METROPOLITAN STATE UNIVERSITY[SM] OF DENVER

Digital Image Processing
Homework 3

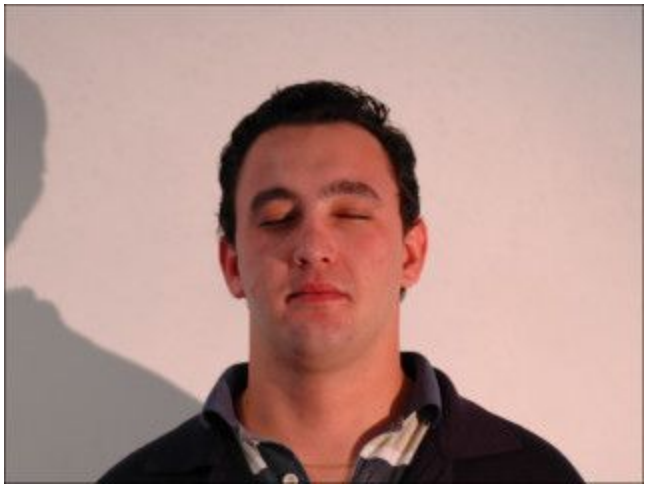
Steven Jennings

26 February 2018

Overview

The goal of this assignment was to understand how skin detection works in numerical computation. These are the two images provided to process:
There are two requirements for this assignment:

The “Dark” Image



The “Good” Image

Procedure

First of all, it's worth mentioning that I didn't use the given code (even though it was helpful), but I used my own approach. Since I have a lot of experience working with photo editing software like GIMP, the HSV colorspace was already familiar to me, and that's all I used to solve this problem. For anyone else who might be reading this, hue is basically the "selection" of color, which is like a color wheel. Saturation is quite literally how "saturated" the color is with the color selection designated by hue. And value is how much "white" or "luminosity" is present in the color.

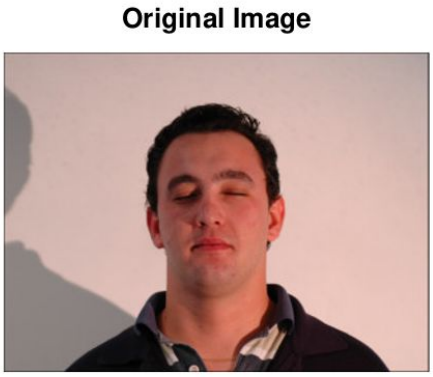
It's important to remember the following about HSV (Hue, Saturation, Value):

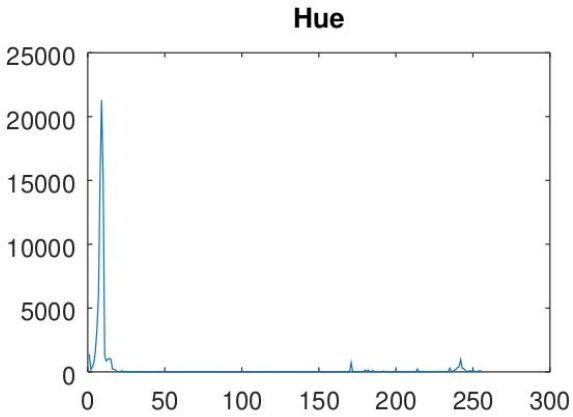
- Maximum saturation and maximum value means full color.
    - (X, Max, Max) = Full color of X
- Zero saturation and maximum value means white color, regardless of hue.
    - (X, 0, Max) = White color
- Zero value means black color, regardless of both hue and saturation.
    - (X, X, 0) = Black color

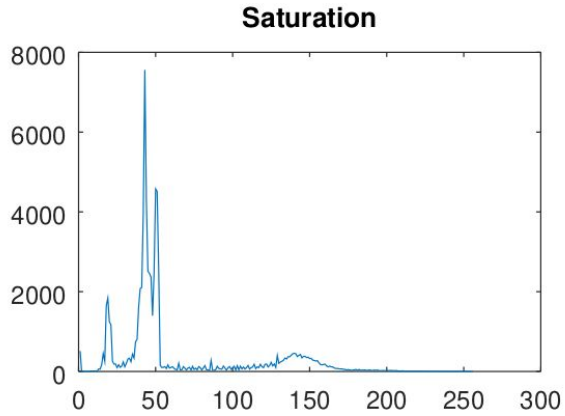
Essentially, the way I solved each of these problems is by acknowledging that the skin of human faces is uniform in color in almost every scenario. By selecting a range of hues, a range of saturations, and a range of luminosity values, (given by histograms) I was able to use this information to detect skin color in simple images like the ones in this assignment.

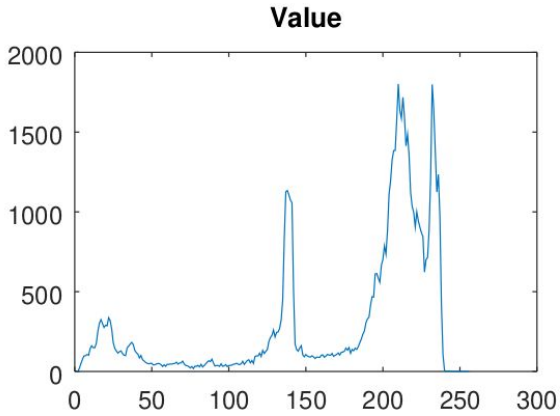
The only problem with my solution is that it does NOT adapt to different images. One range of HSV values that works well for one image might not work as well for another image. But other than that, this solution is very good, as long as the user can analyze the information this program outputs.

Part I: The “Dark” Image









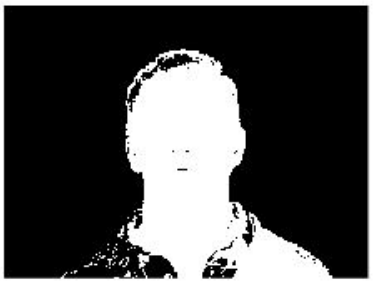
By collecting the information provided by the histograms for the HSV colorspace of the “dark” image, I was able to determine ranges for each of the three histograms which would effectively detect the skin in the image. I chose the following ranges:

- Hue: 0 to 20
- Saturation: 60 to 220
- Value: 100 to 227
    - Values 227 to 255 were mostly just the background luminance

Original Image

Binary Hue

Binary Saturation

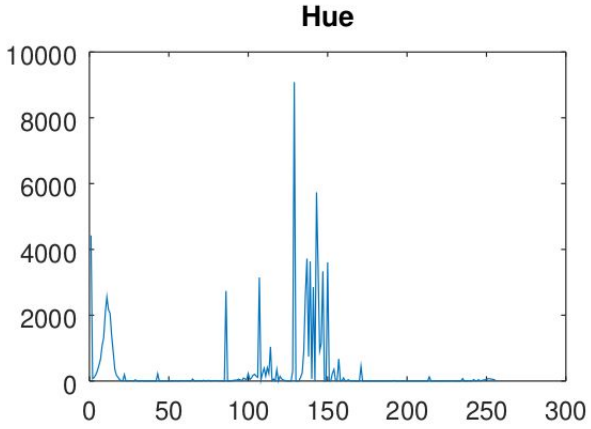


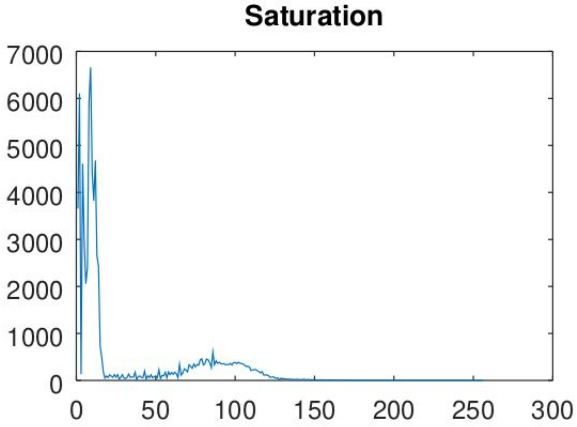
Binary Value

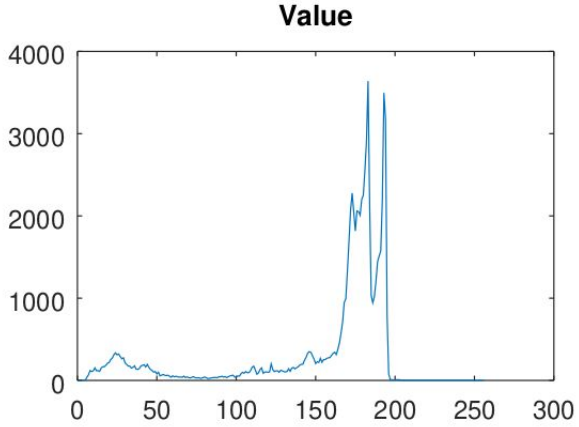
These binary images combined through logic AND to yield the following detection:

Binary Result

Part II: The “Good” Image







Similarly to the “dark” image, these are the ranges I chose for this image:

- Hue: 0 to 80
- Saturation: 50 to 150
- Value: 100 to 200

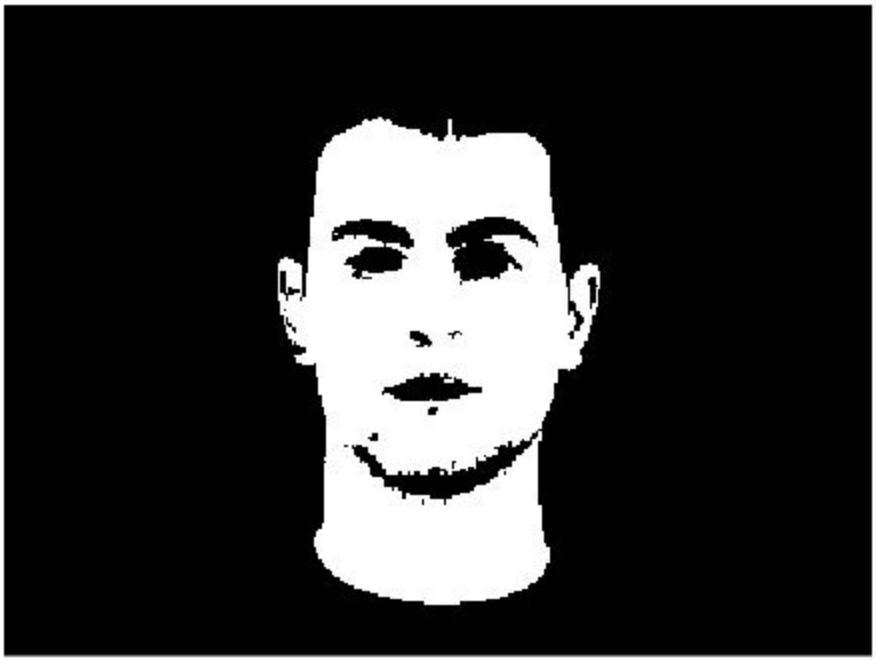
Original Image

Binary Hue

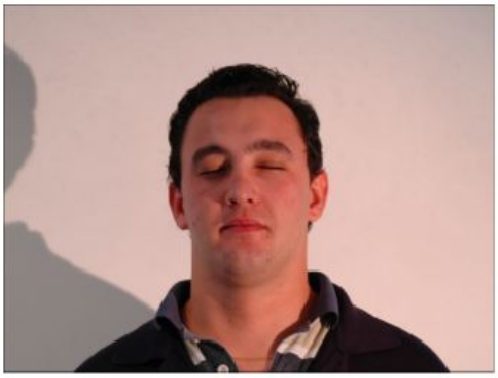
Binary Saturation

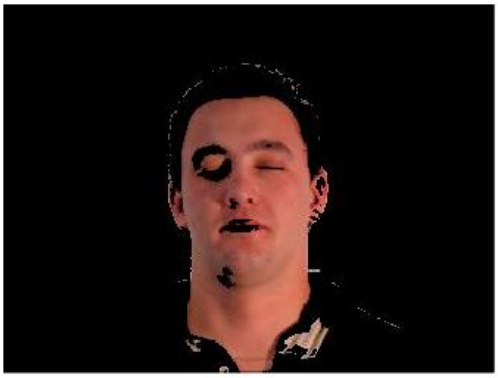
Binary Value

Binary Result



Results and Analysis





So as you can see, the results are not 100% perfect, but in a practical situation, these results would probably be usable. This also goes to show that the basic theory is accurate. Simple HSV manipulation is enough to detect human skin in an image.

Conclusion

This assignment was another nice challenge that encouraged me to study the way image processing works on my own time. It would have been nice if I implemented some form of adaptive feature so that my code would be effective on multiple images with minimal human interaction, but I ran out of time, and analyzing the histograms was a lot even for my brain. I'm sure there are techniques out there I could have used, but I'm not so skilled yet.

Overall, this was the most challenging assignment so far, but also the most rewarding. I'll be sure to keep this code for the upcoming projects.