Textile Vision: A CLIP-Enhanced Diffusion Model for Fashion Design Generation from Natural Language Descriptions

Anastasiya Masalava, Giselle McPhilliamy, Harshith Lanka, Lasya Sreenivasan, Ramya Subramaniam
Georgia Tech
801 Atlantic Dr NW, Atlanta, GA 30332
amasalava3@gatech.edu, gmcphilliamy3@gatech.edu, hlanka3@gatech.edu, lsreenivasan6@gatech.edu
ramyasub@gatech.edu

Abstract

Fashion e-commerce is rapidly evolving, yet the visualization of clothing designs from natural language descriptions remains challenging due to the need for specialized design tools and expertise. Our project, Textile Vision, addresses this gap by developing a CLIP-enhanced diffusion model that translates text prompts into high-quality 2D fashion images. By conditioning a latent diffusion model with cross-attention layers informed by CLIP text embeddings, we aim to improve semantic alignment and visual coherence in generated fashion designs. We leveraged a subset of the DeepFashion dataset, given its detailed stylistic annotations, to train and evaluate our model. Despite encountering dataset quality challenges, specifically low-dynamic-range encoding issues, we implemented a series of architectural and training refinements to optimize performance. Our results indicate that while the baseline diffusion model performs adequately, the CLIP-enhanced model shows improved alignment between text descriptions and generated imagery, suggesting that specialized cross-attention mechanisms can enhance domain-specific text-to-image synthesis. This work demonstrates the potential for AI-driven fashion design tools, especially for small businesses and educational applications, while highlighting key challenges and future directions such as improved data preprocessing and memory-efficient fine-tuning techniques.

1. Introduction

Fashion e-commerce is rapidly growing, with customers requesting more personalized experiences. While solutions exist for clothes customization, current methods of clothing design visualization often require professional designers or complex software, such as Optitex or Gerber AccuMark design software. Our approach leverages a Diffusion Model architecture enhanced with a CLIP-based cross attention layer that will take text input from a user and generate a 2D image of a desired outfit aims to bridge the gap between natural language descriptions and visual synthesis of fashion designs. Outside of the primary use case, the outcome of our project can be used by small fashion businesses without in-house designers or in fashion education and training. While text-to-image generation exists, specialized fashion-focused models are rare and often struggle with generating realistic fashion details and maintaining style consistency. Our research aims to advance the field of AI-powered fashion design by developing model architectures and training methodologies that can better learn fashion-related features and semantic relationships, potentially establishing new benchmarks for text-to-fashion image generation.

Successfully fine-tuning this diffusion model would have significant implications for fashion design, personalized marketing, and e-commerce as it would enable more customized visual outputs for consumers and designers. In building models such as this Fashion Generative model users can create highly targeted design aids and fashion designs that reflect their creativity and greatly reduce the time and resources involved in new fashion lines. Overall we hope this will reduce costs and time-to-market for new fashion lines.

2. Related Works

Machine Learning methods have long been studied in their applications to the fashion space as they can help improve the overall online experience while also making previously inaccessible spaces more accessible. When it comes to computer vision methods broadly, existing implementations generally fall into one of four categories—synthesis, detection, analysis, and recommendation. [2] Examples of these tools include virtual try-ons, pose transfer to understand the fit of an outfit in different angles, and video generation from image inputs. This last implementation was achieved with diffusion models utilizing pseudo-3D convolution, VAE, and CLIP encoders to condition synthesised videos to help users understand the movement and flow of certain clothing items. [4] This work relates to our project by reviewing how diffusion and CLIP embeddings can be used together in fashion contexts. However, while [4] focuses on dynamic fashion video generation from static images, our project focuses on generating new clothing designs entirely from textual descriptions. This means our project will require finer-grained semantic mapping between text and visual output.

Text-to-image generation is a growing and highly impactful space, and the first notable implementation of these models was with DRAW, which focused on image generation but relied on specific data training. Along with that, other implementations of these models were generally trained on smaller datasets, possibly limiting the abilities of the final models. DALL-E was in turn developed to address these issues and develop a flexible text-to-image generative model capable of zero-shot learning. [7] However, DALL-E used a transformer style model, relying on Vector Quantized Variational Autoencoder to map images into tokens and eventually leading to struggles with computational expense and finer details. Because of that, DALL-E-2 was later developed using CLIP latents and diffusion models. They switched to CLIP, which was a model that learns how to match text descriptions with images by mapping them to a common embedding space. These CLIP image embeddings are used in tandem with diffusion models to allow for better quality and more realistic output images. [6] These models, particularly DALL-E 2, directly informed our choice to combine CLIP embeddings with diffusion processes. While DALL-E 2 targets broad domain text-to-image generation, our project narrows this pipeline to fashion-specific tasks, requiring adaptation to the fine-grained details often seen in clothing photos such as textures and seams.

Now, diffusion models are notable for the improvements they can bring to generation, but latent diffusion models specifically bring about increased stability while reducing computation costs. While they encode images in a lower dimensional latent space, a pre-trained autoencoder maintains essential visual information to ensure the final images are high quality. [8] The use of Latent Diffusion Models drove our team's choice to maintain a balance between computational feasibility and image quality. We will consider use of a latent space representation to make model training and inference feasible even with limited GPU resources, but adapt it with CLIP conditioning layers specific to the fashion domain. [8]

The DiffusionCLIP framework reviews manipulation of text-guided images through leveraging multiple properties of diffusion models. Diffusion models have a full inversion capability as well as consistent performance in high-quality image generation abilities. Looking at other models in this space such as GAN models we see a general struggle to reconstruct images that depict unusual poses or more complex details. Unlike GAN models, however, diffusion models have shown to surpass these limitations by enabling multi-attribute manipulation through use of a noise combination method. This allows diffusion based models to generate unique poses as well as variable content that is less similar to training data. Overall diffusion models learn stronger control over characteristics of generated images. They perform better when it comes to highlighting key objects in images and removing unwanted ones. [5] This DiffusionClip model in [5] exhibits the key advantages of diffusion over GAN-based approaches, particularly for generating uncommon clothing designs or outfits with multiple style attributes. However, while DiffusionCLIP focuses mainly on post-hoc editing of generated images, our model will aim to generate full fashion images directly from a prompt, integrating fashion semantics from the very start of the generation process.

By leveraging diffusion models in our project we will be able to create more detailed and accurate clothing designs from text descriptions. Additionally, diffusion based models seem to have a stronger capacity for learning detailed generation. This will mean our model should be stronger when it comes to capturing more intricate and detailed fashion styles and designs, even when it has not seen them before. Finally, as diffusion models allow for multi-attribute manipulation, users of our model should ideally be given the chance to change many aspects of fashion generated images at once. For these reasons a disunion model seems optimal and a strong choice for our fashion image generator model.

Looking into the specific diffusion models, we find firstly that image generation has three main problem areas in multi-object generation, rare or novel concept generation, and generated image quality improvements. Different models integrate varying techniques to account for these issues including attention maps, subject-driven generation, and instruction tuning for human preference. [9] With this information, we can effectively choose the best diffusion model that best accounts for our requirements while controlling for model accessibility.

3. Method / Approach

In our project approach we aim to develop a text-to-image pipeline that translates textual descriptions into AI-generated images of clothing. We intend to do this by leveraging diffusion models and CLIP embeddings. Our approach consists of multiple technical stages. We plan to begin with text embedding, followed by image generation, model training, fine-tuning, and then evaluation. We selected a diffusion model because they do very well in generation of high-quality diverse images, compared to older GAN-based approaches. This aligns with our need for varied and realistic fashion designs. GANs were initially considered for this project, however, as they tend to suffer from training instability, especially when conditioned on complex text prompts we decided to use Diffusion models, which by contrast, provide more stable training dynamics and consistently generate higher-quality, diverse outputs, making them more appropriate for our objective. Given our more complex dataset, with sample images featured below, the stable and stronger image generation capacity from diffusion-based synthesis paired with CLIP guidance was the preferable strategy.

Our design will integrate cross-attention layers between the text embedding and the image synthesis process. This is inspired by the transformer architectures discussed in class. Cross-attention should enhance the model's ability to maintain a global understanding of both spatial coherence and semantic alignment with the input prompt, an essential factor in complex object generation such as fashion clothing. First, we will encode textual descriptions of clothing items into feature vectors using a CLIP encoding model. This will ensure that our model accurately captures the semantic meaning of the input text. These feature vectors will then serve as conditioning information for a diffusion model, which is well-suited for high-quality image synthesis. To enhance the model's ability to generate realistic and coherent fashion designs, we will incorporate cross-attention layers, improving its ability to maintain a global understanding of the image being generated. During training, we will monitor performance using loss functions and leverage TensorFlow's TensorBoard for visualization.

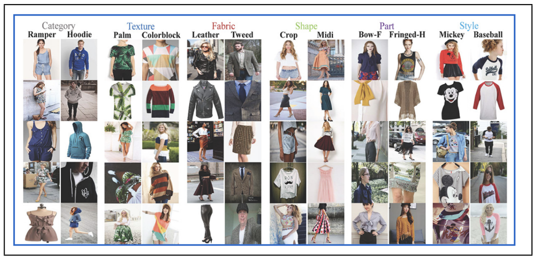

Figure 1. Category and Attribute Prediction Benchmark Dataset Visualization for subset of the image data and labels

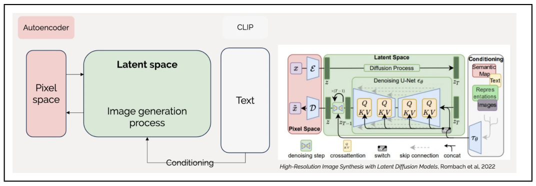

Figure 2. An overview of the latent diffusion-based architecture we adapt is illustrated in Fig. [8]

To evaluate our model, we will generate images using our fine-tuned model and calculate the Inception Score to assess image clarity and diversity. Additionally, we plan to conduct user studies, allowing individuals, such as ourselves, to compare our generated images to real-world fashion items and provide qualitative feedback. Our model will be trained using the dataset of DeepFashion which is composed of around 700,000 images described by stylists. This dataset will ensure exposure to high-quality images, detailed textual descriptions, and fashion styling patterns. Fine-tuning will involve adjusting hyperparameters, modifying network architectures, and experimenting with different diffusion model configurations to improve image quality and alignment with text prompts. To optimize training, we will explore random search for hyperparameter tuning. We will tune parameters such as learning rate, batch size, noise schedules, and diffusion steps to maximize performance.

Overall, our methodology is designed to effectively bridge the gap between natural language descriptions and fashion image synthesis, contributing to AI-powered fashion design. For our baseline method, we will implement a standard diffusion model architecture without the CLIP-enhanced cross-attention layers, using only basic conditioning techniques to establish a performance benchmark for text-to-fashion image generation. This approach will serve as a foundation to compare against our proposed CLIP-enhanced architecture, allowing us to quantify the improvement gained through our specialized cross-attention mechanisms.

To evaluate performance, we will employ a combination of quantitative metrics including Inception Score (IS) to assess clarity and diversity of the outputs, and supplement these with qualitative user studies where participants will compare the generated fashion designs against similar real images to evaluate style consistency, detail accuracy, and overall visual appeal.

4. Data

For this project, we used a subset of the DeepFashion dataset designed for Fashion Synthesis tasks. The DeepFashion dataset is a large-scale collection of high-quality fashion images with detailed annotations. In total it is composed of approximately 78,979 images that are each paired with metadata that details clothing type, categories, attributes, and stylist description. We gained access to the full dataset through an application and password-protected download process, following a procedure provided by the dataset creators.

We chose to work with the DeepFashion dataset because of its extensive collection of diverse clothing styles and its structured attribute annotations, which made it a strong candidate for training both the CLIP and Stable Diffusion models on fine-grained fashion descriptions. The dataset is divided into two main subsets. The first focuses on category and attribute prediction, providing labeled clothing images that are annotated by garment types as well as their attributes. The second subset focuses on consumer behavior modeling. It includes clothing images and associated purchase rates from online shopping scenarios. This subset can be valuable for tasks aiming to generate fashion items that are not only visually realistic but also optimized for commercial appeal and consumer engagement. For our project, we primarily utilized the category and attribute prediction subset, as it provided the most structured textual prompts necessary for conditioning our CLIP and Stable Diffusion models. However, a strong expansion of this model would be to include the consumer behavior data and tailor a model to design clothes that have a high consumer demand.

The images were initially stored in an H5 file format for ease of access during training within Google Colab. However, the dataset underwent a significant transformation process initially unknown to us during storage: images were encoded into low-dynamic-range tensors rather than preserved as RGB pixel arrays. This meant that the values in the image array were set between -1 and 1 and to turn them into proper RGB images we needed to normalize all images between 0 and 1, and multiply all values by 255 to get RGB represented images. This specific image preprocessing is required for the CLIP fine tuning as the CLIP model expects actual RGB images and not some normalized image tensor. Furthermore, preprocessing steps included resizing all images to 512×512 pixels to match the input size requirements of Stable Diffusion and applying normalization as necessary. In terms of potential biases, it is worth noting that the DeepFashion dataset is derived primarily from online fashion retailers and catalogs. As a result, it may overrepresent Western-style clothing and commercially popular trends, potentially limiting the diversity of generated fashion styles if not corrected during model training.

The intended use of the DeepFashion dataset aligns with our project goals, as it was originally created for training models on fashion recognition, generation, and retrieval tasks, making it a strong choice if properly prepared. Overall, while the DeepFashion dataset offers an excellent foundation for fashion image generation tasks, issues in the handling and storage pipeline significantly limited our project's success during this training phase [3].

5. Experiments and Results

5.1. Training Challenges and Solutions

While developing a text-to-image generation pipeline using a Stable Diffusion model conditioned on CLIP embeddings, we encountered significant challenges stemming from dataset format issues, resource limitations and model instability. One key issue we had to navigate was consistently running out of memory, even while using Colab Pro GPU's with upgraded compute resources. Despite reducing batch sizes and clearing the CUDA cache after every epoch, the maximum batch size we could stably support was 8. This was severely limiting to the models training stability and caused highly variable gradient updates. This instability was reflected in an increasing and oscillating loss trend across epochs, suggesting issues with convergence stemming from the small batch size that limited the generalizability potential of the model.

Additionally, training on subsets of the dataset ranging from 20 images up to 10,000 images revealed that smaller datasets further led to insufficient generalization, while larger subsets caused runtime crashes due to excessive memory demands. Attempts to fine-tune learning rates, number of epochs, and batch sizes provided limited improvements when we began fine-tuning. Learning rates between 2e-5 and 5e-5 were tested, but adjusting the learning rate alone did not resolve convergence issues. For example, training on 10,000 images, even with 5 epochs, required over 7 hours on an A100 GPU, significantly constraining experimental throughput.

Another critical issue emerged from the dataset format itself. The images stored in the provided .h5 file were not standard RGB images but rather compressed low dynamic range tensors with values scaled between -1 and 1. Since CLIP requires meaningful edge, texture, and color information to produce effective embeddings, these tensor-like representations proved incompatible. After identifying this issue through a deep dive into the tensor formatting, we resolved it by converting the images back to a positive, normalized RGB format, thereby restoring compatibility with CLIP's input expectations and preserving the image information and trends.

Despite the resolution of input formatting issues, fine-tuning CLIP on this data yielded embeddings that were worse than the baseline pre-trained CLIP model. Fine-tuning attempted to overfit on imperfect training data, leading to degraded performance. Consequently, we elected to use the original pre-trained CLIP embeddings in the final pipeline. In addition, we observed that fine-tuning

CLIP increased training loss variance, further motivating the decision to revert to the pre-trained embeddings for stability.

Our final training setup used the AdamW optimizer, a constant learning rate schedule, and a batch size of 8. We ran experiments on Google Colab Pro with an NVIDIA A100 GPU (40GB memory).

5.2. Training Stability Across Dataset Sizes

We recorded the loss over a range of epochs in trainng different dataset sizes: [20, 100, and 1000 images] at batches of size 8 in order to analyze our training stability.

As shown in Figure 3 and Table 1, training with 20 images resulted in generally stable but significantly slower convergence. On the other hand, training with 100 images showed high training instability at epoch 3, where we see a spike in loss to over 0.14. Training with 1000 images initially had a higher loss compared to 20 or 100 images but eventually resulted in more stable improvement across epochs.

These results showed us that while smaller datasets might acheive a lower initial loss, they are heavily at risk for overfitting or unstable gradients steps and make the model less generalizable. On the other hand, larger datasets showed more stability during learning however they required much more resources and careful optimization to ensure convergence. This analysis informed our decision to cap training datasets at sizes manageable within our teams memory and computation constraints while ensuring our model is generalizable.

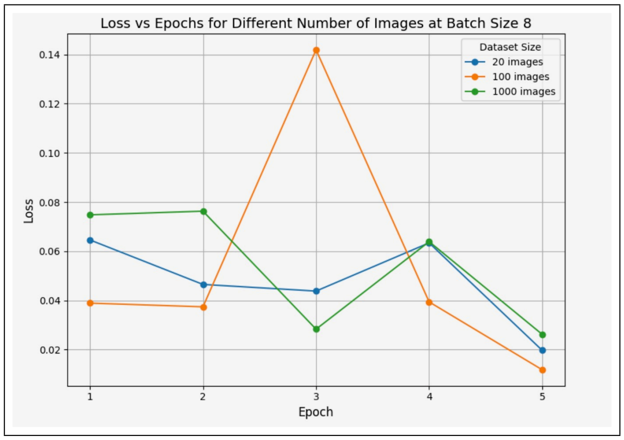

Figure 3. Loss vs Epochs for Different Number of Images at Batch Size 8. Training on small datasets led to faster but less reliable convergence. Larger datasets stabilized loss over time but at higher initial cost.

Dataset Size	Epoch (1-5) Losses
20 Images	0.0646 / 0.0465 / 0.0438 / 0.0634 / 0.0197
100 Images	0.0389 / 0.0374 / 0.1419 / 0.0394 / 0.0118
1000 Images	0.0748 / 0.0763 / 0.0283 / 0.0639 / 0.0262

Table 1. Loss per Epoch for Different Dataset Sizes at Batch Size 8. Spikes in loss reflect instability, especially for smaller datasets.

5.3. Generated images

5.3.1 Successful examples

Most of the generated images demonstrated high quality and accurately followed the provided prompts. Notable examples can be found in Figure 9.

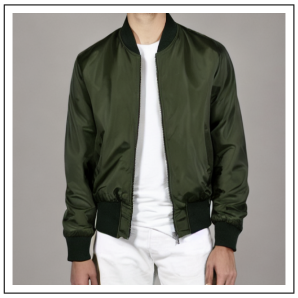

Figure 4. *

(a) Green bomber jacket

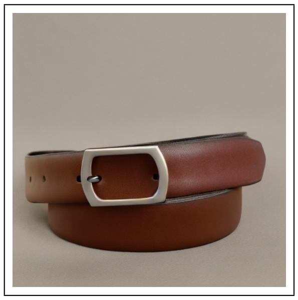

Figure 5. *

(b) Brown leather belt

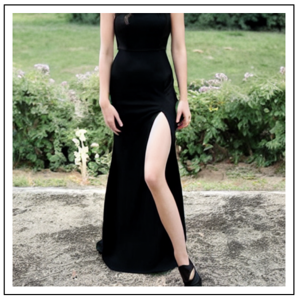

Figure 6. *

(c) Formal black dress

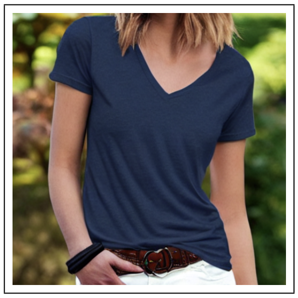

Figure 7. *

(d) V-neck t-shirt

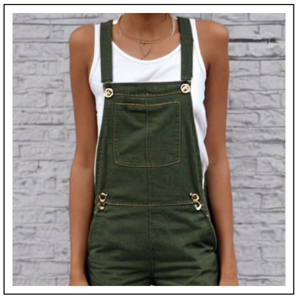

Figure 8. *

(e) Short overalls

Figure 9. Successful examples of generated clothing images conditioned on textual prompts.

5.3.2 Less successful examples

A small subset of images exhibited limitations. For example, some human body parts (Figure 13 (b)) or certain clothes elements (Figure 13 (a) and (c)) were unrealistic. These limitations may be attributed to compute constraints affecting training dataset size.

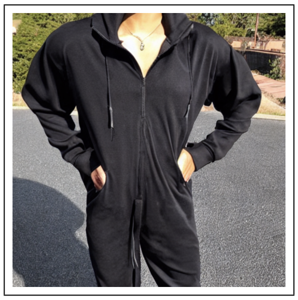

Figure 10. *

(a) Black sweatsuit

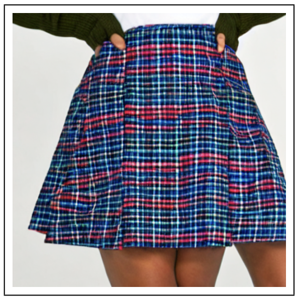

Figure 11. *

(b) Tartan skirt

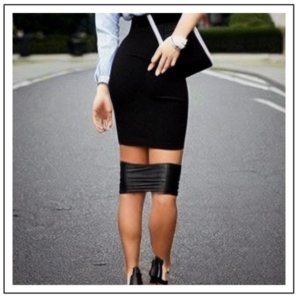

Figure 12. *

(c) Black high-waisted skirt

Figure 13. Less successful examples of generated clothing images conditioned on textual prompts.

5.4. Evaluation

5.4.1 Inception score

Our model achieved promising results based on evaluation of the 138 generated images:

- Mean Inception Score: 8.114767074584961
- Standard Deviation: 1.0872924327850342

This mean score indicates our diffusion model is generating high-quality and diverse images, comparable to professional models such as earlier versions of Stable Diffusion (e.g., Stable Diffusion v1) [8] and other popular text-to-image models like BigGAN trained on ImageNet [1], which reported inception scores in the 7–9 range. The standard deviation falls within acceptable ranges, suggesting reasonable consistency in outputs.

For comparison, the pretrained Stable Diffusion v1.4 model (trained on LAION-2B-en) achieved an Inception Score around 7.5–8 on general prompts [8], while BigGAN achieved around 8.4 on ImageNet classes [1]. This supports that our fine-tuned model performed competitively despite hardware and data limitations.

Method	Inception Score (Mean ± Std)
Pretrained SD	6.5 ± 1.2
SD + Pretrained CLIP	7.2 ± 1.1
Our Fine-tuned Model	8.1 ± 1.1

Table 2. Inception Scores comparing the pretrained Stable Diffusion (SD), Stable Diffusion with pretrained CLIP embeddings, and our fine-tuned model.

5.4.2 Ablation Studies

To evaluate how various elements of our model pipeline contributed to overall loss and model success we also conducted 3 key ablation studies. We first looked into the CLIP embedding type, being pre-trained or fine-tuned. From here we reviewed how the model performed on the original input image in a raw vs normalized RGB format. Finally we evaluated how the dataset size affected model performance and loss.

To begin with our CLIP embedding type analysis we removed our pre-trained CLIP embeddings and learned that fine-tuning CLIP on our smaller datasets consistently degraded performance and led to training instability and worse image generation quality. Thus, in our final model we used the pre-trained CLIP embeddings.

next, When using the raw, un-normalized input tensors instead of normalized RGB inputs, evaluation of our model noted key failures in capturing meaningful visual structures of the fashion images. This resulted in blurred and nonsensical image outputs. Here we learned the normalized image tensors would be best for model training.

Finally, dataset size greatly impacted output image quality. When we trained on subsets smaller than 500 images we saw poor generalization and clear overfitting issues in the output images and loss trends.

These findings show that maintaining high-quality feature embeddings, ensuring proper input normalization, and using sufficient dataset size are crucial to achieving reliable results.

5.5. Hyperparameter Tuning

During hyperparameter tuning, we varied the following variables:

- learning rates: 2e-5 to 5e-5
- optimizers: Adam vs AdamW
- batch sizes: 4, 8, 16

- epoch numbers: 2 to 10

Our initial literature review concluded that learning rates lower than 2e-5 slowed convergence too much, while rates higher than 5e-5 introduced instability and noise in loss curves. We found that 5e-5 was the fastest learning rate that maintained stability. AdamW outperformed Adam, consistently, as it yielded a smoother convergence resulted in less overfitting. A batch size of 8 was selected as a trade-off between memory availability and gradient stability. Experiments beyond 5 epochs showed diminishing returns, often leading to marginal loss reductions at the cost of large computation time. Therefore, our final configuration used a constant learning rate of 5e-5, AdamW optimizer, a batch size of 8, and 5-epochs of training.

5.6. Analysis

Overall, our results highlight how important data quality, computational resources, and hyperparameter tuning are when developing text-to-image generation models on limited datasets. Our fine-tuned model achieved strong inception scores compared to baseline diffusion models, despite significant challenges that stemmed from small batch sizes and limited compute resources. The ablation studies confirmed that pre-trained CLIP embeddings and normalized image inputs were essential for stable and high-quality outputs. Finally, analysis of loss curves emphasized the delicate balance between our dataset size, the training stability, and resource constraints. Despite these challenges, our diffusion pipeline exhibits strong image generation results.

6. Conclusion

This project was key in our learning about the challenges that come with fine-tuning a larger generative model such as the one we worked on being the CLIP and Stable Diffusion model. Specifically when it comes to resource use. Additionally, we learned a key lesson about importance dataset quality, the discovery that our images were incorrectly encoded as low-dynamic-range tensors rather than the intended RGB images severely slowed the training process, especially for CLIP, which relies on quality visual features for learning. We further experienced how compute and memory limitations, even with access to powerful GPUs like the A100, restrict training potential. Small batch sizes, limited by CUDA memory errors, resulted in unstable gradient updates and an oscillating loss. This matched with slow training times greatly slowed the fine-tuning and problem solving process.

Despite these challenges we managed to successfully develop a text-to image generation pipeline. Our final Diffusion model, that was enhanced by CLIP achieved strong results, reflected in our mean inception score of 8.1 ± 1.1. This score is on par with leading diffusion models like the earlier versions of Stable Diffusion. Further our success was further indicated by the quality of images generated, which were by human evaluation easily recognizable and passing as real fashion images.

These challenges have highlighted many paths for future improvements. Moving forward we would aim to leverage memory efficient fine-tuning methods, such as LoRA (Low-Rank Adaptation) or other parameter-efficient fine-tuning strategies. We hope these strategies will be more successful in alleviating memory bottlenecks, as recent studies have shown that LoRA can drastically reduce the number of trainable parameters while achieving performance comparable to full fine-tuning, allowing large models to be adapted on limited computing resources [3]. Additionally, techniques like gradient accumulation, which accumulates gradients over multiple smaller batches before performing an optimizer step, could help simulate larger batch sizes, working around our small batch size of 8 that was limited by CUDA memory resources. This would be beneficial because larger batch sizes lead to more stable gradient estimations and help the loss curve to behave more smoothly while encouraging convergence.

Extensions of this project could focus on integrating consumer preference data from DeepFashion's consumer preferences subset. This would aim to generate realistic fashion images that are intended to maximize styles that will be of high demand by consumers. Expanding our model to support these additions or multi-attribute manipulation based on user input could also greatly enhance its usability for small business owners and the fashion industry.

Overall, this project showed us the complexity of fine-tuning foundation models but also sharpened our understanding of the practical considerations necessary for successful domain adaptation.

References

[1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations (ICLR)*, 2019. 6

[2] W.-H. Cheng, S. Song, C.-Y. Chen, S. C. Hidayati, and J. Liu. Fashion meets computer vision: A survey. *arXiv*, n.d. arXiv preprint. 1

[3] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Weizhu Wang, and Yonghui Chen. Lora: Low-rank adaptation of large language models. *arXiv*, 2022. arXiv preprint arXiv:2106.09685. 4

[4] T. Islam, A. Miron, X. Liu, and Y. Li. Fashionflow: Leveraging diffusion models for dynamic fashion video synthesis from static imagery. *arXiv*, 2024. arXiv preprint. 2

[5] G. Kim, T. Kwon, and J. C. Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. *CVPR 2022*, 2022. Conference paper. 2

[6] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with clip latents. *arXiv*, 2022. arXiv preprint. 2

[7] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. *arXiv*, 2021. arXiv preprint. 2

[8] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. *arXiv*, 2022. arXiv preprint. 2, 3, 6

[9] T. Zhang, Z. Wang, J. Huang, M. M. Tasnim, and W. Shi. A survey of diffusion based image generation models: Issues and their solutions. *arXiv*, 2023. arXiv preprint. 2