Neural Artistic Style Transfer

Gagandeep KN
181IT215, Information Technology
National Institue of Technology, Karnataka)
Surathkal, India
gagandeepkn.181it215@nitk.edu.in

Atharv Belagali
181IT208, Information Technology
National Institue of Technology, Karnataka)
Surathkal, India
atharv910@gmail.com

***Abstract*—Because of its abstraction and originality, art creation is a complicated process. Style transfer employing powerful machine learning technology has been standard in the computer vision sector to make art at a lower cost. On the other hand, traditional transferred images still suffer from color anamorphosis, content loss, and time-consuming issues. In this study, we present a feedforward neural network-based style transfer technique. This makes up the entirety of the network. After training, the style transfer network can immediately translate the content picture into the styled image. The loss network calculates content loss, style loss, and Total Variation (TV) loss to modify the weight of the style transfer network. In addition, a cross-training technique is provided to maintain the content image's features better.**

***Index Terms*—Computer vision, Convolutional neural networks, style-transfer, image synthesis**

I. Introduction

Redrawing an art in a certain style is very difficult job when done manually, it requires an expert and also takes up lot of time to complete this task with perfection. We use Convoltional neural network to perform this task.Separating information from style in natural images is still a challenging task. Deep Convolutional Neural Networks have recently made significant advances, resulting in strong computer vision systems that can train to extract high-level structure information from real images. We employ picture representations generated from Convolutional Neural Networks (CNNs) that are optimised for object recognition and make high-level visual information explicit. The algorithm will enable us to generate a new image of high quality with the combination of content of the style of another image. the results obatined at the end of the project will provide us knowledge on how Convolutional Neural Networks develop deep picture representations and show how they may be used for image synthesis and modification.

II. Literature Survey

A. Deep filter banks for texture recognition, description, and segmentation

They offer numerous advances to texture comprehension in this study in three phases. To begin, they suggest a human-readable texture attribute language to characterise typical texture patterns.Second, they investigate the issue of identifying materials and texture characteristics in realistic imaging environments.

B. Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis

For the purpose of synthesising 2D pictures, this study investigates a combination of generative Markov random field models with discriminatively trained deep convolutional neural networks . They use it for both photographic and non-photorealistic (artwork) synthesis.

C. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture

They use a single basic architecture to tackle three separate computer vision problems in this paper: depth prediction, surface normal estimation, and semantic labelling. They suggest employing a multiscale convolutional network that can quickly adapt to each job with only little changes, regressing straight from the input picture to the output map.

D. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs

For the purpose of "semantic picture segmentation," this article combines approaches from CNNs and probabilistic graphical models. They show that CNN responses at the last layer aren't localised enough for accurate object segmentation.

E. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow

They offer a unique method for estimating optical flow for large displacements with substantial occlusions in this study. There are two steps to it: I dense matching from a sparse set of matches using edge-preserving interpolation; ii) variational energy reduction starting with the dense matches

F. Exploring the Neural Algorithm of Artistic Style

In this paper, They investigate the mechanism of style transfer described in Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge's work "A Neural Algorithm of Artistic Style." On a few cases, they first illustrate the strength of the proposed style space. They next experiment with various hyperparameters and program characteristics not included in the original work.

G. Processing images and video for an impressionist effect

In this paper, they propose a method to convert images and video into animations that look like hand-drawn or painted.

H. Instance Normalization: The Missing Ingredient for Fast Stylization

In this paper, They demonstrate how a simple adjustment in the stylization architecture improves the quality of the produced photos significantly. The change is limited to swapping batch normalisation with instance normalisation, which will be used both during training and testing.

I. Understanding deep image representations by inverting them

They pose the following question in this study to undertake a direct examination of the visual information included in representations: To what degree is it possible to recreate a picture from its encoding? They demonstrate that their technique can more correctly invert representations like HOG and SIFT than recent alternatives, while also being relevant to CNNs.

J. ImageNet Large Scale Visual Recognition Challenge

The construction of this benchmark dataset is described in this study, as well as the advancements in object identification that have emerged as a result. They highlight significant advances in category object identification, offer a thorough study of the current status of large-scale picture categorization and object detection, and compare computer vision accuracy to human accuracy.

III. Motivation

- In the era where all the people are fascinated about animations and graphics,crores of rupees are spent on creating high quality graphics and animations.

- Many films such as Robot,Avatar and marvel movies have revolutionized the film and entertainment industry
- People in India are also realizing the importance of graphics and animation in the field of entertainment
- This can make the graphic designing and animation cost effective and can help the small film producers to afford it.

IV. Problem Statement

The time taken to re draw an image with certain style takes a lot of work and expertise, so to reduce the time and effort. we are exploring the usage of deep neural networks to produce these re drawn image preserving the quality of the original photograph or drawing.

V. Objective

- Produce a re-drawn image with a certain style of the original image
- Preserve the quality of the output image (re-drawn)
- Convert the videos to different style preserving the original structure. Basically applying the same concept to videos
- Understand the applications of the work in producing high quality fake images and image modifications. Find a way to increase the speed of the image transformation to save time.

VI. Methodology

Why is convolutional neural network used? The convolution neural network is a very powerful algorithm when it comes to solving computer vision problems like ours. It is because of the 2 reasons stated below:

- The convolution neural network makes sure that the spatial arrangements of important features during the feature extraction process doesn't matter. Let's take an example to understand this.Suppose we want to detect a face of some animal, then the CNN makes sure that even though the animal in the picture is sleeping, eating, standing or sitting, the face is still detected.So no matter where a particular feature is present in the picture it still makes sure that it is detected.
- Reduces high computations by using techniques like pooling and thus makes the problem solving faster and more practical.

A. Terminologies

Style image- The image whose style has to be transferred to the output image is called the style image. Content image-The image whose content has to be transferred to the output image or the result is known as the content image. Input image- The initial image which is modified to produce an output image is called the initial image.The initial image is equal to the content image initially. Content loss- The euclidean distance between the content image and the input image gives us the content loss.This basically gives us an idea of how different the input image from the content image. When we feed both the above-mentioned images the neural network returns the intermediate layers.The euclidean distance between these intermediate layers gives us the content loss. The mathematical representation of content loss is given below:

$$L_{content}(p,x,l) = 1/2\sum_{ij}(F^l_{ij} - P^l_{ij})^2$$

p and x are the original image and generated image respectively.P and F are the feature maps of the ith layer.

Style loss- Determining the style loss is similar to that of the content loss.Here we find how different is the style of the given image compared to that of the style image.The correlation between different gram matrices of different feature maps gives the content loss.The mathematical expression for content loss is given below:

Total variation loss- This loss is specially considered to improve the texture of the out-

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} \left(G_{ij}^l - A_{ij}^l\right)^2$$

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l$$

Fig. 1: Basic Workflow of the model

put image.That means lesser the TV loss better is the texture of the final image.The TV loss can be computed by the square of the difference between the neighbouring pixels in both the dimensions. The mathematical formula for TV loss is given below:

$$L_{TV} = \sum_{i=1}^{n_H^l - 1} \sum_{j=1}^{n_W^l} \sum_{k}^{n_C^l} \left(c_{i,j,k}^l - c_{i+1,j,k}^l\right)^2 + \sum_{i=1}^{n_H^l} \sum_{j=1}^{n_W^l - 1} \sum_{k}^{n_C^l} \left(c_{i,j,k}^l - c_{i,j+1,k}^l\right)^2$$

The total loss equation is given by the sum of linear combination of all three losses mentioned above. The main idea is to minimize the sum of all three losses or in other words to reduce the sum of euclidean distance between the input image and the style and content image. Minimizing the sum of all three losses gives us the optimised output.The equation to calculate the total loss is given as follows:

$$L_{total} = \alpha L_{content} + \beta L_{style} + \gamma L_{TV}$$

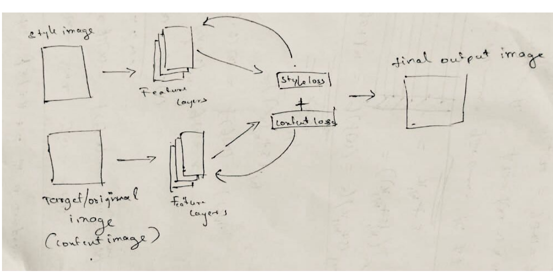

Fig. 2: Basic Workflow of the model

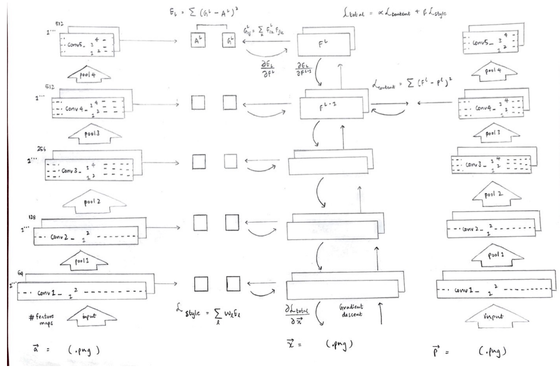

Fig. 3: In depth Architecture of the model

VII. Results and Analysis

We compare our proposed approach to the conventional neural style transfer algorithm to assess it. macOS is the operating system used in the experiment. The relevant software versions are shown in the table below (Table II).

TABLE I: Table of time taken

Software Name	Versions
Python	3.8
NumPy	1.14.6
SciPy	1.1.0
cv2	-

We have compared our model with the traditional style transfer algorithms and these are the results:

TABLE II: Table of time taken

Image size	Traditional	Implemented
256x256	5.4	5.0
512x512	15.1	14.7
1024x1024	30.5	28.6

In Table III, the time is given in seconds. it is observed that after the model was compressed a little bit, the style transfer was faster than the traditional model.

The figures are the output of the two algorithms. The preservance of the content can be easily seen the fig. 4

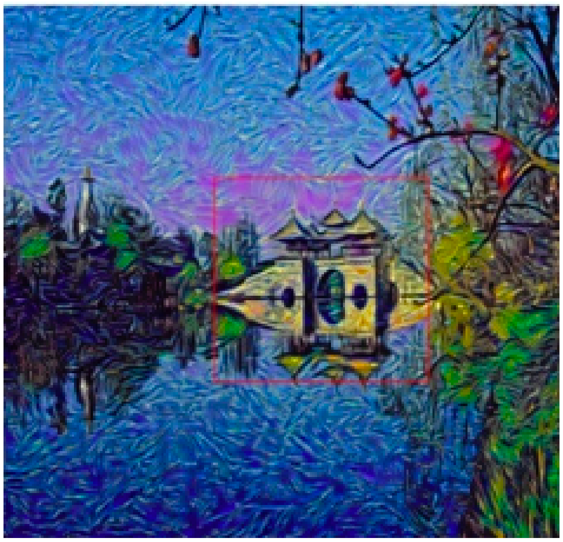

Fig. 4: Output for the algorithm with standard loss function

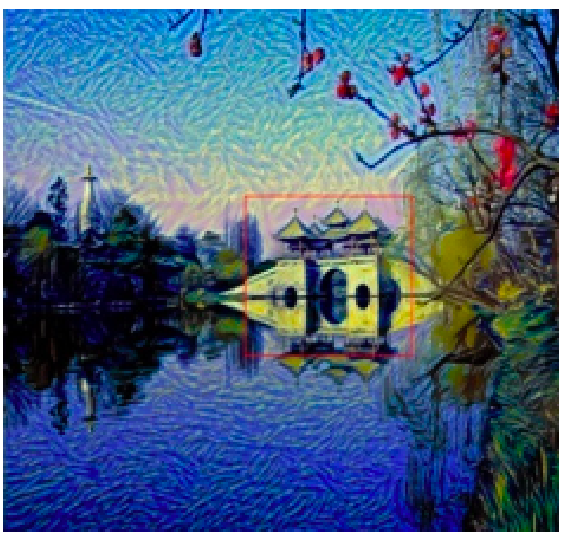

Fig. 5: Output for the algorithm with TV loss

We test the network's performance using the same content and style picture to evaluate two distinct activation functions in the output layer. The outcome of the experiment is displayed below.

Fig. 6: content plus style image

Fig. 7: Output with Tanh activation function

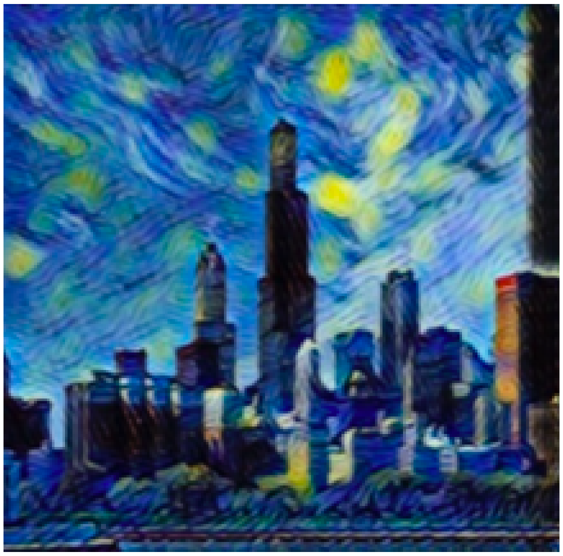

Fig. 8: Output with sigmoid activation function

The output with sigmoid as the activation function has more style embedded in the image than the output with the tanh activation function.

VIII. Conclusion

Traditional neural type transfer has the disadvantages of being time-consuming and losing specifics. In this study, we describe an improved style transfer approach based on a deep feedforward neural network to speed up the stylization process and increase the quality of the stylized picture. A style transfer network made up of many convolution layers is built, and a loss network is based on VGG19. The loss network defines three separate losses that indicate content, style, and smoothness. Then, the style transfer network is trained in advance using the training set, and the loss network determines the loss to update the style transfer network's weight. We tried using the tanh and sigmoid activation functions.

IX. Reference

[1]Adams, A., Baek, J., & Davis, M. A. (2010). Fast high-dimensional filtering using the permutohedral lattice. Computer Graphics Forum, 29(2), 753–762.

[2]Adelson, E. H. (2001). On seeing stuff: The perception of materials by humans and machines. SPIE 4299.

[3]Amadasun, M., & King, R. (1989). Textural features corresponding to textural properties. Systems, Man, and Cybernetics, 19(5), 1264–1274.

[4]Arandjelovic, R., & Zisserman, A. (2012). Three things everyone should know to improve object retrieval. In Proceedings of computer vision and pattern recognition (CVPR).

[5]Arbeláez, P., Pont-Tuset, J., Barron, J.T., Marques, F., & Malik, J. (2014). Multiscale combinatorial grouping. In IEEE conference on computer vision and pattern recognition (CVPR).

[6]Badri, H., Yahia, H., & Daoudi, K. (2014). Fast and accurate texture recognition with multilayer convolution and multifractal analysis. Proceedings of ECCV, 8689, 505–519.

[7]Google Scholar

[8]Bajcsy, R. (1973). Computer description of textured surfaces. In Proceedings of the 3rd international joint conference on Artificial intelligence (IJCAI). Morgan Kaufmann Publishers Inc.

[9]Bell, S., Upchurch, P., Snavely, N., & Bala, K. (2013). Opensurfaces: A richly annotated catalog of surface appearance. In Proceedings of SIGGRAPH.

[10]Berg, T., Berg, A., & Shih, J. (2010). Automatic attribute discovery and characterization from noisy web data. ECCV

[11]Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Pattern Analysis & Machine Intelligence, 24(7), 971–987.

[12]Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and Transferring Mid-Level Image Representations using Convolutional

Neural Networks. In Proceedings of computer vision and pattern recognition (CVPR)

[13]Oxholm, G., Bariya, P., & Nishino, K. (2012). The scale of geometric texture. In European conference on computer vision (pp. 58–71). Berlin: Springer

[14]Parikh, D., & Grauman, K. (2011). Relative attributes. In Proceedings of ICCV.

[15]Parkhi, O. M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). A compact and discriminative face track descriptor. In Proceedings of computer vision and pattern recognition (CVPR)

[16]Patterson, G., & Hays, J. (2012). Sun attribute database: Discovering, annotating, and recognizing scene attributes. In Proceedings of computer vision and pattern recognition (CVPR)