

BILKENT UNIVERSITY SPRING 2023 - CS 484 FINAL
PRESENTATION

Neural Style Transfer Project

Tugberk Dikmen - 21802480



What is Convolutional Neural Network ?

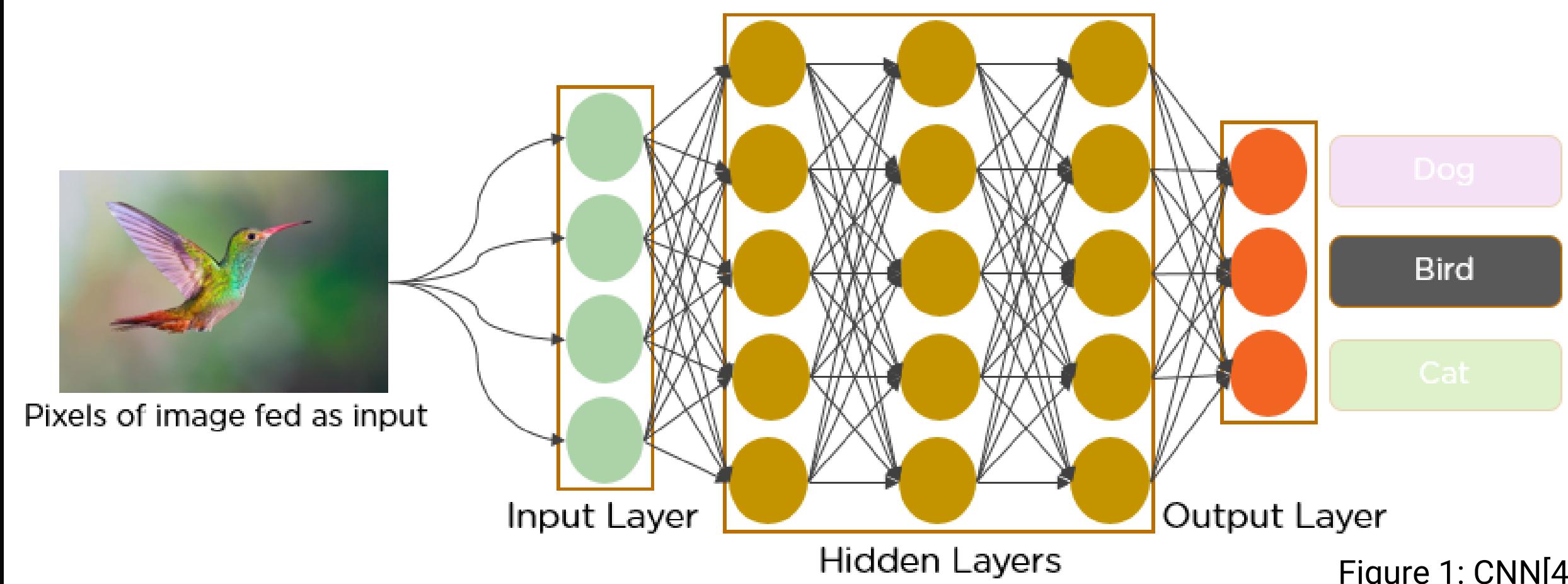
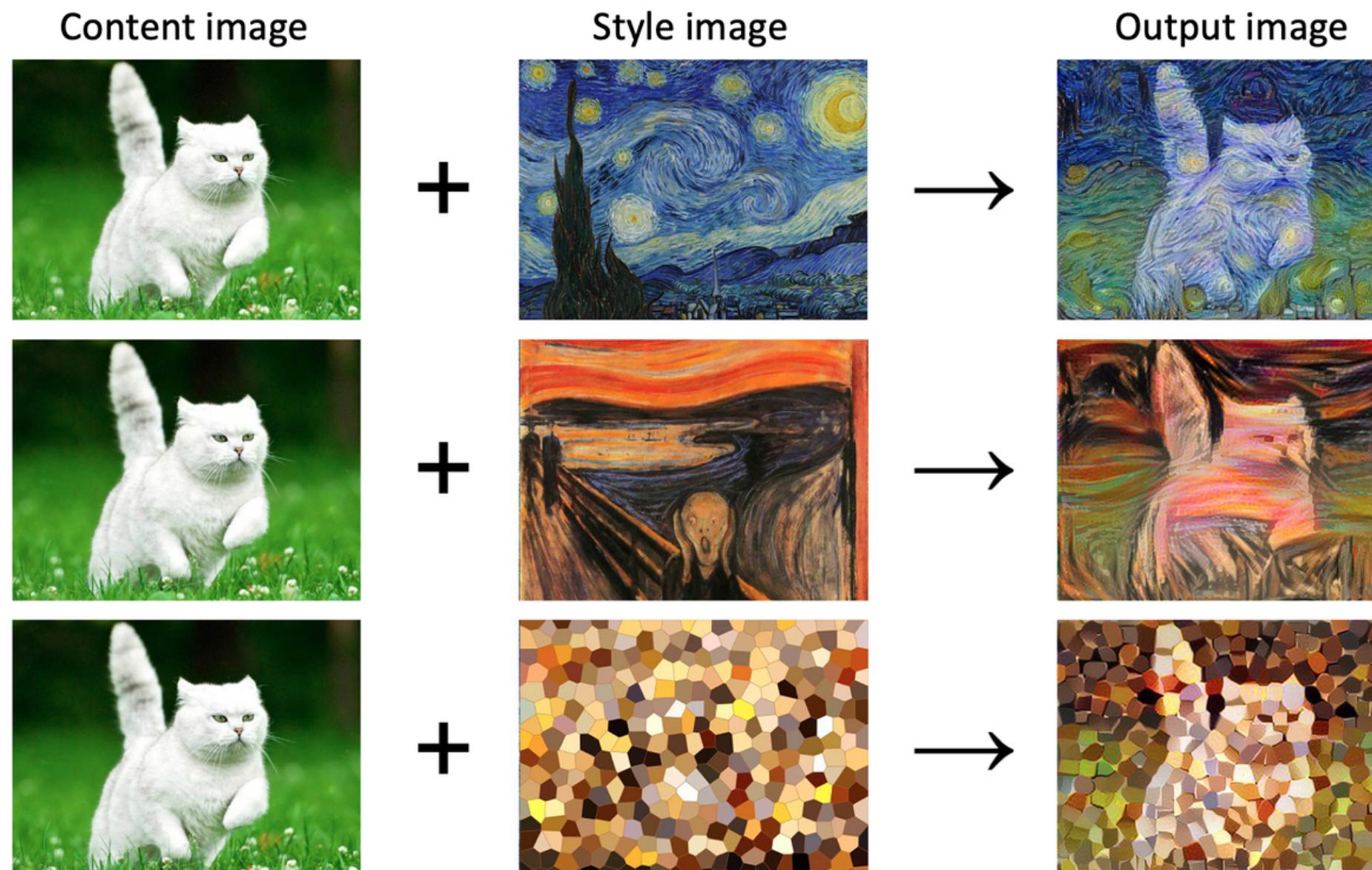


Figure 1: CNN[4]



What is Neural Style Transfer ?

Figure 2: NST[3]

How to Implement a NST Program ?

1. Step



Understanding
what NST is.

2. Step



Choosing a
deep learning
model.

3. Step



Creating a
dataset and
training the
model.

4. Step



Parameter
tuning.

5. Step



Taking the
results.

VGG-19 Deep Learning Model

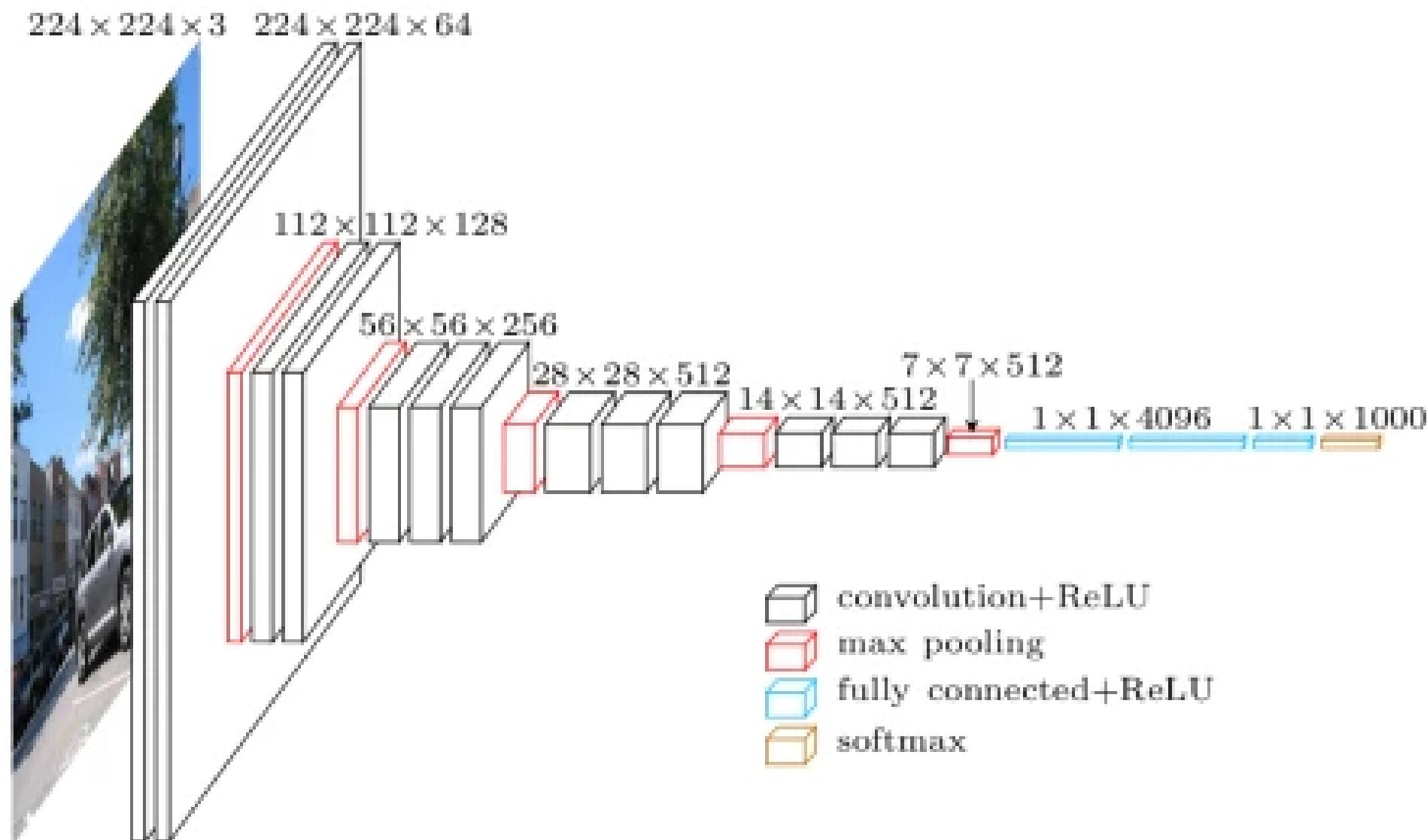


Figure 3: VGG-19[8]



Training VGG-19 Model With Specific Dataset

- Trained for content and style images, 500 each.
- The hyperparameters and architecture should be adjusted and optimized.
- Allows to capture the specific characteristics of the content and style images



Using ImageNet Pretrained VGG-19 Model

- Trained on a large and diverse dataset.
- Already learned to extract low-level features from the images.
- Not always suitable for the specific task may require fine-tuning or re-training

Training VGG-19

Learning Rate (Epochs: 30, Batch Size: 64)

LR: 0.001

- 1/3 of epoch done, **Train Loss: 0.5691 Valid Loss: 0.4940 Accuracy: 82.35%**
- 2/3 of epoch done, **Train Loss: 0.5516 Valid Loss: 0.3493 Accuracy: 85.29%**
- 3/3 of epoch done, **Train Loss: 0.3284 Valid Loss: 0.2217 Accuracy: 97.06%**

LR:0.1

- 1/3 of epoch done, **Train Loss: 0.5691 Valid Loss: 5.4175 Accuracy: 70.59%**
- 2/3 of epoch done, **Train Loss: 0.5012 Valid Loss: 4.8740 Accuracy: 71.35%**
- 3/3 of epoch done, **Train Loss: 0.4109 Valid Loss: 4.0293 Accuracy: 72.63%**

Training VGG-19

Number of Epochs (Learning Rate: 0.001, Batch Size: 64)

NoE: 30

- 1/3 of epoch done, **Train Loss: 0.5691 Valid Loss: 0.4940 Accuracy: 82.35%**
- 2/3 of epoch done, **Train Loss: 0.5516 Valid Loss: 0.3493 Accuracy: 85.29%**
- 3/3 of epoch done, **Train Loss: 0.3284 Valid Loss: 0.2217 Accuracy: 97.06%**

NoE: 60

- 1/3 of epoch done, **Train Loss: 0.5691 Valid Loss: 0.3493 Accuracy: 85.29%**
- 2/3 of epoch done, **Train Loss: 0.3284 Valid Loss: 0.2217 Accuracy: 97.06%**
- 3/3 of epoch done, **Train Loss: 0.2611 Valid Loss: 0.1852 Accuracy: 98.28%**

Training VGG-19

Batch Size (Epochs: 30, Learning Rate: 0.001)

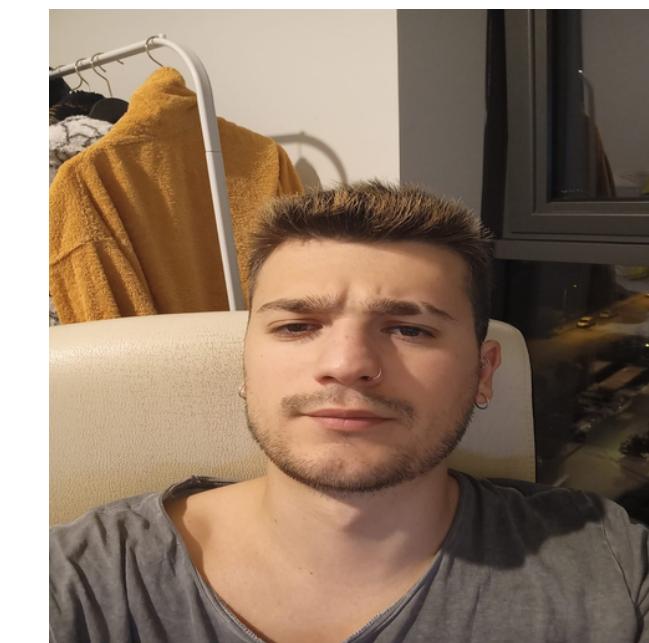
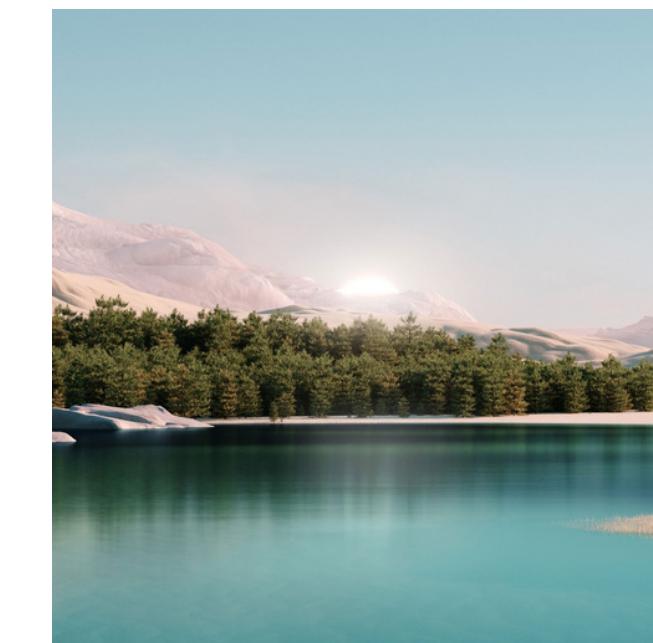
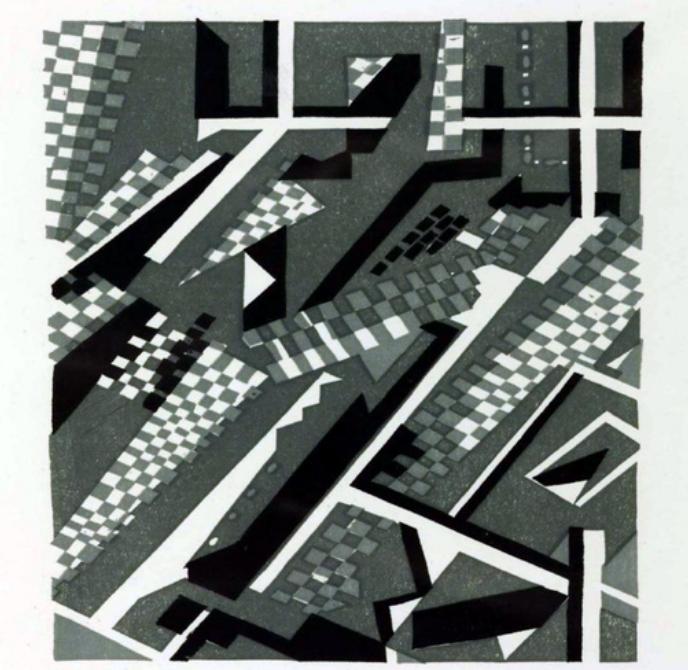
BS: 32

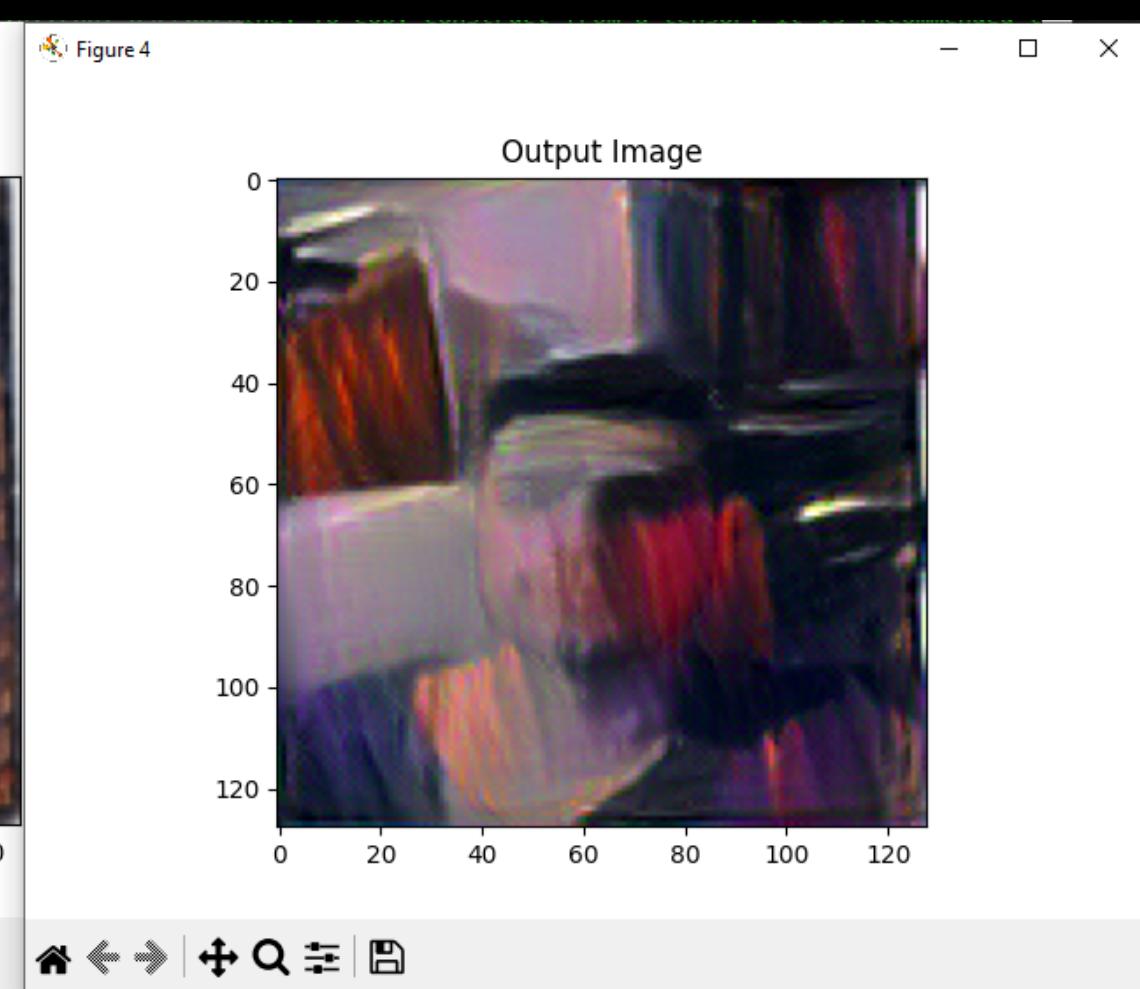
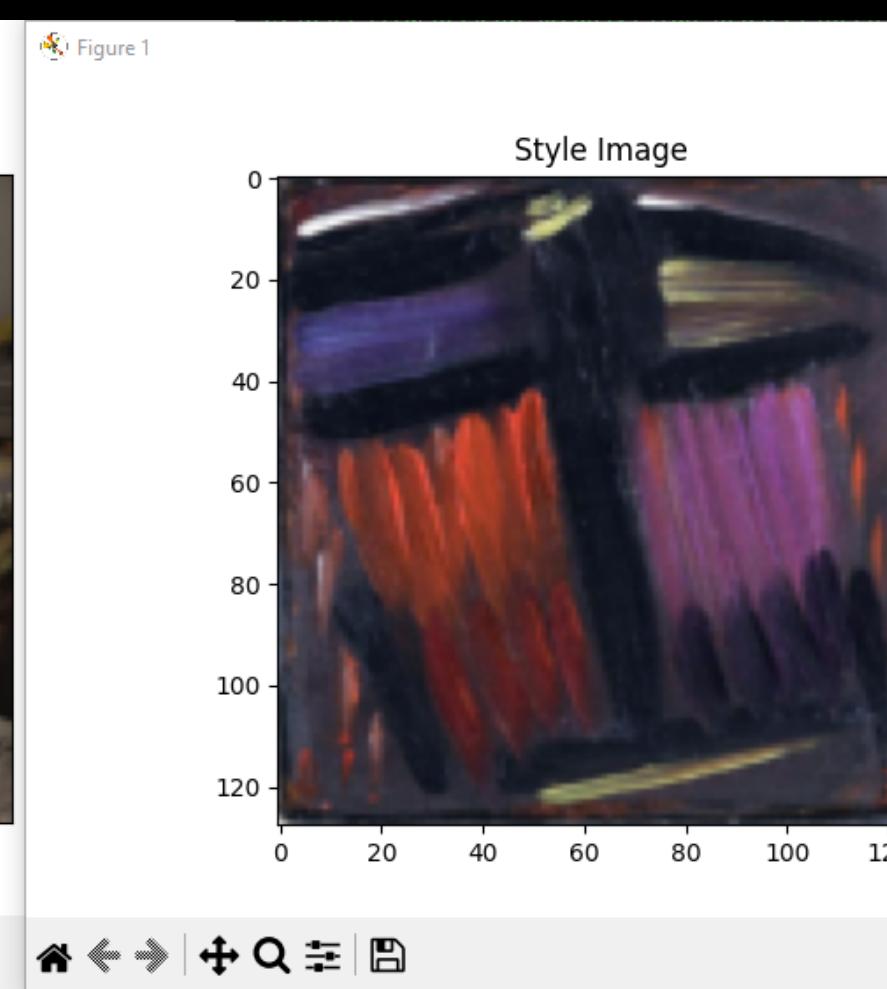
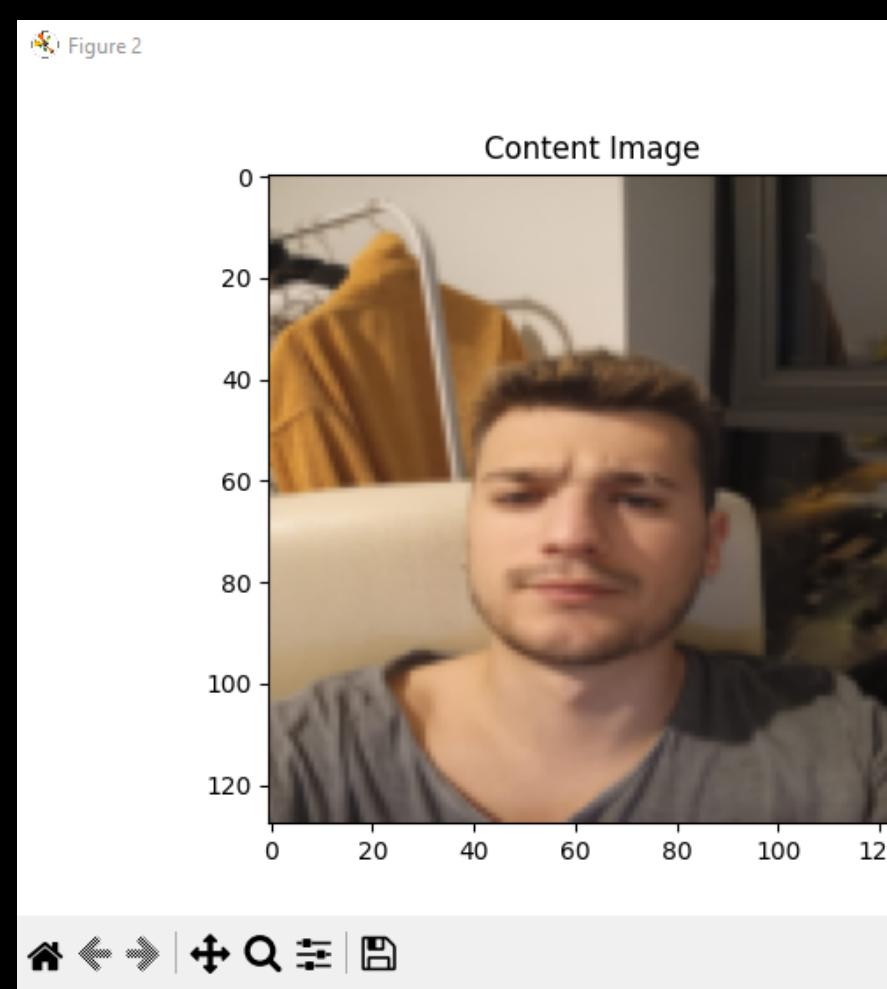
- 1/3 of epoch done, **Train Loss: 0.5426 Valid Loss: 0.3883 Accuracy: 85.29%**
- 2/3 of epoch done, **Train Loss: 0.4040 Valid Loss: 0.1774 Accuracy: 97.06%**
- 3/3 of epoch done, **Train Loss: 0.3219 Valid Loss: 0.0917 Accuracy: 98.86%**

BS: 64

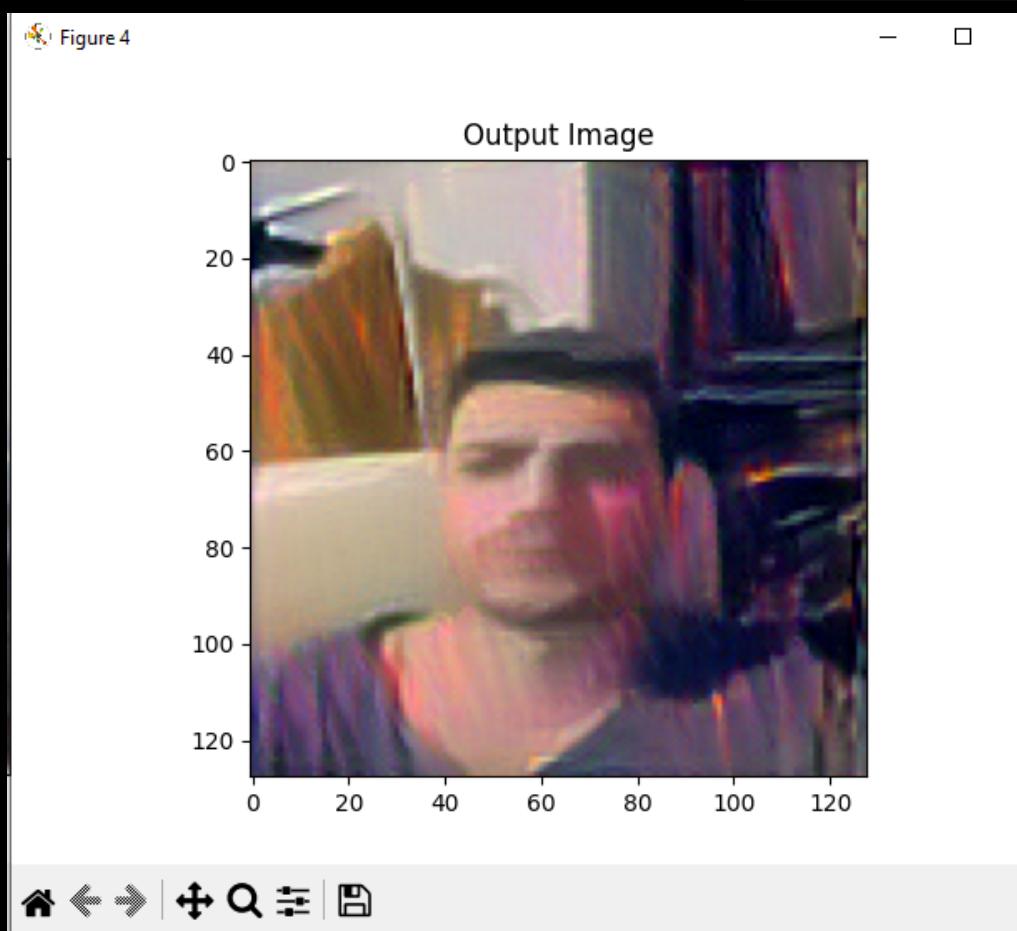
- 1/3 of epoch done, **Train Loss: 0.5691 Valid Loss: 0.4940 Accuracy: 82.35%**
- 2/3 of epoch done, **Train Loss: 0.5516 Valid Loss: 0.3493 Accuracy: 85.29%**
- 3/3 of epoch done, **Train Loss: 0.3284 Valid Loss: 0.2217 Accuracy: 97.06%**

Result Comparison and Analysis for Pretrained Model, LR:0.001 and LR:0.1 Models

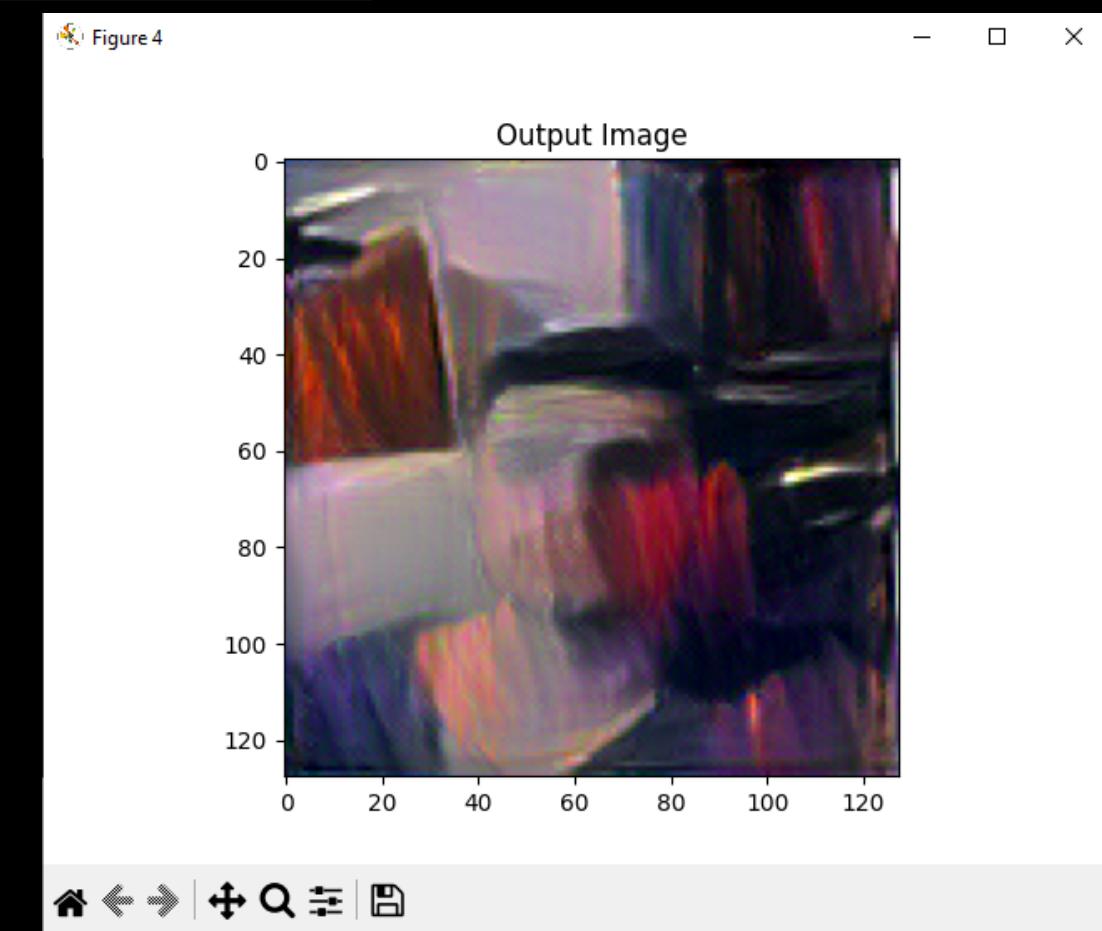




Style Loss : 11.611685 Content Loss: 15.987545



Style Loss : 132.527954 Content Loss: 14.700148



Style Loss : 15.981157 Content Loss: 16.699600

Figure 2

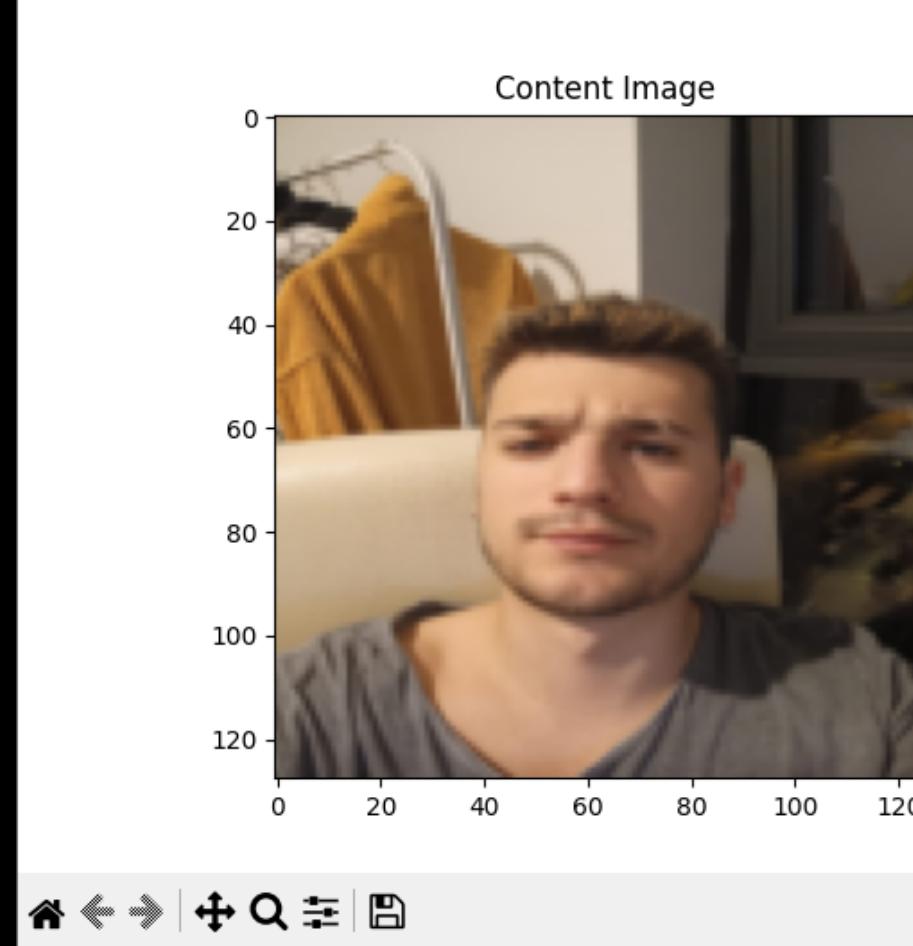


Figure 1

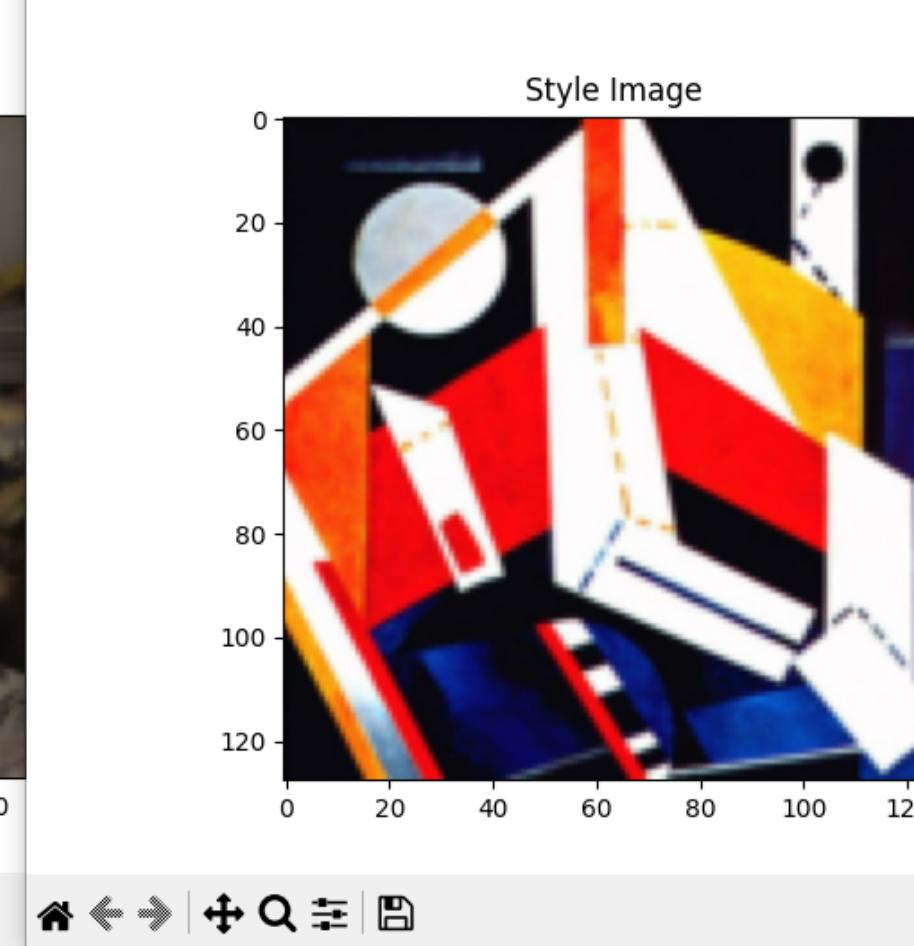


Figure 4

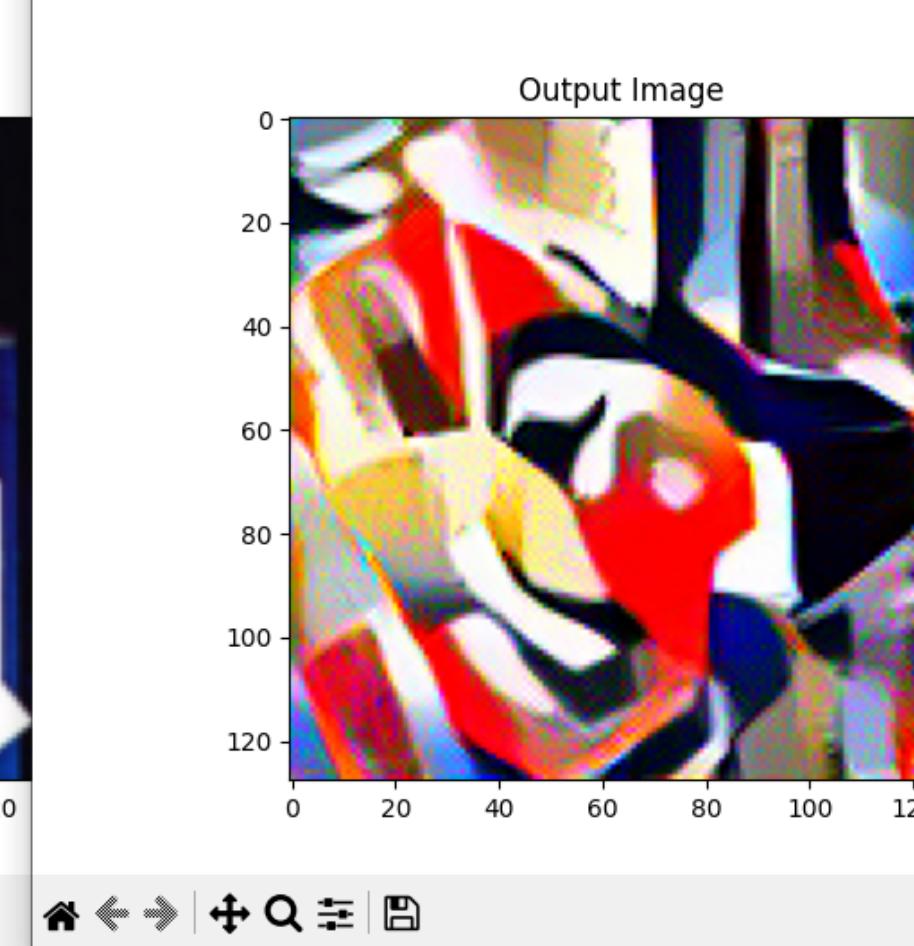


Figure 4

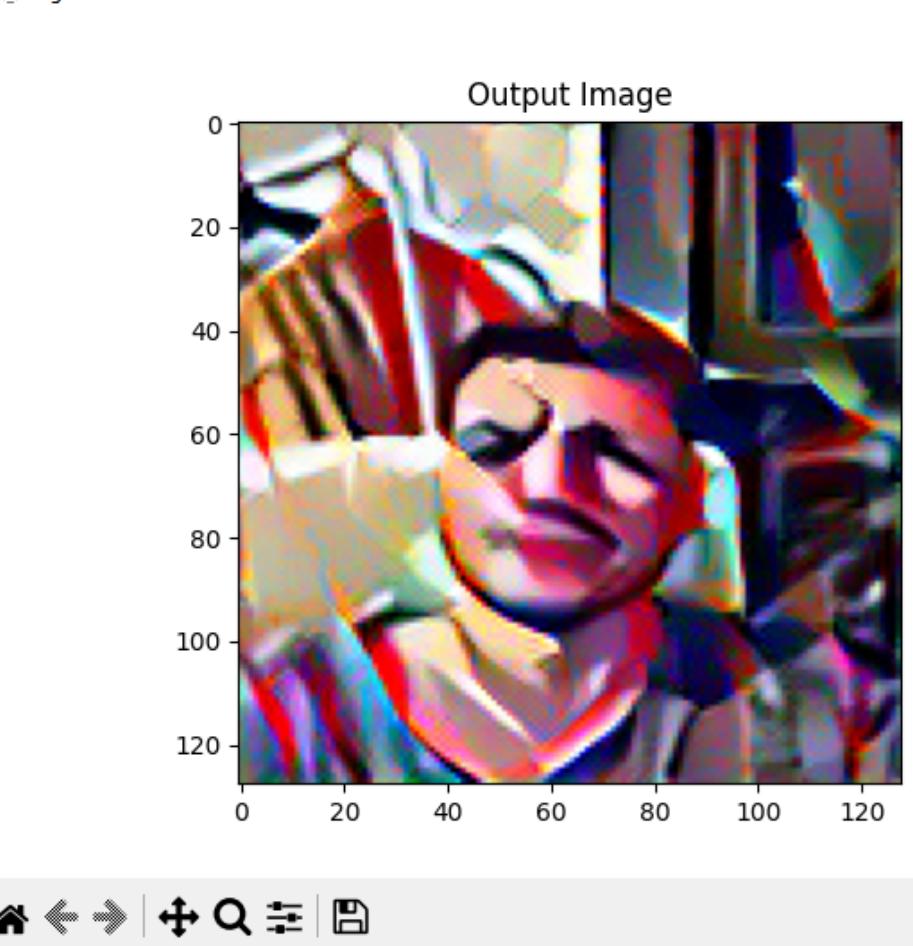
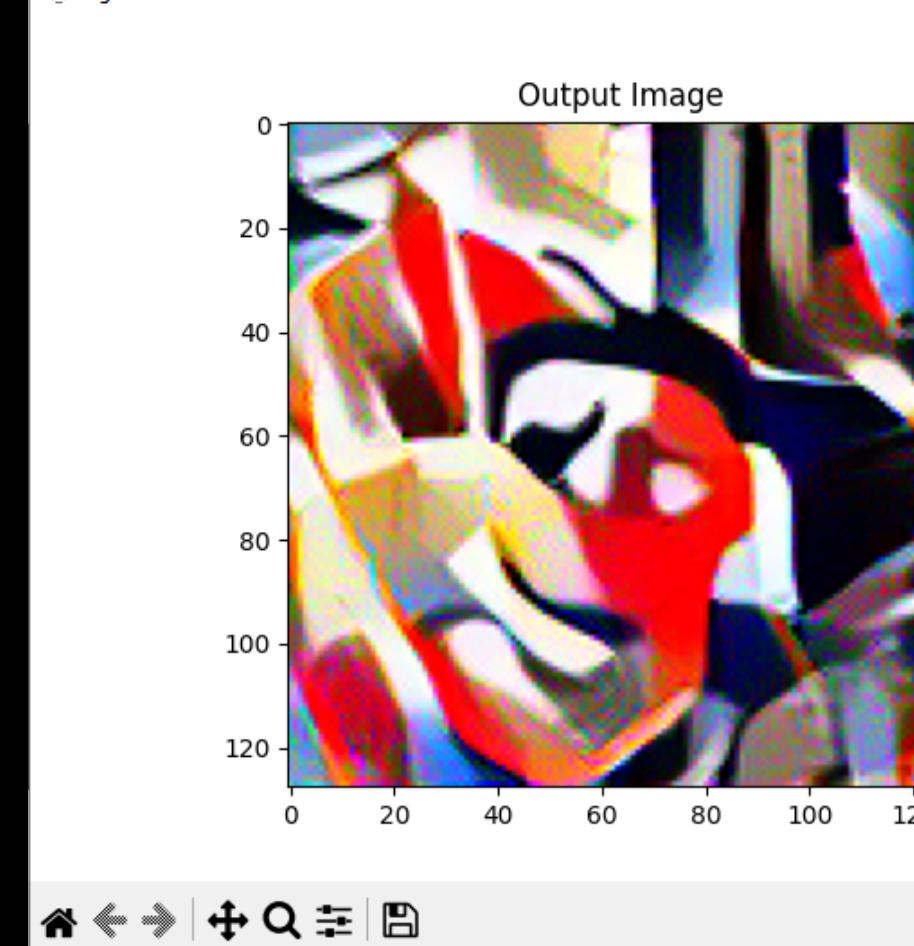
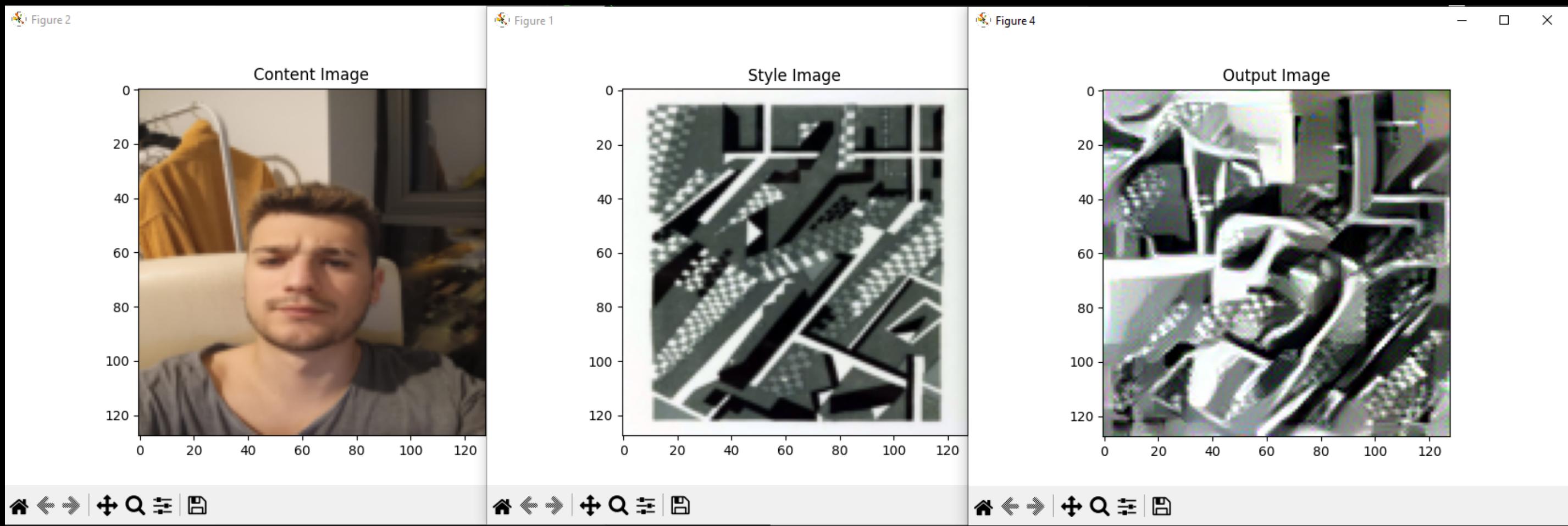


Figure 4

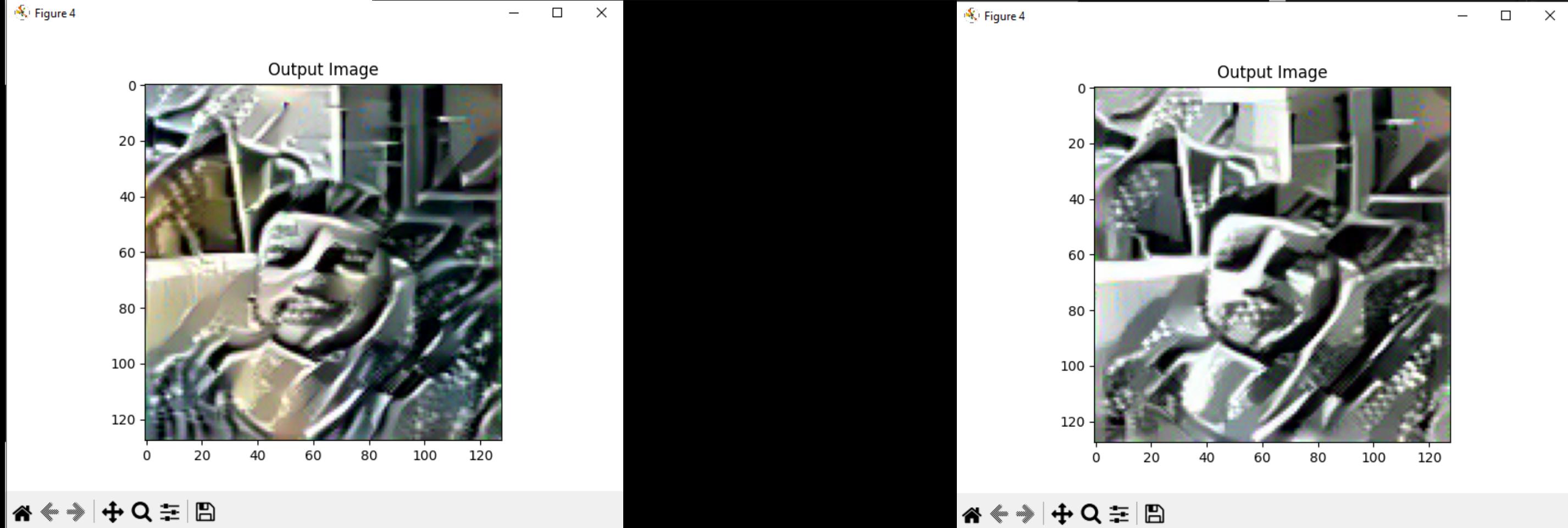


Style Loss : 5883.034180 Content Loss: 61.192142

Style Loss : 879.121338 Content Loss: 76.331711



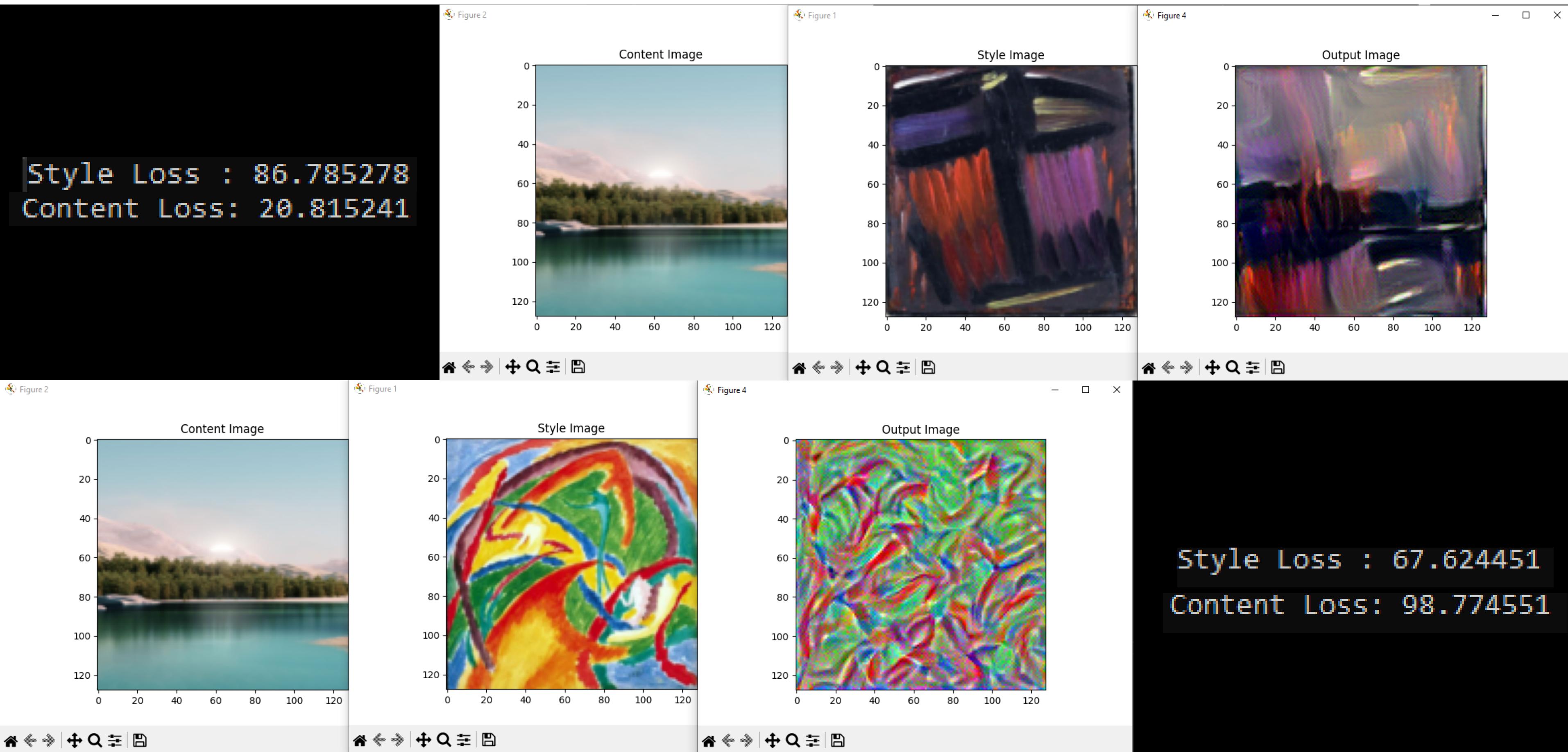
Style Loss : 110.763908 Content Loss: 44.239372



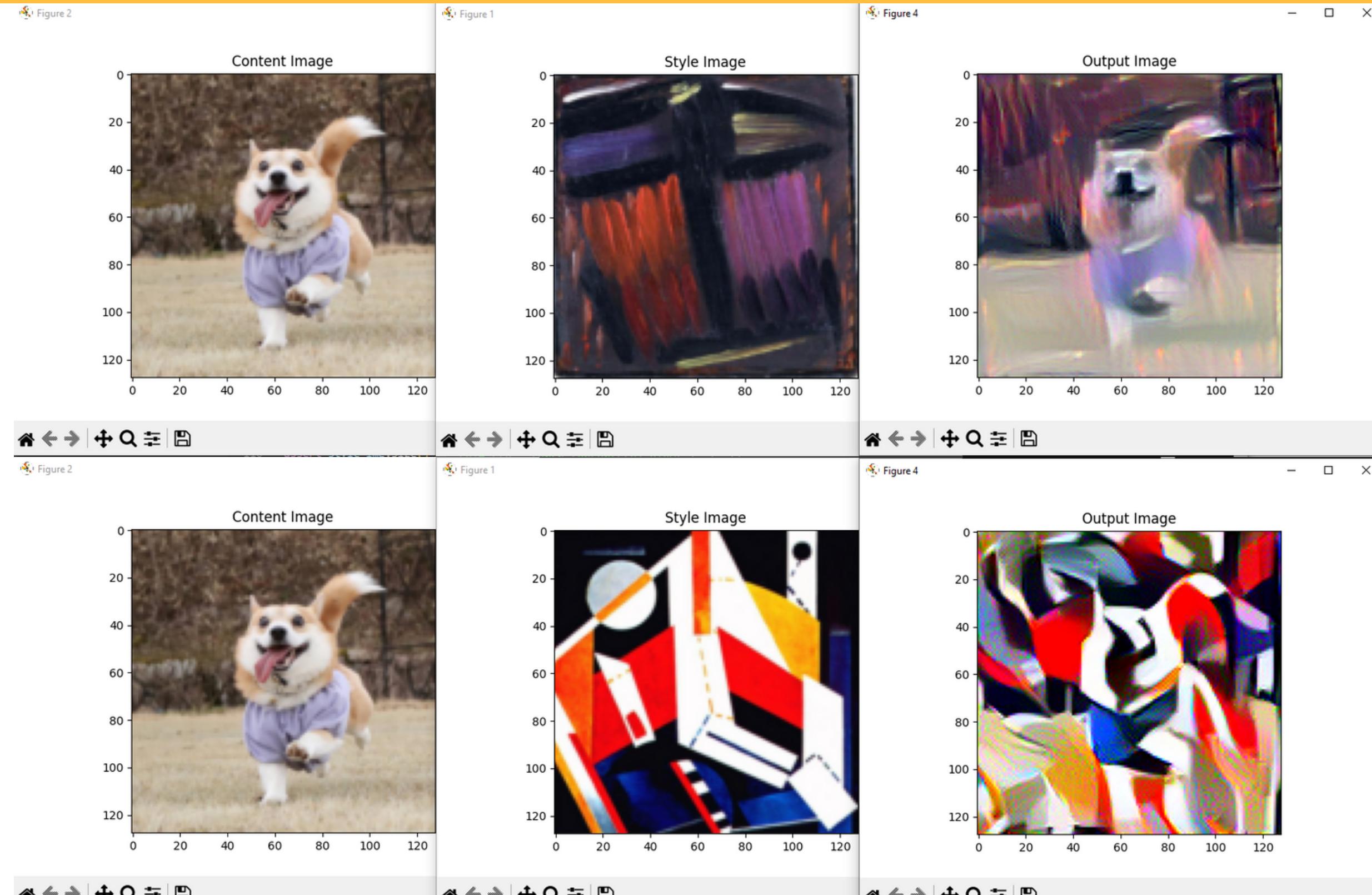
Style Loss : 789.204041 Content Loss: 36.989151

Style Loss : 97.067383 Content Loss: 41.776207

Image Different than Training Content



Conclusion



References

- [1] Computer vision: Algorithms and applications, 2nd ed. [Online]. Available: <http://szeliski.org/Book/>. [Accessed: 07-Apr-2023].
- [2] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," arXiv.org, 02-Sep-2015. [Online]. Available: <https://arxiv.org/abs/1508.06576>. [Accessed: 07-Apr-2023].
- [3] "How to style transfer your own images," GoDataDriven, <https://godatadriven.com/blog/how-to-style-transfer-your-own-images/> (accessed May 8, 2023).
- [4] [1] M. Mandal, "Introduction to convolutional neural networks (CNN)," Analytics Vidhya, <https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/> (accessed May 9, 2023).
- [5] A. Sharma, "Introduction and implementation to neural style transfer - deep learning," Analytics Vidhya, <https://www.analyticsvidhya.com/blog/2020/10/introduction-and-implementation-to-neural-style-transfer-deep-learning/> (accessed May 8, 2023).
- [6] "Neural style transfer tutorial with tensorflow and python in 10 minutes," YouTube, 07-Jan-2021. [Online]. Available: <https://www.youtube.com/watch?v=bFeltWvzZpQ>. [Accessed: 07-Apr-2023].
- [7] "Pytorch Neural Style Transfer tutorial," YouTube, 02-Jul-2020. [Online]. Available: <https://www.youtube.com/watch?v=imX4kSKDY7s>. [Accessed: 07-Apr-2023].
- [8] G. Boesch, "VGG very deep convolutional networks (vggnet) - what you need to know," viso.ai, <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/> (accessed May 8, 2023).
- [9] "Welcome to pytorch tutorials¶," Welcome to PyTorch Tutorials - PyTorch Tutorials 2.0.0+cu117 documentation. [Online]. Available: <https://pytorch.org/tutorials/>. [Accessed: 07-Apr-2023].