Image Style Transfer with (Fast) Neural Style Transfer Model

Team Big Cow King

Feiran (Philip) Huang, Jingwen (Steven) Shi, Hongsheng Zhong

Machine Vision Problem

The aim of this project is to implement a Neural Style Transfer Model (NST) for transferring the art style of a chosen image to another image. This can make people in the future still be able to see the style of old artworks even if the original masterpiece has been lost or broken.

Dataset

COCO 2017 Dataset:

A comprehensive collection of real-life photos from the COCO (Common Objects in Context) dataset, containing various scenes, objects, and contexts captured in everyday life.

Best Art of All Time:

Artworks encompassing various styles, genres, and historical periods, curated as a representative selection of the finest art across different cultures and movements.

Both datasets' images serve as content images and style images.

| Example Input Content Image	Example Input Style Image	Example Output Image

Implementation Decisions, Improvements, and Innovation Approach:

- Detaching Target
  - Content and style images are passed into the model during the generalization of the output image and have been detached from the code. This is because the content and style image should not be updated since we only want to update the generated image to minimize the loss. By calling the detach function, the content and style feather maps will be removed from the computation graph and remain unchanged for the rest of the computation.

- Improve Computation Effenficy
  - The original paper has chosen to insert the style/content layers after a set of convolution layers. However, we found that the image generated by inserting the loss layer after the ReLU activation function can accelerate the image-generating process much faster.
  - Since inserting after ReLU layers will converge much faster, it produces a better quality and visually more appealing image than convolutional layers under the same number of steps taken.

Conv (60 steps with 3s)	ReLU (60 steps with 3s)
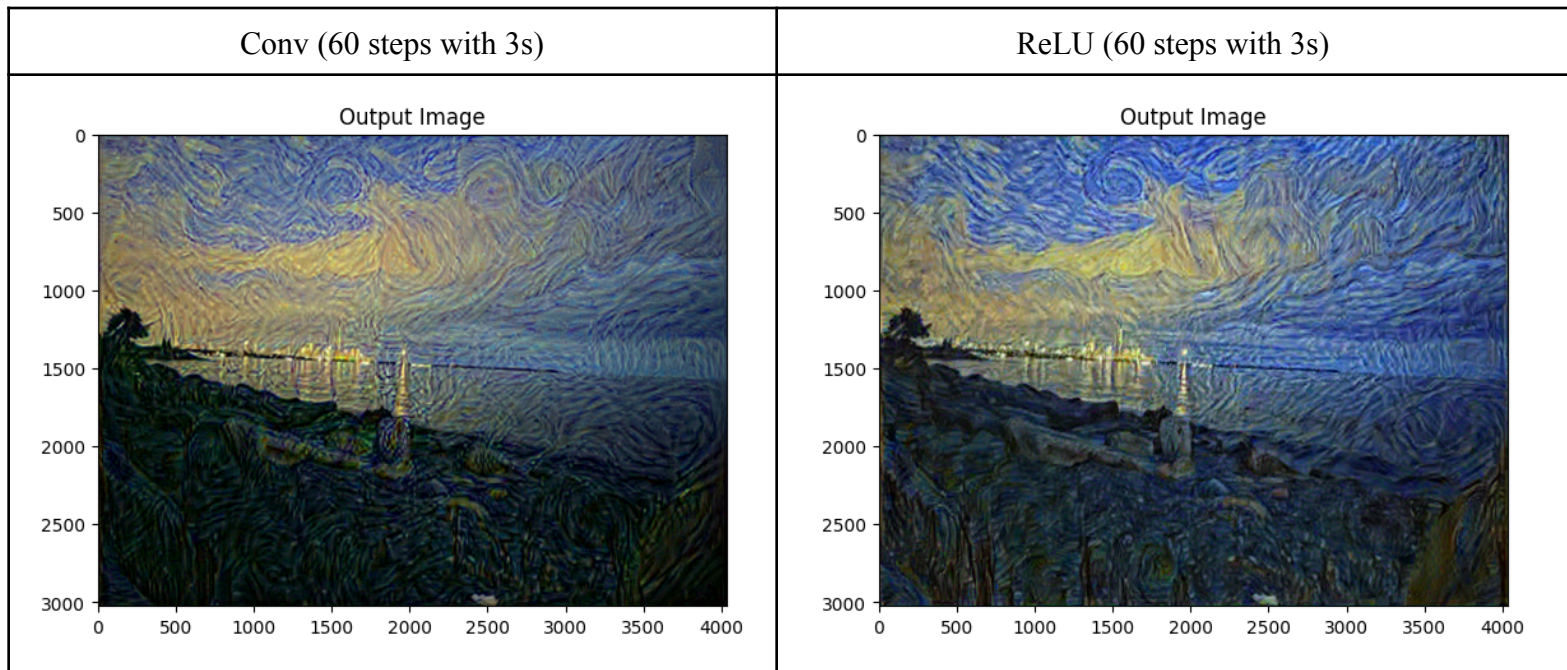
- Improve Image Quality
  - The lower the layer, the better it can capture the details and patterns of an image. The higher the layer, the better it can capture the overall abstraction of an image.
  - In neural style migration, focusing on basic image features such as edges and texture using layers 1-5 of VGG19 may overemphasize texture. Both simple and complex patterns can be obtained from the network by choosing a mixture of layers such as layers 1, 3, 9, 13, and 28. Ensure that the image retains the overall look and structure of the content image while capturing complex details from the style image.
  - Why Insert Style Loss after ReLU 1, 3, 5, 9, 28?
    - Since the input is a copy of the content image, inserting after lower layers is more powerful in matching details and can quickly merge the style patterns to the input image. Therefore, most of the layers used are lower layers.
  - Why Insert Content Loss after ReLU 10?
    - A middle content layer can effectively preserve the input's content. If a lower content layer is used, the output will closely resemble pixel-level copies of the content. However, given that the input is already a replication of the content image, there's no necessity to employ a lower content layer since it might overshadow the style.
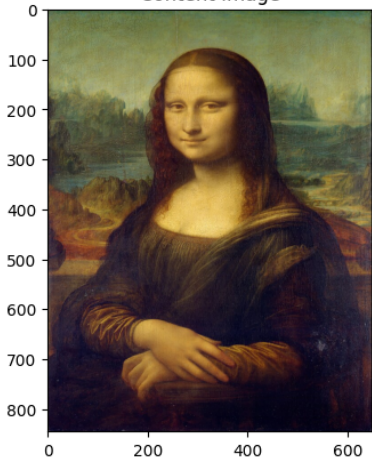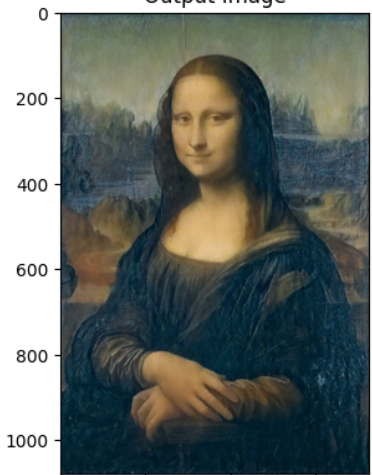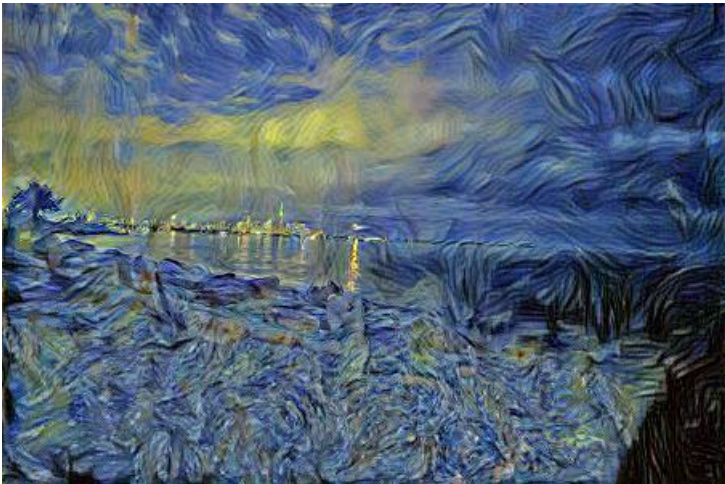
With VGG 19 Layer {1 - 5}	With VGG 19 Layer {1, 3, 9, 13, 28}

- Using Gram Matrix over Covariance Matrix
  - In the covariance matrix, the mean of each feature is subtracted from the feature values which center the data with respect to 0. However, in image processing, these features may be related to pixel intensities (i.e. RGB values) and centers data around the mean will capture the intensity of one feature related to the presence of another.
  - The Gram matrix does not subtract from the mean which only focuses on the overall frequency of feature co-occurrences. Elements like brushstrokes and colour palettes are parts of the style that should not depend on the positioning of objects. Therefore, the ability that the Gram matrix can abstract away from these spatial relationships and only focus on the overall presence of the feature maps, is exactly what we need.

Unsolved Problems:
- Some styles can not be transferred to the content image. The reason behind that is not clear. This usually happens when:
  1. The colours or main objects of the two images are too different.
  2. There is not much obvious line or texture in the style image.

Content	Style	Output

Evaluation Results:

Unlike other traditional models, the Neural Style Migration Model does not have a standard way of assessment. Because it usually depends on aesthetic judgment, here we show some objective data results.

1. Visual Inspection

   By comparing the original image and the image after neural migration, it can be seen that the model retains the key elements of the original image after effectively shifting the style.

<div align="center">Content Image   Output</div>

2. ReLU Converge Speed vs. Conv Converge Speed:

   From the obtained results, it's evident that computing the loss after applying ReLU yields better outcomes compared to computing the loss after convolution. Inserting loss layers after ReLU layers facilitates faster and more efficient convergence compared to inserting them after Conv layers, ReLU layers better preserve information or gradients for faster convergence.

	Insert Loss Layers after Conv		Insert Loss Layers after ReLU	
Step	Style Loss	Content Loss	Style Loss	Content Loss
10	2377.650879	43.962997	17.4932	3.700711
20	682.897217	51.401409	5.970345	3.185289
30	307.438232	49.848648	3.40082	2.763851
40	184.816498	49.311531	2.252596	2.464281
50	126.953033	46.515888	1.753494	2.267855
60	98.597458	44.977612	1.54337	2.128388
70	75.24659	42.910976	1.392532	2.032228
80	61.750702	41.522079	1.292523	1.952586
90	51.640327	40.076035	1.203516	1.90231
100	45.359356	38.968891	1.147376	1.844173

Contribution:
- Jingwen (Steven) Shi: Focuses on Model Building and Report Writing.
  - Responsible for constructing the model and documenting its implementation.
  - Creating comprehensive reports on the progress and findings of the project.
  - Researching and teaching other group members about relevant knowledge related to the project.
- Hongsheng (Ben) Zhong: Focus on Image Style Transform Net building, Training, Fundamental Coding, and Graph Generation.
  - Responsible for developing the Image Transform Net, training the models, and implementing fundamental coding tasks.
  - Responsible for generating graphs or visual representations to illustrate the results and other data visualization tasks.
- Feiran (Phillip) Huang: Focuses on Training function building, Data Processing, Augmentation and Normalization.
  - Responsible for constructing training functions, and implementing augmentation and normalization techniques on the dataset.
  - Ensuring the data is properly prepared and optimized for training the models effectively.

References:

[1]   P. Baheti, "Neural style transfer: Using deep learning to generate art," Apr. 24, 2023. https://www.v7labs.com/blog/neural-style-transfer (accessed Dec. 05, 2023).

[2]   yashchoudhary, "Fast neural style transfer," *Kaggle*, Jul. 04, 2020. Accessed: Dec. 05, 2023. [Online]. Available: https://www.kaggle.com/code/yashchoudhary/fast-neural-style-transfer

[3]   T. Ganegedara, "Intuitive guide to neural style transfer," *Towards Data Science*, Dec. 05, 2020. Accessed: Dec. 05, 2023. [Online]. Available: https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-neural -style-transfer-ef88e46697ee

[4]   Contributors to Wikimedia projects, "Neural style transfer," *Wikipedia*, Mar. 03, 2023. http://en.wikipedia.org/w/index.php?title=Neural%20style%20transfer&oldid=11425552 94 (accessed Dec. 05, 2023).

[5]   M. Dushkoff, R. McLaughlin, and R. Ptucha, "A temporally coherent neural algorithm for artistic style transfer," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, Dec. 2016. Accessed: Dec. 05, 2023. [Online]. Available: http://dx.doi.org/10.1109/icpr.2016.7900142