Original Article

NEURAL STYLE TRANSFER AS DIGITAL REPRESENTATION THROUGH PAINTING USING DEEP LEARNING

Mahesh Vishvakarma [1]*

[1] Freelance Web Developer, Self-Employed, India

ABSTRACT

The intersection of art and technology, particularly the development of deep learning and artificial intelligence, has opened new avenues for creative expression. Neural Style Transfer (NST) is a new approach that uses deep neural networks to add creative style to digital images. This technique overlays visual elements—such as texture, color patterns, and brushstrokes—from one image onto the content of another, producing interesting and imaginative results.

This study explores the fundamental concepts, methodology, and artistic significance of neural style transfer. This technique is primarily based on convolutional neural networks (CNNs), which are capable of extracting high-level content features and low-level style features from images. Through an optimization process that separates and then reassembles these features, NST creates images that retain the structural content of the original image while adopting the artistic style of the reference artwork.

This research highlights the interdisciplinary impact of neural style transfer, demonstrating its relevance not only in digital art and graphic design, but also in fields such as animation, multimedia, and visual communication. Furthermore, NST serves as a bridge between traditional artistic methods and modern computational techniques, allowing artists, designers, and technologists to collaborate and explore new creative possibilities. The results show that neural style transfer is an important step toward integrating deep learning.

Keywords: Neural Style Transfer (NST), Painting, Convolutional Neural Networks (CNNs)

INTRODUCTION

Neural style transfer (NST) is a technique in the fields of deep learning and computer vision that focuses on creating a new image by combining the content of one image with the style of another. The primary objective of NST is to preserve the structural information of the content image by applying aesthetic features such as texture, color, and visual patterns from the style image.

With the rapid advancement of artificial intelligence and convolutional neural networks (CNNs), neural style transfer has emerged as a major application of deep learning in digital arts, image processing, and the creative industries. The concept was first popularized by Gatys et al., who demonstrated that deep neural networks trained for object recognition can also be used to separate and reassemble the content and style of images.

In neural style transfer, a pre-trained convolutional neural network, such as VGG-16 or VGG-19, is used to extract feature representations from both the content and style images. Content representations capture high-level semantic structure, while style representations are constructed using statistical measures such as Gram matrices, which encode texture and visual correlation. By optimizing the target image using a weighted combination of content loss and style loss, a visually appealing output is generated that reflects both content and artistic style.

Neural style transfer has widespread applications in areas such as digital painting, film production, gaming, augmented reality, and multimedia content creation. It not only automates artistic processes but also helps non-artists create visually rich images using simple inputs, bridging the gap between technology and creativity.

EXPERIMENTAL CONFIGURATION AND VISUAL PRESENTATION
Figure 1
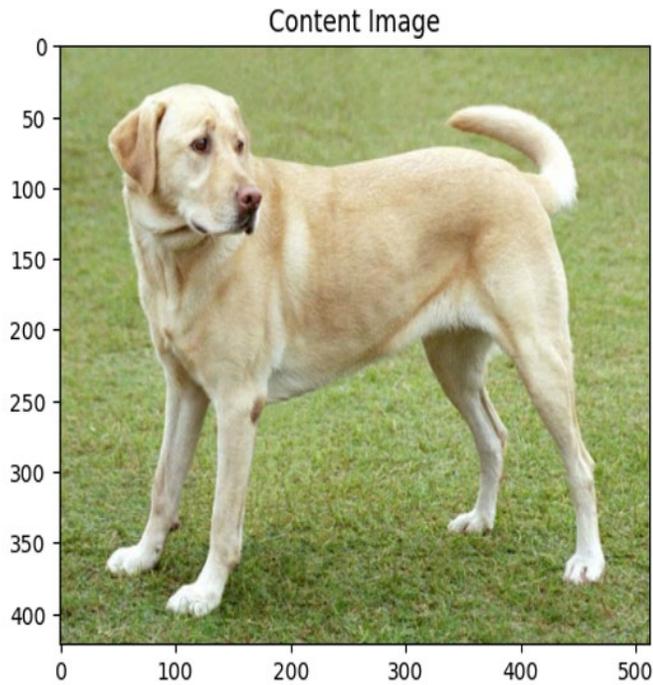
Figure 1 Shows the Original Content Image Used in the Experiment, While

Figure 2
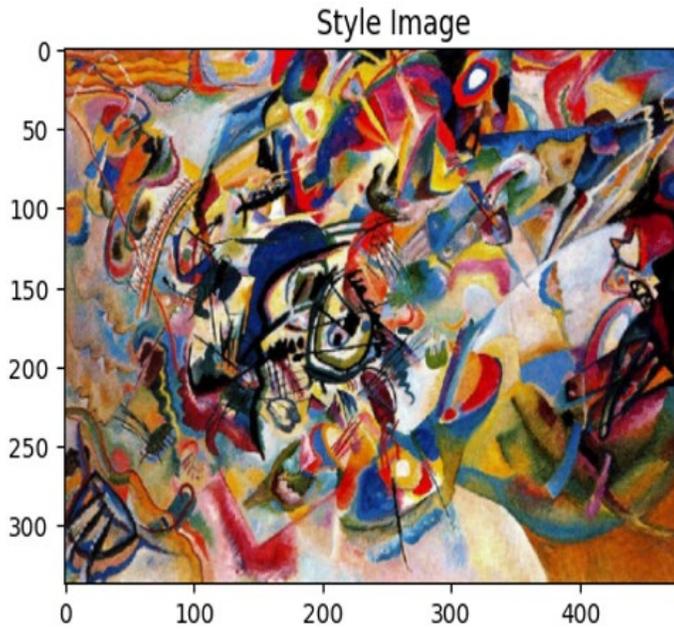
Figure 2 Shows the Reference Style Image.

Figure 3

Figure 3 Shows the Final Stylized Output Generated After Applying Neural Style Transfer.

These images clearly show how the structural elements of the source image have been preserved, while the artistic color of the style image has been successfully transferred.

METHODOLOGY

The proposed methodology implements neural style transfer using a deep learning framework. The system is based on previously developed convolutional neural networks, specifically the VGG-19 model, which serves as a feature

extractor. The methodology includes image selection, preprocessing, RGB normalization, style calculation, loss calculation, and optimization using gradient descent.

IMAGE SELECTION AND PREPROCESSING

In this research, the user manually selects both the content and style of the image through an image upload interface. The system supports common digital image formats such as JPEG and PNG, which makes this approach flexible and user-oriented rather than dependent on a fixed dataset.

After the image is uploaded, it is converted to a standard RGB format to ensure compatibility with the neural network's input dimensions and resized to a fixed resolution. Resizing also helps reduce computational complexity while preserving the necessary visual information.

RGB CALCULATION AND NORMALIZATION:

A digital image is represented using the RGB color model, where each pixel has three channels: red (R), green (G), and blue (B). The intensity value of each channel typically ranges from 0 to 255.

For a pixel at position (x, y), the RGB representation is:

$$Pixel(x, y) = [R(x, y), G(x, y), B(x, y)]$$

To make the data suitable for neural network processing, the RGB values are normalized as follows:

$$R' = \frac{R}{255}, G' = \frac{G}{255}, B' = \frac{B}{255}$$

After normalization, pixel values fall into the [0, 1] range, improving numerical stability and convergence speed.

STYLE REPRESENTATION USING GRAM MATRIX:

The content of an image is represented by the feature activations of the inner layers of the neural network. However, the style of an image is captured using the correlation between different feature maps.

the Gram matrix is calculated as:

$$G_L = F_L \cdot F_L^T$$

where F_L is the matrix of vectorized feature maps at layer L. This operation captures texture patterns and artistic features.

WORKING OF NEURAL STYLE TRANSFER

NST creates an image that matches the content representation of the content image and the style representation of the style image. The created image is initially random or copied from the content image, and is repeatedly updated until it minimizes the total loss.

Figure 4

Figure 4

CONTENT LOSS

Content loss ensures that the structure and meaning of the content image are preserved in the resulting image. It is calculated as the mean square error between the feature maps of the resulting image and the content image.

STYLE LOSS

Style loss is an essential component of neural style transfer, ensuring that the resulting image recreates the statistical texture patterns of the styled image. Unlike content loss, which preserves specific structure, style loss captures visual appearance features such as brush strokes, color distribution, and repetitive textures.

The concept of style loss was introduced by Gatys et al., who showed that the style of an image can be effectively represented by the correlation between feature maps extracted from a deep convolutional neural network.

MATHEMATICAL REPRESENTATION

Let $F\_l \in R^{\wedge}(N\_l \times M\_l)$ be the feature map of the layer l, where:

- N_l= number of feature channels

- M_l= spatial size of each feature map

The Gram matrix for layer l is defined as:

$$G_l = F_l \cdot F_l^T$$

The Gram matrix captures the interactions between different feature channels, which encode texture and style information.

STYLE LOSS FUNCTION

The style loss for a layer is calculated as the mean square difference between the Gram matrix of the generated image and the style image:

Where:

- G^l = Gram matrix of generated image

- A^l = Gram matrix of style image

The total style loss is obtained by aggregating losses across multiple layers:

$$L_{style} = \sum_l w_l L_{style}^l$$

Here w_l represents the weight assigned to each layer.

TOTAL LOSS FUNCTION

The total loss is defined as:

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

Where α controls content importance and β controls style importance.

GRADIENT DESCENT

In neural style transfer, the objective is not to train the neural network itself, but to transfer the pixel values of the resulting image in a way that minimises the total loss. This adaptation problem is solved using a gradient-based adaptation technique.

The total loss is defined as:

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

Where:

- $L_{content}$ preserves semantic structure
- L_{style} preserves artistic texture
- α, β are weighting factors controlling the trade-off

FORMULATION OF GRADIENT DESCENT:

Let x denote the generated image. The goal of optimization is to find:

$$x^* = \arg \min_x L_{total}(x)$$

This is done by repeatedly updating the image generated using gradient descent:

$$x_{t+1} = x_t - \eta \frac{\partial L_{total}}{\partial x_t}$$

Where:

- η = learning rate
- $\frac{\partial L}{\partial x}$ = gradient of loss w.r.t. pixels

WORKFLOW
Figure 5

Generated Image (x_0)

↓

VGG Feature Extraction

↓

Compute Content & Style Loss

↓

Backpropagation $(\partial / \partial x)$

↓

Update Image using L-BFGS

↓

Style Loss $= \|A - G\|^2$
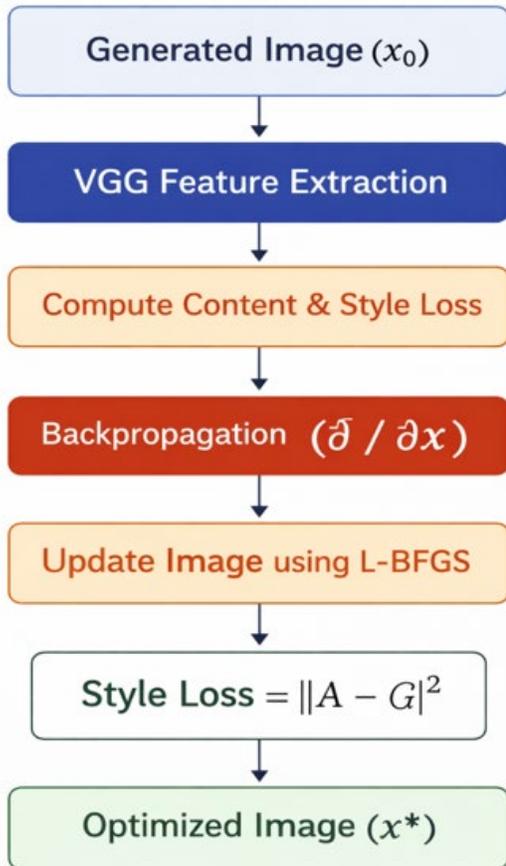
↓

Optimized Image (x^*)

Figure 5 Workflow

COMPARATIVE COLOR TRANSMISSION ANALYSIS

Figure 6 shows a comparative analysis of different color transmission techniques, including Reinhardt, Iterative Distribution Transfer (IDT), MKL, Luminance Transfer, Histogram Matching, Kolsky Decompression, and PCA. The figure shows both the visual output and their corresponding color histograms for each method applied to the same source image.

Histogram Distribution This highlights that Histogram Matching and IDT achieve a closer alignment with the original content image in terms of color intensity and coefficient. These methods preserve the global color structure more effectively than other techniques, such as Reinhardt and PCA, which exhibit noticeable deviations in channel distribution.

Additionally, luminance transfer rate produces intensity transitions but lacks a strong color coefficient, while MKL and IDT with re-graining provide a balanced perceptual quality. Overall, histogram matching and IDT perform better in maintaining perceptual similarities and color consistency with the content image, making them more accurate for true color transmission.
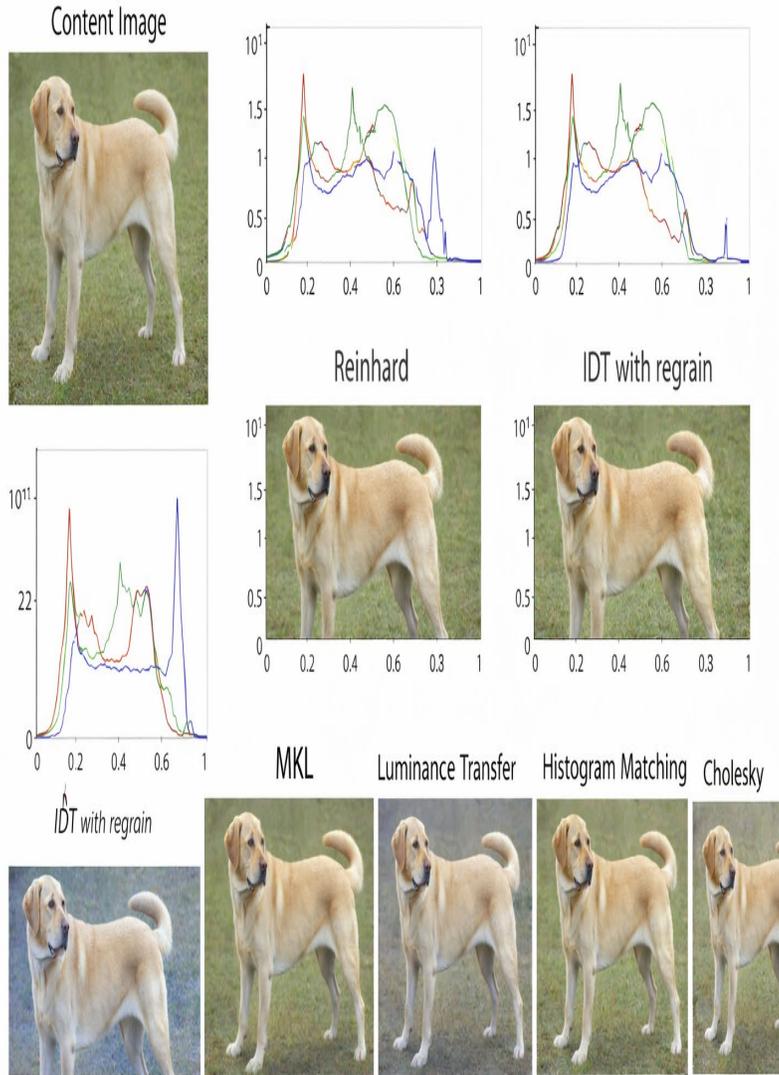
Figure 6

Figure 6 Visual and histogram-based comparison of different color transmission methods (Renard, IDT, MKL, luminosity transform, histogram matching, Cholesky, PCA).

QUANTITATIVE EVALUATION

Table 1

Table 1 Shows the KL-Divergence Values for Different Colour Transfer Techniques Across RGB Channels.

Method	Red	Green	Blue
Reinhard	1.41	1.08	1.65
IDT (10 iter.)	0.10	0.08	0.19
Luminance Transfer	0.82	0.31	0.92
Histogram Matching	0.15	0.26	0.04
PCA	43.63	Inf	1.66

Lower KL values indicate better similarity between colour distributions.

RESULTS AND DISCUSSION

Experimental results demonstrate that neural style transfer successfully incorporates artistic features from style images while maintaining the structural integrity of the content image. Pending results confirm that NST is capable of creating visually appealing digital paintings.

Analysis of the results also shows that histogram-based color transfer techniques outperform linear transformations in terms of perceptual similarity.

CONCLUSION

This research shows that neural style transfer is not just a visual filter, but a mathematically based optimization framework that enables a deep neural network to digitally paint. This approach effectively combines artistic creativity with computational intelligence.

Neural style transfer serves as a powerful tool for digital artists, designers, and multimedia professionals, opening up new possibilities for creative expression while preserving the essence of human imagination.

ACKNOWLEDGMENTS

None.

REFERENCES

Efros, A. A., and Freeman, W. T. (2001). Image Quilting for Texture Synthesis and Transfer. In Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01) (341–346). ACM. https://doi.org/10.1145/383259.383296

Garber, D. D. (1981). Computational Models for Texture Analysis and Texture Synthesis (Doctoral dissertation). University of Southern California, Image Processing Institute. https://doi.org/10.21236/ADA102470

Gatys, L. A., Ecker, A. S., and Bethge, M. (2015). A Neural Algorithm of Artistic Style. arXiv. https://arxiv.org/abs/1508.06576

Harrison, P. (2001). A Non-Hierarchical Procedure for Re-Synthesis of Complex Textures. In WSCG 2001 Conference Proceedings (190–197).

Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In Computer Vision—ECCV 2016 (694–711). Springer. https://doi.org/10.1007/978-3-319-46475-6_43

Khaligh-Razavi, S.-M., and Kriegeskorte, N. (2014). Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. PLOS Computational Biology, 10(11), e1003915. https://doi.org/10.1371/journal.pcbi.1003915

Kriegeskorte, N., Mur, M., Ruff, D. A., Kiani, R., Bodurka, J., Esteky, H., Tanaka, K., and Bandettini, P. A. (2008). Matching Categorical Object Representations in Inferior Temporal Cortex of Man and Monkey. Neuron, 60(6), 1126–1141. https://doi.org/10.1016/j.neuron.2008.10.043

Kwatra, V., Schödl, A., Essa, I., Turk, G., and Bobick, A. (2003). Graphcut Textures: Image and Video Synthesis Using Graph Cuts. ACM Transactions on Graphics, 22(3), 277–286. https://doi.org/10.1145/882262.882264

Kyprianidis, J. E., Collomosse, J., Wang, T., and Isenberg, T. (2013). State of the "Art": A Taxonomy of Artistic Stylization Techniques for Images and Video. IEEE Transactions on Visualization and Computer Graphics, 19(5), 866–885. https://doi.org/10.1109/TVCG.2012.160

Lee, H., Seo, S., Ryoo, S., and Yoon, K. (2010). Directional Texture Transfer. In Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering (NPAR '10) (43–48). ACM. https://doi.org/10.1145/1809939.1809945

Liu, S. (2022). An Overview of Color Transfer and Style Transfer for Images and Videos. arXiv. https://arxiv.org/abs/2204.13339

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (3431–3440). https://doi.org/10.1109/CVPR.2015.7298965

Mahendran, A., and Vedaldi, A. (2014). Understanding Deep Image Representations by Inverting Them. arXiv.

Pitié, F., Kokaram, A. C., and Dahyot, R. (2007). Automated Colour Grading Using Colour Distribution Transfer. Computer Vision and Image Understanding, 107, 123–137. https://doi.org/10.1016/j.cviu.2006.11.011

Popat, K., and Picard, R. W. (n.d.). Novel Cluster-Based Probability Model for Texture Synthesis, Classification, and Compression. In Proceedings of SPIE: Visual Communications and Image Processing.

Ruderman, D. L., Cronin, T. W., and Chiao, C.-C. (1998). Statistics of Cone Responses to Natural Images: Implications for Visual Coding. Journal of the Optical Society of America A, 15(8), 2036–2045. https://doi.org/10.1364/JOSAA.15.002036

Wang, X., and Yu, J. (2020). Learning to Cartoonize Using White-Box Cartoon Representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (8090–8099). https://doi.org/10.1109/CVPR42600.2020.00811

Wei, L.-Y., and Levoy, M. (2000). Fast Texture Synthesis Using Tree-Structured Vector Quantization. In Proceedings of SIGGRAPH 2000 (479–488). https://doi.org/10.1145/344779.345009

Xu, M., and Ding, Y. (2022). Color Transfer Algorithm Between Images Based on a Two-Stage Convolutional Neural Network. Sensors, 22, 1–21. https://doi.org/10.3390/s22207779

Xu, Y., Guo, B., and Shum, H.-Y. (2000). Chaos Mosaic: Fast and Memory Efficient Texture Synthesis (Technical Report MSR-TR-2000-32). Microsoft Research.

Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J. (1997). Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization. ACM Transactions on Mathematical Software, 23(4), 550–560. https://doi.org/10.1145/279232.279236