Midterm report: Controllable Image Style Transfer via Visual Autoregressive Modeling

Liqi Jing Dingming Zhang Peinian Li

November 23, 2025

Abstract

This project studies reference-based image style transfer: given a content image and a style image, the goal is to generate an output that preserves the structural semantics of the content while adopting the artistic texture of the style. We build on the Visual Autoregressive Modeling (VAR) framework and formulate style transfer as conditional discrete sequence modeling in a learned latent space. Images are decomposed into multi-scale representations and tokenized into discrete codes by a VQ-VAE; a transformer then autoregressively models the distribution of target tokens conditioned on style and content tokens. To inject style and content information, we introduce a blended cross-attention mechanism in which the evolving target representation attends to its own history, while style and content features act as queries that decide which aspects of this history to emphasize. A scale-dependent blending coefficient controls the relative influence of style and content at each stage, encouraging the synthesized representation to align with both the content structure and the style texture without breaking the autoregressive continuity of VAR. We fine-tune this StyleVAR model from a pretrained VAR checkpoint on a large triplet dataset of content–style–target images and evaluate it on held-out pairs. Qualitative results indicate that the method can transfer texture while maintaining semantic structure, especially for landscapes and architectural scenes, while a generalization gap on internet images and difficulty with human faces highlight the need for better content diversity and stronger structural priors.

1 Introduction

Reference-based image style transfer aims to generate an image that keeps what is in a content image while changing how it looks according to a style image. Concretely, the spatial layout and object semantics of the content image should be preserved, while colors, textures, and local patterns follow the chosen style. This setting is useful in artistic creation, visual prototyping, and controllable data augmentation, where users wish to restyle an existing scene without altering its high-level meaning. Achieving this objective requires a model that can respect content geometry and semantics while still expressing strong and diverse stylistic effects.

Balancing content preservation and style strength is challenging. If the model focuses too much on content, stylization becomes weak and the output resembles a slightly modified version of the original image. If the model overemphasizes style, it may distort object shapes or introduce artifacts that break semantic coherence. Styles also vary widely: some primarily change global tone, while others rely on fine-grained textures and patterns. A robust method must therefore provide a principled way to combine information from the content and style images, so that the resulting image is structurally faithful yet stylistically rich.

Earlier style transfer frameworks based on feed-forward CNNs and GANs showed that it is possible to learn powerful style priors, but they often come with practical limitations: models may need to be trained or adapted for specific styles, and training can be unstable or sensitive to dataset design. More recently, diffusion-based approaches have become the dominant backbone for high-quality image generation and have been adapted to style transfer as well. However, diffusion models typically require many iterative denoising steps, leading to slow sampling and high computational cost, and they often depend on additional guidance mechanisms or prompt-like controls that are not always ideal when we focus purely on image-to-image style transfer. These factors motivate exploring alternative formulations that retain strong generative capacity while offering better efficiency and more direct conditioning on content and style images.

In this project, we adopt the Visual Autoregressive Modeling (VAR) framework and cast style transfer as conditional discrete sequence modeling in a multi-scale latent space. Each image is decomposed into a hierarchy of feature maps and tokenized into discrete codes by a VQ-VAE encoder. The style image and content image are represented as sequences of tokens across scales, and the target image is generated scale by scale, with each set of target tokens conditioned on the history of previously generated tokens as well as the corresponding style and content tokens. This formulation explicitly encodes the intuition that the target should be consistent with its own past while being guided by both content structure and style appearance.

2 Methodology

2.1 Blended Cross-Attention Autoregressive Modeling

Formulation. In the context of our style transfer task, the objective is to predict a target image that preserves the structural semantics of a content image x_c while adopting the artistic texture of a style image x_s. Adopting the framework of Visual Autoregressive Modeling (VAR), we decompose images into multi-scale representations. Each scale's feature map is tokenized into discrete tokens. Formally, the style image tokens are denoted as $S = \{s_1, s_2, \ldots, s_K\} = \mathcal{E}(x_s)$, and the content image tokens as $C = \{c_1, c_2, \ldots, c_K\} = \mathcal{E}(x_c)$, where K represents the total number of scales and $\mathcal{E}(\cdot)$ denotes the VQ-VAE tokenization process.

The generation of the target image, denoted as $R = \{r_1, r_2, \ldots, r_K\}$, proceeds in a scale-wise autoregressive manner. Consequently, the autoregressive likelihood for StyleVAR is

formulated as:

$$\mathcal{P}(x|x_s, x_c) = \prod_{k=1}^{K} \mathcal{P}(r^k|r^{<k}, s^k, c^k) \tag{1}$$

where r_k denotes the target features at the k-th scale, and $r_{<k} = r_{1:k-1}$ represents the history of generated target features prior to the k-th scale. Crucially, this formulation implies that the generation of the current scale is conditioned not only on the target's own history but also on the corresponding scale-specific features from the style and content conditions.

Model Structure. Figure 1 illustrates the architecture of StyleVAR. Building upon the VAR backbone, we introduce a Blended Cross-Attention mechanism to inject style and content information into the target image generation process. Within each transformer block, the feature update process is expressed as:

$$h_{\text{new}} = h + [\alpha \cdot \text{Attn}(Q = s^k, K = h, V = h) + (1 - \alpha) \cdot \text{Attn}(Q = c^k, K = h, V = h)] \tag{2}$$

where h represents the input target features at stage k (or the output of the preceding transformer block). The term α_k is a heuristic hyperparameter governing the blending ratio between style and content information. Through this mechanism, h passes through the transformer blocks and is iteratively updated via the injection of blended attention, ensuring the synthesized features align with both the content structure and style texture.

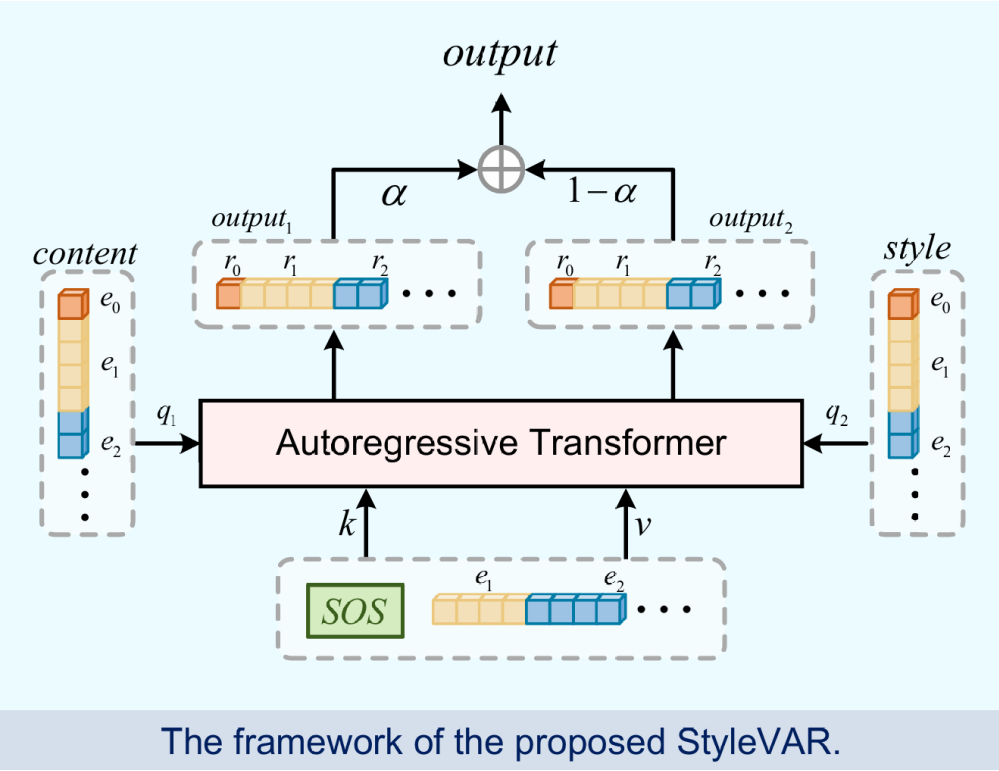

Figure 1: Structure of StyleVAR Transformers.

2.2 Training and Inference

Inference. The inference process begins by initializing the start token at the first scale using content features extracted via a ResNet-18 backbone, which are projected to the embedding dimension via an MLP.

A critical component of the inference logic is the progressive accumulation of features. Unlike the generated target, the style and content features are fully observable; therefore, we pre-calculate their multi-scale ground truth tokens via VQ-VAE decomposition and accumulate them prior to inference. For the target image generation, we maintain a cumulative feature map, denoted as $\hat{f}$. At each step, the generated tokens r_k are quantized and added to $\hat{f}$ in a residual manner. To serve as the input for the subsequent scale $k+1$, $\hat{f}$ is downsampled to the appropriate resolution. This downsampled map acts as the "next-scale input," ensuring that the autoregressive generation maintains structural coherence across resolutions.

Training. During the training phase, we employ a teacher-forcing strategy. The model concatenates the start token with the ground truth tokens of the target image across all other scales. Following the vanilla VAR paradigm, the model predicts the logits for stages 1 to K in parallel. We then calculate the Cross-Entropy loss between the predicted logits and the ground truth codebook indices. Note that, similar to inference, the input to the model at any stage k is the accumulation of ground truth features from all preceding scales (1 to $k-1$), ensuring the model learns to refine coarse-grained features into fine-grained details.

Algorithm 1: StyleVAR Training

Inputs: target image x, style image x_s, content image x_c;
Inputs: VQ-VAE Encoder $\mathcal{E}$, Codebook Z;
Inputs: Transformer $\mathcal{T}_\theta$ (parameters θ);
// 1. Multi-scale Token Decomposition (Ground Truth)
$R = \{r_1, \ldots, r_K\} \leftarrow \text{VQVAE\_Encoding}(x)$;
$S = \{s_1, \ldots, s_K\} \leftarrow \text{VQVAE\_Encoding}(x_s)$;
$C = \{c_1, \ldots, c_K\} \leftarrow \text{VQVAE\_Encoding}(x_c)$;
// 2. Initialize Start Token
$r_{start} \leftarrow \text{MLP}(\text{ResNet}(x_c))$;
// 3. Parallel Prediction (Teacher Forcing)
$R_{input} \leftarrow [r_{start}, r_2, \ldots, r_K]$; // concatenate start token
$L_{1:K} \leftarrow \mathcal{T}_\theta(R_{input}, S, C)$; // Predict logits for all scales
// 4. Optimization
$\mathcal{L} \leftarrow \text{CrossEntropy}(L_{1:K}, R)$;
$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}$;
Return: Optimized parameters θ;

2.3 Discussion: Rationale for Attention Configuration

Assignment of Query, Key, and Value. A critical design decision in StyleVAR is the assignment of the target image features to the Key (K) and Value (V) roles, while assigning the style/content features to the Query (Q) role. This configuration diverges from standard

cross-attention mechanisms where the target usually acts as the Query to retrieve information from the condition (K, V).

Preserving Autoregressive Continuity. The efficacy of the vanilla VAR architecture hinges on its "next-scale prediction" paradigm. To generate the token map at stage k, the model requires a comprehensive aggregation of the entire history of generated tokens, $r_{<k} = r_{1:k-1}$, rather than solely relying on the immediate predecessor r_{k-1}. By designating the target feature history as K and V, we ensure that the attention mechanism explicitly aggregates information from the target's own past. In this setup, the style and content features (as Q) act as a "search query", determining which parts of the target's history are most relevant to emphasize for the current generation step.

Algorithm 2: StyleVAR Inference (Autoregressive)

Inputs: style image x_s, content image x_c;
Inputs: Trained Transformer $\mathcal{T}_\theta$, VQ-VAE Decoder $\mathcal{D}$;
Inputs: Hyperparameters: steps K, resolutions $(h_k, w_k)_{k=1}^K$;
// 1. Prepare Conditions
$S \leftarrow \text{VQVAE\_Encoding}(x_s)$;
$C \leftarrow \text{VQVAE\_Encoding}(x_c)$;
// 2. Initialization
$r_{input} \leftarrow \text{MLP}(\text{ResNet}(x_c))$;
$\hat{f} \leftarrow 0$; // Initialize cumulative feature map
// 3. Stage-wise Generation
for $k = 1$ **to** K **do**
  $Logits_k \leftarrow \mathcal{T}_\theta(r_{input}, s_k, c_k)$;
  $r_k \sim \text{Sample}(Logits_k)$; // Top-k/Top-p sampling
  $z_k \leftarrow \text{lookup}(Z, r_k)$;
  // Accumulate residual to max resolution
  $\hat{f} \leftarrow \hat{f} + \text{interpolate}(z_k, h_K, w_K)$;
  // Prepare input for next scale
  if $k < K$ **then**
    $r_{input} \leftarrow \text{interpolate}(\hat{f}, h_{k+1}, w_{k+1})$;
  end
end
// 4. Image Reconstruction
$\hat{x} \leftarrow \mathcal{D}(\hat{f})$;
Return: Generated image $\hat{x}$;

Contrast with Alternative Configurations. Conversely, if we were to assign the target features to Q and the style/content to K and V, the model would primarily attend to the external conditions (s and c) to construct the next scale. While this maximizes information injection, it risks disrupting the autoregressive consistency fundamental to VAR. The model might rely too heavily on the static conditions and neglect the structural continuity required to evolve r_{k-1} into r_k.

Theoretical Viability. It is worth noting that by setting V as the target features, the

output of the attention block becomes a linear combination of the target's own history. While this does not directly "copy" pixels from the style image, the style-guided re-weighting (via the $Q \times K^T$ score) is theoretically sufficient to modulate the generative trajectory, effectively steering the autoregressive process to adopt the desired stylistic characteristics while maintaining the structural integrity of the target.

3 Experiments

3.1 Implementation Details

Training Setup. We initialized the weights of StyleVAR using the pre-trained vanilla VAR model. To adapt the architecture for style transfer, we froze the VQ-VAE component while fine-tuning the full 600M parameters of the transformer. Since StyleVAR utilizes a dual-stream input (target features and content/style condition features), the original projection layers of the vanilla VAR—responsible for mapping image features to Query, Key, and Value (QKV) - were duplicated to initialize the distinct projection layers for both the target and the condition streams. The Feed-Forward Networks (FFN) were initialized with the original VAR parameters.

We fine-tuned the model for a total of 8 epochs. The learning rate was scheduled at 5×10^{-4} for the first 6 epochs and decayed to 1×10^{-4} for the final 2 epochs. Training was conducted on two NVIDIA A100 (40GB) GPUs. To accommodate memory constraints, we employed a physical batch size of 4 per GPU with 128 gradient accumulation steps, resulting in an effective global batch size of 1024.

Dataset. We utilized the OmniStyle-150K dataset for training, which consists of 143,992 triplets: (content image, style image, target image). The data is structured such that each target image is paired with its corresponding source content and style inputs. To enhance model robustness and force the network to learn fine-grained structural and textural details, we applied data augmentation during preprocessing. Specifically, we applied rotation and brightness adjustments to the content images, and random cropping to the style images.

3.2 Results

Quantitative Analysis. After 8 epochs of fine-tuning, the model demonstrated promising convergence. We evaluated the model on the validation set, achieving a top-1 accuracy of 14.72% averaged across all scales (Mean Accuracy) and 16.26% at the final resolution scale (Tail Accuracy).

Qualitative Analysis. To visually assess the model's performance, we generated samples using the validation set.

3.3 Limitations and Discussion

Despite strong performance in the training and validation sets, our qualitative evaluation revealed a generalization gap when testing unseen images collected from the internet. This indicates a degree of overfitting to the training distribution. Upon further analysis of the

OmniStyle-150K dataset, we identified a data imbalance: while the dataset contains about 150k triplets, these are generated from a limited pool of approximately 1,800 unique content images. Given the StyleVAR's capacity (600M parameters), the model likely memorized the structural priors of this limited content set rather than learning a generalized representation of content structure.

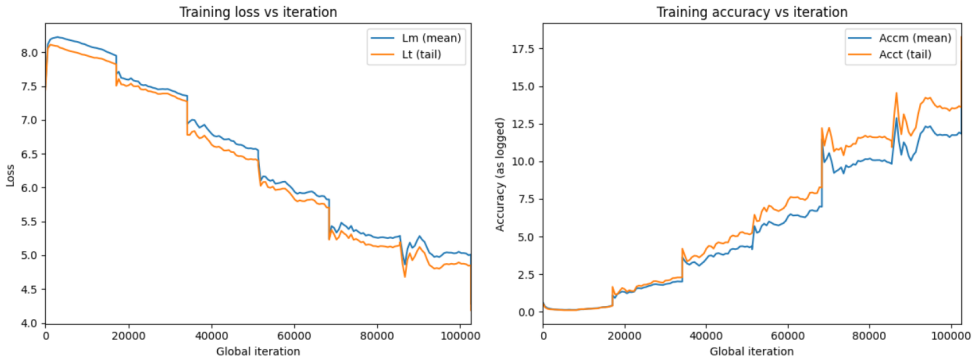

Figure 2: Loss and accuracy of training set across iterations.

Furthermore, we observed a performance disparity between different semantic domains. The model excels at stylizing landscapes and architectural scenes but struggles with human faces. This is likely due to two factors:

1. Complexity. Facial topology is significantly more complex and sensitive to structural deformation than natural scenes (e.g., mountains or buildings).

2. Perceptual Sensitivity. Human visual perception is acutely sensitive to structural anomalies in facial features.

4 Future Plan

Going forward, we plan to address both the generalization and controllability aspects of StyleVAR.

Data Augmentation. On the generalization side, the analysis of our current dataset shows that many triplets are derived from a relatively small pool of unique content images, which makes it easy for a high-capacity autoregressive model to memorize structural priors instead of learning broadly applicable content representations. We therefore plan to expand and rebalance the training data with more diverse content images, especially in challenging semantic domains such as human faces, and to explore regularization and augmentation strategies that encourage the model to focus on transferable structural patterns rather than dataset-specific configurations.

Classifier-free–style guidance. On the controllability side, an important direction is to expose more explicit knobs for adjusting style strength at inference time. One promising idea is to extend StyleVAR with a classifier-free–style guidance mechanism: during training, we can occasionally drop the style conditioning, teaching the model to generate both style-conditioned and "style-agnostic" predictions; at inference, we can interpolate between these

two predictions when sampling tokens, effectively turning style influence into a continuous dial. This would complement the existing blended attention design by giving users a straightforward way to trade off content fidelity against stylistic intensity, while still operating in the same latent autoregressive framework.

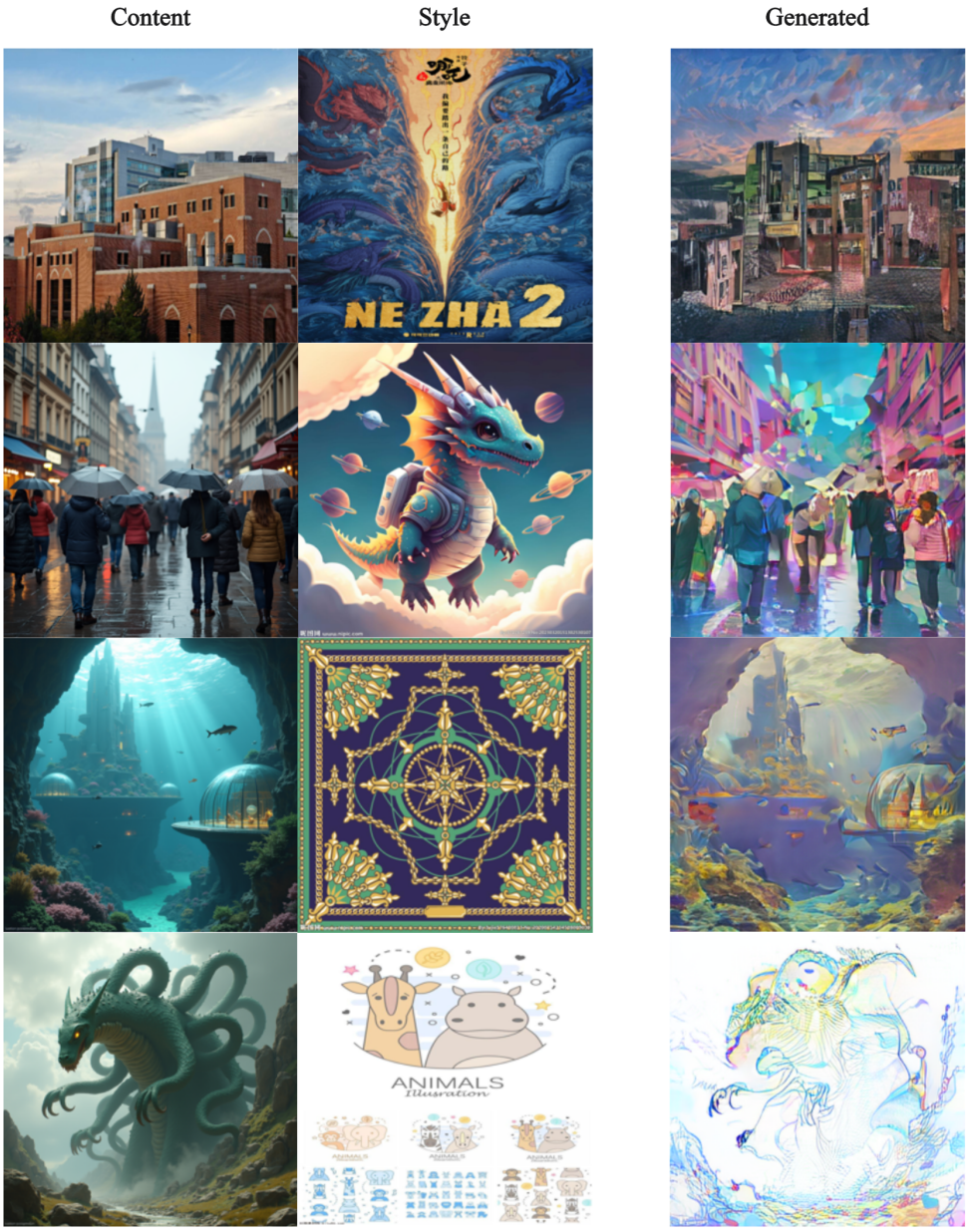

Figure 3: The generated images demonstrate that the model successfully transfers texture while maintaining the semantic structure of the content.

GRPO driven Unsupervised learning. Finally, to transcend the limitations of supervised training—which relies heavily on the availability of high-quality paired ground truth—we plan to explore a second-stage unsupervised fine-tuning phase driven by Reinforcement Learning (RL). Specifically, we aim to adapt the Group Relative Policy Optimiza-

tion (GRPO) algorithm to the visual generation domain. In this framework, the StyleVAR model functions as the policy network. For any given content–style pair, the model samples a group of diverse outputs, which are then evaluated using perceptual metrics (e.g., VGG-based style and content losses) as a dense reward signal. By optimizing the policy to favor outputs with higher relative rewards within the group, the model can learn to minimize perceptual loss directly, even through the non-differentiable discrete sampling steps. Crucially, unlike traditional Actor-Critic methods, GRPO eliminates the need for a separate Critic model; this reduction in memory overhead is particularly advantageous for high-capacity visual autoregressive models, allowing us to allocate more resources to batch size and context length during optimization.

In parallel, we plan to investigate data-driven strategies for setting or adapting the blending between content and style, so that the model can automatically adjust to different types of content–style pairs and reduce failure cases in domains that are currently underrepresented or particularly sensitive to structural distortions.

Team members

Liqi Jing: Implementing the blended attention module for StyleVAR.

Dingming Zhang: Implementing the training framework and lora fine-tuning.

Peinian Li: Implementing the dataset loader.

References

[1] Tian, K., Jiang, Y., Yuan, Z., Peng, B., & Wang, L. (2024). Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*.

[2] Wang, Y., Liu, R., Lin, J., Liu, F., Yi, Z., Wang, Y., & Ma, R. (2025). OmniStyle: Filtering High Quality Style Transfer Data at Scale. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[3] Zhang, Y., Huang, N., Tang, F., Huang, H., Ma, C., Dong, W., & Xu, C. (2023). Inversion-Based Style Transfer with Diffusion Models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 10077–10086). IEEE/CVF. https://doi.org/10.1109/CVPR52729.2023.00978.

[4] DiffSynth-Studio. ImagePulse-StyleTransfer[Dataset]. ModelScope. https://www.modelscope.cn/datasets/DiffSynth-Studio/ImagePulse-StyleTransfer