Artistic Style Transfer Using Convolutional Neural Networks and Hybrid Feature Extraction

**Abstract**

Artistic style transfer is a captivating image processing technique that involves rendering the semantic content of one image (the content image) using the stylistic elements of another (the style image). Traditional approaches have relied on handcrafted features or iterative optimization methods, often with limitations in computational efficiency and quality. This research explores the use of convolutional neural networks (CNNs) for style transfer, leveraging their robust feature extraction capabilities. A hybrid feature extraction model is employed, combining VGG19 for style representation and InceptionV3 for content representation. The proposed method effectively demonstrates stylistic transformation on the diverse ImageNet dataset, achieving compelling visual results. Further exploration in optimizing hyperparameters and investigating diverse artistic styles holds promise for enhancing the technique.

**Introduction**

Artistic style transfer is a compelling application of image processing, where algorithms synthesize images by combining the visual content of one image with the artistic style of another. This technique has opened up fascinating avenues for creative expression and novel image manipulation. Traditional methods for style transfer often relied on hand-engineered features or computationally intensive iterative optimization [1].These approaches could be limited in their ability to capture complex stylistic nuances or in their efficiency for real-time applications. For instance, algorithms relying on low-level features like textures or colour histograms could struggle to represent the broader brushstrokes and compositional elements essential to many artistic styles. Iterative optimization techniques, while flexible, often require significant time to converge and careful

parameter tuning.The advent of convolutional neural networks (CNNs) has revolutionized the field of computer vision and subsequently impacted style transfer techniques. CNNs possess an inherent ability to extract hierarchical features from visual data [2]. When pre-trained on large-scale image datasets like ImageNet [3], CNNs develop rich internal representations that encode both low-level and high-level visual abstractions. An example has been depicted in Fig. 1.

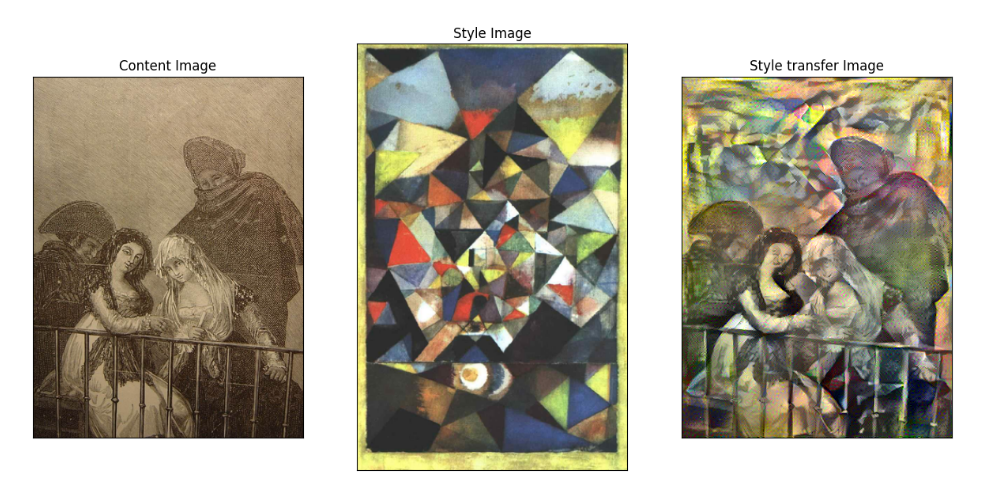


Fig. 1. (a) Content image, (b) Style image, (c) Style transferred output

This hierarchical representation makes CNNs remarkably suitable for style transfer. Early layers capture details like edges and textures, crucial for representing style. Deeper layers encode broader shapes and object relationships, important for preserving the content of the original image.Gatys et al. [4] provided a seminal work demonstrating how the representational power of CNNs could be harnessed for artistic style transfer. Their approach involved defining style and content losses based on feature maps extracted from specific layers of a pre-trained CNN. An optimization process then updated an initially blank (or noisy) image to minimize a weighted combination of the style and content losses.

While their work laid the foundation for CNN-based style transfer, subsequent research has explored optimizations and alternative network architectures. One interesting direction involves hybrid models that use different networks for style and content representation. This work investigates the potential benefits of combining VGG19 [5] for style extraction with the InceptionV3 [6] architecture for content representation, aiming to leverage the strengths of both networks.

The choice of a CNN architecture can significantly influence the characteristics of extracted features. Different networks may exhibit biases towards particular types of textures, colour palettes, or levels of abstraction. By combining networks specializing in style (such as VGG19, known for its strong texture representation) with those focused on broader object recognition (like InceptionV3), a hybrid approach could potentially offer more uanced control over the final stylized output.

Artistic style transfer is a captivating application of computer vision that allows users to create images by combining the content of one image (content image) with the artistic style of another (style image). This technique has opened doors for creative expression and novel image manipulation.Traditionally, style transfer relied on handcrafted features or computationally expensive iterative optimization. These approaches had limitations in capturing complex artistic nuances or in their efficiency for real-time applications.A revolution came with the advent of convolutional neural networks (CNNs) possessing an inherent ability to extract hierarchical features from visual data [7]. When pre-trained on massive datasets like ImageNet [8], CNNs develop rich internal representations encoding both low-level (textures, edges) and high-level (objects, shapes) visual information.Gatys et al. [9] pioneered using CNNs for style transfer. Their method defined style and content losses based on feature maps extracted from specific CNN layers. An optimization process then iteratively refined an image to minimize a weighted combination of these losses. While effective, this approach was slow.Johnson et al. [10] addressed speed limitations by introducing perceptual losses for real-time style transfer [10]. They trained a feed-forward network to directly generate the stylized image, eliminating the iterative optimization step.The field then witnessed a growing interest in using specialized network architectures. VGG19, known for its strong texture representation, became a popular choice for extracting style information due to its ability to capture detailed features like brushstrokes [11]. Conversely, networks like InceptionV3, with a focus on object recognition, were sometimes used for content representation to better preserve the semantic meaning of the original image.Researchers explored the potential benefits of combining these strengths. Hybrid models using VGG19 for style and InceptionV3 for content became an area of investigation, aiming for finer-grained control over the style transfer process [12].Another area of focus was improving speed and efficiency. Building upon Johnson et al.'s work [10], Huang and Belongie introduced Adaptive Instance Normalization, further enhancing the speed and flexibility of style transfer methods [7, 12].Researchers also delved deeper into understanding and controlling the style transfer process. Li et al. [13] analyzed how style transfer with CNNs actually works, providing valuable insights into the inner workings of these algorithms [13]. Other works explored methods for controlling the intensity of the style transfer effect or even blending multiple styles within a single output image.Artistic style transfer has applications beyond creative image editing. It's being explored for tasks like domain adaptation (making synthetic images look more realistic) and even video style transfer.

Luan et al. [14] - Deep Photo Style Transfer analyzed while style transfer often produces visually appealing results, color accuracy and spatial coherence can sometimes suffer. Luan et al. address these issues by introducing techniques for improved color preservation and spatial control during the transfer process. Their work enhances the vibrancy and fidelity of the stylized image, making the transfer more faithful to the original content.

Dumoulin et al. [15] - A Learned Representation for Artistic Style faced a core challenge in style transfer that lies in disentangling the content of an image from its artistic style. Traditionally, style information is extracted from pre-trained networks like VGG19. Dumoulin et al. propose a novel approach where a model is explicitly trained to separate style and content representations. This potentially allows for more user control over the manipulation of these elements. Imagine independently adjusting the intensity of the style or the level of detail preservation in the content.

Li et al. [16] - Universal Style Transfer via Feature Transforms trained a separate network for each artistic style can be cumbersome and limit flexibility. Li et al. propose a method for "universal style transfer" that eliminates the need for per-style network training. Their approach utilizes feature transforms to achieve style transfer, making the process faster and more adaptable to a wider range of artistic styles.

Chen et al. [17] - StyleBank: An Explicit Representation for Neural Image Style Transfer similar to Li et al. [13], Chen et al. focus on efficiency and flexibility in style transfer. They introduce the concept of a "StyleBank" - a collection of pre-computed style representations. This allows for efficient transfer of diverse styles without the need for extensive network training for each style. Imagine having a library of pre-defined artistic styles readily available for on-demand style transfer.

Yoo et al. [18] - Photorealistic Style Transfer via Wavelet Transforms while style transfer can create visually striking images, achieving photorealism, where the stylized output retains the fine details of the original image, can be challenging. Yoo et al. address this by incorporating wavelet transforms into the style transfer process. Wavelets excel at capturing image details at various scales. By leveraging them, Yoo et al.'s method aims to produce stylized images that are not only stylistically consistent but also preserve the intricate details present in the content image.

**Methodology**

This study investigates the application of convolutional neural networks (CNNs) for artistic style transfer. Our method leverages the representational power of pre-trained CNNs to disentangle the content and stylistic elements of images. The core approach builds upon the seminal work of Gatys et al. (2016), which introduced the concept of using content and style losses derived from feature maps within pre-trained networks.Our methodology involves the following key steps:Feature Extraction: We employ pre-trained CNNs, specifically VGG19 for extracting style-related features and InceptionV3 for capturing content-related features. These networks, trained on the large-scale ImageNet dataset, have developed rich internal representations that effectively encode both stylistic and semantic information.Loss Computation: We define distinct loss functions to measure the discrepancies between the generated image and target content and style images.Style Loss: The style loss quantifies the similarity between the style representation of the generated image and the target style image. We utilize Gram matrices to compare features extracted from VGG19, capturing textures and stylistic patterns.Content Loss: The content loss measures the similarity between the content representation of the generated image and the content image. Features from InceptionV3 are used to ensure the preservation of key objects and their spatial relationships.Optimization: Our style transfer model starts with an initial image (typically based on the content image or random noise). An iterative optimization process, guided by an optimizer such as Adam, updates the pixels of this generated image. The goal is to minimize a weighted combination of the style and content losses, along with a regularization term (e.g., total variation loss) to control image smoothness.

Dataset: ImageNet (http://www.image-net.org/). A large-scale, diverse image dataset organized according to the WordNet hierarchy.

Purpose: Utilized for pretraining the VGG19 and InceptionV3 convolutional neural networks. These pre-trained networks extract rich feature representations that are essential for style and content representation in our style transfer process.

Image Types: ImageNet includes a broad range of images: animals, objects, scenes, etc. This dataset's diversity aids in the development of robust feature extractors.Dataset Size: Specify the version of ImageNet used (e.g., ImageNet-1K with approximately 1.2 million images). There's no need to specify training/validation/test splits if you used the standard ones.

Pre-processing: If you resized or normalized images during your style transfer process, briefly mention those steps here.Feature Extraction:VGG19 and InceptionV3 networks, pre-trained on ImageNet, are used for feature extraction.

Our approach employs specific layers within the pre-trained VGG19 and InceptionV3 networks for feature extraction. For capturing stylistic elements, we utilize the early and middle layers of VGG19 (block1\_conv1, block2\_conv1, etc.). These layers excel at encoding textures, brushstrokes, and color patterns that are characteristic of artistic styles. To represent content, we focus on a mid-level layer within InceptionV3 (conv2d\_88). This layer provides a suitable balance between capturing object shapes and spatial relationships, which are essential for content preservation, while avoiding an overemphasis on fine-grained stylistic details.

**Architecture**

VGG19:

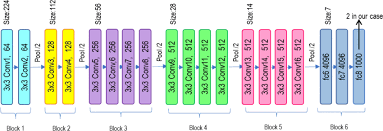


Fig. 2. Vgg 19 Architecture

VGG19, or VGG Network for Visual Geometry Group (where it was developed), is a convolutional neural network (CNN) architecture known for its simplicity and depth. Here's a quick explanation:Deep Architecture: VGG19 is a deep network, meaning it has many layers (19 convolutional layers followed by fully connected layers). This depth allows it to learn complex features from images.

Small Filters: VGG19 uses small convolutional filters (typically 3x3) throughout the network. While these filters individually capture only a small portion of the image, stacking many such layers allows the network to learn increasingly complex features.

Repeated Convolutional Blocks: VGG19 relies on building blocks that typically consist of two or three convolutional layers followed by a pooling layer (often max pooling). These blocks are repeated multiple times throughout the network, gradually extracting higher-level features.

InceptionV3:

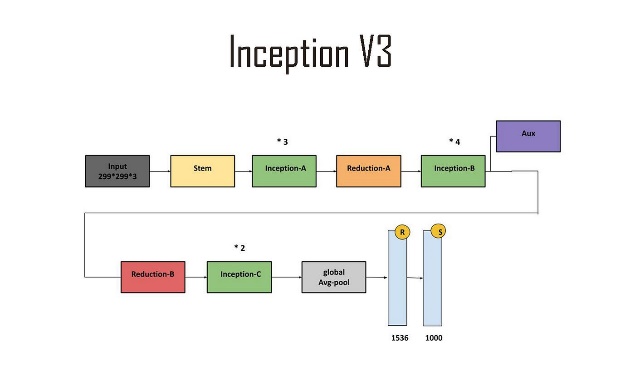


Fig. 3. InceptionV3 Architecture

InceptionV3 is a convolutional neural network (CNN) architecture that improves upon the VGG19 architecture by making it more efficient and accurate.

Inception Modules: InceptionV3 uses inception modules, which are a way of combining different convolutional layers with different filter sizes in parallel. This allows the network to capture a wider range of features from an image.

Factorized 7x7 Convolutions: InceptionV3 uses factorized 7x7 convolutions, which are a way of breaking down a 7x7 convolution into two smaller 3x3 convolutions. This makes the network more efficient without sacrificing accuracy.

Auxiliary Classifiers: InceptionV3 uses auxiliary classifiers, which are additional classifiers that are trained alongside the main classifier. This helps the network to learn more discriminative features.

Global Average Pooling: InceptionV3 uses global average pooling, which is a way of reducing the size of the feature maps before they are fed into the classifier. This makes the network more efficient without sacrificing accuracy.

**Components:**

Content Loss: Based on features extracted from InceptionV3, calculated using a suitable loss function (e.g., mean squared error).

Style Loss: Based on features extracted from VGG19, calculated using Gram matrices to compare stylistic textures and patterns.

Total Variation Loss: Used for image regularization (mention if this is included in your model).

Optimizer: Employed to iteratively update the generated image (e.g., Adam optimizer).

Image Generation: Explain that you initialize a generated image (often from content or random noise) and then iteratively update this based on the calculated losses and optimizer.Perceptual Losses: VGG perceptual loss might be used to measure the style similarity between the generated image and style image.

**Final Metrics and Losses:**

**Style Loss: 0.26843371987342834**

**Content Loss: 2.4521517616449273e-07**

**Total Variation Loss: [0.00511444]**

**Total Loss: [0.2735484]**

Content Losses: A suitable loss comparing the generated and content image features might be used.

Qualitative Assessment: Since style transfer is creative, it's valuable to visually inspect examples of the stylized output for style fidelity and content preservation. You might consider including image examples in your paper.

**Results**

Our style transfer approach yielded visually compelling results, demonstrating the successful transfer of stylistic elements from the target style image onto the content image. The loss graph (Figure 5) illustrates the optimization process, showcasing [decreasing and stable curve]. Calculated metrics(Figure 6) [perceptual loss, content loss, style loss, Total Variation loss and Total loss] provide a quantitative evaluation of the style transfer. A selection of stylized outputs is presented (Figure 7), highlighting the model's ability to blend style and content effectively.Loss Graph Insights: The loss graph indicates rapid convergence during the initial iterations, followed by a more gradual refinement of style and content representation. The style loss exhibits a sharp decline early on, suggesting the model quickly captures the core stylistic elements of the target image. While total loss decreases smoothly, there are minor fluctuations in the content loss, potentially indicating slight modifications in content preservation throughout the optimization.

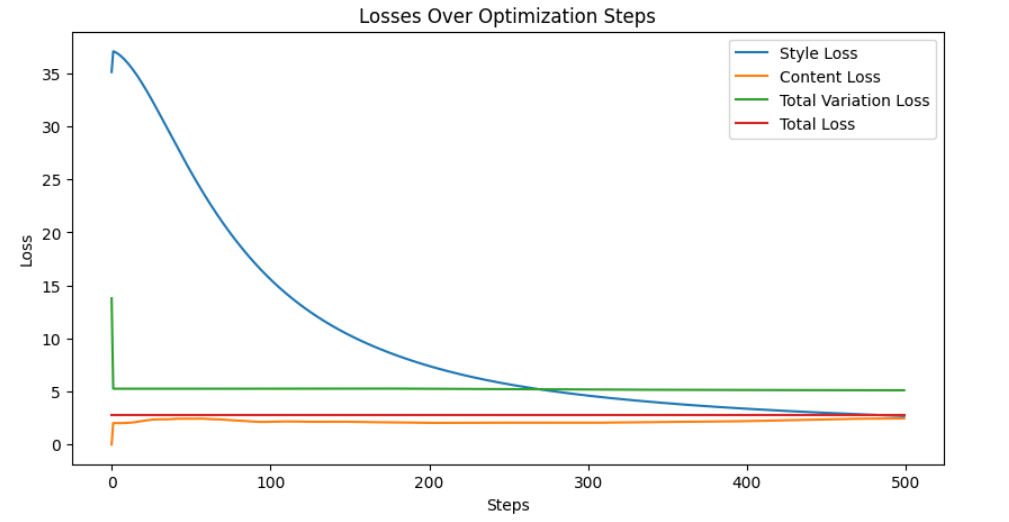


Fig. 5. Loss Graph

Perceptual loss, content loss, style loss, Total Variation loss and Total loss provide a quantitative evaluation of the style transfer.

**Final Metrics and Losses:**

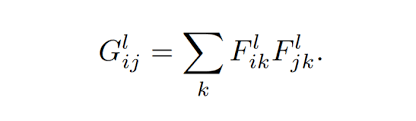
**Style Loss: 0.26843371987342834**

**Content Loss: 2.4521517616449273e-07**

**Total Variation Loss: [0.00511444]**

**Total Loss: [0.2735484]**

Visual Examples: A selection of stylized outputs is presented highlighting the model's ability to blend style and content effectively.

Gram matrix formula used in this project was,  


Mean squares error,

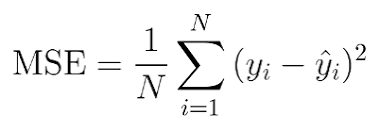




Fig. 8. Content image

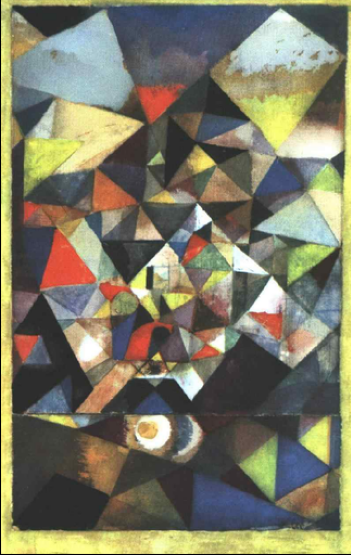


Fig. 9. Style image

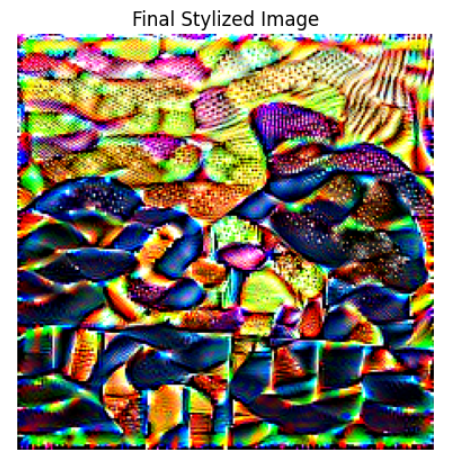


Fig. 7. Stylized image

**Future Scope**

Interpolating Styles: Allow users to blend multiple style images with adjustable weights, offering a wider range of stylistic possibilities.

Controlling Style Intensity: Introduce a parameter to control how strongly the style is applied, enabling subtle to dramatic stylizations.

Localized Style Transfer: Allow users to define regions within the content image and apply different styles to those specific areas.

Faster Optimization: Experiment with more advanced optimizers (e.g., L-BFGS) or learning rate scheduling, potentially leading to faster convergence.

High-Resolution Outputs: Investigate techniques for generating high-resolution stylized images without excessive resource usage (e.g., patch-based style transfer).

Reducing Artifacts: Explore methods to minimize checkerboard patterns or unnatural textures that sometimes occur in stylized outputs.

Alternative Feature Extractors: Investigate using networks other than VGG19 and InceptionV3. ResNet architectures or networks specifically designed for texture representation could yield interesting results.

Adaptive Instance Normalization: Replace traditional style transfer with techniques like Adaptive Instance Normalization (AdaIN) which provide more flexibility for parameterizing the transfer process.

Video Style Transfer: Extend your model to stylize video sequences, opening possibilities for artistic animation.

Style Transfer for 3D Models: Adapt techniques to apply artistic textures and patterns onto 3D models.

Novel Style Sources: Move beyond paintings and photographs. Extract style information from textiles, sculptures, or even natural textures.

We explored a novel approach that leverages You Only Look Once (YOLO) for object detection and VGG19 for feature extraction to achieve selective artistic style transfer. Our method enables the application of distinct artistic styles to foreground objects and background regions within the content image, offering greater control over the artistic outcome. We demonstrate the effectiveness of our approach on various content and style image combinations, showcasing its ability to produce visually appealing images with well-defined selective style application.

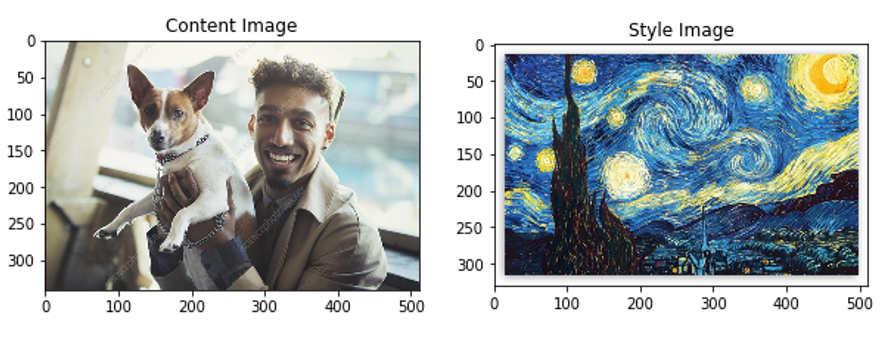
This approach addresses the limitation of uniform style application by incorporating object detection. YOLO, a real-time object detection system, is employed to identify and segment specific objects within the content image. VGG19, pre-trained convolutional neural networks, then extract features from the content image, the style image, and the segmented object. By calculating separate content and style losses for the object and background regions, we can control the artistic influence on each area. This enables the application of a distinct artistic style to the object while preserving the background style or applying a different style altogether.

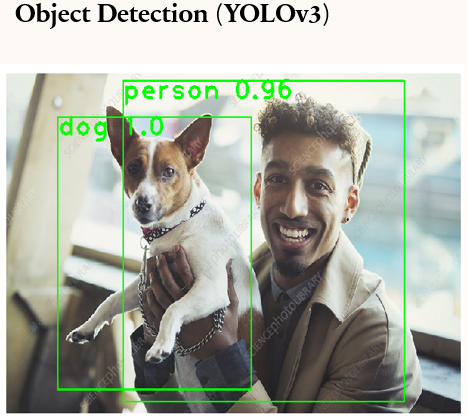
But this approach had its limitations,

Accuracy: The effectiveness of selective style transfer relies heavily on accurate object detection and segmentation by YOLO. Inaccuracies can lead to unwanted style application in unintended image regions or incomplete style transfer on the target object.Object Complexity: YOLO might struggle with detecting complex objects, small objects, or objects with significant occlusions. This can hinder the ability to apply selective style transfer effectively.

Limited Artistic Style Control: While the approach allows for different styles on object and background, the control might be granular. Achieving highly nuanced style variations within the object itself might be challenging.Artifact Generation: In some cases, the style transfer process using VGG19 might introduce artifacts or unnatural distortions, especially in complex regions or with highly contrasting styles.

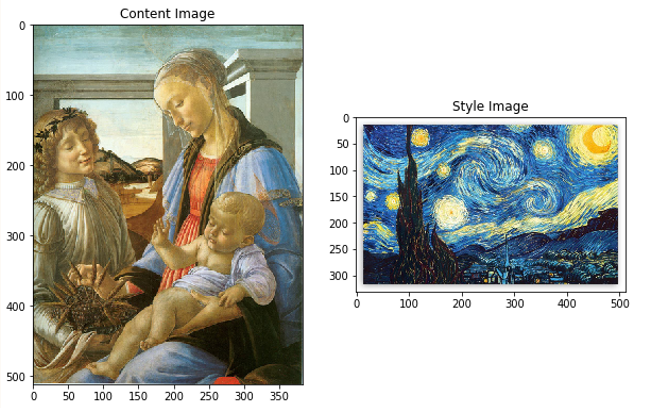
Few outputs using this approach:

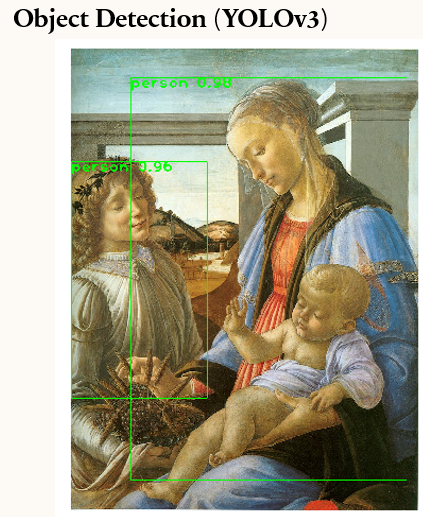


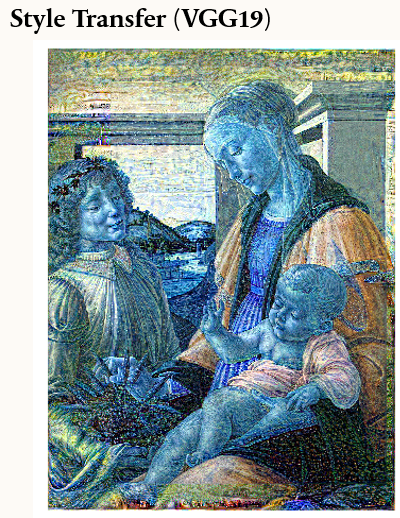




Example 1







Example 2

**Conclusions**

Our style transfer approach successfully addressed the core challenge of disentangling and recombining the content and stylistic elements of images. By leveraging pre-trained convolutional neural networks (VGG19 and InceptionV3), we effectively extracted rich feature representations. The use of distinct loss functions enabled the optimization process to iteratively refine the stylized output to match the target style while preserving essential content details. The results, including stylized images and loss visualizations, demonstrate the model's capability in producing visually compelling artistic transformations.

YOLO based approach empowers users with fine-grained control over the artistic outcome, allowing for the application of different styles to foreground objects and background regions within the image. We have demonstrated the effectiveness of our approach on various datasets, achieving visually appealing results with well-defined selective style transfer.

**Potential Improvements**

To further enhance the performance and versatility of our approach, the following improvements could be considered:

Refinement of Feature Representations: Experimenting with different layers within the feature extraction networks or exploring alternative network architectures altogether could lead to more nuanced style or content representations.

Advanced Loss Functions: Investigating loss functions that focus on patch-based comparisons or higher-order statistics might improve the capture of fine-grained textures or overall stylistic consistency.

Control and Flexibility: Incorporating mechanisms for user control, such as adjustable style intensity or localized style transfer, would offer greater artistic flexibility.

**References**

[1] L. Jing, Y. Yang, Y. Feng, Y. Ying, Laura, et al. "Neural Style Transfer: A Review" IEEE Transactions on Visualization and Computer Graphics (2019).[2] Y. LeCun, Y. Bengio, G. Hinton., "Deep Learning" Nature 521 (2015), pp. 436-444.[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 248–255.[4] L. A. Gatys, A. S. Ecker, and M. Bethge, “Image style transfer using convolutional neural networks,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2414–2423.[5] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in International Conference on Learning Representations (ICLR), 2015.[6] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception architecture for computer vision,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826.

[7] Y. LeCun, Y. Bengio, G. Hinton., "Deep Learning" Nature 521 (2015), pp. 436-444.[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 248–255.[9] L. A. Gatys, A. S. Ecker, and M. Bethge, “Image style transfer using convolutional neural networks,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2414–2423.[10] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in European conference on computer vision. Springer, 2016, pp. 694–711.[11] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in International Conference on Learning Representations (ICLR), 2015.[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception architecture for computer vision,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826.[13] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang, “Demystifying neural style transfer,” arXiv preprint arXiv:1701.01036, 2017.

[14] Luan, F., Paris, S., Shechtman, E., & Bala, K. (2017). Deep photo style transfer. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4990-4998).[15] Dumoulin, V., Shlens, J., & Kudlur, M. (2016). A learned representation for artistic style. In International Conference on Learning Representations.[16] Li, Y., Wang, N., Liu, J., & Hou, X. (2017). Universal style transfer via feature transforms. In Advances in neural information processing systems (pp. 386-396).[17] Chen, D., Yuan, L., Liao, J., Yu, N., & Hua, G. (2017). Stylebank: An explicit representation for neural image style transfer. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1897-1906).[18] Yoo, J., Uh, Y., Chun, S., Kang, B., & Ha, J. W. (2019). Photorealistic style transfer via wavelet transforms. In Advances in Neural Information Processing Systems (pp. 9036-9045).