International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 24 Issue 09, SEP, 2024

IMAGE PROCESSING AND CARTOONIFYING USING MACHINE LEARNING FILTERS

Javeria Nishat[1], Dr. I. Samuel Peter James Nilofer[2], Dr. K. Palani[2]

[1]PG Scholar, Department of CS, Shadan Women’s College of Engineering and Technology, Hyderabad,
javerianishat553@gmail.com

[2]Professor, Department of CSE, Shadan Women’s College of Engineering and Technology,
drisamuelpeterjames@gmail.com

[2]Professor & Head, Department of CSE, Shadan Women’s College of Engineering and Technology,
principalswcet2020@gmail.com

ABSTRACT: Cartoons are illustrations of characters that can be imaginary or are based on a personality. They are semi-realistic or non-realistic, which are popular nowadays to depict a situation or a happening in a funny, satirical way[9]. Earlier, cartoonists used to draw these cartoons by hand and creating a cartoon-like effect is time and space consuming. Need for automation arises. Some solutions have come into existence like installing complex photo editing software like photoshop and others. These softwares are not easy to use and make user tasks hard and complex. Hence, this project plans to create a web application with a simple user interface that is more suitable, space efficient and takes minimum user efforts and allows users to apply the filters to images of their choice. The filters are designed to provide artistically and comically appealing results on a wide range of pictures. The system is made simpler as the code is written in Python language, which is considered the most fun and easy to understand yet with a wide range of applications in every field.

INTRODUCTION: Today social media has become predominant in our daily lives. Various social media apps include Facebook, Instagram, Whatsapp, Snap Chat etc…It’s where people require a lot of edited images to update their display pictures, profile pictures, make memes, blog posts, artworks, tweets, memes, opinions to grab attention of followers or to create influence or to connect with them. This website can be used to edit such images. Also, everyone is fond of cartoons and they are an integral part of our childhood too. Hence this website can be used in the field of multimedia for creating cartoon characters for animated movies and also for producing designs of PC models.

EXISTING SYSTEM: In order to create photos with effects, the majority of photo editing websites offer the various Image Effects where the user has to upload a photo from a computer which then generates a cartoon image and doesn’t capture video to generate an image. Various features like blurring, reducing the color palette, gray scaling and creating neural style images are not available within a single website and need various photo editing softwares like photoshop. It requires users to have minimal technical background and can even misuse the uploaded images.

PROBLEMS IN EXISTING SYSTEM:
It has several drawbacks like colors may be blurred, smoothness deprives and leads to an unsatisfactory result. Some websites may support images of only specific size. The image processing may take a huge amount of time. The users even need to pay subscription charges to use the website. It requires extra time and effort to learn the usage of tools offered by the website.

PROPOSED SYSTEM: In this approach Preprocessing becomes an important part. Here, a user can login to the website by providing his username and password. Then, he can capture a real time photo using a laptop cam or upload a photo from the PC. Now, that generated photo is preprocessed by applying various filters like generating gray scale images, blurring the images (i.e mean blur, median blur, gaussian blur), applying neural style transfer, reducing the color palette. In case the user does not want to capture an image through the camera option, he can upload an existing image from the PC and then use the above functionalities. Later he can download those images if he wants.

ADVANTAGES: The website is user friendly as it has a web interface and easy to use GUI components. Users even with no technical knowledge can easily use the interface. The users need not pay any registration charges. One stop website for various features like blurring the images, generating gray scale images, artistic type ones etc..It ensures security to uploaded images as a user login system is used. Images can be created quickly and easily and can be used as profile pictures on social media websites, to create trending memes. It also reduces the burden of drawing artistic style images manually for comics and magazines.

SYSTEM ARCHITECTURE:

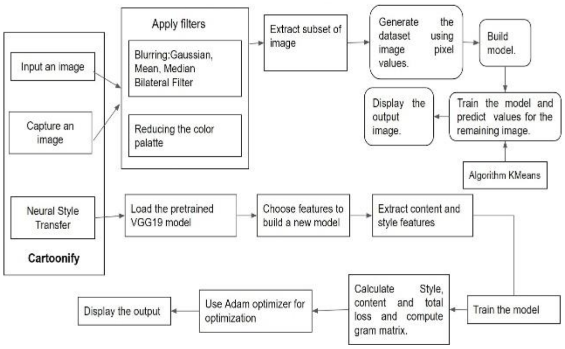

Input an image
Capture an image
Neural Style Transfer
Cartoonify
Apply filters
Blurring:Gaussian, Mean, Median Bilateral Filter
Reducing the color palatte
Extract subset of image
Generate the dataset using image pixel values.
Build model.
Train the model and predict values for the remaining image.
Display the output image.
Algorithm KMeans
Load the pretrained VGG19 model
Choose features to build a new model
Extract content and style features
Train the model
Calculate Style, content and total loss and compute gram matrix.
Use Adam optimizer for optimization
Display the output

Figure 1 System Architecture

IMAGE PREPROCESSING :

Techniques:

Gray Scaling: Here a color image is converted from color spaces like RGB, CMYK, HSV etc to shades of gray. It can be done using the cv2.cvtColor() function of OpenCV.It uses cv2.COLOR_BGR2GRAY as an argument. RGB images are 3 dimensional whereas gray scale images are single dimensional.

Figure 2 Input Image

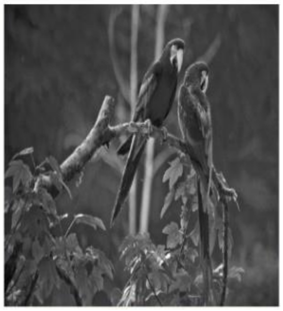


Figure 3 Output Image

Mean Blur Technique: This is done by convolving an image with a normalized box filter. It simply takes the average of all the pixels under the kernel area and replaces the central element. This is done by the function cv.blur() or cv.boxFilter().

Figure 4 Input Image

Figure 5 Output Image

Median Blur Technique: It takes the median of all the pixels under the kernel area and the central element is replaced with this median value. This is highly effective against salt-and-pepper noise in an image.

Figure 6 Input Image

Figure 7 Output Image

Gaussian Blur Technique: In Gaussian Blur operation, the image is convolved with a Gaussian filter instead of the box filter. The Gaussian filter is a low-pass filter that removes the high-frequency components and noise is reduced. Users can perform this operation on an image using the Gaussianblur() method of the imgproc class.

Figure 8 Input Image

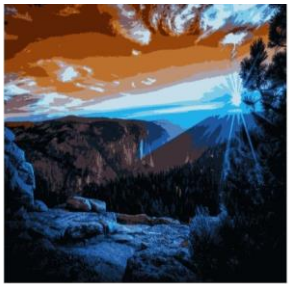


Figure 9 Output Image

Bilateral Filter: Bilateral filter is the key element in the color image processing chain, as it homogenizes color regions while preserving edges, even over multiple iterations.This filter is widely used in noise reduction[4]. The bilateral filter works similarly to a Gaussian filter in that it assigns to each pixel a weighted sum of the pixel values in the neighborhood[6].

Figure 10 Input Image

Figure 11 Output Image

Reducing the Colour Palette: A cartoon sketch will have far fewer colors as compared to an HD photograph. So, a method named color quantization is used. This will reduce the number of colors in the photo. K Means clustering algorithm is used to perform the process. The "k" value is adjusted depending on the number of colors we need.

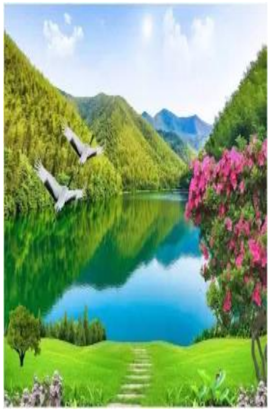


Figure 12 Input Image

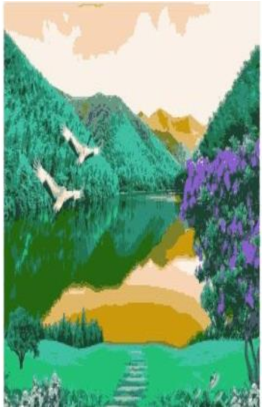


Figure 13 Output Image

Adaptive Thresholding: Adaptive thresholding is the method where a threshold value is calculated for smaller regions. In OpenCV, Adaptive threshold operation can be performed and detect edges on an image using the method adaptiveThreshold() of the Imgproc class.

ALGORITHMS:

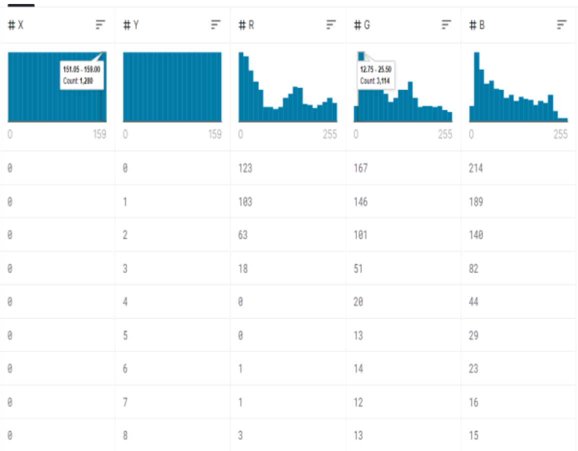
K-means for Color Quantization:K-Means is an unsupervised algorithm which tries to make clusters of input data features and allocates the input data objects to separate clusters based on the relationship among them.

Generation of Dataset : An input image is taken. Extract a subset of the image.

Determine the RGB values of the pixels. Use them to train the model and use it to predict values for the remaining portion.

Figure 14 Input Image for Dataset Generation

What does the dataset contain: For example, the sample dataset contains the CSV representation of an image measuring 160*160 pixels. Each row in the data table represents a pixel, and X and Y columns represent the X and Y coordinates of each pixel, followed by the RGB color values. Each color value is an integer between 0 and
225. These datasets are generated depending upon the input image.



# X	# Y	# R	# G	# B
0	0	123	167	214
0	1	103	146	189
0	2	63	101	140
0	3	18	51	82
0	4	0	20	44
0	5	0	13	29
0	6	1	14	23
0	7	1	12	16
0	8	3	13	15

Fig 15 Sample Dataset

Working of Model: We use the Euclidean distance, which is applied to all pairs of pixels in the RGB space.

$$\sqrt{(R_2-R_1)^2+ (G_2-G_1)^2+ (B_2-B_1)^2}$$

Group pixels in the dataset into clusters based on their distance.

Determine appropriate number of clusters denoted by k. Now all pixels that are assigned to the same cluster will have their RGB values replaced by the mean RGB values of that cluster. After several iterations, it will return the optimal cluster assignments of each point. This process (quantization) reduces the total number of colors in the image to the number of clusters (k) in the model.

Convolutional Neural Networks for Neural Style Transfer: Convolutional Neural Network is a class of Artificial Neural Network(ANN) used in image recognition and processing that is specifically designed to process pixel data. It has one or more convolutional layers and are used mainly for image processing, classification, segmentation and also for other auto correlated data.

Neural Style Transfer: Neural style transfer refers to a class of software algorithms that manipulate digital images, or videos, in order to adopt the appearance or visual style of another image. NST algorithms are characterized by their use of deep neural networks for the sake of image transformation. It is an optimization technique that takes two images- a content and a style image (such as an art work by a painter) and blends them together so that the output image looks like the content image but painted in the style of the style reference image.

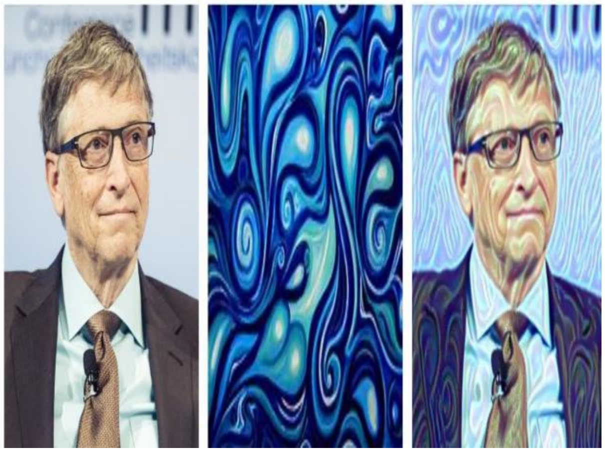
Style Transfusion: Input a style image and content image. A style image (s) — the image we want to transfer the style from A content image (c) — the image we want to transfer a style to. Define loss function which blends two images seamlessly to create visually appealing art. Generated image (g) — the image that contains the final result.

Defining the loss function:
Content Loss= Generated Image Content - Content Image

$$L_{Content} = \frac{1}{2}\sum(T_i - A_i)^2$$

Style Loss =Generated Image Style - Style Image

$$L_{Style} = \alpha \frac{1}{2}\sum wi(T_i - S_i)^2$$



Content Image + Style Image = Output Image
Figure 16 Neural Style Transfer

Using pretrained VGG-19 model for Neural style Transfer: VGG19 is a variant of VGG model (a deep CNN used to classify images) that contains 19 layers : 16 convolution layers, 3 Fully connected layer,
5 MaxPool layers, 1 SoftMax layers.The first layers represent low-level features like edge , colors and textures.The final layers represent higher-level features of objects. For example wheels, tyres,steering etc..

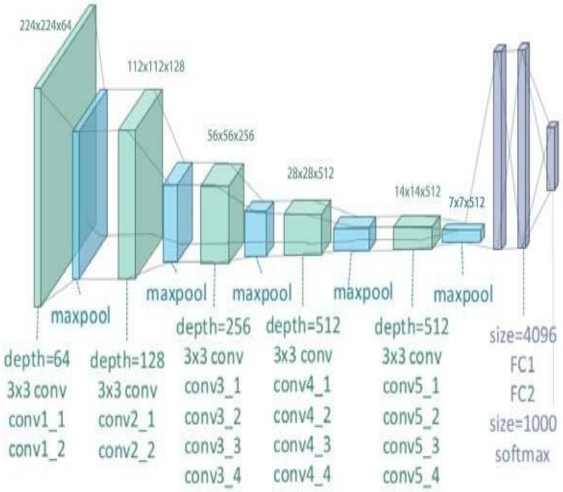


Figure 17 VGG-19 Model

ImageNet Database: VGG model uses the ImageNet Database which is organized according to the WorldNet Hierarchy. It is used for Image classification. It contains more than 14 million images, 1281167 images for training, 50000 images for validation, 1000 classes(species of dogs, cats, household objects, vehicles etc..), 25 attributes (color like red,blue; pattern like spotted, shape like long, texture..), 400 popular synsets.

Building the Model: Create a new model by selecting some of the intermediate layers of the model to get the content and style representations of the image.
Content_layers = ['B5_conv2'] Style_layers = ['B1_conv1', 'B2_conv1', 'B3_conv1', 'B4_conv1', 'B5_1] Calculate Gram matrix, content loss, style loss and total loss. Use Adam Optimizer to reduce the loss. Now import style features into the content image. Display the image once training is done.

CONCLUSION : Cartoonify simplifies the practice of applying filters to images at free of cost with no technical background. It provides a simple user interface allowing users to apply preprocessing techniques of their choice to provide artistically and comically appealing results on various ranges of images.

FUTURE SCOPE: An android application can be designed. Other features can be added such as enabling users to give ratings and feedback about the website, providing scope to apply features to a whole video, enabling the users to post their favorite images on the website, adding more image styling features thereby increasing the functionality.

REFERENCES:

[1] Web Technologies by Uttam K Roy, Oxford University Press.
[2] The Complete Reference Python by Martin C.

Brown, McGraw Hill Education

[3] Javascript from Beginners to Professional by Laurence Lars Svekis, Maaike van Putten, Rob Percival , Packt Publishing Limited.

[4] "Guided Image Filtering" by Kaiming He, Jian Sun and Xiaoau Tang, The Chinese University of Hong Kong , Microsoft Research Asia.

[5] Baggio, Daniel L, "Mastering OpenCv with Practical Computer Vision Projects" Packt Publishing Ltd, 2012.

[6] IEEE Paper on "Toonify: Cartoon Photo Effect Application" by Kevin Dade, Dept of Electrical Eng, ECA) Stanford University Stanford, CA.

[5] ArtsiomSanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bjorn Ommer's "A style-aware content loss for real-time HD style transfer" Published in Proceedings of the European Conference on Computer Vision (ECCV), pages 698–714, 2018.

[6] IEEE Paper on “Neural Style Transfer using VGG19 and Alexnet” by S. Kavitha; B. hanapriya; G. Naveen Vignesh; K.R. Baskaran, published in 2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICA