Image Processing Report

Gaurav Kumar
Roll No: 14240

November 12, 2017

Link for all the codes:- Matlab Codes

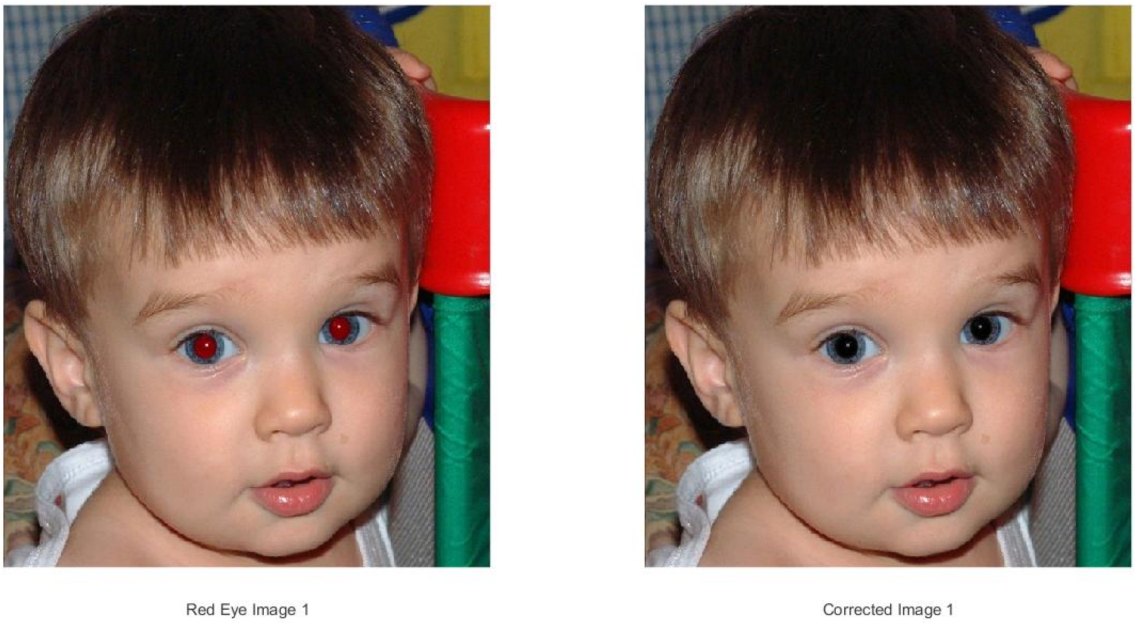
1 Red Eye Correction:-

1.1 Algorithm:

- An RGB image is taken as input and is converted into HSV form for further processing.
- A Vision Cascade Eye Detector that works on Voila- Jones Algorithm is used to build a bounding box around the eye.
- Create a mask of size of image. Loop through each pixel of the bounding box and whenever **Hue** of any pixel is less than 0.02 or greater than 0.9 and **Saturation** value greater than 0.6, we make that pixel of mask 1 (white). Rest of mask remains black.
- Perform **Morphological Transformations** on binary Mask such as opening and dilating to improve the mask. This will make the white portion of the mask more rounded and will make the edges smooth.
- For Correction, Replace the Red channel of the masked portion of image to be the average of blue and green channels.
- For better results, replace blue and green portion of the masked area individually to be the average of their channels.
- Display the final Image and problem of Red Eye is corrected.

1.2 Results:-

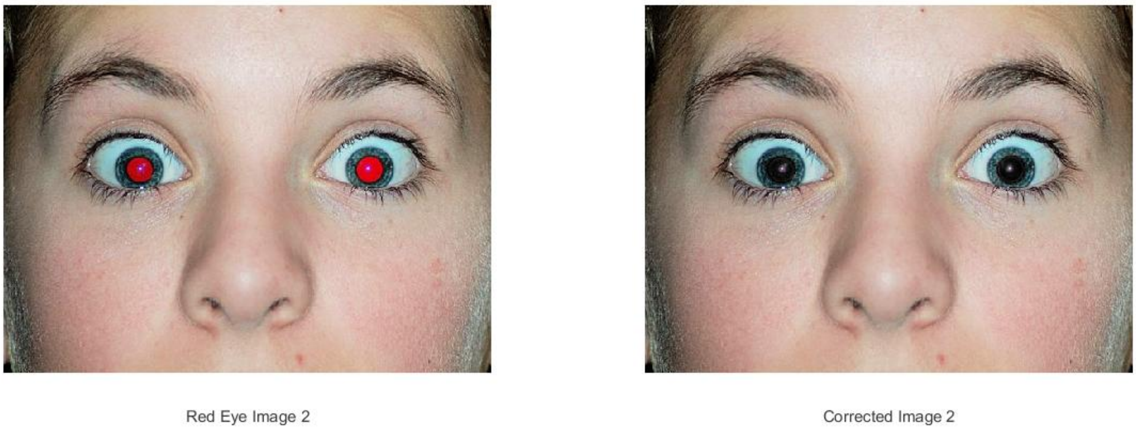
The Results for the Red Eye correction for the two test images are shown below:-



Red Eye Image 1

Corrected Image 1

Figure 1: Red Eye Image(Left), Corrected Image(Right)



Red Eye Image 2

Corrected Image 2

Figure 2: Red Eye Image(Left), Corrected Image(Right)

2 Selective Blur:-

2.1 Algorithm:

- Take a greyscale image as input and convert the pixel values from integer to double.
- Take a **seed point** as input from the user using *ginput* function.
- Apply Motion Blur on the entire image and keep a copy of it.
- Find **Gradient** of the entire image.
- **Binarize** the gradient image using a threshold value which is tuned by testing on multiple grayscale images. In my case, the ideal value of threshold came out to be 0.6. The binary image now has different regions bounded by white contours.
- **Fill holes** in the binary image. A hole is a set of background pixels that cannot be reached by filling in the background from the edge of the image.
- The binary image now has different white regions defined by different objects. Use median filtering to remove small impulse type noise.
- Find all **connected components** to the **seed point** . It makes sure that pixels of only that object remains 1 in which seed point lie. Rest other objects pixels have values 0 only.
- So, after all these steps, a mask is created which is then superimposed on blurred image. The masked part image is not blurred and rest of the image remains blurred.
- Finally sharpen the image using unsharp masking to sharpen the edges of the selected object.

2.2 Results:

The Results for Selective Blur for two test images are shown below:-

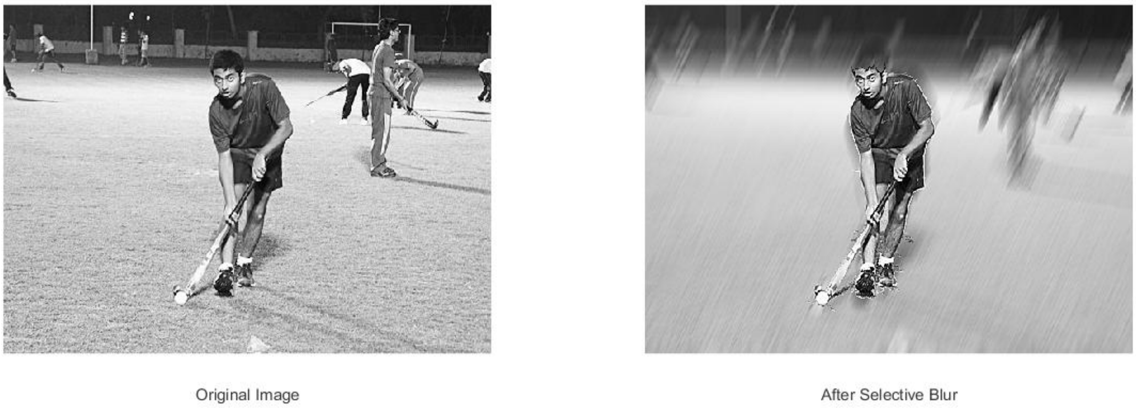


Figure 3: Original Image(Left), Image After Blur(Right)

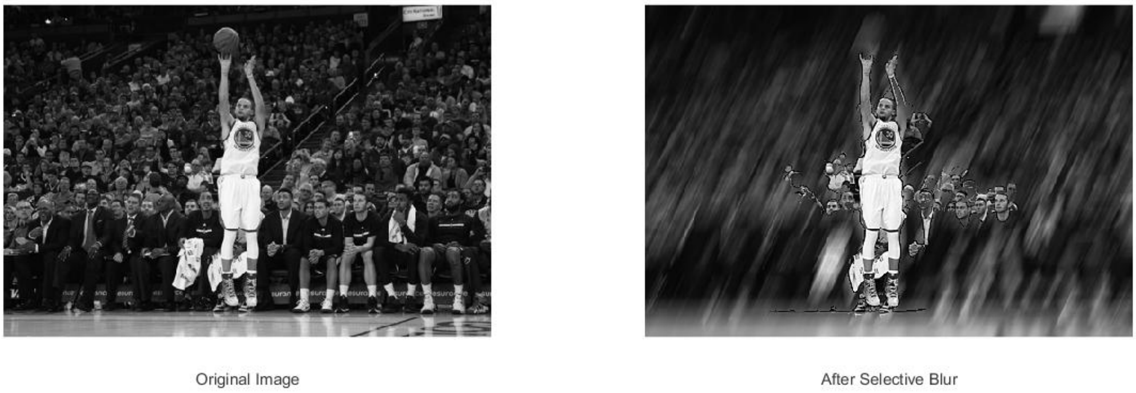
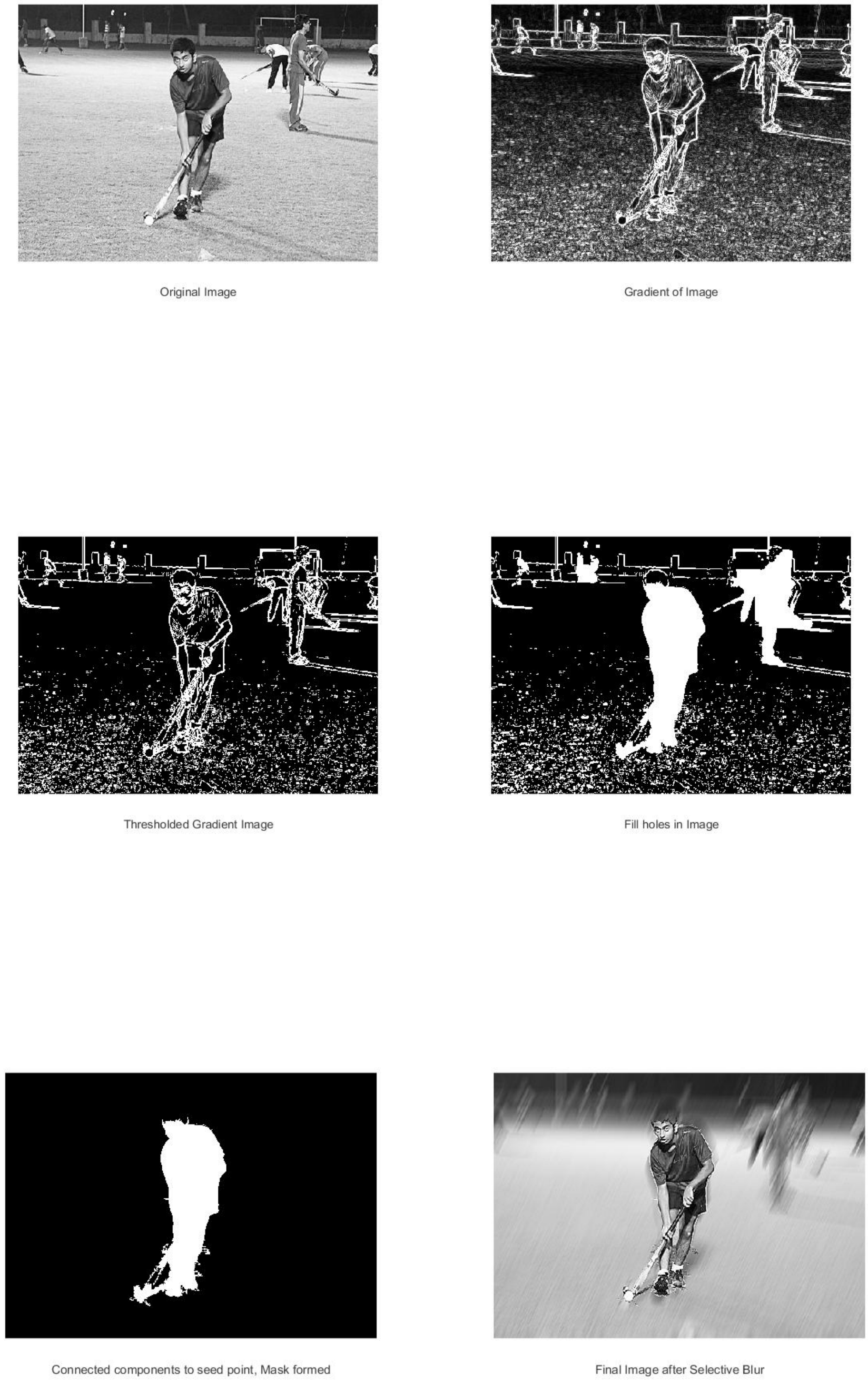


Figure 4: Original Image(Left), Image After Blur(Right)

2.3 Stepwise figures for Selective Blur of Image 1



Original Image

Gradient of Image

Thresholded Gradient Image

Fill holes in Image

Connected components to seed point, Mask formed

Final Image after Selective Blur

Figure 5: Original Image(Top Left), Gradient of Image (Top Right), Thresholded Gradient Image(Middle Left), Holes filled in Image(Middle Left),Connected Components to seed point joined(Bottom Left), Final Image(Bottom Right)

3 GUI in Action:-

The functional GUI was built on Matlab with one button to select an image and display it on screen. Two buttons were present for two different functionalities. Final result would display on the screen.

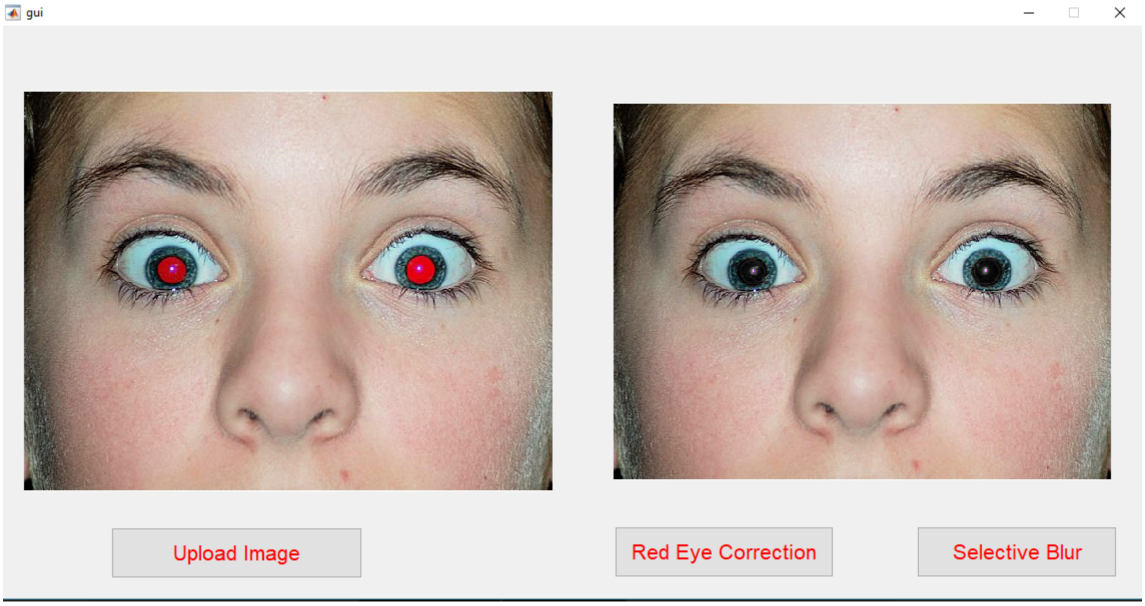


Figure 6: GUI in Action

References

[1] Automatic red-eye detection and removal, Xiao Ping Miao, T. Sim

[2] Image Segmentation and Region Growing Algorithm, Shilpa Kamdi R.K.Krishna

[3] Gui: $https : //in.mathworks.com/help/matlab/creating_guis/about - the - simple - guide - gui - example.html$

[4] Mathworks, https://in.mathworks.com/matlabcentral/answers/120449-how-to-detect-eyes-and-mouth-on-a-single-image-in-matlab