COMPUTER VISION - COUNT SHAPES CARTOONIFY THE FOREGROUND

Sahil Kumar
Indian institute of Technology
Ropar, Punjab

1 Introduction

The assignment has two parts- firstly to implement a function named CountShapesID that would take as input a given image and detect the rings in it, where a ring may be a circle, semicircle or a sphere. The detected rings are to be highlighted on the original image.
Secondly, implement a function CartoonNizerID to cartoonify the foreground and blur the background of the image. The foreground has a human, who is to be cartoonified and the rest is to be blurred.

2 Count Shapes using hough transform

The fundamental principle behind this program is using hough transform to detect the rings. But, some preprocessing of the image is required before applying the hough transform.

2.1 Preprocessing

The original image as give in figure 1. On directly applying hough transform gives bad results where a lot of rings are left out. To solve this problem, I converted the image in binary space to first of all properly carve out the image. This binary image is then dilated and then the connected components having area smaller than a threshold is removed to clean the image. This image is then inverted. This output can be seen in figure 2.

2.2 Hough transform

The parameter space of circle would be three dimensional, (a, b, r). And all the parameters that satisfy (x, y) would lie on the surface of an inverted right-angled cone whose apex is at (x, y, 0). In the 3D space, the circle parameters can be identified by the intersection of many conic surfaces that are defined by points on the 2D circle. This process can be divided into two stages. The first stage is fixing radius then find the optimal center of circles in a

Figure 1: original image

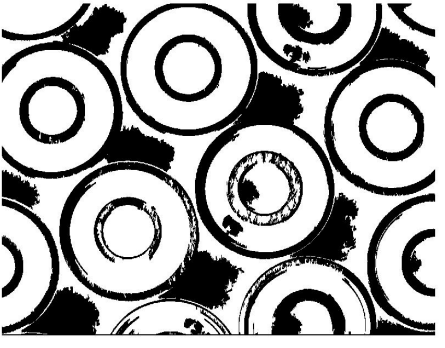

Figure 2: Image after preprocessing

2D parameter space. The second stage is to find the optimal radius in a one dimensional parameter space. The image obtained after preprocessing is subjected to hough transform using empirical values of range of radius and the sensitivity. The hough transform is used multiple times over different range of radius to get better result. Further, the two stage method is used for better accuracy.

2.3 Result

The result is shown in figure 4. Out of 32 rings, 29 are being detected for my empirical values of sensitivity and radius. They are shown using red circles.

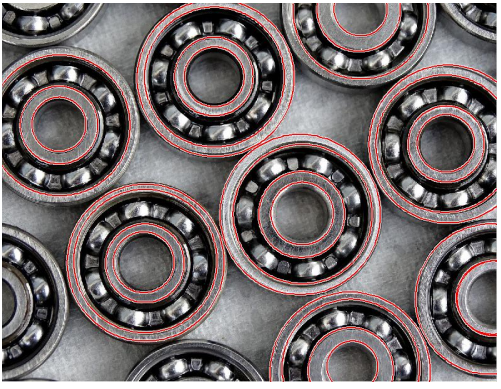

Figure 3: final image

3 Cartoonify an image

The purpose here is to use modified bilateral filtering to cartoonize the image while also conserving the edge information. The foreground is separated from the background by using skin color properties. Firstly, the skin color is detected and thus I obtained a binary image. Now, using dilation and erosion, I obtained a somewhat accurate location of the human being. Now, the entire image was cartoonified and then the detected foreground is replaced in the original image with this cartoonized version. The rest of the image is considered as background and is blurred using gaussian blur. The original image is given in figure 4.

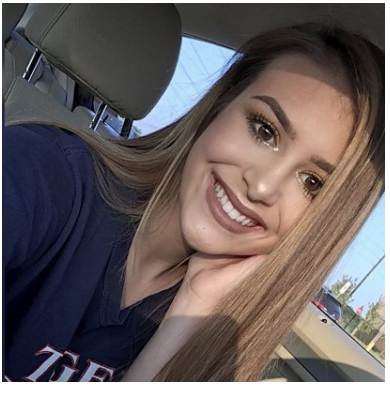

Figure 4: original image

3.1 CartoonNizer

A bilateral filter is a non-linear, edge-preserving, and noise-reducing smoothing filter for images. It replaces the intensity of each pixel with a weighted average of intensity values from nearby pixels. This weight can be based on a Gaussian distribution. Crucially, the weights

depend not only on Euclidean distance of pixels, but also on the radiometric differences (e.g., range differences, such as color intensity, depth distance, etc.). This preserves sharp edges. After applying this bilateral filter, soft luminance quantization is applied to further improve on the cartoonish look. This is done in the lab colorspace and finally converted to rgb colorspace.

3.2 Results

The foreground part is detected and that much part is converted to cartoon and the other parts are blurred. figure 5 shows i) the entire image is cartoon ii) the foreground is cartoon and the background is blurred

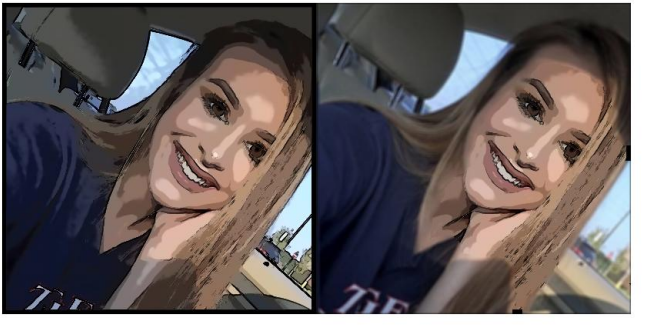

Figure 5: i)Entire image is cartoon ii)Face image cartoon, rest is blurred

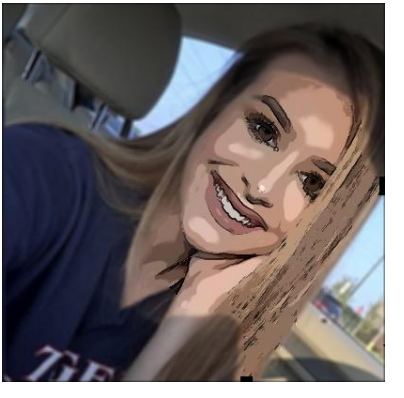

Figure 6: Final image