Segment and Stylize: Applying Neural Style Transfer to Isolated Image Regions

Amy Zhong Sophie Poole

1. Introduction

Some of the most renowned artists, like Van Gogh and Monet, are famous for their distinct brushwork and textures. These can now be algorithmically reproduced onto other images. Neural style transfer (NST) is a technique in computer vision that takes the artistic styles of one image, which is called the "style image," and applies them to another, called the "content image" [5]. Most NST implementations apply style transfer across the entire image, which may obscure or distort important visual details.

To address this limitation, our project aims to bring greater control to neural style transfer by integrating it with image segmentation. The segmentation model enables users to select a specific object of interest in the content image. When combined with NST, this allows the style to be applied selectively — either to the object while preserving the original background, or vice versa. This approach enhances user control and offers greater flexibility compared to the original.

2. Related Works

2.1. Image Segmentation

Image segmentation is a fundamental task in computer vision that involves partitioning an image into meaningful segments, typically to isolate objects or regions of interest. Over the years, segmentation techniques have evolved from classical rule-based algorithms to powerful deep learning models capable of fine-grained, instance-level segmentation. This section reviews key developments in the field, from early heuristic methods to modern transformer-based and foundation models, with particular focus on approaches relevant to our use of the Segment Anything Model (SAM) in this project.

2.1.1 Traditional Image Segmentation

Some early techniques include thresholding, edge-based methods, and clustering techniques [1]. These methods, while foundational, often struggled with complex scenes and varying lighting conditions.

2.1.2 Deep Learning Approaches

Deep learning has revolutionized image segmentation. While traditional methods typically process each pixel or local region with limited context, deep networks capture global context, understanding an object's relationship with its surroundings (e.g., a dog in a park, not just a shape).

One popular segmentation model is Mask R-CNN [8]. This is a region-based convolution neural network approach with two stages. Objects are first detected through a Region Proposal Network (RPN); then in the second stage, the proposed object bounding boxes are refined through classification and bounding box regression. In the Mask R-CNN approach, the second stage also involves a mask prediction branch that runs in parallel.

2.1.3 Segment Anything Model

More recent approaches use not CNN architectures but transformers — Vision Transformers (ViTs) [3]. The advantage of these models is their multi-head self-attention mechanism, and trained on large amounts of data, ViTs have beaten CNN-based approaches in many computer vision tasks. Our focus in this project is specifically on Meta's Segment Anything model (SAM) [10]. SAM is designed to generalize across various tasks and domains without additional fine-tuning. This flexibility is due to its promptable architecture and its extensive training. The model architecture has three main components (figure 1):

1. *Image encoder:* SAM uses a pretrained Vision Transformer to generate an encoding for the input image.

2. *Prompt encoder:* There are four types of possible prompts: mask, points, box, and text. The prompt is encoded as a combination of positional encodings and learned embeddings that capture its semantic intent. This project primarily explores box prompts, which will guide SAM to segment within that specific region of the image.

3. *Mask decoder:* Combining information from the image and prompt encoders, the mask decoder predicts segmentation masks corresponding to the spec-

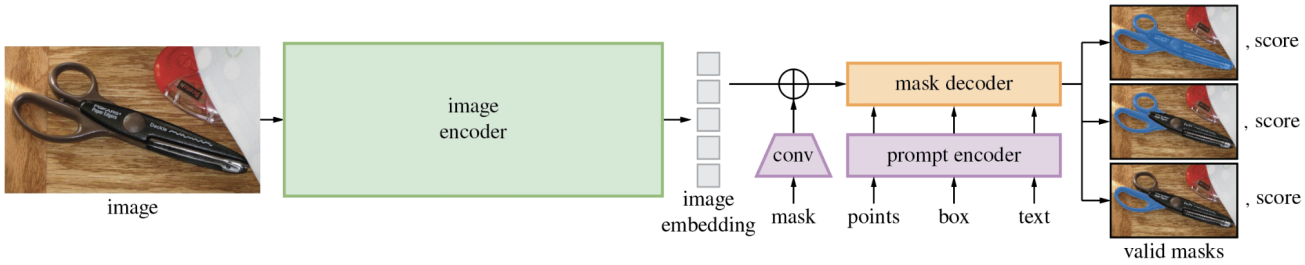

Figure 1. SAM architecture [10].

ified prompts. The decoder is a lightweight modified Transformer-based decoder.

2.2. Neural Style Transfer

Neural Style Transfer (NST) is an image stylization technique that uses deep learning to combine the content of one image with the artistic styles (i.e. colors, texture, brushstrokes, etc.) of another. This allows for the content image to look as if it were painted in the style of the reference art.

While the works of famous artists are often used as style images during NST to mimic the artists' signature aesthetic, they are also chosen because these paintings tend to have strong, distinct visual patterns. If the content and style images are too similar (e.g. a realistic style painting and a photograph), it usually results in less visibly effective results [9].

2.2.1 Traditional Neural Style Transfer

Neural style transfer was first conceptualized in the paper A Nerual Algorithm of Artistic Style [5]. Their approach was to train convolutional neural networks (CNNs) on object recognition. Each layer in the CNNs represent the image's general attributes, which allows the main objects from the content image to be preserved and the main artistic styles of the style image to be transferred, without looking at the images at the pixel level. They used 19 layers in their VGG-Network, referred to as the VGG19 model [12]. The higher layers in the network contain the high-level content, the objects and their arrangement in the content image. This is referred to as content representation. The lower layers reproduce the exact pixel values of the original image for content reconstruction.

NST uses an optimization process to minimize content loss and style loss. The content loss is the "distance in content between the content image and the target image"; the style loss is the "distance in style between the style image and the style of the input image" [4]. The content loss is the mean squared error (MSE) between the two feature vectors representing the generated image and the content image. The style loss is calculated from a Gram matrix of the generated and style images calculated for each layer.

Figure 2. Example of an image from the Cityscapes dataset [2].

2.2.2 Lightweight PyTorch NST

For our project, we used a lightweight PyTorch version of Gatsys et al. neural style transfer model created by Nazia Nafis [11]. This version aims to improve speed and reduce memory usage compared to the original NST.

3. Methodology

3.1. Dataset

We chose to use the Cityscapes dataset [2] throughout the project. This is a benchmark suite of high-resolution images of complex urban street scenes collected across 50 cities in Europe. The images have real-world complexity. For instance in image 2, there are occlusions, varying lighting, and multiple object classes. Importantly, Cityscapes contains 5000 images with pixel-level annotations — pixels are labeled with classes like " bridge," "traffic light," "car," and "person." The ground truth annotation is in the form of a .png, where different colors correspond to differences classes; for instance cars are blue (figure 3).

Because the dataset is so large-scale and we do not aim to measure the performance of our pipeline against its entirety, we created a sample of Cityscapes for easier use. We randomly chose a city (Zurich) and kept only the images that had colored fine-grain annotations. This sample con-

Figure 3. Example of pixel-level ground truth annotation from the Cityscapes dataset [2].

tained 122 images and the corresponding 122 ground truth annotations.

3.2. Image Segmentation

As mentioned in section 2.1.3, we use Segment Anything (SAM) to perform image segmentation. SAM is available in three Vision Transformer (ViT) variants — ViT-H (huge), ViT-L (large), and ViT-B (base) — and our current implementation supports the latter two but not ViT-H, because there is no dramatic difference in performance between ViT-H and ViT-L [10]. We strictly use the ViT-L variant in our sample runs. We leverage SAM's zero-shot segmentation capabilities and use the model off the shelf.

Specifically, we utilize two of SAM's capabilities: automatic mask generation and prompt-based segmentation using box prompts. This dual approach allows us to explore both general-purpose and targeted segmentation scenarios within our pipeline.

3.3. Metrics

For evaluation, we compute Intersection over Union (IoU) and Dice coefficient on the segmentation masks predicted by SAM. These metrics serve primarily as a sanity check to verify the quality of the segmentations. Our goal here is to ensure that the predicted masks used for stylization are reasonably accurate.

Because the ground truth masks are in color, an important step is to binarize them (figure 4). This requires human input to identify the object(s) of interest in the image (e.g. car, person). This list of strings is then mapped to the corresponding color(s) in the ground truth annotations; the mask is binarized by setting pixels of those colors/classes to 1 and all others to 0, enabling comparison with the SAM-predicted binary mask.

3.4. Neural Style Transfer

For neural style transfer, we used the VGG19 convolutional network model, which is commonly used in style transfer tasks due to its strong ability to extract deep visual

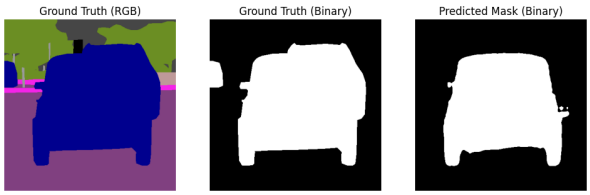

Figure 4. The original ground truth mask, the binarized ground truth, and the predicted mask for image 2 after box-prompted SAM prediction.

features. The model is based on the foundational method proposed by Gatsy's NST model in 2015 [11]. The model takes several key components as inputs: a style image, a content image, and two weighting parameters (content weight and style weight). From the original code, we reduced the style weight to preserve more of the geometric structure of the cityscape images (specifically the shapes of the cars, buildings, etc.).

Before either the content or style images are passed into the model, they undergo a preparation stage. Both images are resized to a common "target size", which is specified by the user, we indicated our target size to be 400 pixels in height. Having both images around the same resolution if important for the effectiveness of the neural style transfer and for the CNN's to compare visual features at a similar scale.

Once resized, the content and style images are processed through the NST pipeline. The content loss is calculated by the "L2 distance between the content image and the generated image," the style loss is the "sum of L2 distances between Gram matrices of the representations of the content image and the style image, extracted from different layers of VGG19," and the total loss is the sum of those losses "multiplied by their respective weights" [11]. The aim is for the total loss function to be minimized using an iterative optimization process. With each iteration, the generative image gradually evolves to better reflect the desired balance between the content structure and stylistic features as distanced by the content and style weights.

3.5. Combining Segmentation and NST

Once we obtain the stylized output from the neural style transfer model, we resize the image back to match the original scale of the input content image. This ensures a consistent, seamless integration with the background of the content image later on. Then the stylized image is applied only to the segmented object identified by SAM. This is so the style only appears on the object that the user has selected earlier. Using this predicted mask, we overlay the stylized object on top of the original content image resulting in a stylized main object among its realistic (original) background.

The key reason we chose to apply the NST model to the entire content image rather than only the masked region was to maintain the visual and structural continuity between the stylized object and the surrounding background, mainly between the main object and the edge features in the surrounding areas.

4. Results

Rather than focusing on large-scale quantitative benchmarking, our project centers on the integration of segmentation and style transfer into a unified pipeline. The primary outcome is the successful combination of SAM with a VGG19-based neural style transfer module, enabling flexible, mask-guided stylization of image regions.

While we do not have large scale quantitative results, we do have qualitative observations from multiple runs. We focus on image 2 as an example throughout this section.

Firstly, while SAM's automatic mask generation is effective from a computational standpoint — capturing distinct shapes, edges, and object boundaries — it often produces masks that are not intuitive or meaningful from a human perspective. As seen in figure 5, some masks over-segment the scene into fine-grained regions that don't correspond to recognizable objects.

This is significantly improved by prompting. Figure 4 shows that with a box prompt (the cropped region), the predicted mask more closely aligns with human perception. In this specific example, the predicted mask had high scores of $IoU = 0.834769932084553$ and $Dice = 0.9099450753873415$. In fact, we often observed high scores around or above the $0.8 - 0.9$ range for both IoU and Dice in the examples we ran; this reflects the effectiveness of SAM.

What's more, in some cases we may argue that SAM's predicted mask outperforms the ground truth. Once more referring to figure 4 as an example, you may notice the ground truth mask treats all visible cars — including those partially occluded or in the background — as part of a single binary class, resulting in a merged segmentation. In contrast the SAM-predicted mask more precisely isolates only the foreground car, demonstrating SAM's instance-awareness. In real world scenarios where occlusions are common, SAM performs reliably.

For the style image, we used a painting titled "Red, Blue, and Yellow geometric Abstraction" by Steve Johnson (this image was included as one of the sample style images provided by Nazia Nafis) [11]. Figure 6 shows the image after the style transfer. The color and similar artistic smudging technique that appears in the painting is transferred to the content image. The main objects in content image have also been preserved, as seen with the car being distinct from the background in figure 6 and the windows in the buildings.

As discussed in section 3, the NST is applied to the entire image for content continuity. This is displayed in the clear boundaries that surround the car that keeps it clearly defined from the other objects in the image (like the buildings and the road). Alternatively if the NST was applied to only the mask, it might have had more detail so the car, but had less of a boundary between itself and the background.

Figure 5. Image 2 after SAM's automatic mask generation.

Figure 6. Image 2 after neural style transfer.

Lastly, figure 7 shows the NST applied to the mask obtained by the segmentation model and then overlaid on the original background. While some of the main features still show on the car after the style transfer is implemented, like the rear-view window, the bottom half of the car is flooded with red.

However the amount of details preserved from the content image is not only dependent on the style and content weights, but also the artistic style of the style image. When the content image is combined with an artistic style with much thinner brushstrokes and smaller areas of color, like Van Gogh's "Starry Night", the details of the car show more prominently.

With using the painting "Red, Blue, and Yellow geometric Abstraction" specifically, we played around with the weights. In figure 6, we used 5000 for the content weight. However increasing and decreasing didn't change the outcome of the image greately.

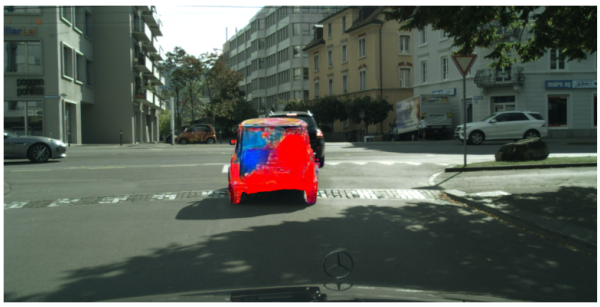

Figure 7. Image 2 after selective neural style transfer to the car.

5. Conclusion

In this work, we apply a segmentation model, Segment Anything, and neural style transfer to add stylization to a specific object or region as defined by the user. This hybrid approach allows the users to have more creative control while also maintaining the realism of the surrounding background. By blending stylized and non-stylized areas, our method differs from traditional style transfer methods which displays the style to the entire content image.

Some potential directions to further this project includes applying our segmentation and NST combination to videos. In this case, the segmentation would generate a mask for the object of interest in each individual frame, and then applying the style transfer to appear within those masks. This results in a video where the object or region of interest appears in a consistent way stylized, while the background remains realistic and unaltered. This application opens possibilities for more creative uses within video and film production. However, this extension comes with its own set of challenges, the main one being computational cost. Videos already require more memory since each frame is essentially an image.

Another potential direction is to experiment with different segmentation and NST models. An alternative segmentation model is Segment Anything V2. While Segment Anything V2 expands the capability of the original Segment Anything model. It has more user control such as object removal and blurring [7]. It also allows for video segmentation, which would be useful for combining segmentation and nst for videos.

The alternative for the NST model is to used Generative Adversarial Network (GAN) based neural style transfer. GANs are considered to be a more successful generative model, "especially in terms of their ability to generate realistic high-resolution images" [6].

Overall, this project aims to allow for more user control when using style transfer and opens the path for more creative use of neural style transfer.

References

[1] Hmrishav Bandyopadhyay. An introduction to image segmentation: Deep learning vs. traditional [+examples]. https://www.v7labs.com/blog/image-segmentation-guide, Aug. 2021. Accessed: 2025-05-08. 1

[2] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 3213–3223, Jun 2016. 2, 3

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020. 1

[4] Aldo Ferlatti. Nerual style transfer (nst) - theory and implementation. *Medium*, Nov 2021. 2

[5] Leon Gatys, Alexander Ecker, and Matthias Bethge. A neural algorithm of artistic style. *Journal of Vision*, 16(12), Sep 2016. 1, 2

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 2020. 5

[7] Mehul Gupta. Sam v2 by meta for video segmentation. *Medium*, 2024. 5

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 1

[9] Zhong Hong. Neural style transfer: Creating artistic images with deep learning. *Medium*, 2023. 2

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 1, 2, 3

[11] Nazia Nafis. A lightweight pytorch implementation of neural style transfer. *Medium*, Dec 2021. 2, 3, 4

[12] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *Computational and Biological Learning Society*, 2015. 2