ASTRO-G: Advancing Style Transfer with Robust Object Detection and Generative Models

Youssef Faragalla
Stanford University
Neurosciences IDP PhD Program
yfaragal@stanford.edu

Miguel Angel Fuentes Hernandez
Stanford University
MS in Computer Science
migufuen@stanford.edu

Abstract

Image style transfer in computer vision applies the artistic style of one image onto another, with applications in art, photography, and medical imaging. Despite progress, achieving high-fidelity style transfer remains challenging due to balancing content structure and artistic style. This project aims to solve this by creating a new framework called ASTRO-G (Advancing Style Transfer with Robust Object Detection and Generative Models). ASTRO-G integrates YOLO for object detection, SAM for segmentation, Stable Diffusion for style generation, and neural style transfer. YOLO detects objects, SAM provides detailed segmentation masks, and Stable Diffusion generates high-quality stylistic images. Neural style transfer then blends the content and style. Combining these advanced techniques enhances precision and quality in style transfer. The LPIPS metric evaluates the similarity between generated and ground truth style images. Experiments using the Egyptian Landmarks Dataset for content and artist datasets show the effectiveness of this approach, with the generated style images and the artist images shown to be perceptually similar (with the vast majority of perceptual distances less than 1), and qualitatively provide similar results for the final output images. The integrated method improves control over image segmentation and style application, promising for various applications.

1. Introduction

Image style transfer represents a captivating and innovative domain within computer vision, where the primary goal is to apply the artistic style of one image (referred to as the style image) onto the content of another image (known as the content image) [3]. This transformative process enables a wide array of applications, from enhancing the aesthetics of photographs to the creation of novel digital artworks, and extending even into the realm of medical imaging where it can assist in visualizing diagnostic images with enhanced clarity and detail [16]. The technique leverages deep learning models to separate and recombine content and style, producing images that are aesthetically pleasing and contextually accurate.

Despite the considerable strides made in this field, achieving high-fidelity style transfer remains a formidable challenge. The primary difficulty lies in preserving the structural integrity and semantic content of the content image while faithfully rendering the intricate artistic style of the style image. This balance is critical for maintaining the recognizability of the original content while introducing the stylistic elements in a coherent and visually appealing manner.

This project aims to push the boundaries of image style transfer by integrating advanced object detection, segmentation and generative models into our new framework ASTRO-G (Advancing Style Transfer with Robust Object Detection and Generative Models). ASTRO-G revolves around four core components: You Only Look Once Open World (YOLO) for object detection, Segment Anything Model (SAM) for object segmentation, Stable Diffusion for style generation, and neural style transfer. The input for our method will be a content image, a prompt for a given artistic style, and a prompt for an object in the image. We then use an object detection model (YOLO) and a segmentation model (SAM) to mask the desired image based on the user object prompt. Then, the image mask will undergo style transfer (with the generated style image from Stable Diffusion with the artistic style prompt) and get reapplied to the original content image. The result yields an image with style transfer applied to a specified object in a specified style. This project endeavors to push the boundaries of image style transfer by integrating advanced techniques in object detection, segmentation and generative modeling.

YOLO is a state-of-the-art object detection system that can detect and classify objects in images with high accuracy and speed. It operates by dividing the image into a grid and predicting bounding boxes and class probabilities for each

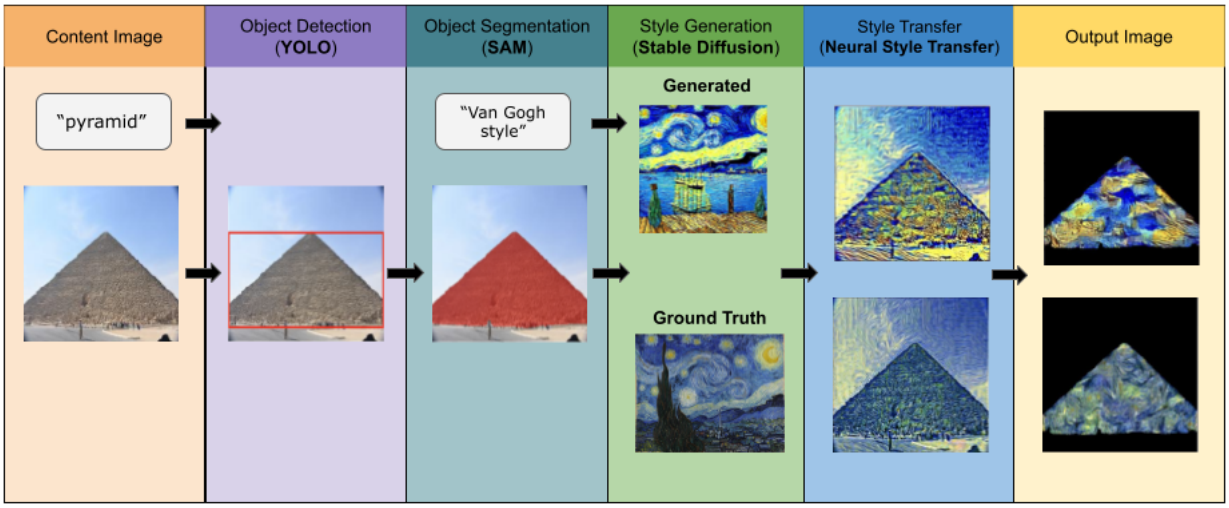

Figure 1: Layout and example of our method. Images are fed into YOLO for object detection with text prompt for the object to detect, then SAM for segmentation, and subsequently whichever image segment is selected is pushed either for 1) a Stable Diffusion-generated style image generated with a text prompt or 2) a ground-truth style image. The mask then undergoes a style transfer and is overlaid with the rest of the original component of the image. An example is shown here for an image of the pyramids of Giza (from our Egyptian dataset) and transformed with style of Van Gogh.

grid cell. This allows for real-time object detection, making it a valuable tool for segmenting images into meaningful components before style transfer.

The Segment Anything Model (SAM) is a highly versatile segmentation model designed to identify and delineate objects and regions within an image accurately. By integrating SAM, we can enhance the granularity and precision of the segmentation process, ensuring that the style transfer is applied selectively to specific regions of interest within the content image.

Stable Diffusion is a generative model that excels at producing high-quality images from textual descriptions or other forms of input data. By leveraging Stable Diffusion, we aim to generate images that maintain high fidelity to both the content and the desired style, resulting in visually stunning and contextually accurate outputs. This model uses a combination of iterative refinement and adversarial training to produce images that are not only realistic but also artistically coherent.

Neural style transfer is the core technique that enables the application of the artistic style from one image onto the content of another. This process typically involves training convolutional neural networks (CNNs) to optimize a loss function that balances content reconstruction and style reconstruction.

In summary, this project seeks to enhance the state-of-the-art in image style transfer through precise object detection, segmentation, and sophisticated generative modeling. The outcomes will not only advance the field but also open up new possibilities for creative and practical applications of style transfer technology.

2. Related Work

While previous efforts have focused on using SAM and neural style transfer together [5, 8, 17], we aim to create a new version with Stable Diffusion that will generate high-quality images, as well as YOLO for object detection to enhance segmentation. A project integrating advanced segmentation, object detection, and generative models like YOLO, SAM, and Stable Diffusion for image style transfer offers significant improvements over traditional methods due to several reasons. Semantic segmentation has been another method that could is useful for style transfer [9, 1, 7], but the masks are still not precise enough for style transfer, as is the case for object detection and style transfer [10, 11]. The use of YOLO and SAM would utilize a version of style transfer that has more precise control over image segmentation by using object detection as another step for style transfer, as has been shown in limited cases [8]. Using Stable Diffusion for high-quality image generation is useful for applications where a ground truth image search would be costly in time and user experience, and an image of a similar style but identical quality may be useful and has been approached before though with not with our current framework [21, 20, 15]. This will be the first time that all of these tools will be combined to our knowledge. The long-term benefits of combining these functions for an automatic pipeline for image and style transfer is ideal for multiple industrial applications.

3. Methods

Our approach involves a multi-step process to effectively transfer styles and evaluate the results. The methods include YOLO for object detection, SAM for segmentation, Stable Diffusion for image generation, and for evaluation. An overview of our approach is shown in Figure 1.

3.1. YOLO for Object Detection

The first step is to use YOLO to detect and locate objects within the content images. This step ensures that key objects are identified and preserved during the style transfer process.

YOLO Open World (You Only Look Once) is a real-time object detection system that applies a single neural network to the full image for a given number of classes. In our case, we used a single class, a single word to define our object. We use the latest YOLO-World that has open-vocabulary detection capabilities and zero-shot learning [2]. This neural network divides the image into regions and predicts bounding boxes and probabilities for each region using the embeddings for text and the visual features.

The YOLO World model includes a YOLO backbone, a text encoder, and a re-parameterizable vision-language path aggregation network (RepVL-PAN) that takes visual and textual input and classifies:

3.1.1 YOLO Backbone

The YOLO backbone is based on earlier version of YOLO, at which it takes an input image and delivers bounding box and image embeddings.

3.1.2 Text Encoder

The next part of the model is the text encoder, which is a pretrained CLIP model [14] that delivers text-embeddings. This takes the text from the prompt and transforms it into a text embeddings that can be taken into

$$\mathrm{W} = TextEncoder(T)$$

W has the dimensions Q, V where Q is the number of nouns and V is the embedding dimension.

3.1.3 RepVL-PAN

The RepVL-PAN is a network that takes the visual and text embeddings from the text encoder and YOLO backbone, and augments them using two methods.

Text-guided CSPL layer Using text embeddings $\mathbf{W}$, the image features $\mathbf{X}_l \in \mathbb{R}^{H\times W\times D}$ are augmented using a method called cross-stage features where augmantation occurs via:

$$\mathbf{X}'_l = \mathbf{X}_l \cdot \delta\left(\max_{j\in\{1..C\}}(\mathbf{X}_l\mathbf{W}_j^\top)\right)^\top, \tag{1}$$

where $\mathbf{X}'_l$ is the adjusted image features that are concatenated with the cross-stage features as output. The δ indicates the sigmoid function.

Image-Pooling Attention layer Then to augment text embeddings, we use an image-pooling attention layer

$$\mathbf{W}' = \mathbf{W} + \text{MultiHead-Attention}(\mathbf{W}, \tilde{\mathbf{X}}, \tilde{\mathbf{X}}) \tag{2}$$

where $\mathbf{W}'_l$ is the adjusted text embeddings that are updated

Then, the object embedding and bounding boxes are extracted from the augmented image features and text embeddings using a text-constrastive head as mentioned in this paper [2].

3.2. Segment Anything Model (SAM)

Next, SAM is used to obtain masks of the image patches in the content images. It is a state-of-the-art image segmentation model that leverages a combination of vision transformers and self-supervised learning. SAM can perform object segmentation without requiring a priori knowledge about the objects in the image. SAM consists of several parts:

3.2.1 Vision Transformer Backbone

The backbone network, a Vision Transformer (ViT) with a masked autoencoder [4], processes the input image and produces a feature map. The masked autoencoder is used to accelerate training and improving accuracy overall for image recognition tasks.

3.2.2 Encoder/Decoder

The prompt encoder takes in prompts of different input types (masks, points, boxes, and text) which are mapped differently. Text is embedded with CLIP, points and boxes with positional encoding, and masks with element-wise convolutions. The embeddings from the prompt encoder and the vision transformer are then integrated using a transformer decoder block as originally described on [18].

In conjunction, this segmentation will allow us to isolate the areas where the style transfer will be applied. SAM is chosen for its robust performance in generating accurate and detailed segmentation masks across diverse images.

3.3. Stable Diffusion

Once the image patches and their subsequent masks are isolated, a pre-trained Stable Diffusion model will be employed to generate images that are stylistically similar to the ground truth images for each artist that is from the dataset below. We used a model from Stability AI's Stable Diffusion XL model, which includes the most [12], which includes a base and refiner model primed to create the best images. The core idea is to reverse a diffusion process that gradually adds noise to an image until it becomes pure noise.

3.4. Style Transfer

Once the masks are generated, neural style transfer (NST) will be used to apply either the style of the ground truth style image or the generated style image to masks created by the earlier parts of the pipeline [3]. NST is a technique used to apply the style of one image (the style image) to another image (the content image) while preserving the original content. We implemented NST by using a pretrained VGG-19 neural network for object recognition. We then optimize the model over a number of epochs to improve the performance of the style transfer. The loss function for NST has two parts: content loss and style loss.

On one hand, content loss measures how much the content of the generated image deviates from the content image. It's computed as the mean squared error between the feature representations of the content image and the generated image at a certain layer l:

$$\mathcal{L}_{\text{content}}(C, G, l) = \frac{1}{2} \sum_{i,j} (F_{ij}^l - P_{ij}^l)^2$$

where F_{ij}^l and P_{ij}^l are the feature representations of the generated and content images at layer l.

The second component, style loss, measures how much the style of the generated image deviates from the style image. It is computed using the Gram matrix G, which captures the correlations between different filter responses at a certain layer l:

$$\mathcal{L}_{\text{style}}(S, G, l) = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - A_{ij}^l)^2$$

where G_{ij}^l and A_{ij}^l are the Gram matrices of the generated and style images at layer l.

The total loss is a weighted sum of these losses:

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{content}} + \beta \mathcal{L}_{\text{style}}$$

where α and β are hyperparameters used to control the importance of content and style when training the neural network.

3.5. Evaluation Metric

To compare both ground truth and generated style images, we used the Learned Perceptual Image Patch Similarity (LPIPS) metric [22]. LPIPS is a perceptual metric used to quantify the similarity between two images by leveraging deep neural network features. Unlike traditional metrics such as Mean Squared Error (MSE) or Peak Signal-to-Noise Ratio (PSNR), LPIPS is designed to align more closely with human perceptual judgments of image similarity. It does this by comparing the activations of a pre-trained deep neural network.

The LPIPS metric can be described mathematically through the following steps:

3.5.1 Feature Extraction

Given two images x and y, we first pass them through a pretrained deep neural network ϕ. The network extracts feature maps from multiple layers:

$$\phi^l(x) \quad \text{and} \quad \phi^l(y) \quad \text{for} \quad l = 1, \ldots, L$$

where ϕ^l denotes the activation from the l-th layer of the network.

3.5.2 Normalization

The feature maps are normalized before computing the distance. This normalization can be achieved by subtracting the mean and dividing by the standard deviation of the activations:

$$\hat{\phi}^l(x) = \frac{\phi^l(x) - \mu^l}{\sigma^l} \quad \text{and} \quad \hat{\phi}^l(y) = \frac{\phi^l(y) - \mu^l}{\sigma^l}$$

where μ^l and σ^l are the mean and standard deviation of the activations at layer l, respectively.

3.5.3 Distance Calculation

The distance between the normalized feature maps of the two images is computed. LPIPS uses a weighted L_2 distance:

$$d^l(x, y) = \sum_{i,j} w_{ij}^l \left(\hat{\phi}_{ij}^l(x) - \hat{\phi}_{ij}^l(y) \right)^2$$

where w_{ij}^l are learned weights that adjust the importance of different channels and spatial locations in the feature maps.

3.5.4 Perceptual Distance

The perceptual distance between the two images is obtained by summing the distances across all layers and taking the square root:

$$\text{LPIPS}(x, y) = \sqrt{\sum_{l=1}^{L} d^l(x, y)}$$

The weights w_{ij}^l are typically learned to further fine-tune the metric based on human perceptual data, thus enhancing its accuracy in assessing perceptual similarity.

3.6. Image Selection for Style Transfer Comparison

In order to benchmark generated style images against the ground truth style images, we computed a pairwise LPIPS metric between a generated style image and a ground truth style image. From these scores, we picked the five best pairs as a minimization function of the LPIPS metric, some of which included repeated images from either of the style image categories.

4. Dataset

4.1. Content Images

We will use the Egyptian Landmarks Dataset from Kaggle which is a subset of the Google Landmarks Dataset V2 [19, 13], which contains images of notable Egyptian landmarks. These images will serve as the base content for style transfer.

4.2. Style Images

The ground truth style images is a dataset from Kaggle that consists of artworks representing various styles, including but not limited to surrealism, pop art, and impressionism [6]. This annotated dataset contains over 16,800 images from 50 artists, where the annotations include nationality, genre, and number of paintings among others. These artworks will serve as the ground truth for style transfer. Each of the images were rescaled to the content image dimensions.

4.3. Generated Images

Stable Diffusion generates images that incorporate the target styles into the content images, using prompts including the artists' name and style (for example, one such prompt for the artist Vincent Van Gogh would be "Vincent Van Gogh style"). These are compared with the ground truth style images using the LPIPS metric and are optimized to pick the best pair of generated and grouth truth images. Each of the images were rescaled to the content image dimensions.

4.4. Prompts

Text prompts that were used for object detection consisted of single-word descriptions of the objects in the image, usually guided by the annotations in the original dataset. Text prompts for generated images also consisted of the the name of artist with the word style next to it (ex: "Camille Pissaro style").

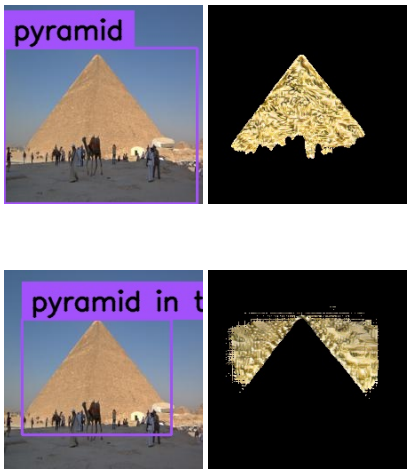

Figure 2: Example of differences in segmentation masks when using different text prompt inputs to YOLO. Top row shows the bounding box in purple with the input text "pyramid" on the left and the subsequently generated mask with style transfer on the right. The bottom row is the same but with the input text being "pyramid in the center".

5. Experiments/Results/Discussion

Our experiments were run on a pipeline where object detection, segmentation, image generation and style transfer took place as shown in Figure 1.

5.1. Segmentation

To segment the content image, we selected one class `pyramid` to be detected by YOLO-World. Once the object was detected, we used the bounding to box as input for SAM to correctly identify the object of interest. This model, SAM, then output a mask matrix of the same dimensions of the image (without considering the RGB channels), with the identified object region. Overall, the most determining factor for accurate object segmentation was the bounding box generated by the detection model. When the bounding boxes are tight, SAM sometimes outputs segmentation masks that don't correspond to the object of interest. For this, we found that general rather than detailed descriptions worked better for this application. In this case, we

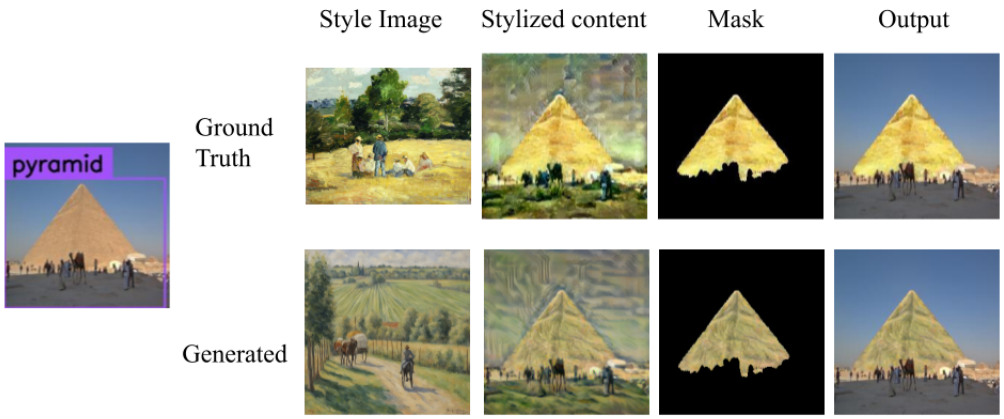

Figure 3: Example of a pair of style images (ground truth and generated in the first two rows), where the content image is transferred to each style (in this case the style of Camille Pissaro), the mask is then applied and then reapplied to the content image. This particular pair is the pair with the best LPIPS metric, the lowest perceptual distance.

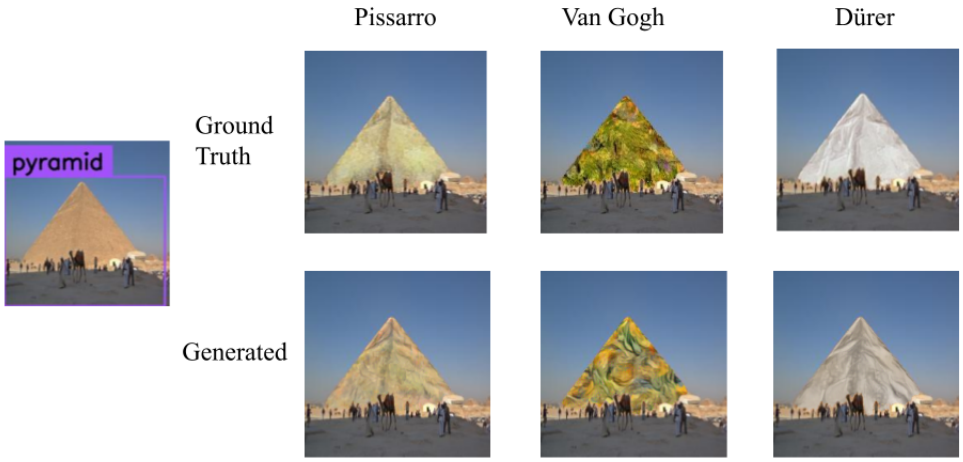

Figure 4: Examples of the best pairs of images for style transfer of the generated and ground truth style images, generated for each artist in each column (Pissarro, Van Gogh, and Dürer). The original content image is presented on the left side for comparison.

compared the classes `pyramid` and `pyramid in the center` as shown in Figure 2.

5.2. Style Generation

Once the object region was identified, we generated an image using Stable Diffusion and scaled it to match the content image's dimensions. To evaluate the stable diffusion model, we generated 25 images using the prompt {`artist name`} `style` for 3 different artists: `Vincent van Gogh`, `Camille Pissarro and Albrecht Dürer`. We then ran the perceptual distance between each of the generated images against their respective artist repertoire, ranking the pairs of images from lowest to largest distance. Figure 5 shows the distribution

of distances per artist.

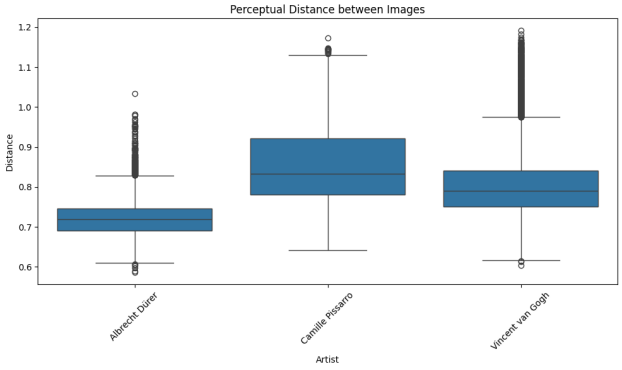

Figure 5: LPIPS distance distribution boxplots for each artist. The model achieves lower distances for less diverse artists.

We selected these artists because of their style differences. Most of Pissarro's work focuses on landscapes, Van Gogh's varies from objects, landscapes and people, and Dürer's are mostly black and white illustrations. As we hypothesized, Dürer was the easiest to generate among the three, mainly because of the limited color palette used on his paintings. By contrast, both Van Gogh and Pissarro showed a larger spread and overall larger distance from the generated styles. In all cases, the distance achieved for the 75 percentile was less than 1, with the best values being just over 0.6.

5.3. Style Transfer

As shown in Figures 4 and 3, we found that the generated and ground truth masked images are qualitatively similar, and can incorporate a range of different styles from different artists. We have used the same $\alpha = 1$ and $\beta = 1000000$ (learned parameters for total style transfer loss) for all three images to maintain consistency of the new method, but we realize this might not be optimal for each style type. For instance, the Van Gogh style in Figure 4 may exhibit high content loss even though style loss is minimized. This is something we would like to optimize for in future experiments. These hyperparameters were chosen by doing manual evaluations of stylized images as we wanted to maintain the content with a significant alteration in the style of the image.

5.4. Discussion

The utility of this method on the artists we have used so far has been promising. We have been able to generate an automated pipeline that can generate style transfer for content images based on explicit object detection. While this is ideal for applications that require explicit object detection (such as interior design, architecture, privacy software, and animation among others), an unsupervised masking method for object detection may be more useful and computationally efficient and this method would be our next natural step to investigate. Future investigations also would likely include understanding how style transfer adapts to objects other than landmarks. Another point to investigate is whether the identification of the object is altered after style transfer as a result of the learned parameters for style transfer loss (particularly content loss).

6. Conclusion/Future Work

In conclusion, we have generated a new method for state-of-the-art in image style transfer through precise object detection, segmentation, sophisticated generative modeling, and rigorous quantitative evaluation in the form of ASTRO-G. We have shown that this method works well compared to ground truth style images and can be used in a number of domains. Future work will look at how to generalize across objects other than landmarks, and how to extend it to multi-object detection and segmentation.

7. Contributions and Acknowledgements

Y.F. and M.A.F.H. contributed equally to this work. They conceived the project, wrote the paper, developed the code, and compiled the illustrations together. The authors gratefully acknowledge their collaboration and shared efforts in the successful completion of this project. Y.F. and M.A.F.H. used GPUs from the Baccus lab computing cluster in the department of neurobiology at Stanford University to run experiments and implement their code (At the time of publication, Y.F. works in this lab, uses this cluster, and received approval from the Baccus lab prior to the start of this project). Code has been adapted from multiple sources, and includes the following links.

YOLO
SAM
Stable Diffusion
LPIPS

References

[1] A. J. Champandard. Semantic style transfer and turning two-bit doodles into fine artworks, 2016.

[2] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan. Yolo-world: Real-time open-vocabulary object detection, 2024.

[3] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style, 2015.

[4] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners, 2021.

[5] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization, 2017.

[6] Icaro. Best artworks of all time, 2023. Accessed: 2024-06-05.

[7] M. Kim and H. Byun. Learning texture invariant representation for domain adaptation of semantic segmentation, 2020.

[8] H. Kulkarni, O. Khare, N. Barve, and S. Mane. Improved object-based style transfer with single deep network, 2024.

[9] L. Kurzman, D. Vazquez, and I. Laradji. Class-based styling: Real-time localized style transfer with semantic segmentation, 2019.

[10] H. Li, W. Wang, C. Wang, Z. Luo, X. Liu, K. Li, and X. Cao. Phrase grounding-based style transfer for single-domain generalized object detection, 2024.

[11] T. Li, J. Chao, and D. An. Style adaptation for domain-adaptive semantic segmentation, 2024.

[12] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations, 2022.

[13] A. Mostafa. Egypt landmarks, 2023. Accessed: 2024-06-05.

[14] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

[15] W. Tang, D. Figueroa, D. Liu, K. Johnsson, and A. Sopasakis. Enhancing fingerprint image synthesis with gans, diffusion models, and style transfer techniques, 2024.

[16] D. Tomar, B. Bozorgtabar, M. Lortkipanidze, G. Vray, M. S. Rad, and J.-P. Thiran. Self-supervised generative style transfer for one-shot medical image segmentation, 2021.

[17] B. Varadarajan, B. Soran, F. Iandola, X. Xiang, Y. Xiong, L. Wu, C. Zhu, R. Krishnamoorthi, and V. Chandra. Squeezesam: User friendly mobile interactive segmentation, 2024.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.

[19] T. Weyand, A. Araujo, B. Cao, and J. Sim. Google landmarks dataset v2 – a large-scale benchmark for instance-level recognition and retrieval, 2020.

[20] Z. Xu, E. Sangineto, and N. Sebe. Stylerdalle: Language-guided style transfer using a vector-quantized tokenizer of a large-scale generative model, 2023.

[21] H. Zhang and K. Dana. Multi-style generative network for real-time transfer, 2017.

[22] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018.