Neural Style Transfer

***Dhawade Sarika, **Gawali Akanksha, **Jathar Pallavi, **Khamkar Snehal,* Bhosale Swati S.**

*****Department of Computer Engineering, HSBPVT's Faculty of Engineering, Kashti

Abstract

The Neural Style Transfer (NST) is the software that allow to the change the Image Environments, Change style, Transform sense using the neural style. Neural Style Transfer is a deep learning technique that combines the contents one images with the style of another creating visually captivating art works that bridge the gap between human creativity and computational process. Introducing the neural network architecture that enables the extraction of content and style features from images. NST finds use in image editing software allowing image stylization based on a general model, unlike traditional methods. The evolution of NST from its inception to the latest advancement, including real-time NST techniques and artistic application. The NST Highlights the various neural network architectures use for NST, comparing their strengths and limitations. The Explore impact of different hyper parameter and techniques on the quality of style transfer result and discuss the potential challenges and future directions in the field. It not only core concepts nut also highlights the numerous possibilities and challenges in the process of creating visually Appling and innovative art works using deep neural network.

Keywords: *Neural Style Transfer (NST), Deep learning, Convolutional Neural Network (CNN), Image Processing, Content-style separation, Style transfer network, Image analysis.*

Introduction

In the computer vision and deep learning, the convergence of art and technology has given rise to a transformative concept known as Neural Style Transfer (NST). NST is an innovative image processing technique that carries the content of one image with the artistic style of another, creating visually captivating and striking compositions. It leverages the power of deep neural networks to reinterpret the essence of an image, reimagining it in the distinct brushstrokes and patterns of iconic artworks, while preserving its essential content. In 2015, introduced a ground-breaking method for synthesizing artistic images by exploring the correlations between content and style representations within a neural network. Since then, NST has garnered widespread attention and found diverse applications, ranging from the creation of museum-worthy artworks to the development of real-time style transfer apps on mobile devices. The fundamental premise of NST is both artistic and scientific, as it bridges the gap between human creativity and computational precision. It enables the democratization of artistry, allowing individuals to transform their photos into masterpieces, democratizing artistic expression in the digital age. Simultaneously, it showcases the remarkable capabilities of deep neural networks in deciphering and emulating complex artistic styles.

Literature Review

The paper [1] combines the benefits of feed-forward image transformation tasks and optimization-based methods for image generation by training feed-forward transformation networks with perceptual loss functions. Perceptual Loss Functions: They define two perceptual loss functions that measure high-level perceptual and semantic differences between images. Feature Reconstruction Loss: Rather than optimistic the pixels of the generated image ˆy = fW (x) to exactly match the pixels of the target image y, we instead encourage them to have related to feature representations as computed by the loss network φ.

The authors [2] have extended their previous work. However the more efficient and real-time style transfer algorithm. Pixel Loss: The pixel loss is the (normalized) Euclidean distance between the output image ˆy and the target y. Total Variation Regularization: To encourage spatial smoothness in the output image ˆy, we follow prior work on feature inversion and super resolution and make use of total variation regularize `T V (ˆy).

In [3] Introduced adaptive instance normalization, enabling real-time and arbitrary style transfer. The degree of style transfer can be composed during training by adjusting the style weight λ in Eqa. Spatial and color control**:** Recently launch user controls over color information and spatial locations of style transfer, which can be easily incorporated into our framework.

The work in [4] introduced a multi-style generative network capable of applying multiple styles to an image simultaneously. Network Architecture: Preceding feed-forward based one-style transfer work learns a Output network that takes only the content image as the input and outputs the transferred image, i.e. the generator network can be show as G(xc), which implicitly learns the feature statistics of the style image from the loss. Up sampled Convolution**:** Standard CNN for image to-image tasks typically adopts an encoder-decoder framework, because it is efficient to put heavy operations (style switching) in smaller feature maps and also important to keep a huge receptive field for preserving semantic coherence.

The authors explored the structure in [5] of style transfer networks and proposed improvements for efficiency and user-friendliness. Two images are related in content if their high-level features as extracted by an image recognition system are terminate in Euclidean distance. Two images are similar in style if their low-level features as extracted by an image recognition system share the same spatial statistics.

Proposed Model:

Convolutional Neural Networks (CNNs):
CNNs are the fundamental building blocks used in neural style transfer. CNNs are used to extract features from both the content and style images. Popular pre-trained CNN architectures like VGG and ResNet are commonly used. A CNN can have many layers, each of which learn to detect the different features of an input image. A filter or kernel is applied to each one image to generate that gets progressively better and more detailed after each layer. In the lower layers, the filters can start as easy features.

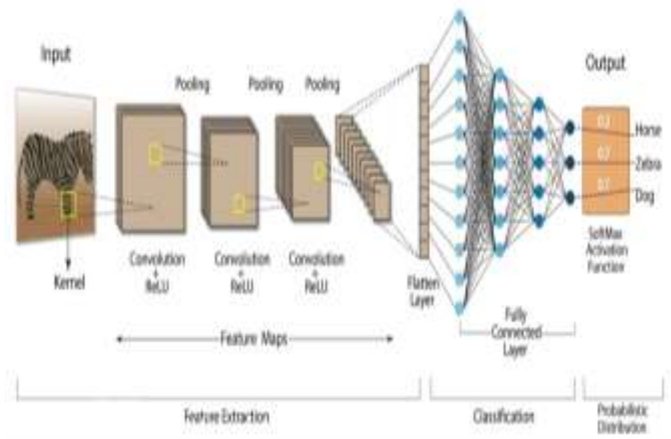
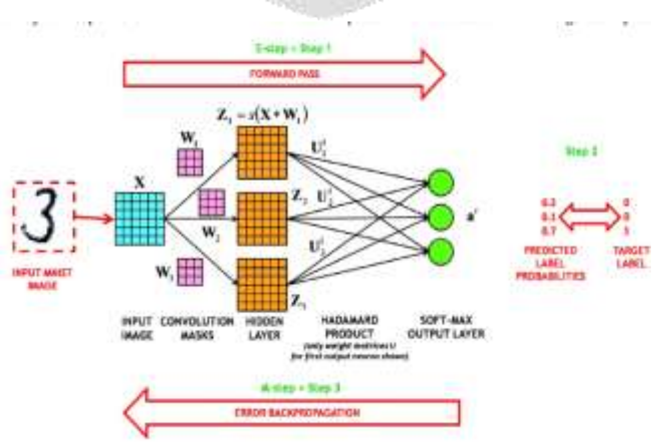


Fig: Convolutional Neural Networks (CNNs):

Content and Style Loss Functions: Neural style transfer uses loss functions to quantify the difference between the content and style of the input images and the generated image.

Content Loss Function: The content loss function quantifies how well the content of the generated image matches the content of the reference content image. It is typically computed as the mean squared error (MSE) or a similar between the feature maps of the content image and the generated image at one or more intermediate layers of a pre-trained convolutional neural network. The content loss encourages the generated image to have similar structures and objects as the content image.
Mathematically, the content loss can be expressed as: Content **Loss = 0.5 * Σ (F_content - F_generated)^2**
Style Loss Function: The style loss function quantifies how well the generated image captures the artistic style of the reference style image. It is computed by comparing the statistics of feature maps (mean and covariance) at multiple layers of the neural network. The style loss encourages the generated image to have similar textures, colours, and patterns as the style image.
Mathematically, the style loss can be expressed as:
Style Loss = Σ (G_style - G_generated)^2
Total Loss: The total loss, which is used for optimization, is a combination of the content loss and the style loss, each weighted by a coefficient. The total loss guides the neural network to find an output image that balances content and style, as well as maintains other image properties.
Total Loss = α * Content Loss + β * Style Loss
Backpropagation and Gradient Descent: To generate the final stylized image, an optimization algorithm (usually gradient descent) is employed to minimize the combined content and style loss.
The gradient of the loss with respect to the generated image is computed, and the image is updated iteratively to minimize this loss.
Backpropagation: Backpropagation, short for "backward propagation of errors," is a supervised learning algorithm used for training artificial neural networks, including deep neural networks. It is a crucial component of the training process because it allows the network to adjust its weights and biases in order to minimize the error in its predictions. Here's how backpropagation works:



Gradient Descent: Gradient Descent is an optimization algorithm used to minimize a loss function by adjusting the parameters of a model (such as neural network weights) iteratively. It works by following the direction of the steepest descent of the loss function.

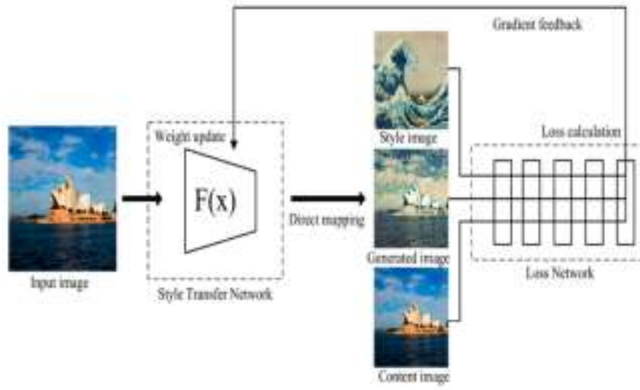


Image Pre-processing and Post processing: Image pre-processing and post processing techniques are used to prepare the input images and render the final stylized output. This may include resizing, normalization, and DE normalization of images to fit the model's requirements and to ensure the output image is visually appealing. **Neural Style Transfer Flowchart:** Neural Style transfer builds on the fact to blend the content image to a style reference image such that the content is painted in the specific style.

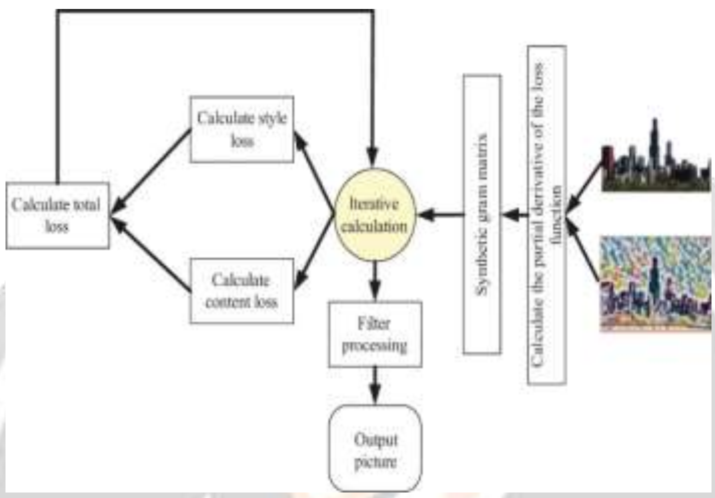
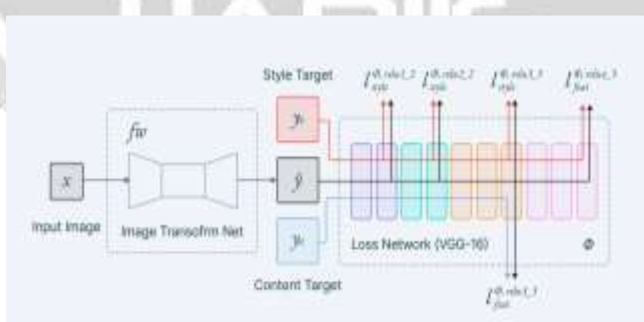


Fig. Flowchart of Neural style Transfer

Mathematical Model:

Ltotal = αLcontent + βLstyle. Thats it! this is all arithmetic we need for style transfer. We need to reduce the total loss and adjust Tstyle & T content to get a target image which has content of the content image and style of out original style image.



Algorithm:

Here are the key algorithms & Technique used in a typical NST:

Pre-trained CNNs: Convolutional Neural Networks, such as VGG-16 or VGG-19, are often used as the backbone for NST. These networks are pre-trained on large image datasets (e.g., ImageNet) and serve as feature extractors for both content and style information. **Content Loss Function:** The content loss is calculated by measuring the difference between the feature representations of the generated image and the content image. It is typically based on the mean squared error (MSE) between the feature maps.

Style Loss Function: The style loss is computed by comparing the statistics of feature maps from the generated image and the style image. This involves calculating the Gram matrix and evaluating the mean squared error between the Gram matrices of feature maps from the style and generated images. **Total Variation Regularization:** To promote spatial coherence and reduce artifacts in the generated image, total variation (TV) regularization is often used. TV regularization penalizes rapid changes in pixel values. **Optimization Algorithm:** Gradient Descent or its variants, such as L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno) or Adam, is employed to iteratively adjust the pixel values of the generated image to minimize the combined content and style loss.

Hyper parameters: Tuning hyper parameters, including the weights for content and style losses, the learning rate, and the number of iterations, is essential to achieving the desired balance between content and style. **Initialization:** Initialization of the generated image, also known as the starting point, can influence the final result. Common initializations include using the content image, random noise, or a combination of both. **Multi-Style Transfer**: Extending NST to incorporate multiple style references, allowing for the creation of images that combine the styles of multiple artists or artworks.

Specification:

Here Tools used:

Tensor Flow: Tensor Flow is a mostly-used open-source deep learning framework developed by Google. It provides a high-level API for building neural networks, making it popular for implementing NST. **PyTorch**: PyTorch is an multiplatform deep learning framework prosper by Facebook's AI Research lab. It is known for its flexibility and dynamic computation graph, which makes it a popular choice for many research projects, including NST. **Keras**: Keras is an open-source high-level neural networks API that runs on top of other deep learning frameworks like TensorFlow and Theano. It simplifies the implementation of neural networks, making it accessible for NST projects.

Fast Neural Style Transfer (NST) Tools: Various pre-built NST tools and implementations are available, which allow users to apply NST without extensive coding. Examples include Fast Neural Style Transfer in Tensor Flow, Artistic Style Transfer with Keras, and more. **Adobe Photoshop and Other Image Editing Software:** Traditional image editing software like Adobe Photoshop can also be used for basic style transfer by manually adjusting the appearance of images.

Jupyter Notebooks: Jupyter notebooks are often used for experimenting with NST and visualizing the results. They provide an interactive and visually rich environment for exploring the code and its effects.

Online NST Services: There are various online platforms and services that offer NST as a service. Users can upload their content and style images, and the service will generate the stylized image. These platforms often use their own implementations of NST.

Advantages:

Artistic Creativity: NST allows artists and designers to create unique, artistic images by blending the content of one image with the style of another.

Automation: NST automates the process of applying a specific artistic style to an image, saving time and effort compared to manual artistic rendering. It can generate stylized images relatively quickly, making it suitable for real-time. Applications like photo and video filtering.

Educational Tool: It can be used in educational contexts to help students understand the relationship between content and style in visual art. NST is a subject of ongoing research, and it provides an avenue for experimenting with deep learning and image processing techniques.

Application:

Artistic Image Generation: NST allows artists and designers to create unique and visually appealing artworks by combining the content of one image with the style of another. This technique can produce stunning paintings, illustrations, and digital art.

Photo Enhancement: NST can be used to enhance and stylize photographs. It's a popular choice for making photos look like famous art pieces or applying different artistic styles to personal photos. Graphic designers use NST to apply specific styles to logos, branding materials, and other visual assets. It can be employed to give a consistent and unique look to a company's visual identity.

Fashion and Textile Design: NST is used in the fashion industry to create unique patterns, textures, and designs for clothing, accessories, and textiles. It helps designers experiment with different styles and combination. Interior designers use NST to envision how different art styles or colour schemes would look in a room. It can assist in the creative process of decorating living spaces.

Education and Learning: NST can be used in educational settings to teach about art, art history, and the concept of artistic styles. It can help students understand and appreciate the work of famous artists.

Conclusion:

We concluded that, neural style transfer (NST) is a powerful and versatile technique that offers numerous advantages for a wide range of creative and practical applications. It combines the content of one image with the artistic style of another, enabling the generation of unique and visually appealing artwork. Whether you're an artist, a designer, a researcher, or a hobbyist, NST can be a valuable tool to enhance your projects. The GANs improvement style, explaining how Spatial, Colour, and Scale control can allow better image generation. Lastly, how NST can be applied over mobile devices in real-time using GANs has been explained.

References:

1]"Perceptual Losses for Real-Time Style Transfer and Super-Resolution" (Johnson et al., 2016).

2]"Fast Neural Style Transfer" (Johnson et al., 2016).

3]"Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization" (Huang and Belongie, 2017).

4]"Multi-style Generative Network for Real-time Transfer" (Deng et al., 2017).

5]"Exploring the structure of a real-time, arbitrary neural artistic stylization network" (Luan et al., 2017).

6] The Smartphone vs. the Camera Industry. Accessed: Apr. 20,2021.[Online].Available:https://photographylife.com/smartphone-vs-cameraindustry/amp

7] Adobe Premiere Pro. Accessed: Apr. 20, 2021. [Online]. Available:https://www.adobe.com/in/products/premiere/movie-and-filmediting.html.

8] DaVinci Resolve. Accessed: Apr. 20, 2021. [Online]. Available:https://www.blackmagicdesign.com/in/products/davinciresolve.

9] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, ''Neural style transfer: A review,'' IEEE Trans. Vis. Comput. Graphics, vol. 26, no. 11, pp. 3365–3385, Nov. 2020.

10] J. Li, Q. Wang, H. Chen, J. An, and S. Li, ''A review on neural style transfer,'' J. Phys., Conf. Ser., vol. 1651, Nov. 2020, Art. no. 012156.

11]. Hao, A. Mallya, S. Belongie, and M.-Y. Liu, ''GANcraft: Unsupervised 3D neural rendering of Minecraft worlds,'' 2021,arXiv:2104.07659.[Online].

12] T. Karras, S. Laine, and T. Aila, ''A style-based generator architecture for generative adversarial networks,'' IEEE Trans. Pattern Anal. Mach. Intell., early access, Feb. 2, 2020, doi: 10.1109/TPAMI.2020.2970919.

13] R. Dhir, M. Ashok, and S. Gite, ''An overview of advances in image colorization using computer vision and deep learning techniques,'' Rev. Comput. Eng. Res., vol. 7, no. 2, pp. 86–95, 2020.[Online].Available:http://www.conscientiabeam.com/journal/76/abstract/6190,doi:10.18488/journal.76.2020.72.86.

14] Y. Deng, F. Tang, X. Pan, W. Dong, C. Ma, and C. Xu, ''StyTr2 : Unbiased image style transfer with transformers,'' 2021,arXiv:2105.14576.[Online].

15] A. Li, C. Wu, Y. Chen, and B. Ni, ''MVStylizer: An efficient edgeassisted video photo-realistic style transfer system for mobile phones,'' in Proc. 21st Int. Symp. Theory, Algorithmic Found., Protocol Design Mobile Netw. Mobile Comput. New York, NY, USA: Association for Computing Machinery, 2020, pp. 31–40, doi: 10.1145/3397166.3409140.f

SARIKA DHAWADE is currently pursuing the BE. Degree in computer science with HSBPVT'S GOI, Savitribai phule pune University.

AKANKSHA GAWALI is currently pursuing the BE. Degree in computer science with HSBPVT'S GOI, Savitribai phule pune University.

PALLAVI JATHAR is currently pursuing the BE. Degree in computer science with HSBPVT'S GOI, Savitribai phule pune University.

SNEHAL KHAMKAR is currently pursuing the BE. Degree in computer science with HSBPVT'S GOI, Savitribai phule pune University.