Migrating of Images Using Neural Style Transfer

[1]**Ali Baig Mohammad***, Assistant Professor, ECE Dept., Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., A.P., INDIA

[2]**K Chiranjeevi Babu***, B.Tech. student, ECE Dept., Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., A.P., INDIA.

[3]**P Sainath***, B.Tech. student, ECE Dept., Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., A.P., INDIA

---***---

Abstract - *Neural Style Transfer is a process of migrating a style from one image (the Style-Image) to another (the Content Image). The goal is to synthesize a brand-new image that is a creative mixture of content and magnificence. Neural style transfer is a very neat idea. It builds on the key concept that, "it is possible to separate the style representation and content representations in a CNN, learnt during a computer vision task (e.g. image recognition task)". Following this concept, NST employs a pretrained convolution neural network (CNN) to transfer styles from a given image to another. This is done by defining a loss function that tries to minimize the difference between a content image, a style image and a generated image, so that output image looks like the content image, but "painted" in the style of the style reference image. This is implemented by optimizing the output image to match the content statistics of the content image and the style statistics of the style reference image. These statistics are extracted from the images using a convolutional network.*

Key Words: *CNN, content loss, style loss, VGG net, Gram matrix.*

1. INTRODUCTION

Convolutional Neural Networks were originally created for classification of images and have lately been used in a variety of other tasks like Image Segmentation, Neural Style and other computer vision and Natural Language Processing tasks as well. CNNs are one of the most interpretable models in Deep Learning because of our ability to visual their representations and understand what they might be learning.

Neural style transfer is one of the most creative application of convolutional neural networks. By taking a content image and a method image, the neural network can recombine the content and the style image to effectively creating an artistic image.

These algorithms are extraordinarily versatile and therefore the nearly infinite doable mixtures of content and magnificence resulted in terribly inventive and distinctive results. In fact, an organization optimized the formula and discharged a mobile application referred to as Prisma, which uses neural style transfer to apply artistic styles to pictures taken from your mobile.

1.1 WHY NST?

Deep neural networks have already surpassed human-level performance in tasks such as object recognition and detection. However, deep networks were lagging far behind in tasks like generating artistic artifacts having high perceptual quality until recent times. Creating better quality art using machine learning techniques is imperative for reaching human-like capabilities, as well as opens up a new spectrum of possibilities. And with the advancement of computer hardware as well as the proliferation of deep learning, deep learning is right now being used to create art.

1.2 USES OF NST

Common uses for NST are the creation of artificial design from pictures, for example by transferring the appearance of famous paintings to user-supplied photographs. Several notable mobile apps use NST techniques for this purpose, including Deepart and Prisma.

Also being recently used for various Data Augmentation techniques. Data Augmentation is the process of finding ways to artificially add data to small datasets to make large datasets.

"The most current kinds of image-based knowledge augmentation embrace geometric distortions like random cropping, zooming, rotation, flipping, linear intensity scaling, and elastic deformation. Whilst these are roaring at teaching rotation and scale changelessness to a model, what of color, texture and complex illumination variation?" and this might be possible by using the concept of Neural Style Transfer.

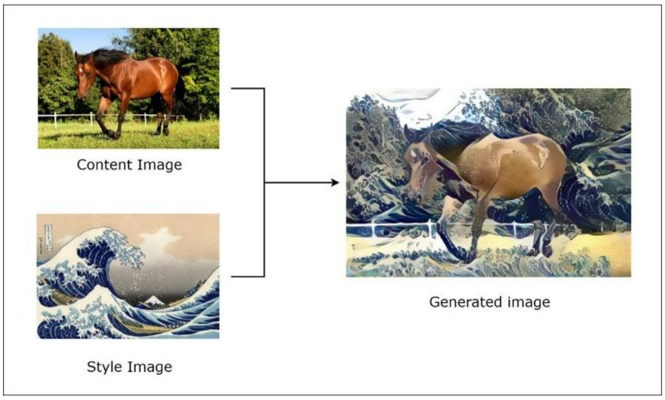


Fig1. Neural style transfer

2. Literature survey

In this paper, they have discussed the trends in current NST field and evolutionary measures to improve the quality and quantity of content and style images.[1]

Finding different ways of generating content images and being observed that CNN is obtaining better results in transferring the style one to other. CNN is better than texture-synthesis algorithm and is suitable for selected regions, visual perceptions and Quality.[2]

There was a discussion in deciding the algorithm which impacts the search results and priority in deciding the neighbors and matching of patterns with neighborhood. The approach in this method is finding the cost function and knowing the values of neighbors and generating iterations with initial value and previous results and approaches are validated.[3]

Improving the speed and processing in style and transfer in images are significant in these days. Prediction and difference between CNN or RGB images in terms of artistic comparison can be known. Patches of images can be small to mega pixels from real images to videos is maintained.[4]

It is the combination of filter banks with convolution layers and distribution of intermediate feature embedding by auto-encoder. It enables incremental learning, which improves efficiency and network-based design using texton mapping methods.[5]

The process of representing new trends and methods in Artificial Intelligence and the process of developing various approaches in learning techniques. The art of exhibiting results in different ways can be derived using the level of representation of problem.[6]

Usage of image processing techniques and other variation of up sampling signals, analytical process approach to find the filtered signals. Different ways of surveillance for monitoring the activities of human and image tools to verify the actions in the system and detect errors.[7]

Deep Dream is the first attempt to produce artistic images by reversing CNN representations with IOB-IR techniques. By further combining Visual Texture Modelling techniques to model style, IOB-NST algorithms are subsequently proposed, which build the early foundations for the field of NST. Their basic idea is to first model and extract style and content information from the corresponding style and content images, recombine them as the target representation, and then iteratively reconstruct a conventionalized result that matches the target illustration. In general, different IOBNST algorithms share the same IOB-IR technique but differ in the way they model the visual style, which is built on the aforementioned two categories of Visual Texture Modelling techniques. The common limitation of IOB-NST algorithms is that they are computationally expensive, due to the iterative image optimization procedure [8].

3. Proposed work

Content and style of images are extracted using functions and the difference of respective content and styles of images are add using a constant. In this method we implemented the combinations of content-style of different images. In order to make the outputs efficient and observe the statistics of transfer of style in generated images. NST mainly contains two images as input, first one is content image and the second is style image, the combination of both images and fixed image using CNN architecture gives Generated image.

It contains different styles of images with different parameters and content of an image are extracted by functions. Transfer of images uses machine learning techniques based on image analogy. Rendering and optimization techniques are used to create visual effects in images.

Transformation of artistic style by visual Texture modelling is being discussed. By using VGG16 architecture, CNN uses different layers to obtain different styles using content images using encoding of parts of image and Loss of image in both content and style images are observed, up to now in Neural Style Transfer there is only change in style from one picture to another. We try to auto adjust the brightness or contrast after the transfer of the style.

3.1. Convolutional Neural Networks

CNN is shown to be able to well replicate and optimize these key steps in a unified framework and learn hierarchical representations directly from raw images. If we take a convolutional neural network that has already been trained to recognize objects within images then that network will have developed some internal independent representations of the content and style contained within a given image.

Here is in fig2 of CNN hierarchy from VGG net where shallow layers learns low level features and as we go deeper into the network these convolutional layers are able to represent much larger scale features and thus have a higher-level representation of the image content.

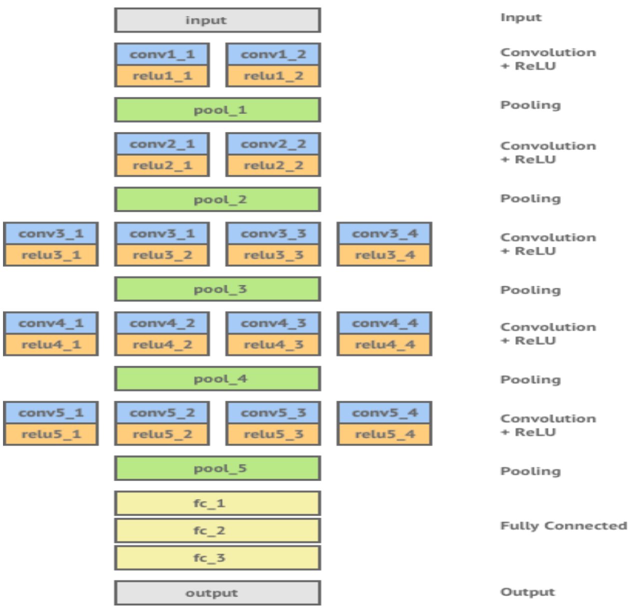


Fig 2.VGG net architectures

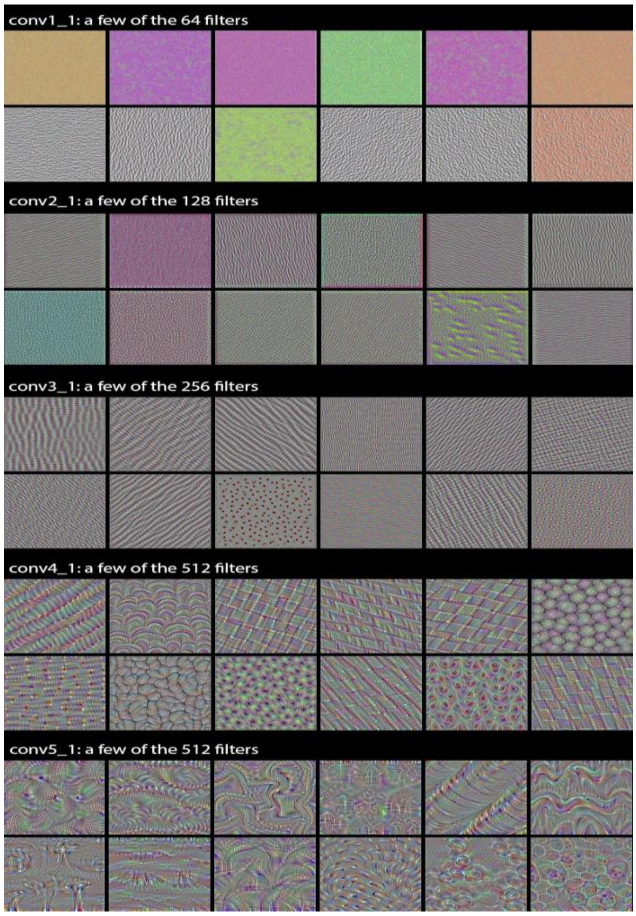


Fig 3. Features visualization of VGG network.

3.2. Loss functions

$$L_{total}(S,C,G) = \alpha L_{content}(C,G) + \beta L_{style}(S,G)$$

In this above equation there are two things we need to calculate to get overall loss i.e. content loss and style loss, alpha and beta hyperparameters which are used to provide weights to each type of loss i.e. these parameters can be thought of simply as knobs to control how much of the content/style we want to inherit in the generated image. So, let's get to understand what each of this loss term entails.

3.2.1. Content loss

Content Loss is easy to calculate, let's take the feature representation of only one of the layers, let's consider 7th convolution layer of vgg16. To calculate the content loss we pass both content image and generated image through vgg16 and get the activation values (i.e outputs) of 7th Conv layer for both of these images which has Relu for its activation, we will denote this layer's output generally as relu_3_3 because it is the output of third Conv layer of third set/block of convolutions (check figure 2 & 3 for reference). Finally, we find the L2 Norm of element-wise subtraction between these two activation matrices as This will help to preserve the original content in the generated image by making sure to minimize the difference in feature representation which logically focuses on the difference between the content of both the images.

To put this loss in mathematical kind or an equation that we will work out. Let's say we've performed Content loss that takes in 3 arguments as input that square measure content image C, generated image G and the layer L whose activation's we are going used to compute loss. Now let's denote every activation layer of a container image as a[L](C) and the activation layer of a generated image as a[L](G).

$$L_{content}(C, G, L) = \frac{1}{2}\sum_{ij}(a[L](C)_{ij} - a[L](G)_{ij})^2$$

3.2.2. Style loss

The style loss while calculating the style loss we will consider feature representation of many convolution layers from shallow to deeper layers of the model. Unlike content loss we can't just find the difference in activation units, What we need is a way to find the correlation between these activations across different channels of the same layer and to try and do this we'd like one thing known as because of the Gram Matrix.

3.2.3. Gram matrix

So let's consider that we pass our style image though vgg16 and we get the activation values from the 7th layer which generates the feature representation matrix of size 56x56x256, you can refer to figure 2 which describes the architecture of vgg16.

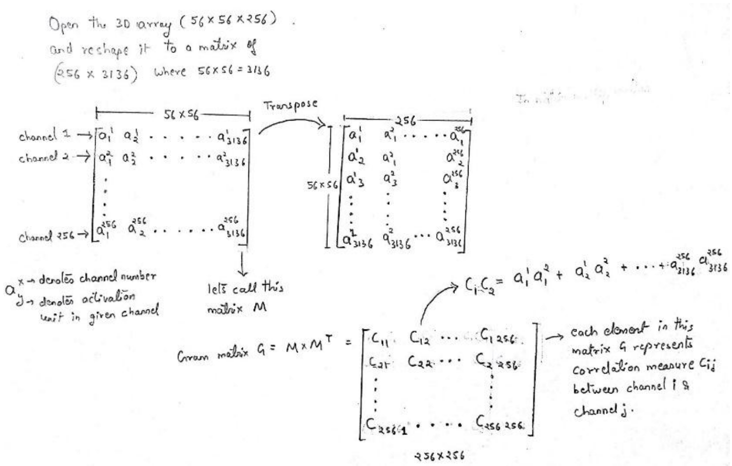


Fig 4. Gram matrix

As you can see in fig 4, how each element of this gram matrix contains a correlation measure of all the channels with respect to each other. Moving forward, how do we use this computed.

Gram matrix G to calculate the style loss. Let's denote the gram matrix of the style image of layer L as GM[L](S) and gram matrix of the generated image of the same layer as GM[L](G). Both the gram matrix were computed from the same layer hence using the same number of channel leading it to be a matrix of size ch x ch, Now if we find some of the square difference or L2_norm of element subtraction of these two matrices and try to minimize it, then this will eventually lead to minimizing the difference between the fashion of fashion image and therefore the generated image. Think about it, it might take some time to settle in but when it does, you will be mesmerized by how simple yet effective this is.

$$L_{GM}(S, G, l) = \frac{1}{4N_l^2 M_l^2}\sum_{ij}(GM[l](S)_{ij} - GM[l](G)_{ij})^2$$

In the above equation, N subscript l represents the number of channels in layer l's feature map / output, and M subscript l represents the height*width of layer l's feature map / output.

$$L_{style}(S, G) = \sum_{l=0}^{L} w_l * L_{GM}(S, G, l)$$

While we use multiple activation layers while loss of computing style, these scenarios lead us to the possibility of assigning different weights to each subloss provided by different layers. Above the formula, sums what I have just said quite elegantly, but in our case, or in most cases, people in general give equal weight to all the layers.

4. Optimizing loss function and styling the image

Using a pre-trained neural network such as VGG-19, an input image, a style image and a random image (output image), one could minimize the losses in the network such that the style loss, content and the total variation loss were at a minimum. In such cases, the output image generated from such a network, resembled the input image and had the stylist attributes of the style image.

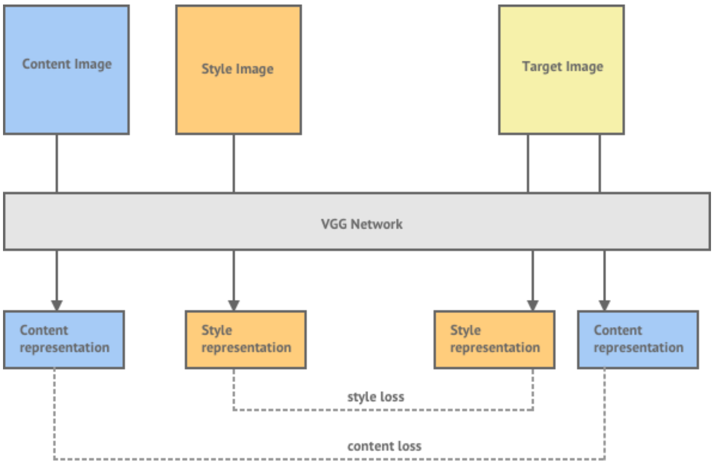


Fig 5. Loss function of style and content image

The total loss can then be written as a weighted sum of the both the style and content losses.

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{content} + \beta \mathcal{L}_{style}$$

We will minimize our total loss by Adam optimizer. As our loss go down we will go close to our goal of producing a style transfer image **Y.**

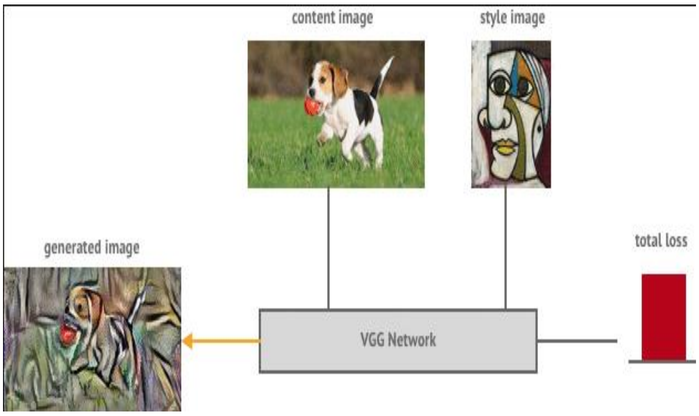
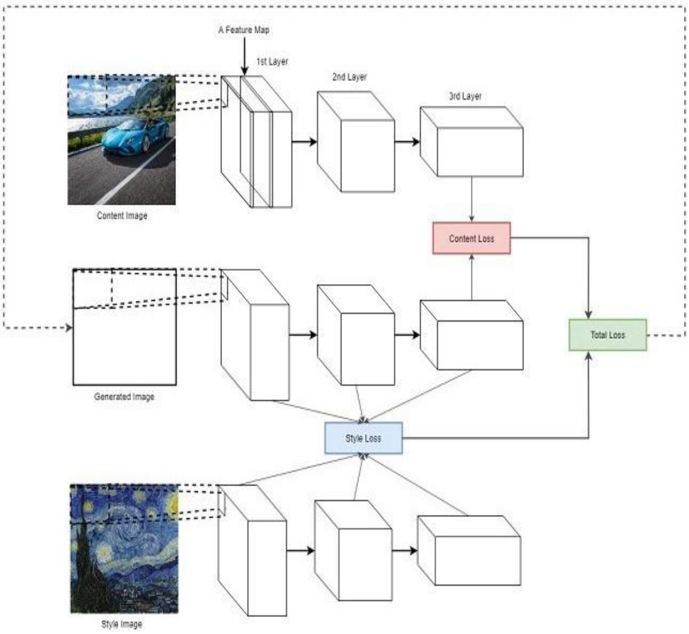


Fig 6. Generated image using VGG network

5. Block diagram



6. Results

6.1. Iteration 1:

In the 1st iteration weight Adj.

- content: 5.18e-11
- Style: 2.14e-29
- Denoise: 5.61-06

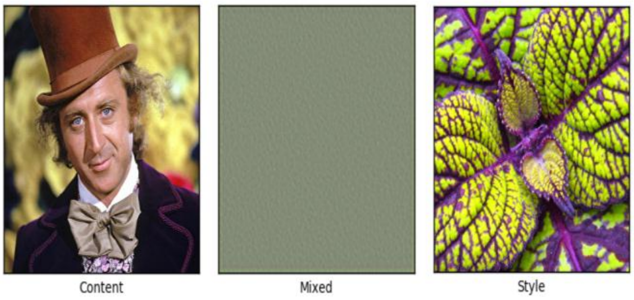


Fig 7. Output 1st Iteration

6.2. Iteration 10:

In the 10nd iteration weight Adj.

- content: 2.79e-11
- Style: 4.13e-28
- Denoise: 1.25e-07

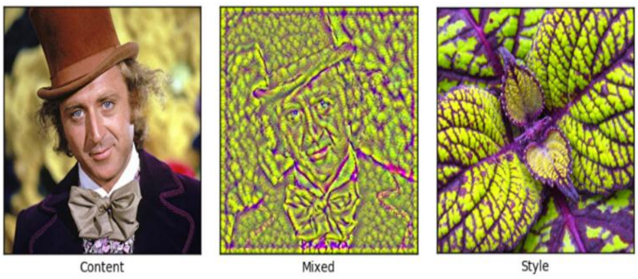


Fig 7.1. Output 10nd Iteration

6.3. Iteration 30:

In the 30th iteration weight Adj.

- content: 2.66e-11
- Style: 1.27e-27
- Denoise: 1.27e-07

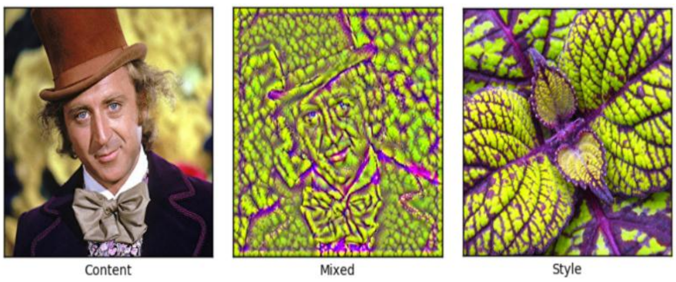


Fig 7.2. Output 30th Iteration

6.4. Iteration 60:

In the 60th iteration weight Adj.

- content: 1.85e-11
- Style: 3.86e-28
- Denoise: 1.01e-07

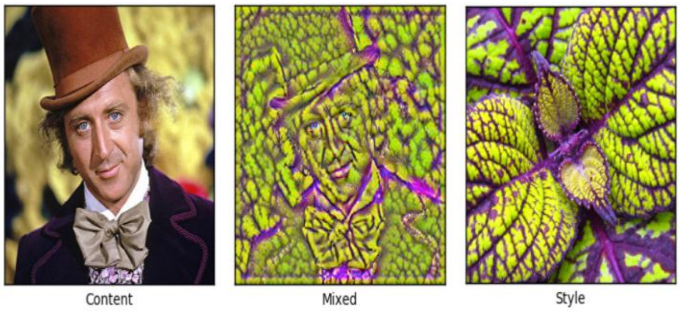


Fig 7.3. Output 60th Iteration

6.5. Final result

After 60th iteration the style from the style image mixed with content of the content image resulting the mixed image as shown below.

Fig 7.4. Final result

7. Conclusion

In this study, we propose a universal transfer algorithm of neural form, which needs no training mechanism for each individual style. By disclosing the process of image creation using an auto-encoder learning for image reconstruction. This paper discussed significant concept-based problems in the neural design transfer approaching state-of - the-art approach. We had overcome in order to show case their similar content by limiting only a single styled image and content training images.

This paper discussed significant concept-based problems in the neural design transfer approaching state-of - the-art approach. We had overcome in order to show case their similar content by restricting only a single styled image and product learning images. We had integrated the whitening and coloring transforms by passing the feed-forward to match the statistical distributions and correlations which is in between the intermediate features of content and style.NST had become an inspirational research area, influenced by both scientifically and industrial demands.

However, it is shown that the proposed method is equally effective for texture synthesis. Experimental results show that the proposed algorithm provides favorable quality in generalizing to arbitrary styles toward state-of - the-art methods. The proposed removal of style-aware content allows images and videos to be stylized in real-time, high-resolution encoder-decoder, and greatly improves stylization by recording how content is influenced by style.

8. References:

[1]. Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu and M. Song, "Neural Style Transfer: A Review," in IEEE Transactions on Visualization and Computer Graphics.

[2]. M. Elad and P. Milanfar, "Style Transfer Via Texture sis," in *IEEE Transactions on Image Processing*, vol.26, no. 5, pp. 2338-2351, May 2017.

[3]. FAST APPROXIMATE NEAREST NEIGHBORS WITH AUTOMATIC ALGORITHM CONFIGURATION Marius Muja, David G. Lowe Computer Science Department, University of British Columbia, Vancouver, B.C., Canada mariusm.

[4]. A Style-Aware Content Loss for Real-time HD Style Transfer,Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, Bjorn Ommer; The European Conference on Computer Vision (ECCV), 2018, pp. 698-714

[5].StyleBank: An Explicit Representation for Neural Image Style Transfer Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, Gang Hua; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1897-1906.

[6]. Y. Bengio, A. Courville and P. Vincent, "Representation Learning: A Review and New Perspectives," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798-1828, Aug. 2013.

[7]. H. A. Aly and E. Dubois, "Image up-sampling using total-variation regularization with a new observation model," in *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1647-1659, Oct. 2005.

[8].Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)

[9]. A Style-Aware Content Loss for Real-time HD Style Transfer Artsiom Sanakoyeu?, Dmytro Kotovenko?, Sabine Lang, and Bj¨orn Ommer Heidelberg Collaboratory for Image Processing, IWR, Heidelberg University, Germany.

[10]. L. A. Gatys, A. S. Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. In NIPS, 2015.

[11]. A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In SIGGRAPH, 2001.

[12]. S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller. Recognizing image style. In BMVC, 2014.

[13]. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science 313(5786),504–507(2006).

[14]. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. ACM Transactions on Graphics (TOG) 36(4), 107 (2017)

[15]. Jing, Y., Liu, Y., Yang, Y., Feng, Z., Yu, Y., Song, M.: Stroke controllable fast style transfer with adaptive receptive fields. arXiv preprint arXiv:1802.07101 (2018)

[16]. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European Conference on Computer Vision. pp. 702–716. Springer (2016

[17]. C. Chen, X. Tan, and K.-Y. K. Wong, "Face sketch synthesis with style transfer using pyramid column feature," in IEEE Winter Conference on Applications of Computer Vision. Lake Tahoe, USA, 2018.

[18]. I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3677–3685.

[19]. W. Zhang, S. Xiao, and X. Shi, "Low-poly style image and video processing," in Systems, Signals and Image Processing (IWSSIP), 2015 International Conference on. IEEE, 2015, pp. 97–100.

[20]. T. Gerstner, D. DeCarlo, M. Alexa, A. Finkelstein, Y. Gingold, and A. Nealen, "Pixelated image abstraction," in Proceedings of the Symposium on Non-Photorealistic Animation and Rendering. Eurographics Association, 2012, pp. 29–36.

[21]. M. He, D. Chen, J. Liao, P. V. Sander, and L. Yuan, "Deep exemplar-based colorization," ACM Transactions on Graphics (Proc. of Siggraph 2018), 2018.

[22]. A. Atapour-Abarghouei and T. Breckon, "Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[23]. I. Korshunova, W. Shi, J. Dambre, and L. Theis, “Fast face-swap using convolutional neural networks,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3677–3685

AUTHORS PROFILE

Ali Baig Mohammad obtained his M.E. in Electronics & Communication Engineering (Systems & Signal Processing) from Osmania University, Hyderabad, B.Tech. from Bapatla Engg. College, Bapatla (affiliated to Acharya Nagarjuna University). He is currently pursuing his Ph.D in Electronics and Communication Engineering at Acharya Nagarjuna University and working as an Assistant Professor in ECE Dept., KLEF deemed to be University. His research interests are Signal processing and Image processing.

K. Chiranjeevi Babu pursing 4th year B. Tech at KLEF.I am interested in design and analysis in signal processing research group.

P. Sainath pursing 4th year B. Tech at KLEF.I am interested in design and analysis in signal processing research group.