CYBERNETICS AND PHYSICS, VOL. 10, NO. 3, 2021, 127–137

ARTWORK STYLE TRANSFER MODEL USING DEEP LEARNING APPROACH

Ngo Le Huy Hien
School of Built Environment, Engineering, and Computing
Leeds Beckett University
England
n.hien2994@student.leedsbeckett.ac.uk

Luu Van Huy
Department of Information and Technology
University of Science and Technology, The University of Danang
Vietnam
luuvanhuy2012@gmail.com

Nguyen Van Hieu*
Department of Information and Technology
University of Science and Technology, The University of Danang
Vietnam
nvhieuqt@dut.udn.vn

Article history:
Received 24.08.2021, Accepted 25.11.2021

Abstract

Art in general and fine arts, in particular, play a significant role in human life, entertaining and dispelling stress and motivating their creativeness in specific ways. Many well-known artists have left a rich treasure of paintings for humanity, preserving their exquisite talent and creativity through unique artistic styles. In recent years, a technique called 'style transfer' allows computers to apply famous artistic styles into the style of a picture or photograph while retaining the shape of the image, creating superior visual experiences. The basic model of that process, named 'Neural Style Transfer,' has been introduced promisingly by Leon A. Gatys; however, it contains several limitations on output quality and implementation time, making it challenging to apply in practice. Based on that basic model, an image transform network was proposed in this paper to generate higher-quality artwork and higher abilities to perform on a larger image amount. The proposed model significantly shortened the execution time and can be implemented in a real-time application, providing promising results and performance. The outcomes are auspicious and can be used as a referenced model in color grading or semantic image segmentation, and future research focuses on improving its applications.

Key words

Image Processing, Style Transfer, Image Transformer Network, Deep Learning, Convolution Neural Network.

1 Introduction

Nowadays, tremendous efforts have been put into accelerating different techniques to assist computers in performing human-like tasks such as classification and communication due to the fast-growing artificial intelligence advancements. Most of the conventional deep learning techniques contribute to improving productivity and quality of life, and artworks are one of the promising areas. Preserving artistic features thereby becomes noteworthy for both human duties and technology, as they represent the heritage and culture through time [Doulamis and Varvarigou, 2012]. The first image transforming algorithm was founded as an innovation to change a picture's style while preserving its shape, which has drawn substantial attention [Gatys et al., 2016b]. The term 'neural style transfer' was introduced to indicate a machine learning technique for converting an image's style from an initial to another by blending a content image into a style reference image, such as artworks from reputed artists. The fundamental purpose is to generate a content-look-alike image but is artificially drawn in the style of the reference image, as illustrated in Figure 1.

Although several researchers have been involved in this trend, there are still many spaces for developing and solving the backlog of limitations [Liu et al., 2017; Chen et al., 2018; Gatys et al., 2016a]. In the scope of this study, the authors have pointed out those remaining limitations in the image transformation models and propose methods to resolve them. Moreover, a web application was implemented to apply the built model, attempted to create a number of exquisite art paintings. The approach

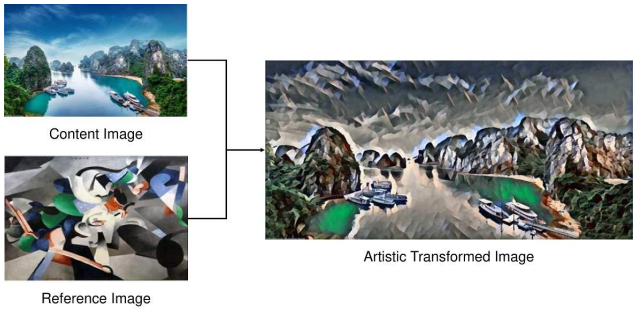

Figure 1. Ha Long Bay image styled in an udnie art.

of this study is divided into 3 steps: building a model to generate uncomplicated images and conveying observations related to the drawbacks of the model; proposing an image conversion network to enhance the pattern conversion model based on the initial model; and applying model transformation and building an artistic image generator website. The proposed model significantly shortened the execution time and can be applied as a referenced model in real-time applications.

2 Literature Review

In recent years, many research authors have been solving style image transformation problems by training convolutional neural networks with loss functions per pixel [Selim et al., 2016; Zhao et al., 2020]. A typical example can be seen from the 'Neural Style Transfer' model proposed by Leon A. Gatys [Gatys et al., 2016b], which has used the content image and reference image for training without any large training dataset directly. The model has produced new images of high perceptual quality that blend the appearance of famous artworks into the content of an arbitrary photograph, with insights into deep image representations. Although this model contains unavoidable shortcomings, it is a prerequisite for later studies on image generation methods [Chen et al., 2018; Jing et al., 2019]. AdaIN model, for example, was developed to match the mean-variance of a content image with a reference image for remodeling features [Huang and Belongie, 2017]. A patch match procedure was introduced in the swap model [Chen and Schmidt, 2016] for alternating content features with the nearest match style characteristic. Li et al. [Li et al., 2017] proposed a multilevel stylization for transforming whiten color recursively, improving the output quality and preserve the content structure. The style transfer problem can be widely used in other neural networks image processing applications, such as text classification, semantic parsing, and information extraction. The deep fake image approach [**?**], for example, might be considered a solution to comparable facial feature transfer challenges. Moreover, it can be seen from these studies that there is a trade-off between content and style losses. Therefore, many researchers have considered image super-resolution and image segmentation methods.

Image super-resolution is a classic problem related to image processing [Bazhanov et al., 2018], and there have been a number of researchers involved in this area [Cheng et al., 2019; Ma et al., 2020]. Yang et al. [Yang et al., 2014] provided a general review of previously standard techniques when applying convolutional neural networks to their research. Some other output quality improvement methods were suggested, such as a model of Chao Dong et al. [Dong et al., 2015], taking a low-resolution content image through a convolutional neural network [Andreev and Maksimenko, 2019] and producing an image with high resolution. While their neural network architecture is uncomplicated and less weight, it exhibits a high quality of image recovery and performs in a short time to apply in practical applications. The above studies are the driving force behind this research to produce the same high-quality transition images as conventional image hyper-resolution.

Image Segmentation methods divide an image into many different image areas [Long et al., 2015]. Image segmentation also has the same objective as the object detection problem: detecting the image area containing the object and labeling them appropriately [Noh et al., 2015]. Although the issue of image segmentation requires a higher level of detail, in return, the algorithm gives an understanding of the image's content at a deeper level. Simultaneously, it reveals the position of the object in the image, the shape of the object, and which object each pixel belongs to [Zheng et al., 2015]. This method generates labels for image regions for the input image to run through a fully convolutional neural network, trained with the loss function per pixel.

The objective of this research is to develop a model that can provide smooth transitions and results as sharp as the study of Chao Dong et al. [Dong et al., 2015]. It is also proposed an image transformation network inspired by two studies of Long J. [Long et al., 2015] and Noh H. [Noh et al., 2015], improving the quality of the output image and shorten the transition time. The results are promising for transferring artwork style to other images, contributing to the applications in many natural science fields, including materials science and physics.

3 Standard Image Style Conversion Model

In the standard image style conversion model proposed by Leon A. Gatys et al. [Gatys et al., 2016b], its input data consists of a content image and a reference image. The resulting (target) image is initialized as white noise before applying convolutional neural networks (CNN) to transform the white noise image closer to the content and visual style.

As illustrated in Figure 2, this standard model comprises 2 steps:

1. Training by a convolutional neural network for extracting features.
2. Calculating the loss functions of the content image and the reference image to update the target image

closer to the content and reference image. The result is obtained when the total loss is minimum.

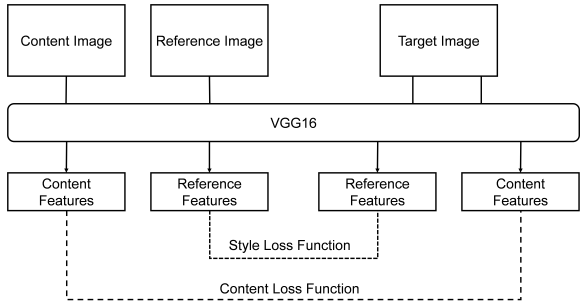

Figure 2. Standard Conversion Model.

3.1 Convection Neural Network Features Extraction

The standard model applies the VGG-16 pre-trained neural network to extract the content image and reference image features. Despite using the same VGG16 network, content images and reference images have different extraction features. When passing an image through the convolutional neural network, the higher-order layers in the convolutional neural network capture the 'high value' content about the objects and their arrangement from the input image without correct pixel values. In contrast, the lower layers reproduce the exact pixel values of the original image. Therefore, features extracted from higher-order layers are considered image content features, while features at lower layers are considered image style features, as shown in Figure 3.

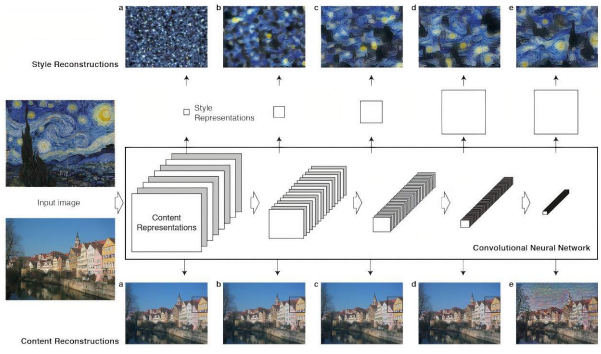

Figure 3. The CNN Network extracts the features of content and style images.

Based on the study of Leon A. Gatys et al. [Gatys et al., 2016b], the class of the VGG16 network was used to extract the characteristic of:

– Content image: conv5_2;
– Reference image: conv1_1, conv2_1, conv3_1, conv4_1, conv5_1.

3.2 Building Loss Function

This section calculates the loss functions of the content image and the reference image to update the target image closer to the content and reference image, as presented below.

3.2.1 Loss Function for Recreating Content To extract content features, the content image was passed through filters of a convolutional neural network. For each class of the CNN network with N_l filters, N_l feature maps sized M_l (M_l = width $\times$ height of the image) were generated. Therefore, each class l stores the matrix $F_l \in R^{N_l \times M_l}$, in which F^l_{ij} is the trigger function of the i$^{\text{th}}$ filter at position j in class l. Let $\overrightarrow{p}$ and $\overrightarrow{x}$ is the input and output image, and P^l and F^l represent their respective features in the class. The loss function is based on content characteristics of the input and output image is:

$$L_{\text{content}} = \frac{1}{2}\sum_{i,j}\left(F^l_{\text{ij}} - P^l_{\text{ij}}\right)^2 \tag{1}$$

3.2.2 Loss Function For Recreating Style Calculating the style loss function is relatively more complicated, whereas it follows the same principle. Instead of comparing the intermediate outputs of the content image and reference image, a Gram matrice was handled to compare the two outputs.

Calculating the Gram matrix: After the target image and reference image passed through a convolutional neural network, a n_C feature map was obtained with $n_H \times n_W$ dimensions. To calculate the similarity of the 2 images, n_C feature maps were taken before computing the scalar product of two feature vectors (each feature map is flattened to a feature vector) on each corresponding pair of maps. As a result, a Gram matrix with $n_C \times n_C$ dimensions was created, as shown in Figure 4. Given a set n_C of $n_H \times n_W$ vectors, the Gram matrix G is the matrix of all possible inner products of n_C:

$$G_{\text{ij}} = F_{\text{i}}^{\intercal} F_{\text{j}} \tag{2}$$

The Gram matrix determines the vectors F_i up to isometry and indicates the correlation between filters.

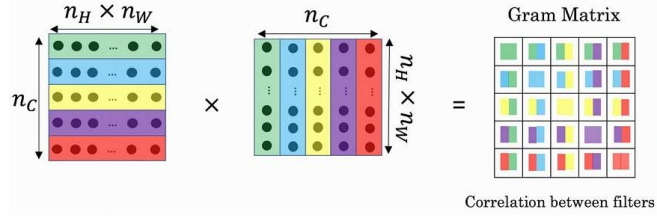

Figure 4. The Gram Matrix is created from a target image and a reference image.

The style features are given by the Gram matrix $G_l \in R^{N_l x N_l}$, in which G^l_{ij} is the inner product between the feature maps, are vectors i and j in class l:

$$G^l_{ij} = \sum_k F^l_{ik} F^l_{jk} \tag{3}$$

Let $\overrightarrow{a}$ and $\overrightarrow{x}$ is the input image and target output image, and A^l and G^l are their corresponding styles in class l. Then, the contribution of class l to the total loss is:

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} \left(G^l_{ij} - A^l_{ij}\right)^2 \tag{4}$$

And the sum of the style loss function is:

$$L_{style} = \sum_{l=0}^{L} w_l E_l \tag{5}$$

3.2.3 Synthesizing the Loss Function To convert the style of a work of art $\overrightarrow{a}$ to an image $\overrightarrow{p}$, it is needed to synthesize a new image while having a representation of its content of $\overrightarrow{p}$ and perform the style of $\overrightarrow{a}$. Therefore, this study uses the content and style loss aggregation function.

$$L_{total} = \alpha L_{content} + \beta L_{style} \tag{6}$$

in which α and β are the weights for content reproduction and style reproduction, respectively. The selection of α and β also affects the quality of the final output image.

3.3 Results

Figure 5. Applying Standard Transformation Model with the content image of Golden Bridge in Da Nang and the reference image of Candy Artwork.

In general, the standard model was able to apply the style of the given image to the content image. However, it can be seen from Figure 5 that the quality of the output image remains relatively low, although it has been through a long period of training. The color arrays were changed disorderly in the resulting image, containing noise, and its resolution was severely reduced. Not only that, on average, a transfer takes 10 to 15 minutes for 100 loops, and it increases correspondingly with the number of loops and the resolution of the input image. This is comparatively a long time to attain a possible pattern conversion when applying to the real practices.

In the upcoming section, the study proposes a new type conversion model based on the standard type conversion model to improve image generation speed and image quality.

4 Proposed Image Style Conversion Model

From the standard image style conversion model outlined above, this section proposes a new image style conversion model. The concept, architecture, and experiment results of the proposed model are presented below.

4.1 The Concept

The standard conversion model initializes the target image as white noise, then attempts to match it in a typical way of the content image and reference image, leading to a time-consuming process and low-quality output. Accordingly, the concept of this model is to propose an image conversion network that possesses the ability to learn features similar to a content image. When having an image that needs to be transformed, its features can be obtained by the model faster and more accurately, thereby reducing the conversion time to the target image, as indicated in Figure 6.

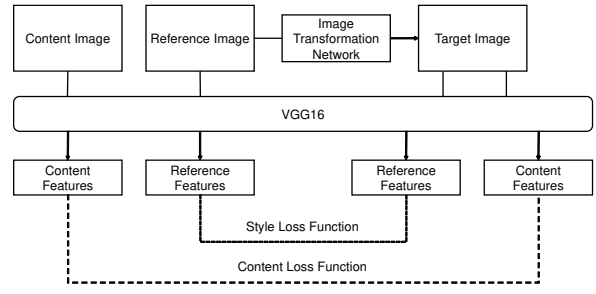

Figure 6. Proposed image style conversion model.

4.2 Model Architecture

4.2.1 Image Transformation Network The image transformation network is a convolutional neural network with the residual mass parameterized by the weights of W; it converts the input image x to the output image $\hat{y}$ through mapping $\hat{y} = f_W(x)$. Each loss function l_i ($\hat{y}$,y_i) measures the differences between output image $\hat{y}$ and input image y_i. The image transformation network was trained using stochastic gradient descent to optimize weights of the loss function:

$$W^* = \text{argmin}_W E_{x,\{y_i\}} \left[\sum_{i=1} \lambda_i l_i \left(f_W \left(x \right), y_i \right) \right] \quad (7)$$

Using the above-mentioned loss function, the following details of the convolutional neural network have been used in the proposed image transformation network, as presented in Table 1.

There are a few points of interest in this image transformation network model; the stridden convolutions downsampling and upsampling blocks in the network were used instead of pooling layers. The network body consists of five residual blocks by using the architecture in the research of Kaiming H. et al. [He et al., 2016]. All convolutional classes (no residual block) were followed by the normalization class batch and the ReLU non-linear classes, as presented in Figure 7.

4.2.2 The Function of the Image Conversion Network Layers Downsampling with strided convolutions: The first convolution in the proposed network has stride = 1, but the following two layers have stride = 2. This means that every time the filter was relocated, it shifts 2 pixels instead of 1 and the output image has the size of n/2 × n/2. With a convolution layer stride = 2, it reduced half of the input size, called downsampling. Since the input image can be sampled down; each pixel of the input results from a calculation involving the larger number of pixels from the original input image. This approach allows the kernel filters to access a much larger portion of the original input image without increasing the kernel size. Applying a certain conversion pattern to the entire image creates more information for each filter related to the original input image, making the conversion network performs better.

Upsampling with strided convolutions: Upsampling is the contrary of downsampling, which is used with stride = 2. After applying 2 layers of downsampling, the image size is reduced by 1/4 compared to the original. The desired output of the converter network is a typed image with the same resolution as the original content image. To achieve this, 2 convolution layers were applied with stride = 1/2. These layers, which increase the output image size, are called upsampling.

Residual convergence class [He et al., 2016]: Between the downsampling and upsampling classes, there are 5 residual blocks. These are first-order convolution layers, but the difference between these layers and the conventional convolution layers is that the network's input directly contributes to the output.

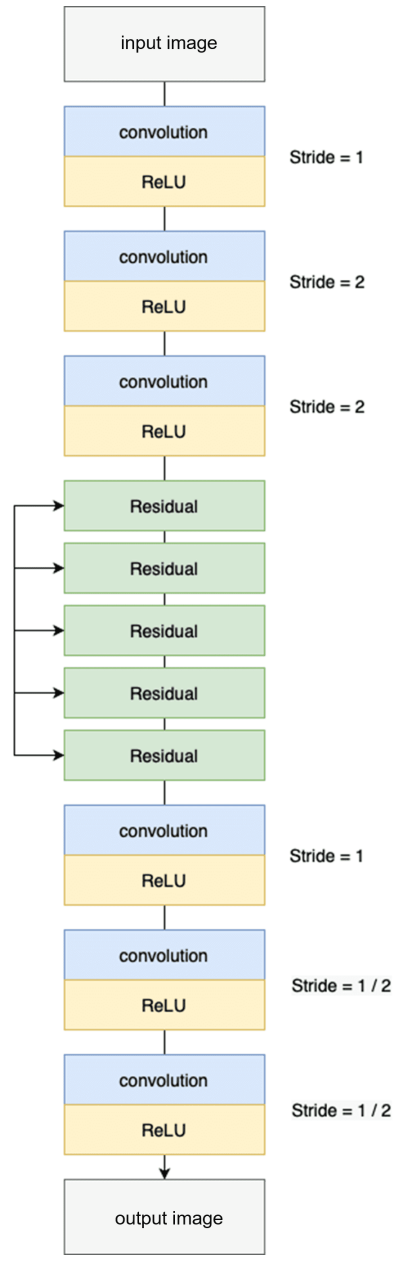

Figure 7. Image transformation network architecture.

4.2.3 Constructing Loss Function Like the standard model, the proposed model also utilizes the VGG16 pre-trained neural network [Van Hieu and Hien, 2020a; Van Hieu and Hien, 2020b] to measure the loss in the input images' content or art style. However, the content-feature reconstruction loss function was computed at class relu2_2, and the style-feature reconstruction loss function was calculated in classes relu_2, relu 2_2, relu 3_3, and relu 4_3.

Table 1. Details of proposed image transformation network.

Operation	**Kernel size**	**Stride**	**Feature maps**	**Padding**	**Activation**
Network – $256 \times 256 \times 3$ input					
Conv2d	9	1	32	ReflectionPad2d	
InstanceNorm2d			32		
Conv2d	3	2	64	ReflectionPad2d	
InstanceNorm2d			64		
Conv2d	3	2	128	ReflectionPad2d	
InstanceNorm2d			128		
Residual block			128		ReLU
Residual block			128		ReLU
Residual block			128		ReLU
Residual block			128		ReLU
Residual block			128		ReLU
Upsampling			64		
InstanceNorm2d			64		
Upsampling			32		
InstanceNorm2d			32		
Conv2d	9	1	3	ReflectionPad2d	ReLU

Feature Reconstruction Loss: Instead of forcing pixels of the output image $\hat{y}$ = f_W (x) exactly matches the pixels of the target image, they have the same feature representations as computed by the loss network ϕ. Let ϕ j (x) are the features of the j layer of the network ϕ when processing image x; if j is a convolution class, ϕ_j (x) will be a feature map of the shape $Cj \times Hj \times Wj$. The loss on feature reproduction is the Euclidean distance between the object's features:

$$l^{\phi}_{\text{feat}}(\hat{y}\,, y) = \frac{1}{C_j H_j W_j} \left\| \phi_{\,j}\,(\hat{y}) - \phi_{\,j}\,(y) \right\|^2_F \qquad (8)$$

Style Reconstruction Loss: The idea of calculating the style loss function is the comparison of the Gram matrix of the visual outputs as in the standard model.

$$G^{\phi}_{j}\,(x)_{c,c'} = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'} \qquad (9)$$

Then, the style loss function is the Frobenius standard square of the difference between the Gram matrix of the output image and the input image:

$$l^{\phi}_{\text{style}}(\hat{y}\,, y) = \left\| G^{\phi}_{j}\,(\hat{y}) - G^{\phi}_{j}\,(y) \right\|^2_F \qquad (10)$$

To perform style reproduction from a set of J classes instead of a class j, let define $l^{\phi\,,J}_{\text{style}}(\hat{y}\,, y)$ is the sum of the losses per class j ∈ J.

$$l^{\phi\,,J}_{\text{style}}(\hat{y}\,, y) = \sum_{j}^{J} l^{\phi\,,j}_{\text{style}}(\hat{y}\,, y) \qquad (11)$$

4.3 Results

Figure 8. Proposed style conversion model from the content image of Golden Bridge in Da Nang and the reference image of Candy artwork.

Figure 9. Comparison of results generated by the Standard transformation model (top) and Proposed style conversion model (bottom).

It is undoubtedly noticed from Figure 8 and 9 that the improved conversion network of the proposed model is significantly better than the standard model, in general. The improved conversion is many times faster than the basic transitions model, and small images can even be executed in real-time. The essence of artwork generated from the proposed model is also more artistic and sharper than the first model. Moreover, proposed models with a trained style through one training can apply its style for any content image in later usage.

5 Experiments and Results

5.1 Dataset Preparation and Configuration

In this study, the dataset used for implementation includes the Flirck8k dataset (8100 images, 1Gb) and the Microsoft Coco dataset (80000 images, 13Gb). The content of these image datasets is mostly scenes of diversified environments such as mountains, rivers, and trees, as illustrated in Figure 10. Given the diversity of these datasets, it was expected the model to be able to style any content image even if it has never been trained before.

Figure 10. Flickr8k Dataset.

The Flirck8k dataset was trained on Google Colab GPU Tesla K80, while the Microsoft Coco dataset was trained on a personal computer configured with Intel Core i7 - 4700MQ. The standard conversion model was configured with the loss function shown in Figure 11 and the following characteristics.

- Content_weight : $1e^3$ loss weight of content;
- Style_weight: $1e^{-2}$ loss weight of style;
- Content_image: golden_bridge.jpg the content image prepared to be styled;
- Style_image: candy.jpg reference image.

And the proposed conversion model was configured with the loss function shown in Figure 12 and the following characteristics.

- Content_weight: $1e^5$;
- Style_weight: $5e^{10}$;
- Style_image: candy.jpg;
- Learning_rate: $1e^3$ the learning rate of the neural network.

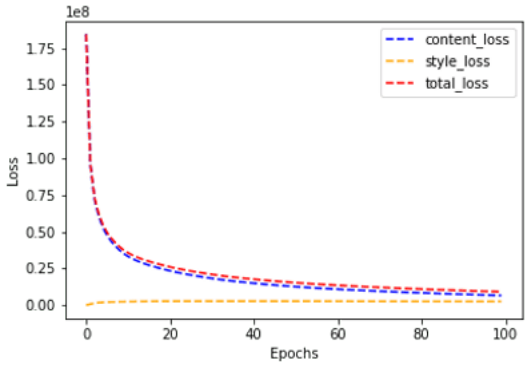

Figure 11. Loss function diagram of the standard loop-based conversion model.

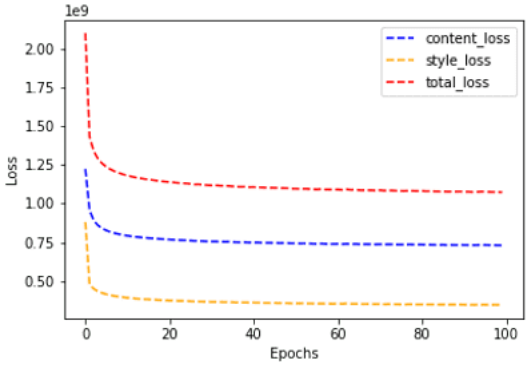

Figure 12. Loss function diagram of the proposed loop-based conversion model.

5.2 Results

Figures 13, 14 and Table 2 showcase the results and comparations of the models through different criteria, from which the advantages and disadvantages of each model will be concluded. Traditional metrics were used to evaluate output image quality, Peak signal to noise ratio (PSNR), and structural index similarity (SSIM) [Wang et al., 2004], both of which represent the human view of image quality [Hanhart et al., 2013; Huynh-Thu and Ghanbari, 2008; Sheikh et al., 2006]. The goal of these analyses was not to achieve the best PSNR or SSIM results but simply to show differences in output image quality, characteristic loss from the original image between different models trained.

5.3 Discussion

It is noticed from Figure 14 that the PSNR index of the image from the standard and proposed models is relatively close; sometimes, the standard model is higher than the proposed model (the higher the PSNR, the less interference effect). However, this index does not precisely determine the quality of the output image. Specifically for images from the standard model, the noise was calculated on the whole picture while it is measured in the picture's unimportant parts in the proposed model; therefore, the image quality of the proposed model was still better than from the standard model. Furthermore, the proposed model's image always retains the main image characteristics compared to the original image, performing much better than the standard model. This can be indicated that the SSIM of the proposed model was always higher than the standard one.

Figure 15. With the same content and reference image. The proposed style conversion model (right below) provides better quality than the standard conversion model (left below).

Figure 15 proves that with the same content and reference image, the proposed model produces a completely better image quality than the standard model, which proves the effectiveness of using additional conversion networks.

Figure 16. The proposed model was trained on the Flick8k dataset (left, bottom) and Coco dataset (the right side, below) using the content image of the homeland of Vietnam and the reference image of Candy artwork.

Whereas Figure 16 shows that with the proposed model, the model trained on a larger data set gives superior results compared to the model on the small dataset (the Microsoft Coco dataset is 10 times higher than the Flick8k dataset). This supremacy once again confirms the efficiency of using a larger image conversion network of the proposed model compared to the smaller training dataset.

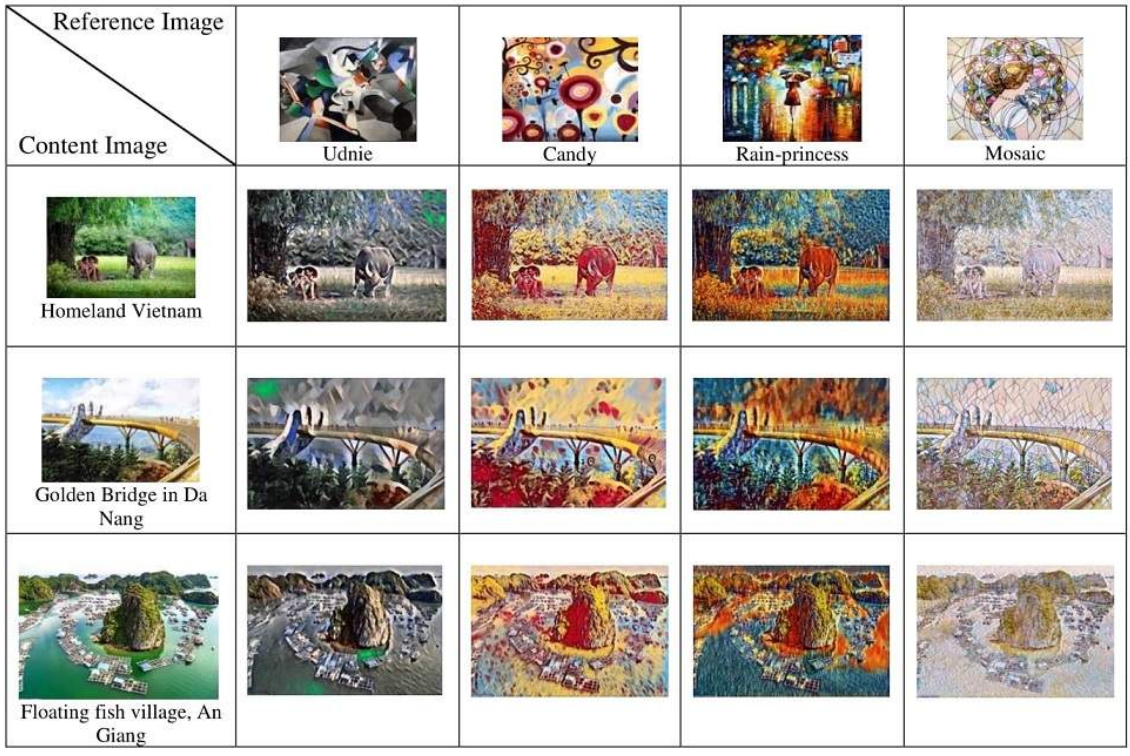

Figure 13. Some of the artwork results produced by the proposed model.

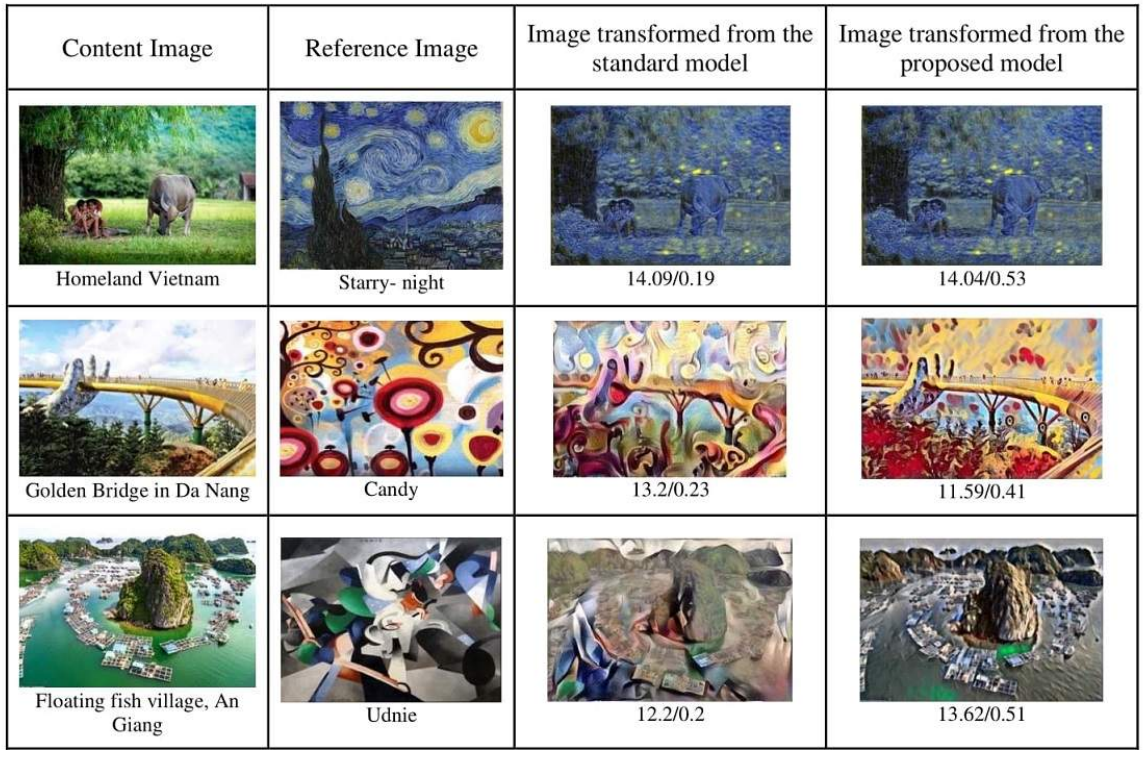

Figure 14. Comparison of PNSR / SSIM of the images generated from the models (under each image).

Table 2. Comparison of the execution times of the models.

Model / Image size (pixel)	Standard Model	Proposed Model on Flirck8k Dataset	Proposed Model on Microsoft Coco Dataset
256 × 256	5 minutes	0.6 second	0.6 second
512 × 512	15 minutes	5 seconds	5 seconds
1024 × 1024	31 minutes	30 seconds	30 seconds

5.4 Artwork Generator Website

To create exquisite artistic paintings and demonstrate the effectiveness of the prosed image transformation model that can be applied in practice, an artistic photo generator website was built with the following main functions.

– Art exhibitions: (Figure 17) View pictures, collections of famous artists, exhibition events, and related information.
– Artwork Generator: (Figure 18) Transforms images from exhibited artworks or images uploaded from user devices.

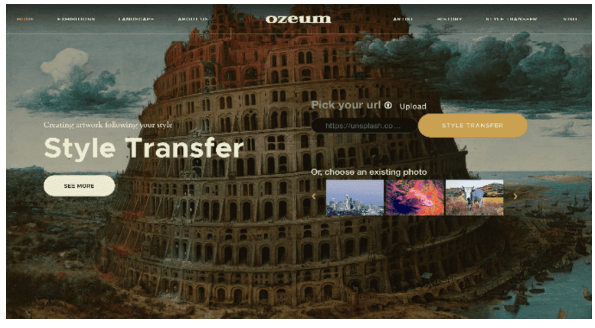

Figure 17. Layout of the artwork generator website.

Figure 18. Artistic Image Generator Website – several style-transformed images.

6 Conclusion and Future Work

In this paper, a new image conversion model was proposed and compared with a previous conversion architecture, using a convolutional neural network for transformation, pre-trained VGG16 model for measuring the loss in the input images' content. Despite substantial efforts that have been put in the past ([Huang and Belongie, 2017; Chen and Schmidt, 2016; Bourached et al., 2021]), this research introduced a high-efficiency model for image style transfer in large-scale and time-efficient focus. The proposed model generates a remarkable outcome on output image quality, performing much better than the standard model while also retaining the main image characteristics compared to the original model in both PSNR and SSIM index. Moreover, it also reduced more than 90% the execution time, which is significant for executing in real-time practice. And these study outcomes open up new avenues for future research and can play as a crucial source for future image style transfer systems. The new image conversion model can not only be applied in arts but also in various areas such as geotechnics, civil engineering, materials science, and physics.

Future work may gear towards improving applications of the model, such as the ability to convert more artwork styles in a single training session or the flexibility to adjust the degree of transformation. This is also a promising and potential image conversion model that can be applied in color grading or semantic image segmentation.

7 Acknowledgement

This research was funded and implemented for the Mercury project of Est Rouge Technologies JSC, Vietnam. This work was also supported by the People's Committee of Danang and The University of Danang, Vietnam.

References

Andreev, A. and Maksimenko, V. (2019). Synchronization in coupled neural network with inhibitory coupling. *Cybernetics and Physics*, **8** (4), pp. 199–204.

Bazhanov, P., Kotina, E., Ovsyannikov, D., and Ploskikh, V. (2018). Optimization algorithm of the velocity field determining in image processing. *Cybernetics and Physics*, **7** (4), pp. 174–181.

Bourached, A., Cann, G., Griffiths, R.-R., and Stork, D. G. (2021). Recovery of underdrawings and ghost-paintings via style transfer by deep convolutional neural networks: A digital tool for art scholars. *arXiv preprint arXiv:2101.10807*.

Chen, D., Yuan, L., Liao, J., Yu, N., and Hua, G. (2018). Stereoscopic neural style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6654–6663.

Chen, T. Q. and Schmidt, M. (2016). Fast patch-based style transfer of arbitrary style. *arXiv preprint arXiv:1612.04337*.

Cheng, M.-M., Liu, X.-C., Wang, J., Lu, S.-P., Lai, Y.-K., and Rosin, P. L. (2019). Structure-preserving neural style transfer. *IEEE Transactions on Image Processing*, **29**, pp. 909–920.

Dong, C., Loy, C. C., He, K., and Tang, X. (2015). Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, **38** (2), pp. 295–307.

Doulamis, A. and Varvarigou, T. (2012). Image analysis for artistic style identification: A powerful tool for preserving cultural heritage. *Emerging technologies in non-destructive testing V*, **71**.

Gatys, L. A., Bethge, M., Hertzmann, A., and Shechtman, E. (2016a). Preserving color in neural artistic style transfer. *arXiv preprint arXiv:1606.05897*.

Gatys, L. A., Ecker, A. S., and Bethge, M. (2016b). Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423.

Hanhart, P., Korshunov, P., and Ebrahimi, T. (2013). Benchmarking of quality metrics on ultra-high definition video sequences. In *2013 18th International Conference on Digital Signal Processing (DSP)*, IEEE, pp. 1–8.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Huang, X. and Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. pp. 1501–1510.

Huynh-Thu, Q. and Ghanbari, M. (2008). Scope of validity of psnr in image/video quality assessment. *Electronics letters*, **44** (13), pp. 800–801.

Jing, Y., Yang, Y., Feng, Z., Ye, J., Yu, Y., and Song, M. (2019). Neural style transfer: A review. *IEEE transactions on visualization and computer graphics*, **26** (11), pp. 3365–3385.

Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., and Yang, M.-H. (2017). Universal style transfer via feature transforms. *arXiv preprint arXiv:1705.08086*.

Liu, X.-C., Cheng, M.-M., Lai, Y.-K., and Rosin, P. L. (2017). Depth-aware neural style transfer. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, pp. 1–10.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. pp. 3431–3440.

Ma, W., Chen, Z., and Ji, C. (2020). Block shuffle: A method for high-resolution fast style transfer with limited memory. *IEEE Access*, **8**, pp. 158056–158066.

Noh, H., Hong, S., and Han, B. (2015). Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pp. 1520–1528.

Selim, A., Elgharib, M., and Doyle, L. (2016). Painting style transfer for head portraits using convolutional neural networks. *ACM Transactions on Graphics (ToG)*, **35** (4), pp. 1–18.

Sheikh, H. R., Sabir, M. F., and Bovik, A. C. (2006). A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on image processing*, **15** (11), pp. 3440–3451.

Van Hieu, N. and Hien, N. L. H. (2020a). Automatic plant image identification of vietnamese species using deep learning models. *International Journal of Engineering Trends and Technology*, **68** (4), pp. 25–31.

Van Hieu, N. and Hien, N. L. H. (2020b). Recognition of plant species using deep convolutional feature extraction. *International Journal on Emerging Technologies*, **11**, pp. 904–910.

Wang, Z. and Bovik, A. C. (2009). Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, **26** (1), pp. 98–117.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, **13** (4), pp. 600–612.

Yang, C.-Y., Ma, C., and Yang, M.-H. (2014). Single-image super-resolution: A benchmark. In *European conference on computer vision*, Springer, pp. 372–386.

Zhao, H.-H., Rosin, P. L., Lai, Y.-K., and Wang, Y.-N. (2020). Automatic semantic style transfer using deep convolutional neural networks and soft masks. *The Visual Computer*, **36** (7), pp. 1307–1324.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. (2015). Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1529–1537.