CARTOONIFY

MINI PROJECT – I

SYNOPSIS

GLA UNIVERSITY
ESTD. 2010
ऋते ज्ञानान्न मुक्तिः

GLA
UNIVERSITY
MATHURA
Established vide U.P. Act 21 of 2010.

Department of Computer Science & Application

Institute of Engineering & Technology

SUBMITTED TO: -

Abhishek Kumar Tiwari
(Technical Trainer)

SUBMITTED BY: -

Aryan Gupta (201500148)
Rishabh Sharma (201500569)

Acknowledgement

It gives us a great sense of pleasure to present the synopsis of the B.Tech mini project undertaken during B.Tech III Year. This project is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals. We owe special debt of gratitude to Mr. Abhishek Kumar Tiwari, Technical Trainer , for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal and for his constant support and guidance to our work.

His sincerity, thoroughness and perseverance has been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also taught us about the latest industry-oriented technologies. We also do not like miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and co-operation.

Aryan Gupta (201500148)

Rishabh Sharma (201500569)

ABSTRACT

Since the client for animation pictures recovery framework focuses to get applicable pictures to inquiry picture from information base inside same item (for example a client has animation picture with object, for this situation the client will focus to get all significant picture with, along these lines A significant advance in animation picture recovery is characterizing the item inside animation picture .In this paper, a proficient technique for objects extraction from animation pictures is presented; it depends on broad suppositions identified with shading and areas of items in animation pictures, the items are commonly gravitated toward the focal point of the picture, the foundation tones is the all the more much of the time gravitated toward the edges of animation picture, and the item colors is less touch for the edges. The cycles of shading quantization, seed filling and found the item apparition have been utilized. The aftereffects of led tests showed that the framework have promising effectiveness for extricating both single or multi object(s) lay in straightforward and complex foundations of animation pictures.

Contents

INTRODUCTION

Animation pictures assume basic parts in our regular day to day existences particularly in diversion, training, and promotion, that become an inexorably escalated research in the field of media and PC designs. The naturally animation object extraction is exceptionally helpful in numerous applications; one of the most significantly is the animation pictures recovery, where the client for animation pictures recovery framework focuses to get comparable pictures to question picture from information base in character (i.e., a client has animation picture with object Dora, so the client will focus to get all applicable picture with Dora character). Today, various analysts have misused the ideas identified with content based pictures recovery (CBIR) to look for animation pictures containing specific object(s) of interest . A few area based recovery techniques proposed, for additional subtleties see . A portion of the programmed techniques, which separate the region(s) of premium from the other less helpful areas in a picture, have been adjusted to recover animation characters [7-8]; they utilize incomplete highlights for perceiving locales and additionally angles which are reasonable for animation portrayal or signal acknowledgment. A few endeavors go past separating focal articles [3], others utilized Salient Object Detection (SOD) [9-12]. In this paper, a basic programmed strategy for objects extraction from animation picture is proposed; it is based on the suspicion that the needed item is established inside or near the focal piece of picture.

SOFTWARE AND HARDWARE REQUIREMENTS

Python: We use python as a programming language for building the application.

cv2: We use cv2 for image processing.

Numpy: Mainly NumPy is used for dealing with arrays. Here the imagesthat we use are stored in the form of arrays. So for that, we use NumPy.

easygui: easygui is a module used for GUI programming in python. In our application easygui is used to open the file box to upload images from the local system.

Imageio: Imageio is a python library that reads and writes the images.

Matplotlib: Matplotlib is used for visualization purposes. Here we plot the images using matplotlib.

OS: Here in our application os is used for dealing with paths like reading images from the path and saving the image to the path.

Tkinter: Tkinter is a standard Graphical User Interface(GUI) package.

512 MB Ram

Window 10

PROJECT DESCRIPTION

Image to Cartoon Python OpenCV Machine Learning . This Project web app project you can directly select image then you can convert any image to cartoon . its very interesting project .

WORKING

To convert an image to a cartoon, multiple transformations are done. Firstly, an image is converted to a Grayscale image. Then, the Grayscale image is smoothened, and wetry to extract the edges in the image. Finally, we form a color image and mask it with edges. This creates a beautiful cartoon image with edges and lightened color of the original image.

IMPLEMENTATION

We will import the following modules:

CV2, easygui, Numpy, Imageio, Matplotlib, OS

We will use the pop-up box to choose the file from the device, which opens every time you run the code. fileopenbox() is the method in easyGUI module which returns the path of the chosen file as a string. Using Imread , is a method in cv2 which is used to store images in the form of numbers. This helps us to perform operations according to our needs. The image is read as a numpy array, in which cell values depict R, G, and B values of a pixel. cvtColor(image, flag) is a method in cv2 which is used to transform an image into the colour-space mentioned as 'flag'. Here, our first step is to convert the image into grayscale. Thus, we use the BGR2GRAY flag. This returns the image in grayscale. A grayscale image is stored as grayScaleImage. After each transformation, we resize the resultant image using the resize() method in cv2 and display it using imshow() method. This is done to get more clear insights into every single transformation step. cvtColor(image, flag) is a method in cv2 which is used to transform an image into the colour-space mentioned as 'flag'. Here, our first step is to convert the image into grayscale. Thus, we use the BGR2GRAY flag. This returns the image in grayscale. A grayscale image is stored as grayScaleImage.

After each transformation, we resize the resultant image using the resize() method in cv2 and display it using imshow() method. This is done to get more clear insights into every single transformation step. we will work on the first specialty. Here, we will try to retrieve the edges and highlight them. This is attained by the adaptive thresholding technique. The threshold value is the mean of the neighborhood pixel values area minus the constant C. C is a constant that is subtracted from the mean or weighted sum of the neighborhood pixels. Thresh_binary is the type of threshold applied, and the remaining parameters determine the block size. we finally work on the second specialty. We prepare a lightened color image that we mask with edges at the end to produce a cartoon image. We use bilateralFilter which removes the noise. It can be taken as smoothening of an image to an extent. The third parameter is the diameter of the pixel neighborhood, i.e, the number of pixels around a certain pixel which will determine its value. The fourth and Fifth parameter defines signmaColor and sigmaSpace. These parameters are used to give a sigma effect, i.e make an image look vicious and like water paint, removing the roughness in colors. We will combine the two specialties. This will be done using MASKING. We perform bitwise and on two images to mask them.

REFERENCES;

BOOKS

Hands-on ML with Scikit-Learn, Keras & TensorFlow

Machine Learning for Absolute Beginners: A Plain English Introduction.

Websites:

- www.pythonprojects.com
- www.geeksforgeeks.org
- www.google.com
- www.projectdeveloper.com

Faculty Guidelines:

Mr. Abhishek Kumar Tiwari (Technical Trainer in GLA University)

GitHub Repository link:

https://github.com/aryangupta777/cartoonify.git

Google Drive link:

https://drive.google.com/drive/folders/15q45izQZbS8zYUtOu7C1QofqSpGKL2la?usp=sharing