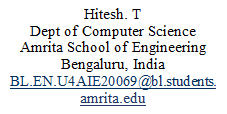
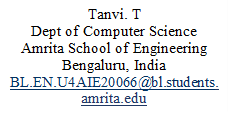
**IMAGE STYLE TRANSFER**

**USING CONVOLUTIONAL NEURAL NETWORKS**

 Text

Description automatically generated

***B. Convolutional Neural Networks***

Pictures can be modelled using convolutional neural networks (CNN), which are used to model spatial information. Because of their greater capacity to extract features from images, such as barriers and road signs, CNNs are characterised as universal non-linear function approximators. There are various algorithms which are used to transfer the texture of one image to another based on artist’s choice. Hertzman et al. use visual analogies to transfer the texture from an already styled image onto a target image. To limit the texture creation process, Freeman develop a correspondence map that incorporates target picture characteristics like image intensity.

Few of the research are limited to low-level images which can’t be used to transfer the texture at times. To render the semantic picture content of the target image in the style of the source image, a style transfer algorithm should ideally be able to extract it from the target image and then provide information to a texture transfer procedure.

In this study, we demonstrate how the content and style of natural photographs may be independently processed and altered using the generic feature representations learnt by high-performing convolutional neural networks. It is a very challenging topic to usually distinguish between substance and style in natural photographs. Deep Convolutional Neural Networks have made recent strides, and these developments have led to the development of potent computer vision systems that can train to extract high-level semantic information from real-world images.

***Abstract* - Despite having various image processing techniques and different applications based solely on Image transferring, changing the style of a particular image to another form considering the interests of users has been complicated. Portraying the style of our choice to any image given as an input is made possible by using Convolutional Neural Networks. Artistic style of a particular image can be taken out from famous paintings merged with user’s choice. Neural Algorithms based on Image style helps to portray images of perceptual quality. Convolutional neural network results show that high level image synthesis and manipulation are possible.**

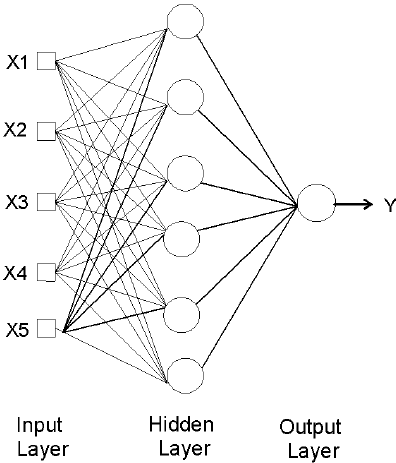
**Keywords – CNN, Image Style Transfer**

1. **INTRODUCTION**
2. ***Style Transfer***

Style transfer is a basic and widely used Computer Vision Technique in the current world. It reassembles an image's content in another image's style. Image processing has developed rapidly in the past years, many operations and processes are taking place in this specific field. There are numerous effective non-parametric texture synthesis algorithms available that can resample the pixels of a given source texture to create lifelike natural textures. To retain the semantic content of a target image, the purpose of texture transfer is to synthesise a texture from a source image while imposing restrictions on the texture creation. Pre-sets for image processing are set using picture styles. We may crop, desaturate, resize, rotate, and scale photographs using image styles.

Dr. Peeta Basi Pati  
CSE-AI  
Amrita Vishwa VidyapeetamBangalore, India  
 [bp\_peeta@blr.amrita.edu](mailto:%20bp_peeta@blr.amrita.edu)

Dr. Suja  
CSE-AI  
Amrita Vishwa VidyapeetamBangalore, India  
p\_suja@blr.amrita.edu



***Figure 4 – MLP Architecture***

**III. Training and Testing**

Create training and validation datasets, with validation being used to avoid overfitting. Using an optimization approach, divide the hyperparameters into a testing dataset after establishing an instance of the MLP Classifier. The weights are iteratively changed while assessing the data to reduce function loss between real and useful outputs. The validation dataset should be evaluated after each iteration of the MLP to check for overfitting.

Use performance indicators like accuracy, precision, recall, and F1-score to assess the MLP Classifier's effectiveness. The chosen picture dataset is divided into training, validation, and testing datasets after the necessary libraries have been imported. Accuracy is predicted using several algorithms for each dataset.A picture containing text, diagram, line, plot

Description automatically generated

***Figure 5 – Training & Testing***

Convolutional neural networks' image representations (CNN). A particular input image is represented as a group of filtered images at each processing level in the CNN. The size of the filtered images decreases as the number of distinct filters rises throughout the processing hierarchy. The style transfer method elegantly simplifies to an optimization issue inside of a single neural network because the texture model is also based on deep image representations. Fresh images are generated by performing a pre-image search to match feature representations of example images.

**II. MULTI-LAYER PERCEPTRON**

It is a type of Artificial Neural Network that consists of multiple layers of interconnected neurons. In this network, the information flows from the input to the output layer through a series of intermediate layers. An artificial neuron has the capacity to compute the weighted sum of its inputs, which is followed by the application of an activation function to produce a signal that will be sent to the following neuron.

The basic Outline of MLP architecture comprises of Input Layer, Hidden Layer, and Output Layer in which neurons in each layer are connected to neurons of another layer through weighted connections. A layer applies a activation sum to the input received from previous layer and forwards the output to next layer.

The input layer consists of neurons that receive the input data whereas the Output Layer consists of neurons that produce the output of the network

The Hidden Layer has neurons that are not physically connected to the input or output layer which may vary depending on the problem. MLP’s are basically used for learning classification and regression along with various machine learning frameworks such as PyTorch, Keras, and TensorFlow.

1. ***Fully Connected Layer***

This layer takes the flattened output from the previous layer and applies a set of weights to produce a final output. The fully connected layer is usually followed by a SoftMax activation function, which produces a probability distribution over the output classes.

The train test validation is calculated based on the input dataset provided using CNN network

A screenshot of a computer

Description automatically generated

***Figure 7 – Plots***

1. **Visualize Filters and Feature Maps in CNN**

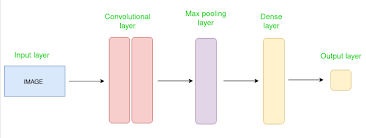
Understanding how a convolutional neural network processes the incoming data and extracts features is made easier by seeing filters and feature maps in the network. Filters in CNN are small matrices of weights that are applied to the input image to extract features. The "get weights()" layer technique lets you see the learned filters in each convolutional layer.

Convolution is the method by which a tiny number matrix is slid over the input image to create a filter. The filter multiplies the picture values that overlap with it at each place before summing the results to produce a single output value. We may create a new set of feature maps that emphasize various facets of the image by applying the filter to the entire picture. The size of the filter is typically small, such as 3x3 or 5x5, but the number of filters can be large. In fact, in modern CNN architectures, the number of filters can be in the hundreds or even thousands.

A total of 25 images are considered initially before starting the process. Training and Validation are divided in the ratio of 50:20 whereas Training and Testing are divided in the ratio of 70:30. Accuracy is predicted based on the operations performed on the original dataset.

**IV. CNN Architecture**

CNN architecture consists of multiple pooling layers and three fully connected layers. The fully connected layer is usually followed by a SoftMax activation function, which produces a probability distribution over the output classes.



1. ***Convolutional Layer***

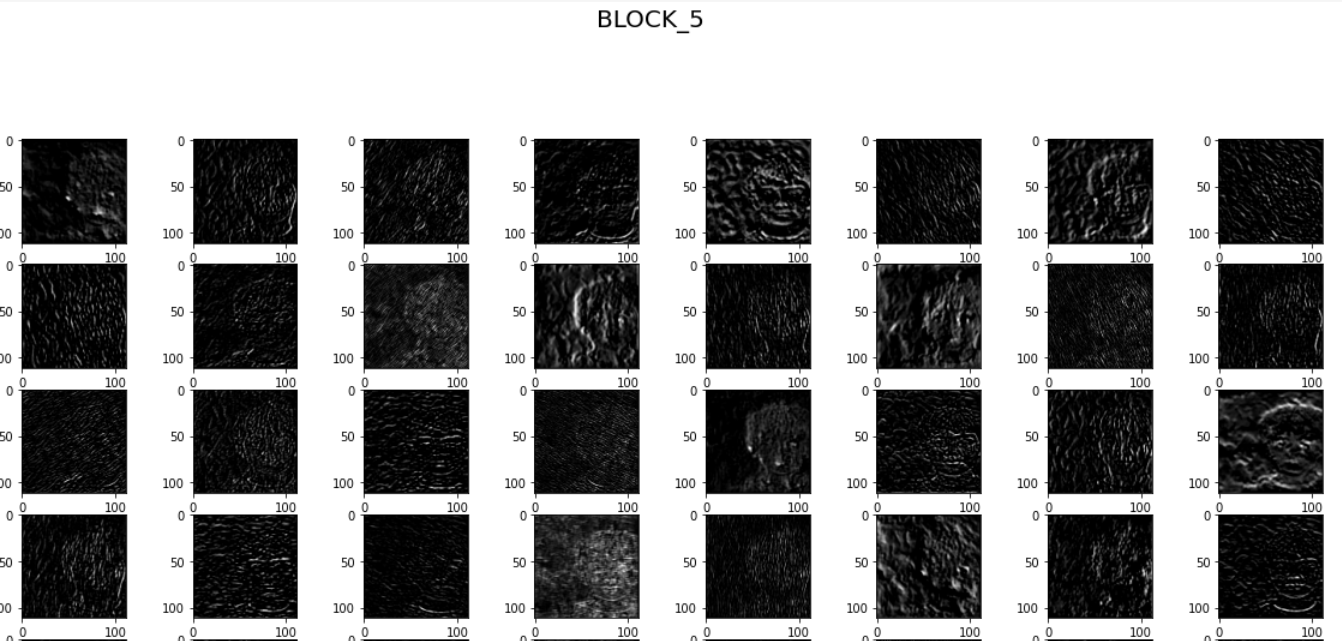
Features can be extracted by applying different filters to input images. Feature map is produced by highlighting specific patterns when each filter slides over performing a dot product between its weights and pixel values.

1. ***Activation Layer***

It introduces non-linearity by applying an activation function. The most common activation function used in CNNs is the Rectified Linear Unit (ReLU) function which sets all negative values to zero.

1. ***Pooling Layer***

This layer performs down sampling by reducing the spatial dimensions of the output from the previous layer while preserving the most essential features. The most common pooling operation used in CNNs is max pooling, which takes the maximum value in each pooling window.



***Figure 10 – Feature Extraction\_Blocks***

**2. CNN with Keras**

Keras is a high-level deep learning library that allows you to easily build and train deep learning models, including CNNs. Keras is a powerful deep learning library that can be used for various image classification tasks.

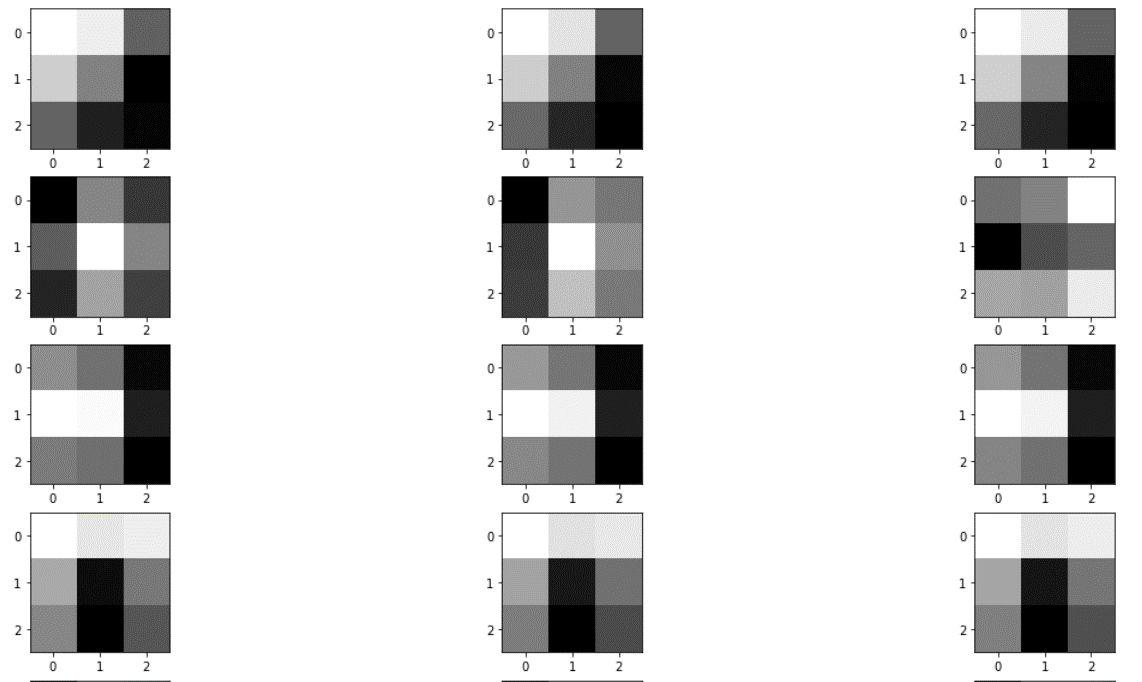
Diagram

Description automatically generated

Diagram

Description automatically generated

***Figure 11 – Keras***



***Figure 8 – Filters Visualization***

Feature maps in Convolutional Neural Networks (CNNs) refer to a convolutional layer's output. A set of features that the network has come to recognize in an input image are represented by each feature map. To convolve over the input image, the network learns a set of filters during training that is sometimes referred to as kernels or weights. Every filter pulls out a certain aspect from the image, like edges, corners, or blobs. A feature map that depicts the presence or absence of that feature in various areas of the input image is the result of this convolutional procedure.

1. **Visualize CNN with Keras**

Typically, each convolutional layer in CNN has multiple feature maps each representing a different set of learned features.

A picture containing square, screenshot, black and white, monochrome

Description automatically generated

***Figure 9 – Feature Extraction***

These feature mappings are then run through an activation function, such as ReLU, to introduce non-linearity and boost the model's representational power. Convolutional layers create feature maps, which are then input into other layers like pooling or fully connected layers for additional processing until the final prediction of the CNN is produced.

This typically involves resizing the images to a standard size, converting them to tensors, and applying normalization to match the input format used during the VGG-19 training phase. These feature maps capture the information about content and style in their respective layers. Compare the feature maps of the generated image with the feature maps of the content image. The weights control the importance of each loss component in the final total loss. Adjusting these weights can result in different stylization effects. Repeat the optimization process for a certain number of iterations or until convergence, adjusting the generated image in each iteration to better match the desired style.

**IX. RESULTS**

**A picture containing cloud, sky, outdoor, building

Description automatically generated**

***Figure 14 – Input image1***

A painting of a starry night

Description automatically generated with medium confidence

***Figure 15 – Input image2***

The CNN model's architecture needs to be specified. With the help of Keras' high-level API, you can quickly construct a deep learning model utilising layers like Conv2D, MaxPooling2D, Dropout, and Dense. Using the test data, you may assess the model's performance after training.A black text on a white background

Description automatically generated with low confidence

***Figure 12 – Accuracy***

**VIII. VGG – 19**

VGG-19 (Visual Geometry Group-19) is a convolutional neural network architecture that is commonly used for various computer vision tasks, including image style transfer. Image style transfer refers to the process of combining the style of one image with the content of another image, resulting in a new image that exhibits the content of the latter image but with the artistic style of the former. VGG-19 is a deep convolutional neural network that has 19 layers, including convolutional layers, max-pooling layers, and fully connected layers. It was originally developed for image classification tasks and has achieved remarkable performance on large-scale image recognition benchmarks, such as the ImageNet dataset.

* Deep architecture
* Pre-trained weights
* Computational efficiency
* Broadly adopted.

This model is usually trained on a large dataset like ImageNet and contains learned weights that capture useful image features. The content layers represent the intermediate feature maps in the network that capture the content of an image, while the style layers represent feature maps that encode texture and visual patterns.

The process involves selecting specific layers in the VGG-19 network to capture content and style information. The content loss measures the difference between the feature maps of the generated image and the content image, while the style loss compares the Gram matrices of the generated image and the style image. By optimizing the generated image using gradient descent or another optimization algorithm, the total loss is minimized, resulting in a stylized image that retains the content of the original image while adopting the style characteristics of the reference image.

**XII. REFERENCES**

[1] [Jin Qiu](https://ieeexplore.ieee.org/author/37089286318), [Jian Liu](https://ieeexplore.ieee.org/author/37089286126), [Yunyi Shen](https://ieeexplore.ieee.org/author/37089284839), “Computer Vision Technology Based on Deep Learning” 17-19 December 2021, [10.1109/ICIBA52610.2021.9687873](https://doi.org/10.1109/ICIBA52610.2021.9687873)

[2] Lin Wang, Kuk Jin-Yoon, “State-of-the-Art and Future Developments in Deep Learning for HDR Imaging” 29 October 2021, [10.1109/TPAMI.2021.3123686](https://doi.org/10.1109/TPAMI.2021.3123686)

[3] Shervin Minaee, Yuri Boykov, Antonio Plaza, “Image Segmentation using Deep Learning” 17 February 2022, [10.1109/TPAMI.2021.3059968](https://doi.org/10.1109/TPAMI.2021.3059968)

[4] Yannis Panagakis, Jean Kossaifi, James Oldfield, “Tensor Methods in Computer Vision and Deep Learning” 30 April 2021, [10.1109/JPROC.2021.3074329](https://doi.org/10.1109/JPROC.2021.3074329)

[5] Yanying Zhang, Xiaolin Zheng, “Development of Image Processing based on Deep Learning Algorithm” 16 April 2022, [10.1109/IPEC54454.2022.9777479](https://doi.org/10.1109/IPEC54454.2022.9777479)

[6] Dang Bich Thuy Le, Alexander Nacev, Ram Narayanan, “Deep Learning Methods for Denoising and Image Enhancement in Low-field MRI” 26 August 2021, [10.1109/IST50367.2021.9651441](https://doi.org/10.1109/IST50367.2021.9651441)

[7] Purbaditya Bhattacharya, “A Convolutional Neural Network with two channel input for Super Image Resolution” 19 July 2019, [10.1109/IJCNN.2019.8852039](https://doi.org/10.1109/IJCNN.2019.8852039)

[8] [Agnieszka Mikołajczyk](https://ieeexplore.ieee.org/author/37085897344), [Michał Grochowski](https://ieeexplore.ieee.org/author/37085704280), “Style transfer-based image synthesis as an efficient regularization technique in deep learning” 29 August 2019, [10.1109/MMAR.2019.8864616](https://doi.org/10.1109/MMAR.2019.8864616)

A cityscape with a tower in the distance

Description automatically generated with low confidence

***Figure 16 – Output image***

**X. GUI**

**A screenshot of a computer

Description automatically generated with medium confidence**

***Figure 17 – Home page***

. A screenshot of a computer

Description automatically generated with medium confidence

***Figure 18 – Result Page***

GUI is created based on the input images and the number of iterations required for the model to give accurate output. It is a user-friendly interface where the user can choose several iterations required to get an image of their satisfaction. For each iteration, it takes 9 seconds.

**XI. CONCLUSION**

By leveraging the pre-trained VGG-19 model, content and style features can be extracted from images, enabling the generation of visually appealing stylized images.

[9] [Hui-Huang Zhao](https://ieeexplore.ieee.org/author/37086501506), [Yu-Kun Lai](https://ieeexplore.ieee.org/author/38575316500), [Qin-Yun Liu](https://ieeexplore.ieee.org/author/37086884218), “Image Neural Style Transfer With Global and Local Optimization Fusion” 13 June 2019, [10.1109/ACCESS.2019.2922554](https://doi.org/10.1109/ACCESS.2019.2922554)

[10] [Yi-Sheng Liao](https://ieeexplore.ieee.org/author/37089238889), [Chun-Rong Huang](https://ieeexplore.ieee.org/author/37068864300), “Semantic Context-Aware Image Style Transfer” 10 February 2022, [10.1109/TIP.2022.3149237](https://doi.org/10.1109/TIP.2022.3149237)

[11] [Haiwen Du](https://ieeexplore.ieee.org/author/37088704546), Yu An, Lu Liu, “Disentangling Noise Patterns From Seismic Images: Noise Reduction and Style Transfer” 3 November 2022, [10.1109/TGRS.2022.3219117](https://doi.org/10.1109/TGRS.2022.3219117)

[12] [Bingyang Niu](https://ieeexplore.ieee.org/author/37089206622), [Yuxuan Ma](https://ieeexplore.ieee.org/author/37089207744), “Style Transfer of Image Target Region Based on Deep Learning” 13 June 2021, [10.1109/ISCIPT53667.2021.00104](https://doi.org/10.1109/ISCIPT53667.2021.00104)

[13] [Luoyifan Zhong](https://ieeexplore.ieee.org/author/37089435011), Na Li, Yu Sun, “Research and Application of Image Style Transfer Method” 8 May 2022, [10.1109/BigDataSecurityHPSCIDS54978.2022.00035](https://doi.org/10.1109/BigDataSecurityHPSCIDS54978.2022.00035)

[14] Yuan Liu, [Francisco Emmanuel T. Munsayac](https://ieeexplore.ieee.org/author/37086282594), “Image Style Transfer with Feature Extraction Algorithm using Deep Learning” 30 November 2021, [10.1109/HNICEM54116.2021.9731923](https://doi.org/10.1109/HNICEM54116.2021.9731923)

[15] [Jyothsna Cherapanamjeri](https://ieeexplore.ieee.org/author/37089668898), [B. Narendra Kumar Rao](https://ieeexplore.ieee.org/author/37089669232), “Neural Networks based Object Detection Techniques in Computer Vision” 21 September 2022, [10.1109/ICIRCA54612.2022.9985581](https://doi.org/10.1109/ICIRCA54612.2022.9985581)

[16] [Cameron P. Kyle-Davidson](https://ieeexplore.ieee.org/author/37086807046); [Adrian F. Clark](https://ieeexplore.ieee.org/author/37335112100), “Neural Network Components for Evolved Vision Systems” 19 September 2019, [10.1109/CEEC.2018.8674219](https://doi.org/10.1109/CEEC.2018.8674219)

[17] [Vishal Monga](https://ieeexplore.ieee.org/author/37285111300); [Yuelong Li](https://ieeexplore.ieee.org/author/37086061928); [Yonina C. Eldar](https://ieeexplore.ieee.org/author/37284203200), “Algorithm Unrolling: Interpretable, Efficient Deep Learning for Signal and Image Processing” 25 February 2021, [10.1109/MSP.2020.3016905](https://doi.org/10.1109/MSP.2020.3016905)

[18] [Wenjie Yang](https://ieeexplore.ieee.org/author/37089543540), [Shouheng Peng](https://ieeexplore.ieee.org/author/37089621432), “A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects” 10 June 2021, [10.1109/TNNLS.2021.3084827](https://doi.org/10.1109/TNNLS.2021.3084827)

[19] [Dhanesh Ramachandram](https://ieeexplore.ieee.org/author/37394572200), [Graham W. Taylor](https://ieeexplore.ieee.org/author/37085420871), “Deep Multimodal Learning: A Survey on Recent Advances and Trends” 9 November 2017, [10.1109/MSP.2017.2738401](https://doi.org/10.1109/MSP.2017.2738401)

[20] Ge Wang, “A Perspective on Deep Imaging” 3 November 2016, [10.1109/ACCESS.2016.2624938](https://doi.org/10.1109/ACCESS.2016.2624938)