An Intelligent approach to Pic to Cartoon Conversion using White-box-cartoonization

Vidya Kudale[1], Harshada Sable[2], Kajal Bankar[3], Pravin Shingade[4]

[1,2,3] Student, Dept. of Computer Engineering, SVPMCOE Baramati, Maharashtra, India
[4]Student, Dept. of Computer Engineering, SVPMCOE Baramati, Maharashtra, India

---

Abstract - *Recent years have witnessed increasing attention in cartoon media, influenced by the strong demand of business application. This paper presents an intelligent approach towards pic to cartoon conversion. Have observing the cartoon painting behavior, this paper proposes application based on one by one establish three completely different representations from images: the surface illustration that contains sleek surface of cartoon pictures, the structure representation that refers to the thin color-blocks and flatten surface content within the celluloid vogue advancement, and the last be texture illustration that reflects high-frequency texture, contours and details in cartooned pictures. And finally, A Generative Adversarial Network (GAN) framework is employed to be told the extracted representations and to cantoonize images.*

Key Words: **cartoonization, gan, structure, surface, texture, white-box-cartoonizatio**n

1. INTRODUCTION

Social media is extensively used these days. And standing out in this online crowd has always been a to-do on every user's list on these social media platforms. Be it images, blog posts, artwork, tweets, memes, opinions and what not being used to seek attention of followers or friends to create influence or to connect with them on such social platforms.

We aim to provide one such creative solution to their needs, which is applying cartoon like effects to their images. Users can later share these images on any social media platforms, messengers, keep it for themselves, share it with loved ones or do whatever they like with it. Nowadays almost everyone is registered in social networks.

A Generative Adversarial Network (GAN) framework is employed to be told the extracted representations and to cantonize images. The educational objectives of our technique as one by one supported every extracted representations, creating our framework manageable and adjustable. This allows our approach to fulfill artists' needs in numerous designs and numerous use cases. Qualitative comparisons and quantitative analyses, moreover as user studies, have been conducted to validate the effectiveness of this approach, and our technique outperforms previous ways all told comparisons. Finally, the ablation study demonstrates the influence of every element in white-box framework.

We have created an application based on this framework made cartoonization happened in smooth way, within few seconds on single key. You may choose image from your device or live capture one.

2. Related works

Following are few remarkable works mentioned which are related in this area. All have shown difference in performance and remarkable solutions.

2.1 Cartoon-GAN:

Cartoogan is GAN i.e. Generative Adversarial Networks based Photo cartoonization method. However, existing methods do not produce satisfactory results for cartoonization, due to the fact that firstly, cartoon styles have unique characteristics with high level simplification and abstraction, and Secondly, cartoon images tend to have clear edges, smooth color shading and relatively simple textures, which exhibit significant challenges for texture-descriptor-based loss functions used in existing methods.

This method takes unpaired photos and cartoon images for training, which is easy to use. Two novel losses appropriate for cartoonization area unit proposed: (1) a semantic content loss, that is developed as a thin regularization within the high-level feature maps of the VGG network to address substantial vogue variation between photos and cartoons, and (2) an edge-promoting adversarial loss for conserving clear edges. we tend to more introduce an initialisation part, to boost the convergence of the network to the target manifold. The proposed methodology is additionally way more economical to coach than existing strategies. Experimental results show that our methodology is in a position to get high-quality cartoon pictures from real-world photos (i.e., following specific artists' designs and with clear edges and swish shading) and outperforms progressive strategies.

2.2 SGAN-based Multi-Style Photo Cartoonization

This research proposes a multi-style generative adversarial network (GAN) architecture, called MS-

CartoonGAN, that can transform pictures into multiple cartoon styles. They have developed a multi-domain architecture, where the generator consists of a shared encoder and multiple decoders for different cartoon styles, along with multiple discriminators for each and every individual style.

2.3 Stylized Neural Painting

This paper proposes an image to painting transformation methodology that generates vivid and realistic painting artworks with controllable styles. Different from previous image to image translation methods that formulate the translation as pixel wise prediction, proposed method deals with such an artistic creation process in a vectorized environment and produce a sequence of physically meaningful stroke parameters that can be further used for rendering. Since a typical vector render is not differentiable, it design a new neural renderer which imitates the behavior of the vector renderer and then frame the stroke prediction as a parameter searching process that maximizes the similarity between the input and the rendering output.

2.4 Image-to-Image Translation Using Generative Adversarial Network

Image to Image translation is one of the application of GANs as a data augmentation which we have used in this proposed framework. Generative Networks makes the mapping between source image and target image easier and it calculates the loss function also to improve the quality of generated target image. In this paper, Conditional GANs are used which translates the images based upon some conditions. The performance is also analyzed of the model by doing hyper-parameter tuning[18].

2.5 Image Cartoonization

This paper presents an approach to cartoonize digital images into cartoons. The proposed method may differ from that previously used. This paper focuses on various techniques involved during the whole process, which, when used layer by layer, gives a suitable balanced output.

There are four filters applied: - 1. Pencil Sketch – Coverts the contents of an image as if it were drawn from a pencil. 2. Detailed enhancement – Sharpens the image, adds detailed noise in the image to improve details. 3. Bilateral filter – Smoothen the image by removing noise while keeping the edges sharp. 4. Pencil edge – Converts the image into one which has only significant edges and fills the insides with white color[19].

3. Approach

We have created a web-application that can capture image as well as can take input image through device storage files and applying cartoonifying effects give cartooned image output. Images are decomposed into the surface representation, the structure representation, and the texture representations, and three independent modules are introduced to extract corresponding representations. A generative adversarial networks with G generator and, D_s and D_t as two discriminators, where aim of discriminator D_s is to differ between surface representation extracted from model outputs and cartoon pictures, and discriminator D_t is used to differ among texture representation extracted from model outputs and cartoon images. Pretrained VGG-network is used to extract features and to impose spatial constrains on global elements between input images and output images. Weight for each element can be adjusted in the loss function, which allows users to control the output styles and adapt the model to diverse use cases.

3.1 GAN Model

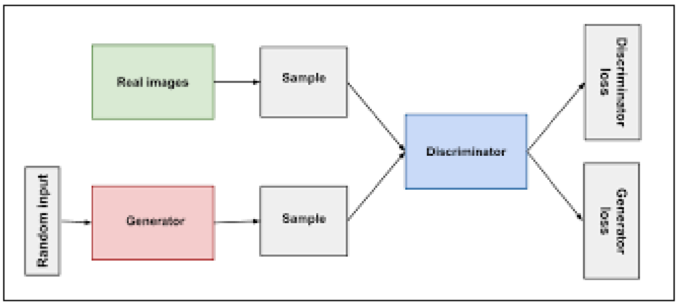

Fig-1: GAN Model Architecture

Generative Adversarial Network(GAN) is a state-of-the-art generative model that can generate data with the same distribution of input data by solving a min-max problem between a generator network and a discriminator network. It is powerful in image synthesis by forcing the generated images to be indistinguishable from real images. GAN has been widely used in conditional image generation tasks, such as image inpainting, style transfer, image cartoonization, image colorization. In proposed method, we have adopted adversarial training architecture and use two discriminators to enforce the generator network to synthesize images with the same distribution as the target domain.

3.2 Learning Through the Surface Representation

The surface illustration imitates cartoon painting style wherever artists roughly draw drafts with coarse brushes and have sleek surfaces quite like cartoon pictures. To smooth pictures and in the meantime keep the worldwide semantic structure, a differentiable guided filter is adopted for edge conserving filtering. Denoted as F_{dgf} , it takes associate image I as input and itself as guide map, returns extracted surface illustration F_{dgf} (I_c, I_c) with textures and details removed. A discriminator Ds is introduced to judge whether model outputs and reference cartoon pictures have similar surfaces, and guide the generator G to seek out data stored within extracted surface representation. Let scientific discipline denote the input picture and I_c denote the reference cartoon images, we have a tendency to formulate the surface loss as:

$$\mathbf{L}_{surface}(G,D_s)= \log D_s(F_{dgf}(I_c,I_c))+\log(1D_s(F_{dgf}(G(I_p),G(I_p))))$$

3.3 Learning Through the Structure Representation

The Structure illustration emulates flattened global content, thin color blocks, and clear boundaries in celluloid style cartoon progress. we have a tendency to initially use Felzenszwalb algorithm to segmented pictures into separate regions. As superpixel algorithms solely considered the similarity of pixels and ignore semantic data, we have a tendency to introduce selective search to merge each regions and extract a thin segmentation map. Standard super-pixel algorithms color each region with an mean of each pixel value . By analyzing the processed dataset, we have a tendency to found this lowers global contrast, darkens pictures, and causes hazing result on the ultimate results . we have a tendency to propose an adaptive coloring algorithm, , wherever we discover 1 = 20, 2 = 40 and μ = one.2 generate sensible results. The colored segmentation maps and also the final results trained with adaptive coloring, this resultively enhances the contrast of pictures and reduces hazing effect.

$$S_{i,j} = (\theta_1 * S+\theta_2 * \tilde{S})\ \mu$$

$$(\theta 1,\theta 2) = \begin{cases} (0,1)\ \sigma(S) < \gamma 1,\ (0.5,0.5) \\ \gamma 1 < \sigma(S) < \gamma 2,\ (1,0) \\ \gamma 2 < \sigma(S) \end{cases}$$

We use high-level options extracted by pre-trained VGG16 network to enforce spatial constrain between our results and extracted structure illustration. Let us, First denote the structure illustration extraction, the structure loss $L_{structure}$ is developed as:

$$\mathbf{L}_{structure} = ||VGG_n(G(I_p))\ VGG_n(F_{st}(G(I_p)))||$$

3.4 Learning Through the Texture Representation

The high-frequency feature of cartoon pictures are key learning objectives, but luminance and color data build it simple to differentiate between cartoon pictures and real-world photos. We propose a random color shift algorithmic F_{rcs} to extract single texture illustration from color pictures, that retains high-frequency textures and reduces the influence of color and brightness.

$$F_{rcs}(Irgb) = (1-\alpha)(\beta_1 * I_r +\beta_2 * I_g +\beta_3 * I_b) + \alpha *Y$$

In above equation , I_{rgb} represents three-channel RGB color pictures, I_r, I_g, and I_b represent three color channels, and Y represents common grayscale image regenerate from RGB color image. We set = 0.8, 1,2 and 3, U(1, 1). As is shown in Fig-2 , the random color shift will generate random intensity maps with brightness and color information removed. A discriminator D_t is introduced to differentiate texture representations extracted from model outputs and cartoons, and guide the generator to learn the clear contours and fine textures stored within the texture representations.

$$L_{texture}(G,D_t) = \log D_t(F_{rcs}(I_c)) +\log(1-D_t(F_{rcs}(G(I_p))))$$

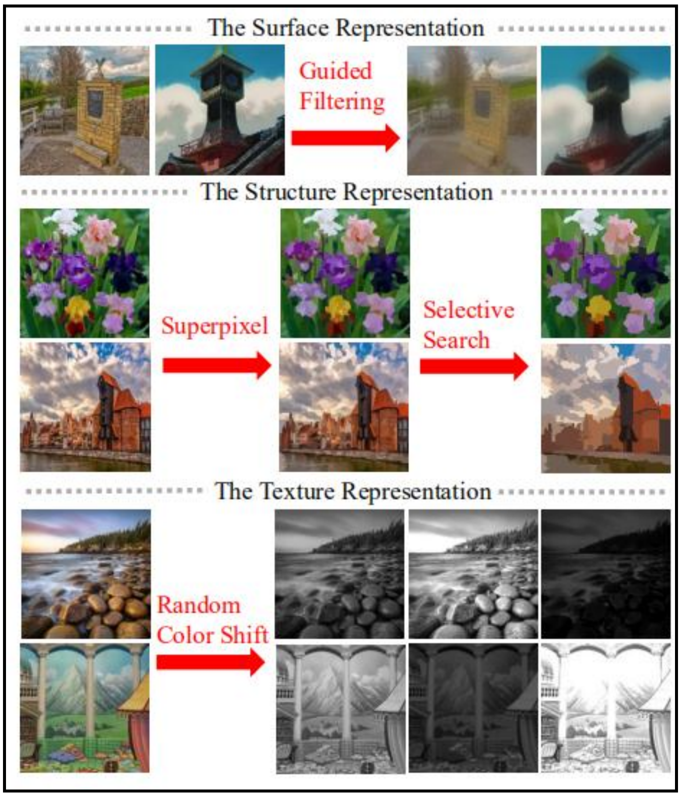

Fig- 2: The Surface Representation, The Structure Representation and Texture Representation

3.5 Flask Application

We created a web based application for performing cartoonization using white box framework. Flask python framework comes in handy for easy deployment of website design. The task of application is to provide a user interface to capture image or let user select image from device storage and performing cartoonization process on it. After completing the process flask will render html page with cartoonized image shown on it and will also allow user to download it. This web-app is user friendly and does not required to provide any user credentials to use it i.e. thers no need for login or sign up.

4. Flow Chart of System:

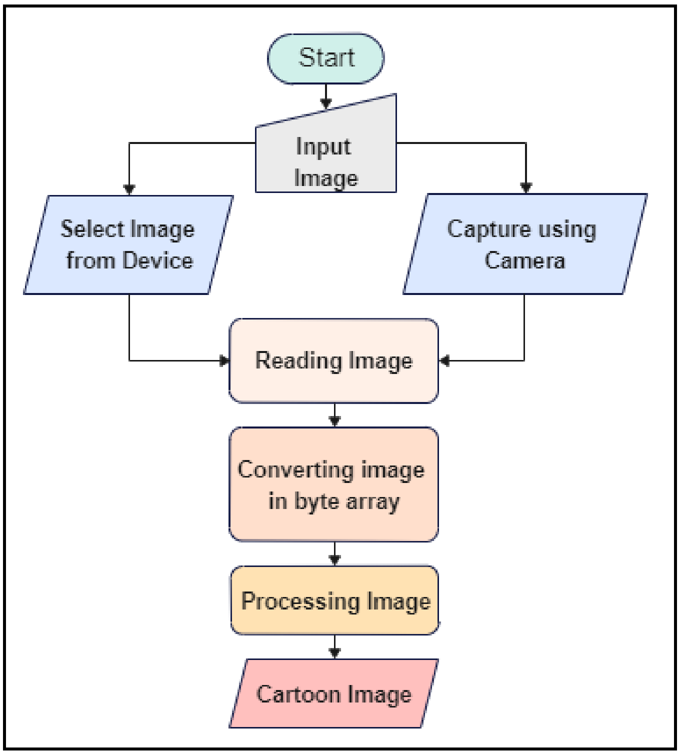

Fig- 3: Flow chart of Our System

5. OUTPUT

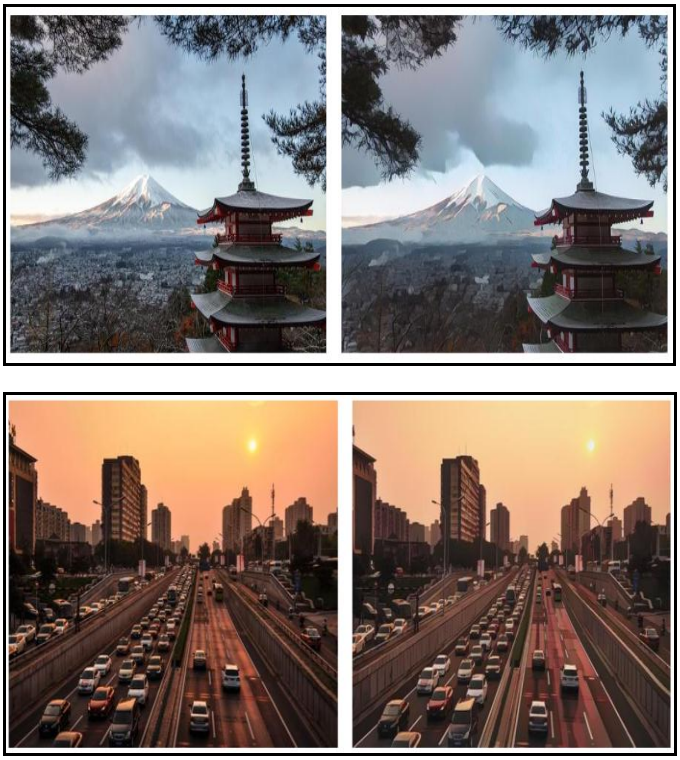

Fig- 4: Original vs Cartooned Output

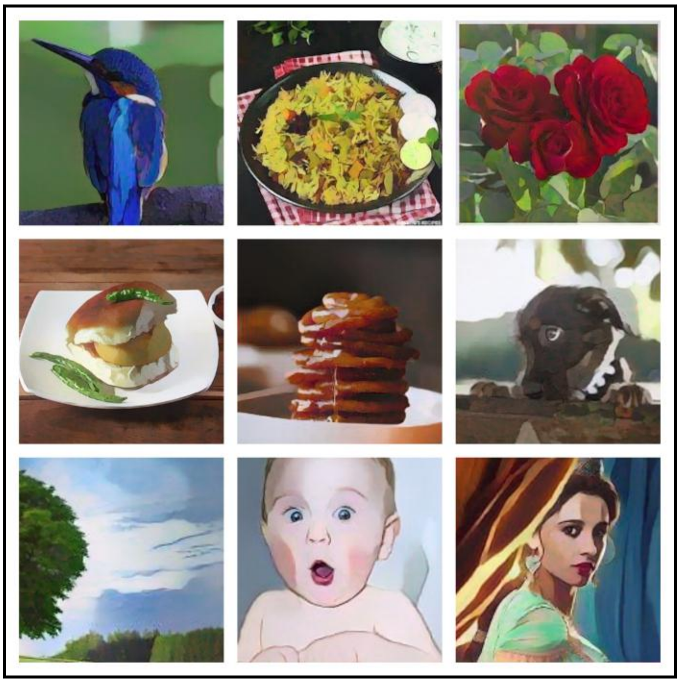

Fig- 5: Different Types of Photos Converted to Cartoon

6. CONCLUSIONS

Thus we have shown that however actual image will be converted to cartoon. The proposed white-box controllable image cartoonization framework supported GAN, which generate high-quality cartoonized pictures from real-world photos. Images classified into Three cartoon representations: the surface illustration, the structure illustration, and the texture illustration. Corresponding image processing modules are extract three representations for network training, and output designs could be controlled by adjusting the load of every illustration within the loss function. In depth quantitative and qualitative experiments, as well as user studies, are conducted to validate the performance of our technique.

REFERENCES

[1] Zheng, Yifan Zhao, Mengyuan Ren, He Yan, Xiangju Lu, Junhui Liu, Jia Li. "Cartoon Face Recognition: A Benchmark Dataset". arXiv:1907.13394v3 [cs.CV] 27 Jun 2020

[2] Haozhi Huang, Hao Wang, Wenhan Luo, Lin Ma, Wenhao Jiang, Xiaolong Zhu, Zhifeng Li, Wei Liu. "Real-Time Neural Style Transfer for Videos". 2017 IEEE Conference on Computer Vision and Pattern Recognition DOI 10.1109/CVPR.2017.745.

[3] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexi A. Efros Berkeley AI Research (BAIR) laboratory, UC Berkeley "Unpaired Image-to-Image Translation using Cycle-

Consistent Adverserial Networks", ICCV paper arXiv:1703.10593 2017.

[4] Henry Kang, Seungyong Lee, Charles K. Chui. "Flow-Based Image Abstraction" 2009 IEEE Computer Society

[5] Hamidreza Sadreazami, Amir Asif, Arash Mohammadi. "Iterative Graph-based Filtering for Image Abstraction and Stylization". 1549-7747(c) 2016 IEEE.

[6] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua and Sabine Susstrunk, "Slic superpix-els compared to state-of-the-art superpixel methods", ¨ IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 11, pp. 2274-2282, 2012.

[7] 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.

[8] Xinrui Wang, Jinze Yu, ByteDance, The University of Tokyo, Style2Paints Research "Learning to Cartoonize Using White-boox Cartoonize Representations". CVPR 2020 IEEE.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets. In Advances in Neural Information Processing Systems", pages 2672–2680, 2014.

[10] 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.

[11] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh,and Ming-Hsuan Yang. "Diverse image-to-image translation via disentangled representations". In Proceedings of the European Conference on Computer Vision (ECCV), pages 35–51, 2018.

[12] Zhengxia Zou, Tianyang Shi, Shuang Qiu, Yi yuan, Zhenwei Shi. "Stylized Neural Painting" DOI: 10.1109/CVPR46437.2021.01543

[13] Y. Shu et al., "GAN-based Multi-Style Photo Cartoonization," in IEEE Transactions on Visualization and Computer Graphics, doi: 10.1109/TVCG.2021.3067201.

[14] Martın Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. "Tensorflow: A system for large-scale machine learning". In 12th Symposium on Operating Systems Design and Implementation, pages 265–283, 2016.

[15] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super resolution: Dataset and study. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, July 2017

[16] Milan Sonka, Valcav Hlavac, Roger Boyle, "Image Processing, Analysis, and Machine Vision".

[17] Y. Chen, Y. -K. Lai and Y. -J. Liu, "CartoonGAN: Generative Adversarial Networks for Photo Cartoonization," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 9465-9474, doi: 10.1109/CVPR.2018.00986.

[18] K. Lata, M. Dave and K. N. Nishanth, "Image-to-Image Translation Using Generative Adversarial Network," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), 2019, pp. 186-189, doi: 10.1109/ICECA.2019.8822195.

[19] Yilin Ouyang; Yunbo Rao; Dawei Zhang; Jiajun Cheng "Cartoon Colorization with Gray Image Generated from Sketch" 10.1109/PRAI53619.2021.9551084

[20] Yuxiang Xie; Xidao Luan; Xin Zhang; Chen Li; Liang Bai "A Cartoon Image Annotation and Retrieval System Supporting Fast Cartoon Making" 10.1109/CSE.2014.314

[21] Xinyu Li, Wei Zhang, Tong Shen, Tao Mei "Everyone is a Cartoon-ist: Selfie Cartoonization with Attentive Adversarial Networks" arXiv:1904.12615 [cs.CV]