not enough data and too much noise. Whether you're working with medical images, environmental sensors, or even rare wildlife footage, these constraints can cause your model to overfit or learn patterns that just don't generalize. That's where generative models come in. These models can learn from the limited data you have and generate more examples that (ideally) capture the essence of your dataset. One promising family of these models is the Denoising Variational Autoencoder (DVAE). Think of DVAEs as two-in-one learners: they can clean up noisy inputs while also learning a latent structure that helps them generate realistic new samples. But here's the thing—DVAEs aren't the only players in the game. They're part of a much bigger design space that includes tools like VQ-VAEs, GANs, and Diffusion Models (like DDPMs and DDIMs). So we asked ourselves a question:

Which image generation model produces the most useful synthetic data for improving classification performance, and how does this depend on the amount of training data available?

And not just in terms of generating pretty pictures—we wanted to know which models help downstream classifiers (like CNNs) perform better. More importantly, what's going on under the hood? This prompts us to look at the following question.

How do different image generation models affect the learned latent representation of a CNN, and can differences in representation space help explain variations in downstream classification performance?

What has already been done

Before diving into how we tackled our two core questions, it's worth reviewing prior work. Rather et al. [1] propose a GAN-based framework to boost CNN performance. Others use generative models to address class imbalance [2, 3]. Agrawal [4] compares DVAEs and VAEs on performance and latent structure. Khazrak et al. [5] and Bauer et al. [6] evaluate diffusion models against GANs, though without analyzing learned representations. Kim et al. [7] show structured synthetic noise can drive useful visual representations. Fan [8] explores fully synthetic pipelines for representation learning that rival real-data performance. Our study builds on these by comparing several generative models and explicitly examining how their outputs reshape CNN latent representations under varying data regimes—an angle we believe is missing from existing work.

Data and Models

Let's dive into the methodology behind our experiments, starting with an overview of the data and models we used. Much of our work builds on the excellent blog post [9], so we highly recommend checking that out—this project can be thought of as an extension of his research.

Data

The first decision we had to make was which dataset to use for training. We chose the FashionMNIST dataset, a popular benchmark introduced by Xiao et. al. FashionMNIST contains grayscale images of clothing items from 10 different categories. with examples of each shown below.

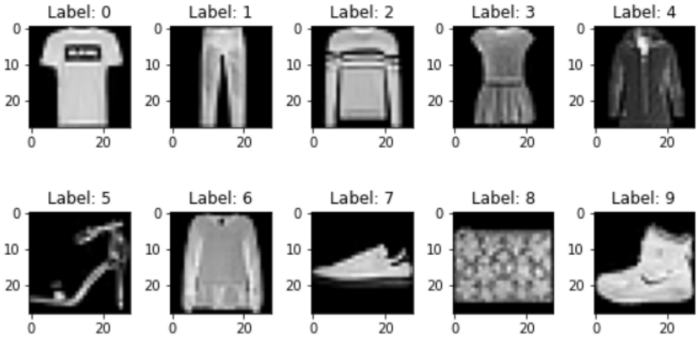


Figure 1: Example of images from the dataset

Each image in the FashionMNIST dataset is 28x28 pixels, resulting in a 784-dimensional vector when flattened. The full dataset contains 70,000 grayscale images of clothing items, divided into 60,000 training and 10,000 testing examples. There are 10 total categories: *T-shirt/top*, *trouser*, *pullover*, *dress*, *coat*, *sandal*, *shirt*, *sneaker*, *bag*, and *ankle boot*.

To explore the impact of data distribution on generative model performance, we create three different subsets of the training data, each containing 5,000 samples:

- **Balanced:** Each class has exactly 500 samples, ensuring a uniform distribution.
- **Semi-Balanced:** Two randomly selected classes (in our case, *pullover* and *shirt*) have only 100 samples each, while the remaining classes have 600 samples each.
- **Highly Imbalanced:** The dataset is deliberately skewed with varying numbers of samples per class to simulate extreme imbalance.

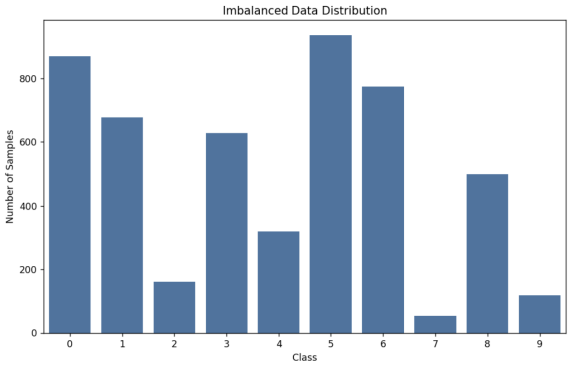


Figure 2: The distribution of the highly imbalanced data set

Models

In our experiments, we evaluate 12 different models using a Convolutional Neural Network (CNN) across 3 datasets. Each model generates an equal number of images as the training set size, which are then combined with the original images and used to train the CNN. The goal is for the CNN to classify the test set images into their respective 10 categories from the FashionMNIST dataset. The models are tested on a set of 10,000 images, with training sets ranging from 1,000 to 5,000 images (1k, 2k, 3k, 4k, and 5k).

Classifier Architecture (CNN)

For each of the 12 generative models, a CNN is trained to classify images into one of the 10 FashionMNIST categories. The CNN begins with two convolutional layers to extract hierarchical visual features, followed by a max-pooling layer that reduces spatial dimensions and enhances robustness. The output is then flattened and passed through three fully connected layers with ReLU activations, enabling the model to learn abstract representations and refine class boundaries. A final output layer maps these features to the 10 target classes.

VAE

The VAE consists of an encoder and a decoder. The encoder takes the input, processes it through two hidden layers (both with ReLU activations), and outputs two vectors: the mean (mu) and log-variance (logvar), which represent the distribution of the latent space. The decoder starts with the latent variable z, passing it through a hidden layer, followed by another hidden layer, and finally outputs the reconstructed image. The output is passed through a sigmoid activation to ensure pixel values are in the range [0, 1].

DVAE

The architecture of the DVAE is the same as the VAE described earlier. The only difference is that various types of noise are applied to the input images during training to encourage the model to learn to denoise and handle noisy data. These corruptions include: Gaussian noise, salt-and-pepper noise, rotation, brightness adjustments, contrast adjustments, and blur.

VQ-VAE

The VQ-VAE architecture consists of an encoder, a vector quantizer, and a decoder. The encoder is composed of three fully connected layers (with ReLU activations) that transform the input into a latent representation. The output is passed through the vector quantizer, which maps the latent variables to the closest vectors in a learned codebook and computes a loss based on the distance between the embeddings and the codebook vectors. The decoder then takes the quantized latent vectors, processes them through three fully connected layers (with ReLU activations), and outputs the reconstructed image. The loss function includes the quantization loss and the commitment loss, which encourages the encoder to use the codebook effectively.

Conditional GAN

The Conditional GAN model generates class-conditioned images. The generator takes a latent vector and label, embeds the label, and passes the combined input through multiple dense layers with residual connections to produce an image. The discriminator evaluates real vs. generated images, also conditioned on the label, using spectral normalization for stability and self-attention for improved feature extraction. The model is trained adversarially to generate realistic images specific to the given class label.

Big GAN

This BigGAN-inspired model is a step up from the earlier Conditional GAN (cGAN) in both scale and sophistication. While the cGAN uses fully connected layers and simpler residual blocks, this model incorporates deeper architectures, transposed convolutions for image upsampling, and self-attention mechanisms—all of which help it model more complex patterns. Additionally, it uses ResBlocks to stabilize training during resolution changes and better preserve information. These improvements allow it to generate higher-quality, more detailed class-conditional images, especially in low-data settings.

Diffusion

Denoising Diffusion Probabilistic Models (DDPMs) generate images by starting from pure noise and iteratively denoising through hundreds of small steps, giving them stable training, excellent mode coverage, and state-of-the-art visual fidelity that often surpasses GANs. However, they shine only when trained on very large, diverse datasets—small training sets leave them blurry or off-class—and each sample requires dozens to thousands of network evaluations, making inference comparatively slow and costly in production.

CLIP Guided Diffusion

CLIP-guided diffusion marries a diffusion generator with CLIP's text–image similarity scores: at each denoising step the model is nudged toward images that CLIP judges closer to a target prompt. We adopted it hoping that rich descriptions—e.g., "winter coat" or "professional button-up shirt" instead of the bare FashionMNIST labels coat, pullover, shirt—would push the generator to create visually distinct, class-specific samples, improving CNN precision and recall on those frequently confused categories. While CLIP guidance indeed offers finer semantic control, its effectiveness still hinges on large training data and has the same multi-step sampling cost as vanilla diffusion.

A summary of the models and the number of parameters that each of the models have can be viewed here.

Model	**Trainable Parameters**
VAE / DVAE	367,024
VQ-VAE	16,804
cGAN	3,950,965
BigGAN	15,233,346
Diffusion Model	4,210,224
CLIP Diffusion	5,262,896
CNN	1,658,890

Table 1: Trainable parameters for each generative model used in the study.

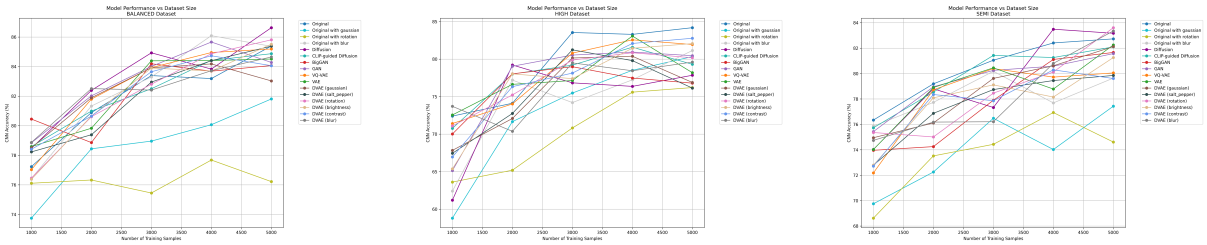
We aimed to test the models under reasonable computational constraints, focusing on accessible setups. Most models, such as VAE/DVAE, VQ-VAE, cGAN, and BigGAN, were efficiently trained on local CPUs, while the Diffusion Model required Google Colab due to its slower inference times. This approach allowed us to conduct the experiments without relying on high-end GPUs.

Experimental Setup and Results

Answering Question 1

Our first experiment investigates whether certain models perform better at generating images with varying amounts of training data, and how these effects change across our three datasets. We outline our methodology using the balanced dataset and apply the same approach to the other two datasets. For the balanced dataset, we trained each image generation model using different amounts of training data, ranging from 1,000 to 5,000 images. We then paired the generated images with the corresponding original data points and fed them into a CNN to assess how the number of training examples impacted the performance of each model in generating useful images. To evaluate the consistency of our findings, we ran this experiment twice for each of the three datasets. However, we found that the results were not reproducible across runs, suggesting instability in how training data volume affects model performance. We will discuss the results from both experiments in detail. We will denote them as Iteration 1 and 2.

Results and Analysis For Iteration 1



(a) Balanced Dataset (b) Highly Imbalanced Dataset (c) Semi Imbalanced Dataset

Figure 3: Number of Imaged used in Training vs Classification Accuracy per Model for Each Dataset

Across all models, we observe a general upward trend in CNN classification accuracy as the number of training samples increases. The most significant improvements occur between 1,000 and 3,000 samples, after which performance tends to plateau, suggesting that some generative models are already effective with just 3,000 samples. Performance across models is highly variable—different models take the lead at different sample sizes, and results are somewhat volatile. Notably, models trained with corrupted data using Gaussian noise or rotation tend to perform the worst overall; however, this trend does not consistently hold when such corruptions are applied within the DVAE framework, where certain noise types—like rotation—can actually lead to competitive performance depending on the dataset. For future experiments, evaluating performance at larger intervals (e.g., 1k, 10k, 50k) may offer clearer trends and reduce noise in comparisons. DVAE(blur) and DVAE(rotation) both tend to do especially well at 5k, indicating DVAE may be great especially with larger datasets.

On the balanced dataset, Conditional GAN and BigGAN performed well with a low amount of data (1k images), but did not match the performance of DVAE, VAE, or Diffusion models at 5000 images. On the highly imbalanced dataset, VAE, DVAE, and VQ-VAE consistently outperformed other models, likely due to their stability and ability to model complex distributions without relying on adversarial training.

Note that for the figures below, the top row shows original images, bottom row shows corresponding reconstructions from a model trained on a specified number of images.

In the balanced dataset, VQ-VAE and DVAE with rotation corruption achieved the highest classification accuracies at 5,000 samples, while BigGAN led at just 1,000 samples.

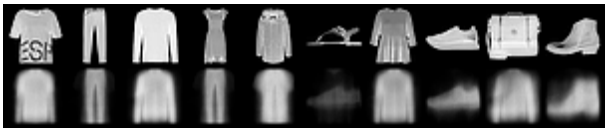


Figure 4: VQ-VAE sample output (5,000 images)

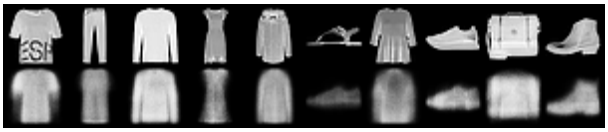


Figure 5: DVAE (rotation) sample output (5,000 images)

In the highly imbalanced setting, the DVAE with blur and the standard VAE performed best at the lowest sample count of 1,000.

For the semi-balanced dataset, DVAE with rotation and blur noise, along with the diffusion model, reached the top performance at 5,000 samples.

Figure 9: Diffusion sample output (1,000 images)

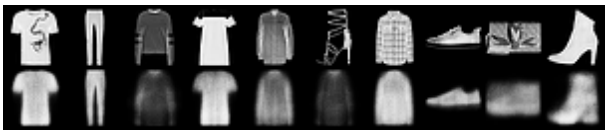


Figure 10: DVAE (blur) sample output (5,000 images)

These results highlight how certain generative models and even specific noise strategies can be more effective depending on the distribution and quantity of the data, and that even with 1,000 images to train on they can generate pretty good reconstructions.

Results and Anlysis For Iteration 2

An initial review of model performance reveals that diffusion-based models consistently underperformed in terms of CNN accuracy. In fact, among the bottom 20 models, nearly all are variants of diffusion or CLIP-guided diffusion, scoring even lower than the CNNs trained on the baseline datasets. This suggests that despite the theoretical strengths of diffusion models in generative tasks, their utility in this specific data augmentation context—namely FashionMNIST—was limited. A likely explanation is qualitative in nature: the images generated by these diffusion models did not closely resemble the true distribution of FashionMNIST classes.

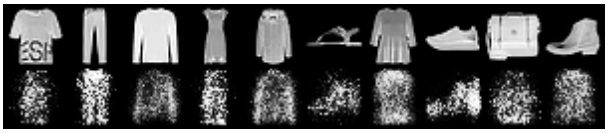


Figure 6: GAN sample output (5,000 images)

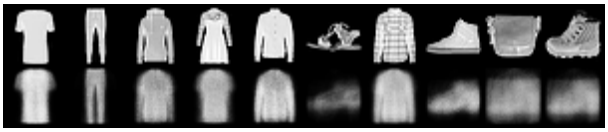


Figure 7: VAE sample output (1,000 images)

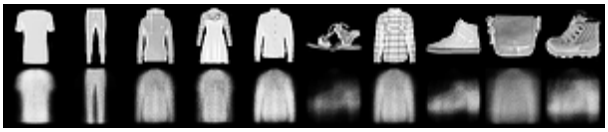


Figure 8: DVAE (blur) sample output (1,000 images)

In contrast to the diffusion models, the top 20 performing models were dominated by GAN and VQVAE-based augmentations. These generative methods consistently produced CNN accuracies above 0.75, suggesting that the synthetic images they generated were not only realistic, but also class-consistent and highly useful for training. Notably, VQVAE appeared across dataset types—Balanced, Semi-Imbalanced, and Highly-Imbalanced—indicating strong robustness to data distribution shifts.

Interestingly, we also observed that several baseline models trained on 4000 and 5000 samples ranked among the top performers. This reinforces the idea that for a relatively simple dataset like FashionMNIST, sufficient real data alone can enable strong classifier performance, especially when sample sizes reach the 4000+ range. In these cases, augmentation may provide limited additional value, as the data coverage is already high.

Additionally, many of the imbalanced datasets—when paired with GAN or VQVAE augmentation—led to better downstream CNN accuracy than their balanced counterparts. This suggests that generative models can not only supplement low-data regimes, but also help mitigate class imbalance by enriching underrepresented classes, ultimately improving the model's ability to generalize.

The overall results, and comparison on when it is valuable to augment limited or imbalanced data, is described in the table below.

Table 2: Baseline vs. Best-Model CNN Accuracy Across Dataset Types and Sample Sizes

Dataset	Size	Baseline	Best (Model)	Δ Acc.	Take-away
			Balanced		
Balanced	1 000	0.406	0.709 (GAN)	+0.302	Huge lift when data are scarce.
Balanced	2 000	0.648	0.775 (VQVAE)	+0.127	Benefit shrinks as data double.
Balanced	3 000	0.654	0.786 (VQVAE)	+0.132	Plateau begins; still useful.
Balanced	4 000	0.756	0.809 (GAN)	+0.053	Marginal gain; plenty of real data.
Balanced	5 000	0.757	0.807 (GAN)	+0.050	Similar story—synthetic optional.
			Semi-imbalanced		
Semi-imbal.	1 000	0.546	0.721 (DVAE_brightness)	+0.175	Scarce & skewed → augmentation pays.
Semi-imbal.	2 000	0.661	0.675 (GAN)	+0.014	Almost no gain; real data sufficient.
Semi-imbal.	3 000	0.488	0.747 (GAN)	+0.260	Big jump; baseline collapses on minorities.
Semi-imbal.	4 000	0.413	0.810 (VQVAE)	+0.396	**Largest lift**; imbalance dominates.
Semi-imbal.	5 000	0.715	0.792 (VQVAE)	+0.077	Added data narrows benefit.
			Highly-imbalanced		
High-imbal.	1 000	0.224	0.722 (DVAE_blur)	+0.499	Extreme skew + low data: augmentation critical.
High-imbal.	2 000	0.550	0.772 (VQVAE)	+0.222	VQVAE adds value.
High-imbal.	3 000	0.443	0.779 (GAN)	+0.337	GAN excels once size grows.
High-imbal.	4 000	0.801	0.817 (VQVAE)	+0.016	Baseline already strong; little headroom.
High-imbal.	5 000	0.667	0.759 (GAN)	+0.092	Moderate boost; skew still hurts.

Answering Question 2

To investigate the second research question—how synthetic data impacts internal representations and downstream performance—we analyzed the latent space of CNNs trained on different datasets. Specifically, for each vision model and dataset size, we passed the generated images through the corresponding trained CNN. From each CNN, we extracted the 128-dimensional embeddings produced by the final fully connected (FC) layer, just before classification. These embeddings represent the model's internal encoding of the input data.

To visualize the structure of this high-dimensional latent space, we applied t-distributed Stochastic Neighbor Embedding (t-SNE) to project the 128-dimensional embeddings onto a 2D plane. This allowed us to examine how well the representations of different classes clustered together, and whether the inclusion of synthetic data reshaped the decision boundaries or introduced overlap between classes.

To go beyond visual inspection, we trained CART (Classification and Regression Tree) models on these 128-dimensional embeddings, using the true class labels as targets. The performance of the CART models provided an interpretable, quantitative measure of the separability of the embedding space: higher CART accuracy implies that class boundaries are more distinct in the learned representation.

We first analyzed the embedding spaces of the baseline datasets—Balanced, Semi Imbalanced, and Highly Imbalanced—at varying sample sizes (1000, 2000, 3000, 4000, 5000). We then augmented each of these datasets with synthetic images generated by our vision models and repeated the embedding extraction, t-SNE projection, and CART analysis. By comparing the "before" and "after" states of the embedding spaces and CART performance, we aimed to understand whether synthetic data improved class separability, led to better generalization, or potentially introduced overfitting.

Results For Question 2

We conclude with an examination of the CNN's latent-space representations—specifically, the 128-dimensional embeddings produced by the final layer before classification. Our original hy-

pothesis was straightforward: datasets whose embeddings form well-separated class clusters should yield higher downstream accuracy. To test this, we trained CART models on the embeddings of each experiment; CART accuracy served as an interpretable proxy for class separability in latent space.

The results, shown in the figure below, challenge that intuition. Most CNNs achieved consistently high CART accuracy—indicating strong geometric separation—yet this separation often failed to translate into equally high test accuracy.

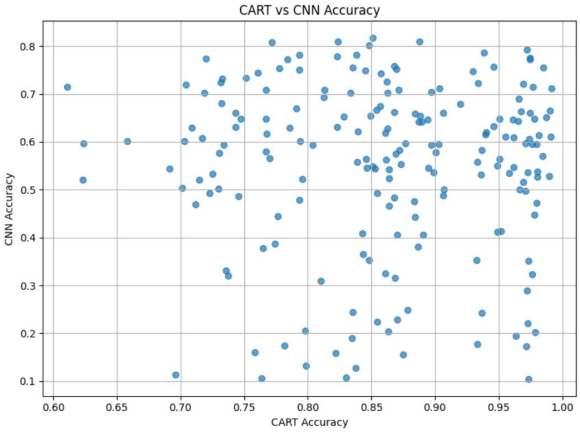


Figure 11: CART Accuracy vs. CNN Accuracy

In other words, even when the network's internal features cleanly partitioned FashionMNIST classes, the final classifier sometimes under-performed. The disconnect suggests that latent separability alone is not a sufficient predictor of overall performance; the realism and class fidelity of the augmented images remain critical.

Finally, in reviewing the 128-dimensional CNN embeddings, we found that every experiment fell into one of **three archetypal patterns**:

1. **Single, well-separated clusters.** Some datasets—almost always the larger imbalanced sets—showed a single, tight cluster for each FashionMNIST class, with ample margin between classes. Interestingly, these single clusters were not particularly seen in the balanced datasets, just the imbalanced.

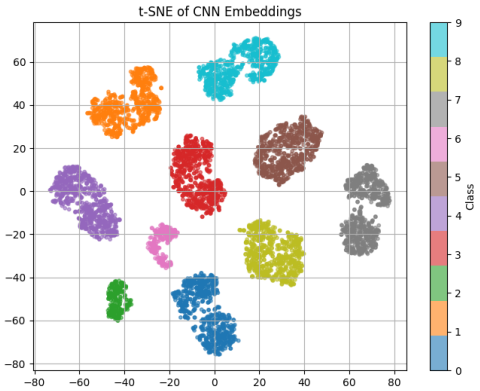


Figure 12: Example of well-separated single clusters.

2. **Twin clusters per class (GAN / Diffusion).** GAN and Diffusion augmentations often produced *two* distinct clusters for every class—one anchored near the original data. This suggests the generator created samples that were visually consistent with their class yet occupied a new sub-manifold of the latent space.

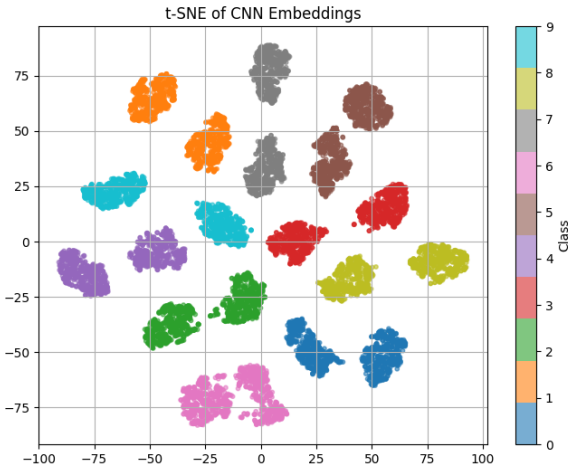


Figure 13: Twin-cluster behaviour characteristic of GAN / Diffusion.

3. **Overlapping garment trio (VAE / DVAE).** Only the VAE and DVAE variants collapsed the embeddings for *shirt* (class 2), *pullover* (class 4), and *coat* (class 6) into a single region, making them largely inseparable. Although shirts, pullovers, and coats share a broadly similar silhouette, their embedding overlap suggests that the VAE and DVAE decoders failed to capture the finer visual cues that distinguish these garments. Figures 7 and 8 demonstrate this, showing how the generated images for those classes all looked very similar due to their blurry quality. Consistent with this, the same three classes recorded the lowest precision and recall scores across nearly every CNN we trained, indicating that—even when the latent space appears separable—the generated images remain too visually alike for reliable classification.

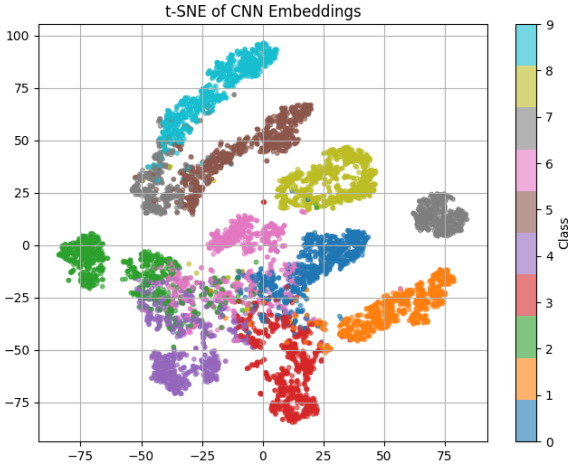


Figure 14: Class-2/4/6 overlap in VAE / DVAE embeddings.

The first two archetypes align with higher CNN accuracies: either the classes are clearly partitioned or the generator augments them in a coherent, separable fashion. By contrast, the VAE/DVAE overlap correlates with a sharp drop in accuracy on those three classes, underscoring that *latent separability at the class level remains a prerequisite for reliable downstream performance.*

Limitations and Future Work

Inconsistent Results Across Runs For Some Models: The observed inconsistencies in model performance, particularly when comparing runs with 1k and 5k samples, likely stem from the inherent variance in model training. Some models perform better on certain runs with few samples and other runs with more samples. This may be due to unstable learning dynamics, where slight differences in initialization, optimization paths, or data splits lead to drastically different results. As a result, the reliability of downstream linear probe evaluations (e.g., CART classifiers) becomes compromised, making it difficult to draw consistent conclusions from these experiments for all models.

Lack of Correlation Between CNN Accuracy and CART Accuracy / Embedding Separability: The absence of a clear relationship between end-to-end CNN classification accuracy and the performance of CART probes on embeddings suggests that high task performance does not necessarily imply semantically meaningful representations. CNNs might rely on distributed, entangled features or shortcut cues that suffice for the final prediction but do not translate well into structured, easily probed embedding spaces.

Future Work: Future work should address these limitations by:

- One limitation of our study is the instability in model performance across training runs, which made it difficult to draw consistent conclusions. We suspect that this may be due in part to the absence of normalization layers, which are commonly used in many of the models we employed and are known to improve training stability. While we do not explore this further in the current work, incorporating normalization layers, improved initialization methods, and other training regularization techniques could be a promising direction for future research.
- Investigating representation quality more directly using a wider range of probing techniques and intrinsic dimensionality metrics, beyond accuracy alone.

Conclusion

Our exploration into synthetic image generation reveals that not all generative models are equally helpful when it comes to boosting downstream classification performance. While models like cGAN and BigGAN can offer early performance gains in low-data regimes, their effectiveness plateaus or falls behind as more data becomes available. In contrast, simpler models such as DVAEs—especially those trained with certain types of input corruption (e.g., rotation or blur)—prove remarkably robust across both balanced and imbalanced data settings. VQ-VAE also shows strong potential, particularly in stable, balanced scenarios.

Perhaps most critically, our findings underscore that latent space representations of images alone do not equate to "classifier usefulness." The "visual realism" of the images plays a pivotal role in how well the downstream classifier is able to classify images, demonstrating the importance of not just augmenting limited or imbalanced data, but augmenting it with high quality images. Future work should further dissect these learned feature spaces and extend evaluations to more

complex datasets and architectures. Synthetic data isn't just filler—it's a powerful tool, but only when wielded with care.

References

1. Zeng, X., Zhang, W., Huang, Y., & Wu, W. (2023). An improved denoising diffusion probabilistic model based on feature fusion and noise estimation. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-023-15747-6

2. Kingma, D. P., & Salimans, T. (2023). Variational Diffusion Models. *arXiv preprint arXiv:2302.10910*. https://arxiv.org/abs/2302.10910

3. Sharma, P., & Chawla, R. (2024). Enhanced diffusion models using memory-augmented transformer networks. *Applied Soft Computing*. https://doi.org/10.1016/j.asoc.2024.111955

4. Hu, Y. (2023, February 18). Denoising Variational Autoencoder. *Deep Learning MIT Blog*. https://deep-learning-mit.github.io/staging/blog/2023/denoisingVAE/

5. Le, T., Bui, H., Tran, D., & Phung, D. (2024). Guided Diffusion Models with Learnable Guidance. *arXiv preprint arXiv:2412.12532*. https://arxiv.org/abs/2412.12532

6. Batzolis, K., Vahdat, A., & Van Gool, L. (2024). Conditional Flow Matching: Provably Better and Cheaper Diffusion Models. *arXiv preprint arXiv:2401.02524*. https://arxiv.org/abs/2401.02524

7. Austin, J., Johnson, A., Ho, J., Tarlow, D., & Brooks, D. (2022). Structured Denoising Diffusion Models in Discrete State-Spaces. In *Advances in Neural Information Processing Systems, 35*, 24602–24616. https://proceedings.neurips.cc/paper_files/paper/2022/file/e8507db80464ced5658d16b49bd458b9-Paper-Datasets_and_Benchmarks.pdf

8. Wang, T. (2024). *Improving representation learning in variational autoencoders through structured priors* (Master's thesis, Massachusetts Institute of Technology). https://dspace.mit.edu/handle/1721.1/156315