Style-Transfer with Diffusion Models, Project overview

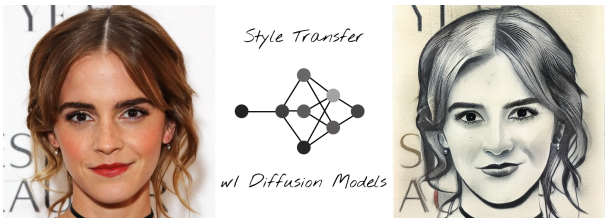
Filip Dabkowski, Zuzanna Warchoł

March 6, 2025

1 Overview

1.1 Style Transfer

For non-artists, recreating their favorite photo in the style of Van Gogh or making it look like a hand-drawn sketch is challenging. Style transfer solves this problem by extracting the artistic style from a reference image (e.g., Starry Night) and applying it to another image that contains the desired content (e.g., a portrait of Emma Watson). This allows for artistic transformations without requiring manual painting or drawing skills.



1.2 Goal of the Project

This project was developed as a learning experience for new members of the Gradient student circle. The final version of the project will be showcased at FOKA during the Open Days at PG on March 21st.

During the Open Days, we aim to demonstrate style transfer using multiple artistic styles applied to the faces of high school students or anyone visiting our station. The demo will either be connected to a camera for real-time transformations or allow users to upload their own photos for style transfer.

1.3 Why use Diffusion Models?

The concept of style transfer is not new. Initially, it was introduced using Convolutional Neural Networks (CNNs) [3], where it was discovered that higher convolutional layers primarily capture the content of an image, while lower layers capture its style properties.

Subsequently, new image generation models emerged:

- Generative Adversarial Networks (GANs) were the first major improvement.
- Diffusion Models (DMs) [4] later surpassed GANs and became the state-of-the-art in image generation.

Since our goal was to explore the most relevant and cutting-edge techniques, we chose to focus on Diffusion Models for this project.

2 Methodology

Diffusion Models (DMs) are a type of score-based generative model [7] that learn the probability distribution of a dataset, enabling the generation of new samples from this distribution. In the context of image-generation Diffusion Models, the data space consists of all training images of a given resolution, treated as data points. The underlying probability distribution that the model seeks to learn assigns high probability to realistic images and low probability to everything else.

During the sampling process, a random point is drawn from the data space, which initially represents random noise. Score-based models then learn to predict a vector that, when added to this data point, increases its probability within the estimated distribution. In the context of Diffusion Models, this process effectively transforms random noise into a visually coherent image.

The model learns this probability distribution through a two-step process:

1. Noise Addition: Gaussian noise is added to a data point (image), moving it further from high probability points, closer to random noise that will be generated during sampling.
2. Denoising The model learns to predict the negative of the noise from the noisy image, removing the noise increases its probability (brings it closer to realistic images).

This denoising process allows Diffusion Models to generate high-quality images by learning how to reverse the degradation introduced during training.

2.1 Pre-trained Models Used

Due to the limited timeframe for this project, only two months, with half of that period overlapping with the exam session, we decided to utilize an existing research paper that incorporates pre-trained components. We selected the DreamStyler [1] paper, which builds upon Stable

Diffusion and Textual Inversion for training while leveraging ControlNet to assist with style-transfer tasks.

Stable Diffusion [6] is a groundbreaking Diffusion Model (DM) and the first capable of generating high-resolution images on consumer hardware. It is an implementation of the Latent Diffusion Model (LDM), which operates in the latent space rather than directly in pixel space. Traditional pixel-based diffusion models consume significant computational resources to generate detailed pixels, even when such detail may not contribute much to semantic or object-level understanding. In contrast, LDM generates images in the latent space, focusing on semantic information, while pixel-level details are added during the final conversion from latent space back to pixel space.

Contrastive Language-Image Pre-Training (CLIP) [5] is integrated into Stable Diffusion as its conditioning input for text prompts. CLIP itself was designed for multiple tasks, with its core purpose being to map the embedding of an image into the same space as the embedding of its textual description.

CLIP consists of two encoders:

- A Vision Transformer (ViT) for processing images.
- A Transformer for processing text.

Within diffusion models, CLIP processes text prompts provided by the user or during training. The prompt is first tokenized into individual components, which are then transformed into corresponding embedding vectors. These embeddings pass through the encoding transformer, which encodes them into a single vector. This vector is then used as a conditioning factor in the diffusion process.

Textual Inversion [2] is a technique for fine-tuning large text-guided image generation models to learn new concepts. Retraining such models every time a new object needs to be generated is impractical, as it would require a vast dataset and substantial computational resources, such as a GPU farm. Fortunately, these models already exhibit strong zero-shot image-generation capabilities.

Instead of fine-tuning the entire model, Textual Inversion freezes the diffusion model (DM) and trains a new token embedding vector. This allows the model to generate images of a specific object or concept with minimal data. Typically, only three to five images are needed for training using a prompt like *"Picture of S^*"*, where S^* represents the new learned token. Once trained, this token can be used in various prompts, such as:

- *"Crochet S^*"*
- *"A S^* themed lunchbox"*

This enables the model to generate novel variations of the concept in different contexts.

ControlNet [8] is a method for fine-tuning Stable Diffusion to accept additional conditioning inputs, such as depth maps, bounding boxes, and edge detections. Fine-tuning Stable Diffusion directly is feasible but computationally expensive. Instead, ControlNet freezes Stable Diffusion's weights and trains only a small additional network, referred to as ControlNet.

A trained ControlNet enhances the denoising process of Stable Diffusion by incorporating extra conditioning data. This approach significantly reduces computational costs by several orders of magnitude while maintaining high-quality results.

2.2 Training / Fine-tuning Approach

DreamStyler Model leverages Stable Diffusion (SD) for image generation and Textual Inversion (TI) to learn an embedding vector that represents the style of a given image. During training, we iteratively generate the style image using the prompt: *"Picture of C^* in the style of S^*"*

Where:

- S^* represents the learned style token and its corresponding embedding vector.
- C^* is a description of the training image, emphasizing objects and composition.

Since the Stable Diffusion weights remain frozen throughout training, this approach helps prevent overfitting and avoids distorting the pre-trained SD model. However, the training process is highly dependent on the quality of the C^* prompt, if the prompt does not accurately describe the image, the generated outputs may contain artifacts.

Moreover, a single S^* token may not always be sufficient to encode complex style details. To address this, the authors of the original paper introduced Multi-Stage Textual Inversion, where N different S^* embedding vectors are trained at different stages of the denoising process, allowing for a more nuanced representation of the style.

2.3 Sampling and Style Transfer

During the sampling and image generation process, we use a prompt similar to the one used during training: *"Picture of a cat in the style of S^*"*

Where S^* represents the learned style embedding vectors.

For style transfer, in addition to conditioning the model with a text prompt, we utilize ControlNet to provide additional conditioning information derived from the content image. This allows for better preservation of structure while applying the learned style, resulting in more coherent and visually accurate outputs.

**The whole text was written by a human with Generative and not Generative AI being used as tools for spell checking, grammar checking, and suggestions of better phrasing.*

References

[1] Namhyuk Ahn, Junsoo Lee, Chunggi Lee, Kunhee Kim, Daesik Kim, Seung-Hun Nam, and Kibeom Hong. Dreamstyler: Paint by style inversion with text-to-image diffusion models, 2023.

[2] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022.

[3] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style, 2015.

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[6] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

[7] Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation, 2019.

[8] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023.