e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:01/January-2021 Impact Factor- 5.354 www.irjmets.com

CONVERSATION OF REAL IMAGES INTO CARTOONIZE IMAGE FORMAT USING GENERATIVE ADVERSARIAL NETWORK

Rudresh Joshi[*1], **Shreya Asoba**[*2]

[*1]UGC, Department of Electronics, Medicaps University, Indore, MP, India.

[*2] UGC, Department of Electronics, Medicaps University, Indore, MP, India.

ABSTRACT

Our aim is to creating animated images (Cartoon images) with the help of real images in high resolution that considered very challenging and valuable in today's computer vision and computer graphics area. The Traditional technique was quite time taking and it require an artistic mindset, but sometime it do not produce satisfactory results. Then we analysis the behavior of cartoon paintings after then, we propose three separately identify sections. (1) The surface representation (2) The structure representation (3) The texture representation. Where our model will first decompose the real image and then start learning from that decompose image. For these we are using Generative Adversarial Network (GAN) framework. Cartoon GAN will be used for the learning real–world image and extract them into Cartoonized image.

Keywords: GAN, CNN.

I. INTRODUCTION

Cartoons one of the most popular and entertaining art in the world. Cartoon is a form of 2D or 3D illustrated visual art. While the specific definition has changed over time, modern usage refers to a typically nonrealistic or semi-realistic drawing or painting intended for satire, caricature or humor or to the artistic style of such works. In the 20th century and onwards it referred to comic strips and animated films. The process of conversion of real-world image into cartoon materials of some product is called image cartoonization. As we know cartoons are artistically made it requires elegant and fine human artistic skills.

To Obtain high quality cartoonize images as same as real world Images, an artist needs to draw every single line along with each shade and color region of the image. Meanwhile exiting tools and software for editing images is not up to the mark for cartoonization. CartoonGAN is designed for image cartoonization, in which a Generative Adversarial Network framework with a loss, is proposed and reach good results in some use cases. While if we provide the training data directly to the model it will decrease its quality and stylization, result into bad cases in some scenario. To remove the above problem we have to make deep observation on artists (human) painting behavior and cartoon images of different format for image creation.

For that we perform our first identified section i.e. **Surface Representation**, it extract the smooth surface of the images. Given an image of $I \in \mathbb{R}^{W*H*3}$, we extract a weighted low frequency component $I \in \mathbb{R}^{W*H*3}$, where the color composition a0nd surface texture are preserved with edges, texture and ignore rest of the details . This method is inspired by the paintings of0 cartoon designed by cartoon artist for drawing composition drafts and tries to achieve a flexible representation for smooth surf0aces. The **Structure Representation** is method to compress the global structural information and scattered color blocks in cartoon style. The surface representation provides great importance for visual effects as well as this approach is embedded in the cartoon workflow.

The third one, **Texture Representation** where we enhance the details and edges. We simply reduce the channel from 3 to 1 I.e. the input image $I \in \mathbb{R}^{W*H*3}$converts into single channel $I \in \mathbb{R}^{W*H*1}$. The purpose behind this process is to remove luminance and color of the image for preservation of pixel intensity

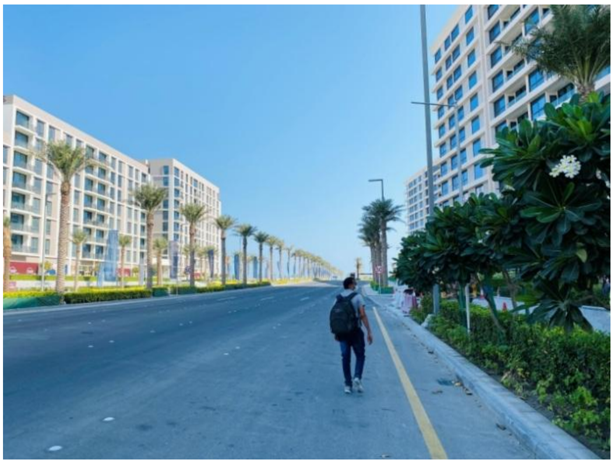


Fig.-1: Real World Image

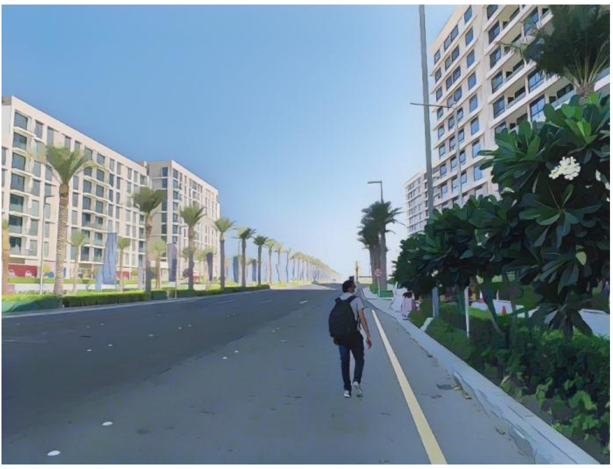


Fig.-2: GAN Model Generated Image

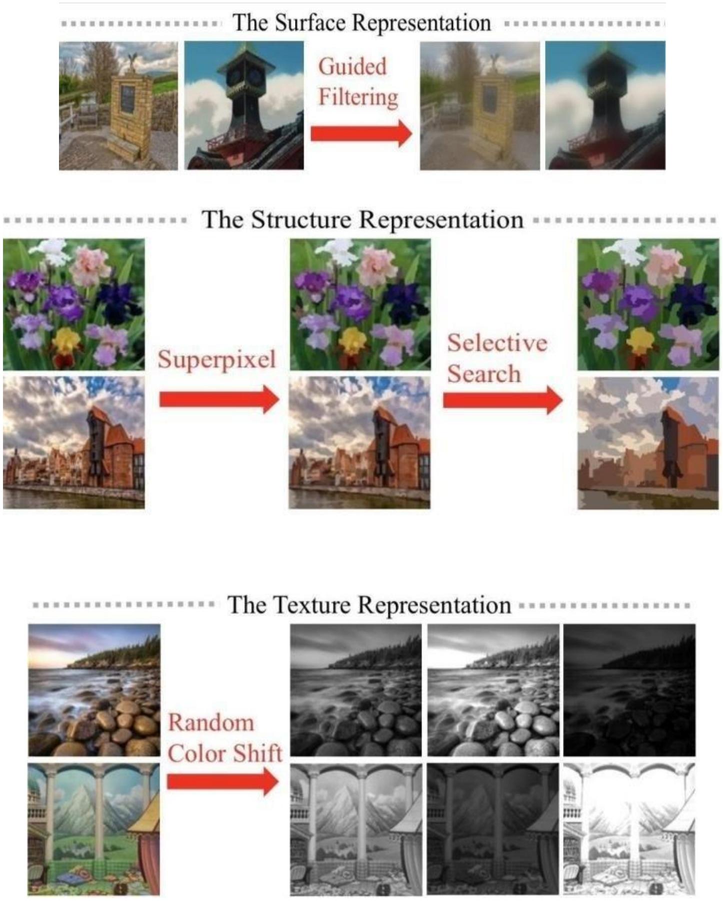

The Surface Representation
Guided Filtering
The Structure Representation
Superpixel
Selective Search
The Texture Representation
Random Color Shift

Fig.-3: Representation of Surface, Structure and Texture Extraction Process

II. LITERATURE REVIEW

2.1 Smoothing and Extraction

Image smoothing is one of the finest areas for study [12, 13, and 14]. The previous approaches are based on filtering and optimization, after then Farbman et al. [7] utilized weighted least square to constrain the edge-preserving operator. Xu and

Fan et al. [8] introduce end to end network for image smoothing. In our project work we use a differentiable guided filter to extract smooth, cartoon like surface from images, enabling our model to learn structure-level composition and smooth surface that artist have created in cartoon artwork.

2.2 Non-Photo Realistic Rendering

There are many methods for mimicking cartoon style that method uses only two ways either filter or formulation in optimization problem. However it is tough to achieve artistic styles using simple mathematical calculations and formulas. The NPR methods are mainly used in image abstraction. It simply high light semantic edges during image filtering and it extracts the details of the images. The purpose of the nonrealistic rendering is to mimic specific artistic style including cartoon. NPR algorithms are generally developed in automatic or semi-automatic. The neural style transfer method is popular among NPR algorithms.

2.3 Role of Neural Nets

Convolutional Neural Networks (CNN) is always considering as a problem solver in case of image or computer vision areas.

According to traditional style transfer algorithms, which require both style / non style images, in last few researches shows that VGG network trained for object recognition has better ability to carry out semantic features of objects and it is one of the important part of stylization. Another format is Image to Image translation where it deals with transferring image from one domain to another domain. It provide image quality enhancement, stylizing photos into paints, cartoon images and sketches. Bi- directional models are proposed for inter domain translation before few days. Zhu et. Al performs transformation of Rain to winter and sketch to paint of unpaired images.

2.4 Use of Generative Adversarial Network

Generative Adversarial Network is types of neural networks capable of generating new data that confirms to learned patterns. It basically consists of two parts generator and discriminator, generator trained to produce output that manipulates the discriminator where discriminator which classifies the created image is real or not. Generative Adversarial Network is very powerful in image synthesis by managing the generated image to be varied from the real image.

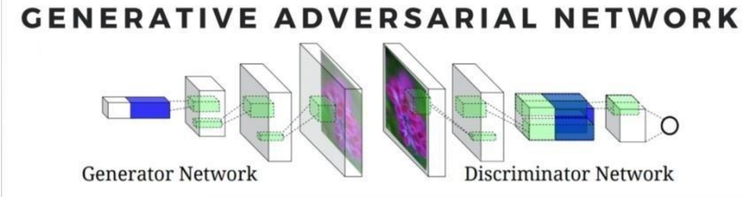
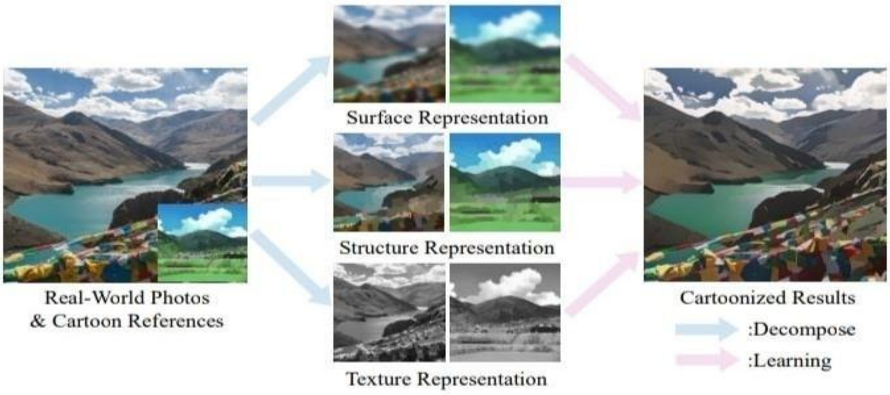


Fig.-4: Network Architecture of Generative Adversial Network, specifying generator and de-generator

III. METHODOLOGY



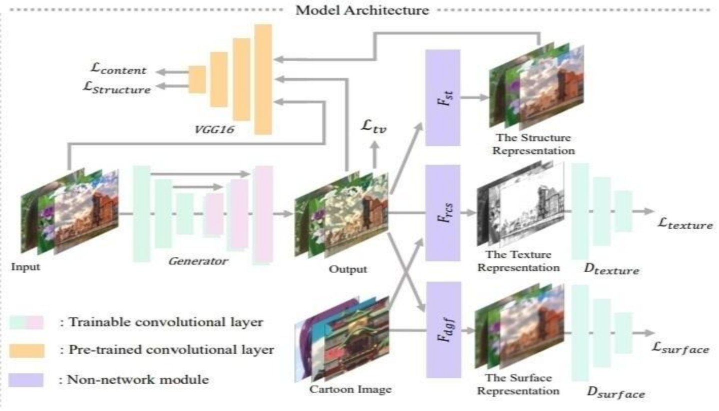


Fig.-5: Model Architecture

Our approach is to decompose the image into three individual modules i.e. surface representation, structure representation and texture representation as shown in the above figures (3, 4, and 5) and they are used to extract corresponding information. Generally GAN having two sections generator G and discriminator where in our model we use two discriminators Ds and D2, Where D1 is for surface discrimination to extract surface details from the image and D2 is for texture discrimination to extract texture details from the image. For extracting the high resolution features and extracts global content using structure representation. Adjustment of weight is done by loss function; using these we can control the output image style.

3.1 Role of Surface Representation in Model:

In surface representation we simple apply the approach of artist. Where artist draw a rough draft and have smooth surface. Our purpose behind the smoothing image is to preserve the global semantic and differentiable guided filter is used for preserving edges of the image and it is represented by $\mathcal{F}_{dgf}$. It requires an input image **I** and the output of the filter will provide us surface representation $\mathcal{F}$(Ic,Ip)With guided map and remove details from the image.

For cross check we apply discriminator D1, it will check the output of the model and reference cartoon images and make sure that they have similar surface and according to that instruct the generator to learn information which is stored in the extracted surface representation. So according to above theory let represent the input image and represent the cartoon image, so the loss formula will be:

$$Ls(G, Ds) = logDs(\mathcal{F}_{dgf} (Ic, Ic)) + log(1 - Ds(\mathcal{F}_{dgf} (G(Ip), G(Ip)))) \qquad (1)$$

3.2 Role of Structure Representation in Model:

It provides the sparse color blocks and clear boundaries in cartoon workflow. For segmentation of image we use felzenswalb algorithm and Super Pixel algorithm is used for similarity of pixel and it will ignore the semantic information of the image. For merging of the entire region in to one single format we use selective search. We found this lower global contrast by analyzing the processed data set; darken images and causes hazing effect on the final results. Thus, we propose an adaptive coloring algorithm, and formulate it in Equation 2, where we find γ1 = 20, γ2= 40 and μ = 1.2 generate good results.

$$Si,j = (\theta1 \, * \, S^{-} + \theta2 \, * \, S^{\sim})^{\mu} \qquad (2)$$

For high level feature extraction we apply VGG16 per trained network to enforce spatial constrain between our result and extracted structure. Now $\mathcal{F}_{st}$ represent the structure extracted image, so the loss formula for structure representation is

$$Lst = || VGGn(G(Ip)) - VGGn(Fst(G(Ip))) || \qquad (3)$$

3.3 Role of Textural Representation:

The key learning objective of cartoon images are high frequency, but the light, color and contrast distinguish it from real image. We use random color shift algorithm $\mathcal{F}_{rcs}$ to extract the single channel from the color images.

$$Frcs(Irgb) = (1-\alpha)(\beta1*Ir+\beta2*Ig+\beta3*Ib)+\alpha*Y \quad (4)$$

In Equation 4, represents 3-channel RGB color images, and represent three color channels, and Y represents standard grayscale image converted from RGB color image. We set = 0.8, 1, 2 and 3~U (-1, 1). the random color shift can achieve random intensity maps along with removal of luminance and color information. A discriminator is proposed to find out the difference between texture representations extracted from model outputs and cartoons, and guide the generator to learn the lucid contours and fine textures stored in the texture representations.

$$Lt(G, Dt) = logDt(Frcs(Ic)) + log(1 - Dt(Frcs(G(Ip)))) \quad (5)$$

IV. ENVIRONMENTAL SETUP

We implement our project using GAN framework with tensorflow and all experiments were performed on Google colab GPU. Hyper-parameters and all results shown in this paper, except specially mentioned are proposed with $\lambda1 = 1$, $\lambda2 = 10$, $\lambda3 = 2 * 10^2$, $\lambda4 = 2 * 10^3$, $\lambda5 = 10^4$. Having batch size of 20 at the time of training and learning rate is set to be 2.3 *10−4. We use Adam Optimization algorithm to optimize our networks. As we are using GAN approach so we first pre- train the generator with loss of approx 60000 iterations and then optimize according to generative adversarial approach. After these the training of the model is stopped on convergences. Our model is data driven so the neural network is easily learn, even large number of parameter defined. In training dataset we created a collection of real world images and cartoon images and in the test data set we only collect real World images. It contains human face and landscape images for generalization. We collect 12,000 human face image from FFHQ dataset and 8,000 images of landscape. We collect 10,000 images from animations for the human face for cartoon images and 10,000 images for landscape. During training, every images are resized to 256*256 resolution, and face Images are feed only once in every five iterations.

We compare our model with four different algorithms that represent Image-to-Image Translation [10], Neural Style Transfer [9], Image Abstraction [3] and Image Cartoonization [11] respectively.

V. RESULTS AND DISCUSSION

The time of execution of our model is quite efficient than other four methods which are describe early in environment setup and are experimented on different hardware using different images in terms of quality. As shown in the Table 1 where LQ means 256*256 and HQ means 720*1280 size of image. Our model perform very well on HQ image the execution time on GPU devices is 18.22 ms which is very helpful in real time video cartoonization task.

Table-1: Comparison of Performance evolution metrics on different device

Methods	[17]	[11]	[16]
LQ, CPU(ms)	640.3	1950	1332.50
LQ, GPU(ms)	16.55	14.02	8.99
HQ, GPU(ms)	49.22	147.96	105.8
PARAMETERS(M)	1.78	149.5	11.4

VI. CONCLUSION

In this paper we perform transformation of high definition real world images in to high definition Cartoon images using generative adversial network .Our main goal is to recreating exact characteristics of the real image in to cartoon images.

Where we use three modules to extract different – different information from the input image for model training and controlled output image using hyper parameters and loss functions. Loss function also provides fine edges to the images.

VII. REFERENCES

[1] Y. Chen, Y.-K. Lai, and Y.-J. Liu. Transforming photosto comics using convolutional neural networks. In International Conference on Image Processing, 2017.

[2] Yijun Li, Chen Fang, Aaron Hertzmann, Eli Shechtman, and Ming-Hsuan Yang. Im2pencil: Controllable pencil illustration from photographs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1525–1534, 2019.

[3] Tero Karras, Samuli Laine, and Timo Aila. Style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4401–4410, 2019.

[4] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image to image translation using cycle-consistent adversarial networks," arXiv preprint arXiv:1703.10593, 2017.

[5] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In Proceedings of the European Conference on Computer Vision (ECCV), pages 172–189, 2018.

[6] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Wespe: weakly supervised photo enhancer for digital cameras. In Proceedingsof the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 691–700, 2018

[7] Zeev Farbman, Raanan Fattal, Dani Lischinski, and Richard Szeliski. Edge-preserving decompositions for multi-scale tone and detail manipulation. In ACM Transactions on Graphics (TOG), volume 27, page 67. ACM, 2008.

[8] Qingnan Fan, Jiaolong Yang, David Wipf, Baoquan Chen, and Xin Tong. Image smoothing via unsupervised learning. In SIGGRAPH Asia 2018 Technical Papers, page 259. ACM, 2018.

[9] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision, pages 694–711. Springer, 2016.

[10] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cleconsistent adversarial networks. In Proceedings of IEEE International Conference on Computer Vision, 2017.

[11] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. In European Conference on Computer Vision, pages 1–14. Springer, 2010

[12] C. Ledig, L. Theis, F. Husz´ar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017

[13] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle consistent adversarial networks. In Proceedings of IEEE International Conference on Computer Vision, 2017

[14] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision, pages 694–711. Springer, 2016.

[15] Dongbo Min, Sunghwan Choi, Jiangbo Lu, Bumsub Ham, Kwanghoon Sohn, and Minh N Do. Fast global image smoothing based on weighted least squares. IEEE Transactions on Image Processing, 23(12):5638–5653, 2014