ARBITRARY STYLE TRANSFER

Morteza Mogharrab
Department of Computing Science
University of Alberta
Edmonton, Canada
mogharra@ualberta.ca

M Mahmud Hasan
Department of Computing Science
University of Alberta
Edmonton, Canada
mmahmud3@ualberta.ca

Md Solehin Islam
Department of Computing Science
University of Alberta
Edmonton, Canada
solehin1@ualberta.ca

1 Short Abstract

This paper focuses on arbitrary style transfer using machine learning to apply the visual style of one image onto another while maintaining the original content. The project aims to develop an efficient and flexible deep-learning model for style transfer, emphasizing distortion reduction. The methodology involves dataset collection, preprocessing, and CNN-based algorithm implementation for universal style transfer. The model is trained to learn diverse styles for transforming ordinary images into artistic renditions. The project begins with a VGG-19 pre-trained model on ImageNet, using the Gram matrix for style blending. To overcome VGG-19 limitations, the ResNet-34 model is explored, and the paper details its implementation, including Gram matrix utilization, Loss and Weights Calculations, and outcomes. The results highlight advancements in arbitrary style transfer, addressing challenges in pre-existing models and suggesting future avenues for deep learning and image processing exploration.

2 Introduction

This report delves into the realm of computer vision and image processing, with a primary focus on achieving nuanced and arbitrary style transfer through machine learning. The goal is to seamlessly apply the visual style of one image to another while preserving its inherent content. The research involves dataset collection, preprocessing, and the development of a Convolutional Neural Network (CNN)-based algorithm for universal style transfer.

The study begins by leveraging the VGG-19 pre-trained model on ImageNet, utilizing the Gram matrix for style blending. To address limitations, the exploration extends to the ResNet-34 model, providing insights into its implementation, including Gram matrix utilization, Loss and Weights Calculations, and outcomes.

The literature review section contextualizes the research by chronicling advancements and challenges in deep learning-based style transfer over the past decade. It covers various methodologies, such as iterative optimization, instance normalization, and binary selection units, exploring their contributions and constraints.

The core of the paper details the proposed model, dissecting the roles of VGG-19 and ResNet-34 in style transfer. The Gram matrix is highlighted as crucial for robust feature extraction. The exploration of individual layer style weights, content and style weights, loss calculations, and the iterative update process provides insights into achieving optimal style transfer outcomes.

In the domain of arbitrary style transfer, the research not only presents a model but contributes through critical analysis of existing literature, benchmarking against prior methodologies, and suggesting future exploration avenues in deep learning and image processing. The research unfolds new horizons for the synthesis of art and technology, enriching visual storytelling and creative expression.

3 Literature Review

The literature on deep learning-based style transfer in image processing has witnessed significant advancements and encountered various challenges. Initially, Gatys et al. proposed a method relying on iterative optimization, but its computational cost was high [16]. To address this, fast feed-forward approaches were introduced, training networks to minimize feature and style reconstruction loss, although they were initially limited to fixed styles. Dumoulin et al. introduced instance normalization, allowing for 32 styles, and Li et al. encoded 1,000 styles using a binary selection unit [6] [19]. However, both methods had constraints in transferring arbitrary styles.

Several techniques aimed to match mean and variance using feature vectors, neglecting covariance and impacting image synthesis. Li et al. addressed this by applying whitening and coloring transform with pre-trained auto-encoders, albeit at a high computational cost [10]. Shen et al. proposed a meta-network, but its substantial memory requirements and lack of explicit modeling of second-order image statistics posed challenges [25].

Traditional stroke-based rendering and filtering methods, though accurate in producing style patterns, were computationally expensive. Unified models like AdaIN [17] and DIN [18], utilizing conditional instance normalization, simplified output quality. WCT converted features using whitening and coloring with covariance but faced computational challenges [19]. Avatar-Net employed a patch-based style decorator, maintaining global and local styles, yet encountered difficulties [23].

SANet, similar to Avatar-Net, used soft attention for style decoration, incorporating identity loss to preserve content structure. However, the dynamic production of affine parameters in the instance normalization layer could lead to distortion artifacts [20]. Various approaches adopted encoder-decoder frameworks, integrating feature transformation and/or fusion. GANs showed success by treating style images as a separate domain [24]. CUT proposed patch-wise contrastive learning, addressing content preservation and mode collapse [22]. TUNIT applied contrastive learning but struggled with arbitrary style transfer due to the assumption of semantic similarity [14]. IEST used feature statistics as style priors and contrastive learning, but the contrastive loss was computed only on generated results [15].

Deep Photo Style Transfer presented a method for photographic style transfer, acknowledging its issues and proposing an algorithm to address them [1]. A discussion on CNNs highlighted their role in developing image recognition models with a simple yet precise architecture [1].

Line Search-Based Feature Transformation introduced a transformation for photorealistic style transfer, allowing control over the balance of content preservation and style infusion [3]. Photorealistic Style Transfer via Wavelet Transforms proposed a wavelet-corrected transfer that preserved structural information and statistical properties, achieving high-resolution processing speed [4].

Learning Linear Transformations for Fast Image and Video Style Transfer presented an effective linear propagation model, quicker and more GPU-friendly than other approaches, suitable for a wider range of applications [5]. A Learned Representation For Artistic Style constructed a scalable deep network capturing various artistic styles, providing a valuable step towards rich models of paintings [6].

High-Resolution Network for Photorealistic Style Transfer employed a high-resolution network for better results in photo-realistic style transfer with less image distortion [7]. A Neural Algorithm of Artistic Style introduced a synthetic system using a deep neural network for creative image generation [8]. Image Style Transfer Using Convolutional Neural Networks proposed a structural algorithm for artistic style construction based on convolutional neural networks [9].

The literature also addressed image compression using Semantic Learning for Image Compression (SLIC), emphasizing the advantageous trade-off between distortion and compression ratio using neural networks [SLIC].

Table 1: This table provides a summary of the advancements and trending technologies in Style Transfer from 2011 to 2022.

METHOD	YEAR	TECHNOLOGY
Mathematical techniques to extract and transfer style features	2012, 2011	Mathematical algorithms, image processing techniques
Neural networks for style transfer with iterative optimization	2015	CNNs (VGG - ResNet)
Fast feed-forward approaches for direct style translation	2016	CNNs
Instance normalization for enhanced quality and wider style range	2016	CNNs, instance normalization
Binary selection unit and pre-trained auto-encoders for 1,000 transferable styles	2017	Autoencoders, binary selection unit
Improved loss functions for realism and quality	2017	CNNs, loss functions
Attention mechanisms for controlled style transfer	2017	CNNs, attention mechanisms
Multi-scale style transfer for natural stylizations	2018	CNNs, multi-scale processing
Style-aware normalization for consistent results	2018	CNNs, style-aware normalization
Unsupervised style learning for style representation	2018	CNNs, unsupervised learning
StyleGAN for synthesizing images with various styles	2018	Generative adversarial networks (GANs)
Style adaptation for specific content images	2019	CNNs, style adaptation

Continued on next page

Table 1 – continued from previous page

METHOD	YEAR	TECHNOLOGY
Hierarchical style transfer for multiple styles	2019	CNNs, hierarchical processing
Video style transfer for temporally consistent stylizations	2020	CNNs, video processing techniques
Text-to-style transfer for stylizing images based on descriptions	2020	CNNs, natural language processing (NLP), transformers
Style editing for manipulating style features	2021	CNNs, style editing techniques
Style transfer in the wild for real-world images	2021	CNNs, image enhancement techniques
Real-time style transfer for interactive applications	2022	CNNs, real-time processing techniques
Hardware acceleration for real-time performance	2022	GPUs, specialized hardware
Diffusion-based style transfer	2022	Diffusion models

4 Proposing Model

This section delves into the use of VGG-19 and ResNet-34 convolutional neural network architecture. Originally designed for image classification, these models are repurposed to convey style, focusing on their design principles. This section compares VGG-19 and ResNet-34 considering factors such as simplicity and computational efficiency. The Gram matrix has been highlighted for its critical role in robust feature extraction, capturing correlations in essential feature maps for style translation. This method introduces a different style transfer approach with adjustable stylization levels through layered style weights. The iterative process of image updating and loss calculations is explained. This section lays the foundation for testing, emphasizing architecture selection, warm matrix, and innovative weight tuning for different style outputs. The following sections will review the experimental setup, results, and discussions.

4.1 VGG-19 Model

The VGG-19 model, devised by the Visual Geometry Group (VGG) at the University of Oxford, is a convolutional neural network renowned for image classification, denoting its 19 layers. Despite its original purpose, VGG-19 has extended utility in various computer vision domains, particularly in style transfer. The architecture comprises 19 layers, including 16 convolutional and 3 fully connected layers, with compact 3x3 filters and ReLU activation. Max-pooling layers enable spatial downsampling with 2x2 filters and a stride of 2. The distinctive layer stacking, using small filters, enhances the network's capacity to discern intricate features.

The strategic use of 3x3 filters and minimal stride enhances efficacy in capturing details, and fully connected layers facilitate amalgamating high-level features for precise classification. In style transfer, VGG-19 serves as a feature extractor, leveraging intermediate layers to separate and recombine content and style. A pre-trained VGG-19 model, initially trained on ImageNet, is repurposed for style transfer, involving the optimization of content and style losses computed from VGG-19 feature representations. The goal is to minimize losses for generating aesthetically enhanced stylized images. In summary, VGG-19's pre-trained 19-layer CNN architecture excels in image classification and is instrumental in style transfer applications, offering robust feature extraction capabilities for manipulating and combining content and style in visual compositions.[11]

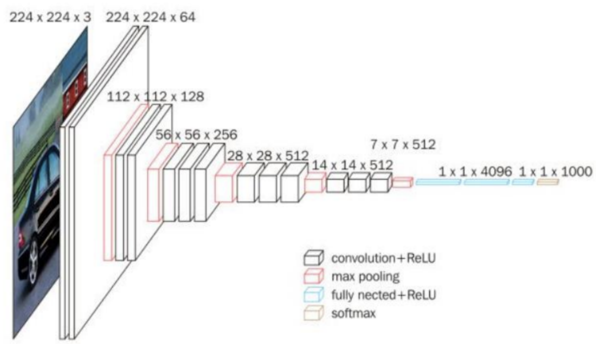


Figure 1: VGG-19 Network Model Architecture

The image stylization algorithm mainly includes three important parts: content reconstruction, style representation, and style transformation. In this proposed system, the output of the middle and high-level activation function of the VGG network is used to represent the content features of the image, mainly including its macrostructure and contour. Then, the Gram matrix is used to describe its style features. Image style transfer can be realized by minimizing the difference between the generated and input image's content features and style features.

4.2 ResNet 34 Model

ResNet-34, belonging to the ResNet family, is a robust convolutional neural network (CNN) architecture widely utilized for image classification and object detection. With 34 layers and pre-training on the ImageNet dataset, ResNet-34 employs residual blocks featuring shortcut connections to address challenges like vanishing or exploding gradients, enabling effective training of deep networks.

Key features of ResNet-34 include its use of residual blocks, integrating input and output to learn residual functions, and an architecture consisting of convolutional layers, batch normalization, ReLU activations, and residual connections, ensuring enhanced trainability despite increased depth. Skip connections in residual blocks facilitate gradient flow, mitigating the vanishing gradient problem.

Global average pooling is employed to reduce parameters and prevent overfitting. Originally designed for image classification, ResNet-34 demonstrates versatility in computer vision applications, including style transfer. In a project utilizing a pre-trained ResNet-34, features from different convolution blocks were extracted to compute losses for generating styled images.

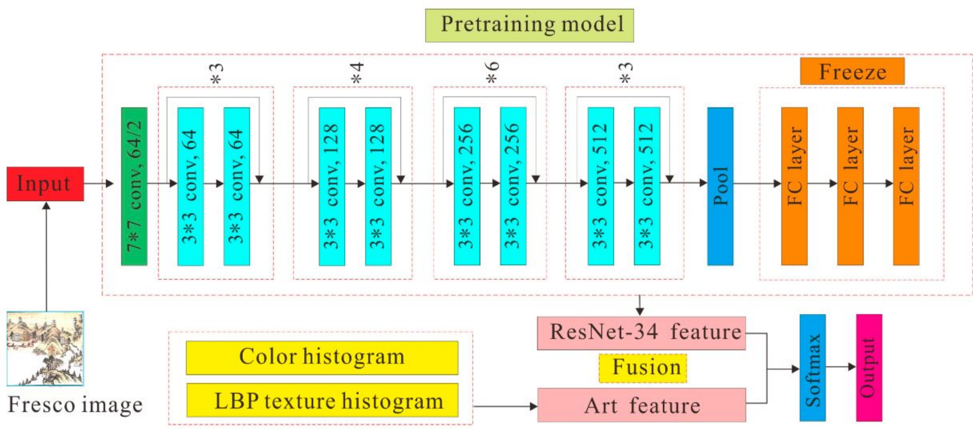


Figure 2: ResNet 34 Architecture[27]

Emphasis on deeper layers, particularly Convolution 2 on Layer 3 Block 5, preserves content features crucial for style transfer. Leveraging ResNet-34's pattern recognition capabilities, the project successfully transferred the reference image's style to content images, highlighting the model's efficacy in pattern recognition and style transfer applications.

4.3 VGG-19 and ResNet-34 Comparison

The choice between VGG-19 and ResNet-34 for style transfer involves weighing various factors associated with each architectural framework. VGG-19 is characterized by a simple design using 3x3 convolutional filters and max-pooling layers, making it easy to understand and implement. It excels in capturing hierarchical features, making it suitable for style transfer, and has widespread adoption with available pre-trained models. However, its computational cost, especially as a feature extractor, may limit real-time or resource-constrained applications, and the potential for vanishing gradients persists in deep networks.

On the other hand, ResNet-34 incorporates residual connections to address the vanishing gradient problem, allowing for the training of deep networks with improved efficiency. It demonstrates versatility beyond image classification and offers advantages in training efficiency. However, its increased complexity poses challenges in interpretation and implementation, and there is a risk of overfitting, especially with limited training data.

The decision between the two architectures depends on specific style transfer requirements, the availability of computational resources, and the preference for simplicity or complexity. The presence of pre-trained models is a crucial consideration to save on training time and resources. Computational constraints, particularly for real-time or resource-limited scenarios, should be assessed, with VGG-19 potentially demanding more computational resources. Additionally, task-specific evaluations are recommended, considering variations in image characteristics and artistic styles that may impact performance. In practical applications, both VGG-19 and ResNet-34 have utility, and the selection is guided by trade-offs and constraints relevant to the specific application.

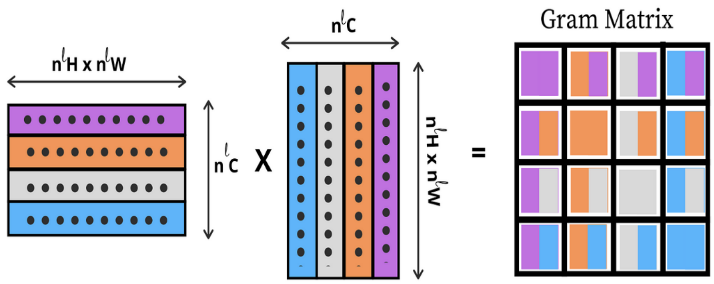


Figure 3: Gram Matrix Architecture

4.4 Gram Matrix

The Gram matrix is a pivotal component in facilitating the transfer of style for robust feature extraction after extracting features from content and style images. In the context of convolutional layers in a neural network, each layer produces a tensor characterized by dimensions corresponding to batch size, depth (d), and spatial dimensions (height, width - h, w).

To compute the Gram matrix for a given convolutional layer, the following steps are undertaken:

- Obtain the depth, height, and width of a tensor using batch size, depth (d), height (h), and width (w).
- Reshape the tensor to flatten its spatial dimensions.
- Calculate the Gram matrix by multiplying the reshaped tensor by its transpose.

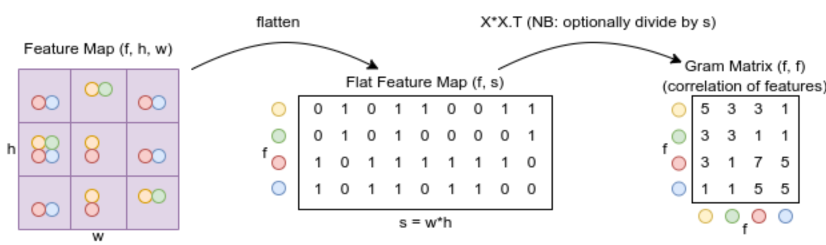


Figure 4: Gram Matrix Example

Having established the procedures for feature extraction and Gram matrix computation, features are extracted from images, and Gram matrices are computed for each layer in the style representation. The Gram matrix serves as a mathematical representation of an image's style, constructed from the feature maps of a specific neural network layer. This matrix is generated by computing the inner product (dot product) of vectorized feature maps, encoding correlations between different features and capturing statistical information about textures and patterns present in the style image.

The Gram matrix underscores the importance of capturing not only individual features but also their relationships, as the inner product operation reflects how different features within a layer interact. This nuanced understanding significantly contributes to the style transfer process.

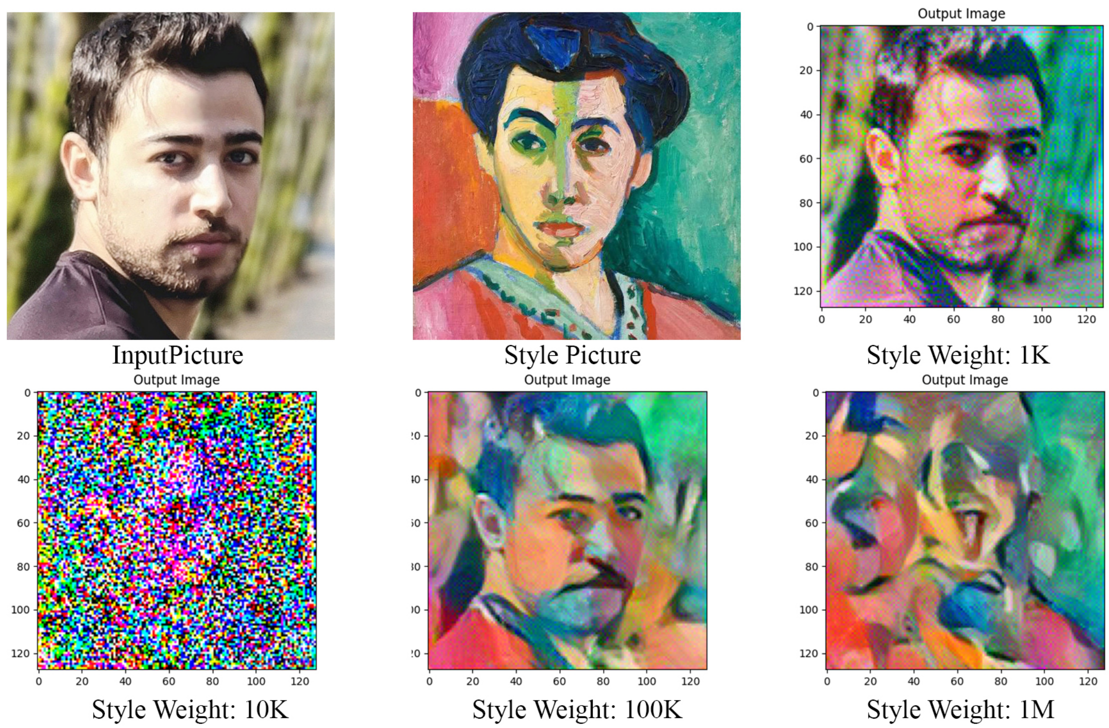


Figure 5: Effects of varying style weights on the input image.

As a foundational element for style transfer, the Gram matrix plays a crucial role in creating visually appealing outputs. The style transfer algorithm leverages the Gram matrix to combine the content of one image with the stylistic elements of another, achieving a harmonious blend of textures and patterns. This comprehensive approach results in images that seamlessly integrate content and style, producing aesthetically pleasing and engaging visual compositions.

4.5 Loss and Weights Calculations

Individual Layer Style Weights

Our proposed system presents a novel approach to style transfer by introducing weights on the style representation at each relevant layer, utilizing a 0-1 scale. By assigning emphasis to earlier layers (conv1-1, conv2-1), the resulting target image exhibits more pronounced style artifacts, prioritizing subtle features. This is due to the multi-scale style representation contributed by each layer, which varies in size.

Content and Style Weight

To achieve a desired balance in stylization, users can adjust content and style weights, influencing the level of stylization in the final image. Our approach maintains a fixed content weight of 1 while allowing users to modulate the style weight. This flexibility empowers users to attain a customized ratio aligned with their artistic vision.

Updating the Target and Calculating Losses

Image updates occur incrementally, resembling the training loop, where the target image undergoes changes while VGG19 and other images remain unchanged. Optimal results were observed after 1000 steps, but users experimenting with weight values or different images can use fewer steps. Within the iteration loop, content and style losses are computed, and the target image is updated accordingly.

Content Loss

Content loss is computed as the mean squared difference between target and content features at layer conv4_2, defined as follows:

$$Content\_loss = 1/2 \sum (T_i - A_i)^2$$

Where:
$\mathbf{T}_i$ is output image

$\mathbf{A}_i$ is the input image

Our goal is to minimize the loss and reduce difference. It will make the generated image more similar to the content image.

Style Loss

The style loss is calculated similarly, iterating through layers specified by name in the style weights dictionary. Gram matrices for the target image (target gram) and style image (style gram) at each layer are compared, yielding layer style loss.

$$Style\_loss = \alpha 1/2 \sum w_i (T_i - A_i)^2$$

Where:
α is the weight

$\mathbf{T}_i$ is output image

$\mathbf{A}_i$ is the input image

Total Loss

The total loss is determined by adding up style and content losses, weighted by the specified alpha and beta:

$$Total_Loss = \alpha(ContentWeight) * Content\_Loss + \beta(StyleWeight) * Style\_Loss$$

Adjusting content and style weights provides control over the influence of each component, offering a diverse range of stylized outputs.

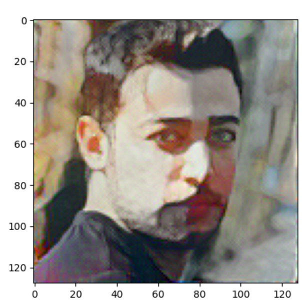
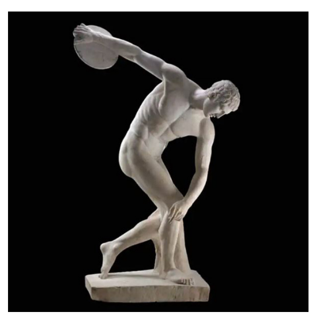
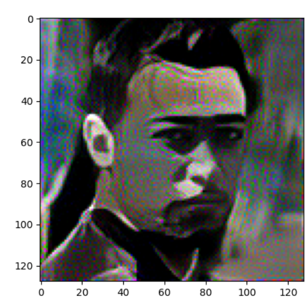
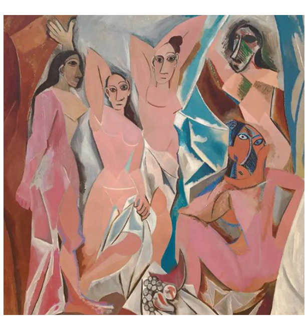
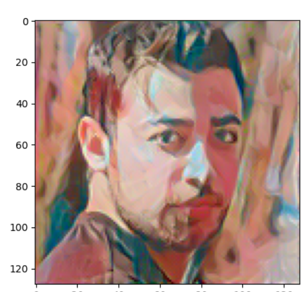
Understanding Weight Adjustments in Style Transfer

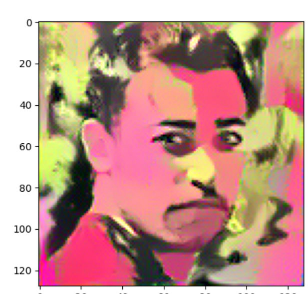
In style transfer, the weights assigned to content and style losses during optimization influence the preservation of content and style in the final image. Different weights for content and style losses allow users to balance the influence of content and style on the stylized result. Adjusting content weight emphasizes content preservation while adjusting style weight prioritizes transferring the artistic style.
Weight choice is subjective and depends on the desired outcome. Some styles may require a strong emphasis on style, while others prioritize faithful content reproduction. The choice of weights impacts the visual appearance, necessitating experimentation for the desired balance.

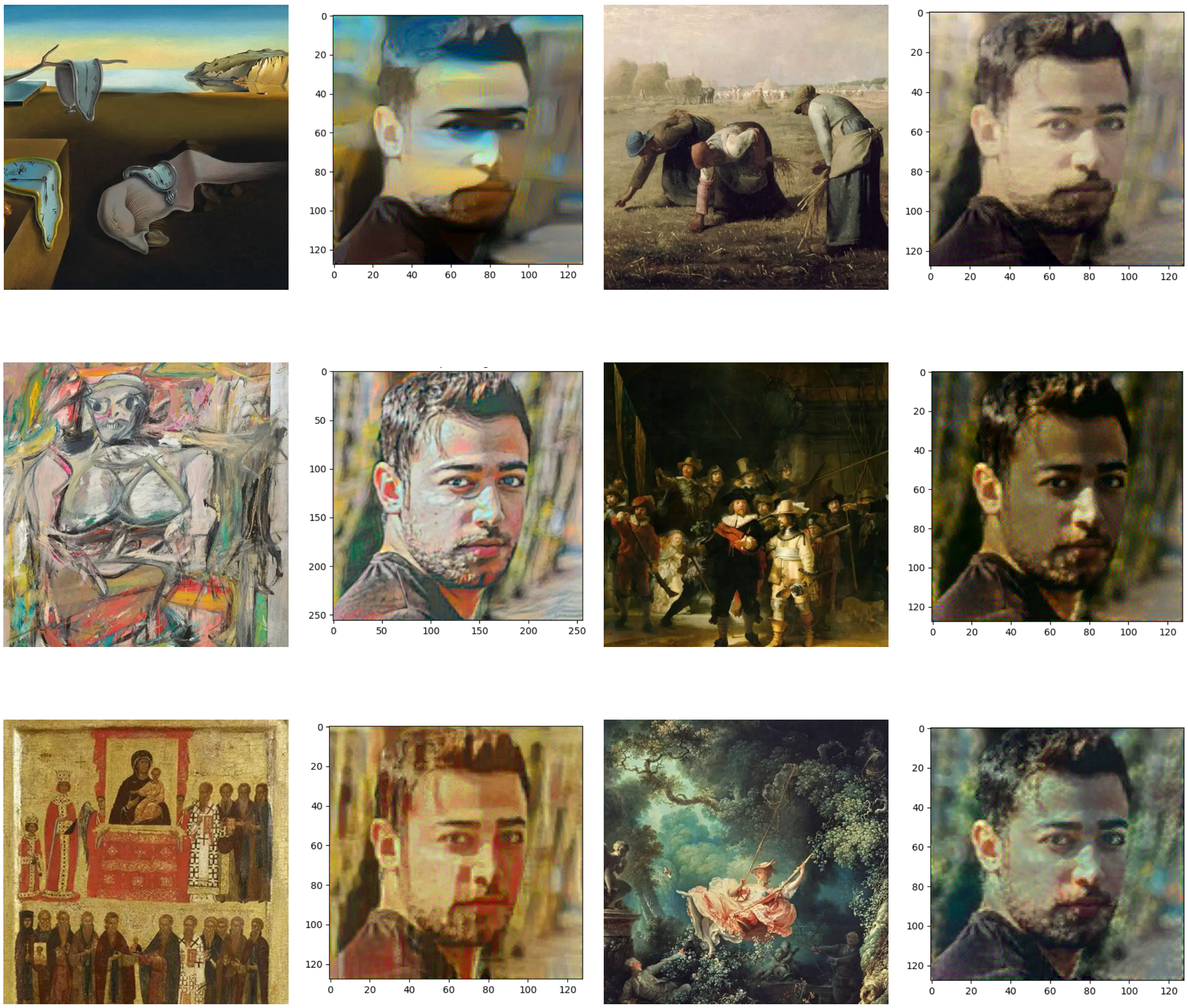
- Content Weight:
  - Higher content weight preserves more content from the content image.
  - Lower content weight allows flexibility in adjusting content to match the style.
- Style Weight:
  - Higher style weight emphasizes transferring style from the style image.
  - Lower style weight focuses more on preserving content.

5 Results

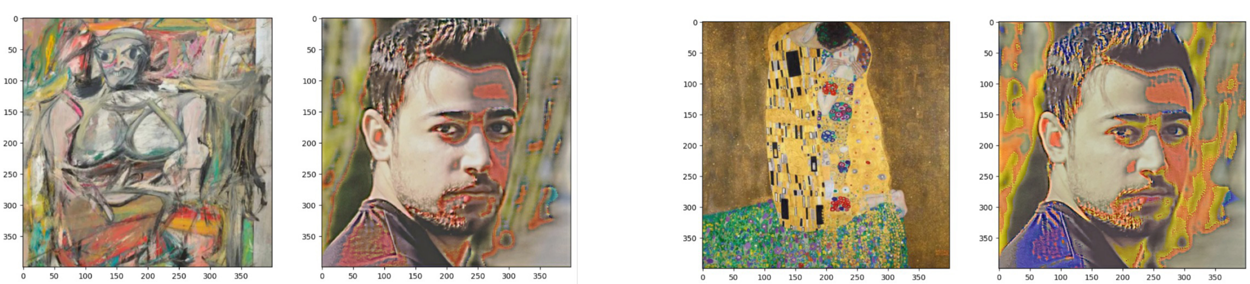
5.1 VGG-19 Results





5.2 ResNet-34 Results



6 Future Work

While we explored arbitrary style transfer using VGG-19 and ResNet-34, the ever-evolving landscape of deep learning and image processing opens up promising avenues for future research. One intriguing direction is the integration of Generative Adversarial Networks (GANs) into the style transfer framework. GANs have demonstrated remarkable capabilities in generating realistic and high-quality images by capturing intricate patterns and textures. Incorporating GANs into our model could potentially enhance the generation of visually compelling stylized images, pushing the boundaries of artistic expression.

Moreover, the adaptation of transformer-based architectures for style transfer presents an exciting prospect. Transformers, with their attention mechanisms and parallel processing capabilities, have shown great success in various computer vision tasks. Investigating how transformer models, such as the widely-used BERT or more recent vision transformers (ViTs), can be tailored to address the challenges of arbitrary style transfer would be a valuable exploration. Their ability to capture long-range dependencies and contextual information may contribute to more coherent and nuanced stylization.

Real-time video-style transfer stands as another compelling area for future work. Extending our current model to process video streams in real time introduces additional challenges, such as temporal coherence and computational efficiency. Techniques like temporal consistency regularization and adaptive stylization mechanisms can be explored to ensure a smooth and visually pleasing transition of styles across consecutive video frames.

Furthermore, the integration of user interaction and customization features could make style transfer more interactive and user-friendly. Allowing users to dynamically manipulate style parameters during the generation process could result in a more personalized and creative output, fostering a deeper connection between the user and the artistic synthesis process.

7 Discussion and Conclusion

We explored arbitrary style transfer using machine learning, particularly Convolutional Neural Networks (CNNs) such as VGG-19 and ResNet-34. The focus has been on achieving nuanced and flexible style transfer while minimizing distortion. The Gram matrix, loss calculations, and weight adjustments played crucial roles in the proposed model, offering a detailed and adaptable approach. The comparative analysis of VGG-19 and ResNet-34 highlighted their strengths and limitations, aiding practitioners in choosing suitable architectures. We tried to contribute to the field by addressing challenges in existing models, suggesting improvements, and paving the way for future research at the intersection of art and technology. The detailed experimental setup, results, and discussions can potentially enhance understanding and provide a foundation for continued advancements in arbitrary style transfer.

8 Team Members' Responsibilities

The success of the project depended on the collaborative efforts of each team member, with everyone playing crucial roles and shouldering distinct responsibilities. Throughout every stage of the project, close collaboration was emphasized, ensuring effective and efficient teamwork. Our collective objective as a team was to attain the project goals and successfully deliver a crop disease detection system.

8.1 Morteza Mogharrab

- Conducted literature review
- Analyzed available datasets for potential data training
- Implemented VGG-19 model and Gram Matrix
- Calculated Loss and Weights of the model

8.2 Md Solehin Islam

- Conducted literature review
- Explored and implemented ResNet-34 architecture
- Calculated Loss and Weights of the model

8.3 M Mahmud Hasan

- Conducted literature review
- Explored and implemented ResNet-34 architecture
- Calculated Loss and Weights of the model

9 References

[1] Eli Shechtman Kavita Bala Fujun Luan,Sylvain Paris. ”Deep Photo Style Transfer”. 2017.

[2] O'Shea, Keiron, Nash, Ryan. (2015). An Introduction to Convolutional Neural Networks.ArXiv e-prints.

[3] T. -Y. Chiu and D. Gurari, ”Line Search-Based Feature Transformation for Fast, Stable, and Tunable Content-Style Control in Photorealistic Style Transfer,” 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2023, pp. 249-258, doi: 10.1109/WACV56688.2023.00033.

[4] Sanghyuk Chun Byeongkyu Kang Jung-Woo Ha Jaejun Yoo,Youngjung Uh. ”Photorealistic Style Transfer via Wavelet Transforms”. 2019.

[5] U. Obu, Y. Ambekar, H. Dhote, S. Wadbudhe, S. Khandelwal and S. Dongre, ”Crop Disease Detection using Yolo V5 on Raspberry Pi,” 2023 3rd International Conference on Pervasive Computing and Social Networking (ICPCSN), Salem, India, 2023, pp. 528-533, doi: 10.1109/ICPCSN58827.2023.00092.

[6] J. Shlens V. Dumoulin and M. Kudlur. ”A learned representation for artistic style”. 2016.

[7] Wei Li Ming Li,Chunyang Ye." High-Resolution Network for Photo realistic Style Transfer". 2019..

[8] Matthias Bethge Leon A.Gatys,Alexander S.Ecker. "A Neural Algorithm of Artistic Style". 2015.

[9] Matthias Bethge Leon A.Gatys, Alexander S.Ecker. "Image Style Transfer Using Convolutional Neural Networks". 2016.

[10] J.Yang Z.Wang X.Lu Y.Li,C.Fang and M.-H.Yang. "Universal style transfer via feature transforms. InNIPS". 2017.

[11] Zhao, Zhe, Zhang, Shifang. (2023). Style Transfer Based on VGG Network. International Journal of Advanced Network, Monitoring and Controls. 7. 54-72. 10.2478/ijanmc-2022-0005.

[12] Li, Z., Zhou, F., Yang, L. et al. Accelerate neural style transfer with super-resolution. Multimed Tools Appl 79, 4347–4364 (2020). https://doi.org/10.1007/s11042-018-6929-x.

[13] Gatys LA, Ecker AS, Bethge M (2016) Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2414–2423

[14] K. Baek, Y. Choi, Y. Uh, J. Yoo, and H. Shim, "Rethinking the Truly Unsupervised Imageto- Image Translation," arXiv.org, 2020, doi: https://doi.org/10.48550/arXiv.2006.06500.

[15] H. Chen et al., "Artistic Style Transfer with Internal-external Learning and Contrastive Learning," Advances in Neural Information Processing Systems, vol. 34, pp. 26561–26573, Dec. 2021, Accessed: Feb. 25, 2023. [Online].

[16] L. A. Gatys, A. S. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," arXiv.org, 2015, doi: https://doi.org/10.48550/arXiv.1508.06576.

[17] X. Huang and S. Belongie, "Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization," arXiv.org, 2017, doi: https://doi.org/10.48550/arXiv.1703.06868.

[18] Dynamic Instance Normalization for Arbitrary Style Transfer, ResearchGate, 2020, doi: https://doi.org/10.1609//aaai.v34i04.5862.

[19] Y. Li, C. Fang, J. Yang, Z.Wang, X. Lu, and M.-H. Yang, Universal Style Transfer via Feature Transforms, arXiv.org, 2017, doi: https://doi.org/10.48550/arXiv.1705.08086.

[20] D. Y. Park and K. H. Lee, Arbitrary Style Transfer with Style-Attentional Networks, arXiv.org, 2018, doi: https://doi.org/10.48550/arXiv.1812.02342.

[21] An Introduction to Convolutional Neural Networks, ResearchGate, 2015. https://www.researchgate.net/publication/285164623

[22] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, Contrastive Learning for Unpaired Imageto-Image Translation, arXiv.org, 2020, doi: https://doi.org/10.48550/arXiv.2007.15651.

[23] Avatar-Net: Multi-scale Zero-Shot Style Transfer by Feature Decoration ResearchGate, 2018, doi: https://doi.org/10.1109//CVPR.2018.00860.

[24] Domain Enhanced Arbitrary Image Style Transfer via Contrastive Learning, Acm.org, 2020. https://dl.acm.org/doi/fullHtml/10.1145/3528233.3530736.

[25] S. Y. Falong Shen and G. Zeng. "Neural style transfer via meta networks". 2018.

[26] https://towardsdatascience.com/understanding-and-visualizing-resnets-442284831be8

[27] Gao, Le, Xin Zhang, Tian Yang, Baocang Wang, and Juntao Li. 2023. "The Application of ResNet-34 Model Integrating Transfer Learning in the Recognition and Classification of Overseas Chinese Frescoes" Electronics 12, no. 17: 3677. https://doi.org/10.3390/electronics12173677