Alex Iacob

Prof. Kinsman

CSCI 431

September 16, 2022

***Question 1:***

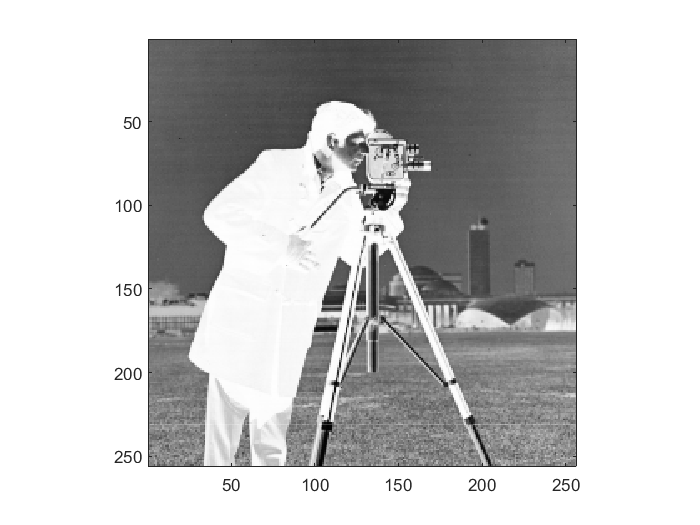
Matlab version = 9.12.0.2009381 (R2022a) Update 4

Matlab License number = 364896

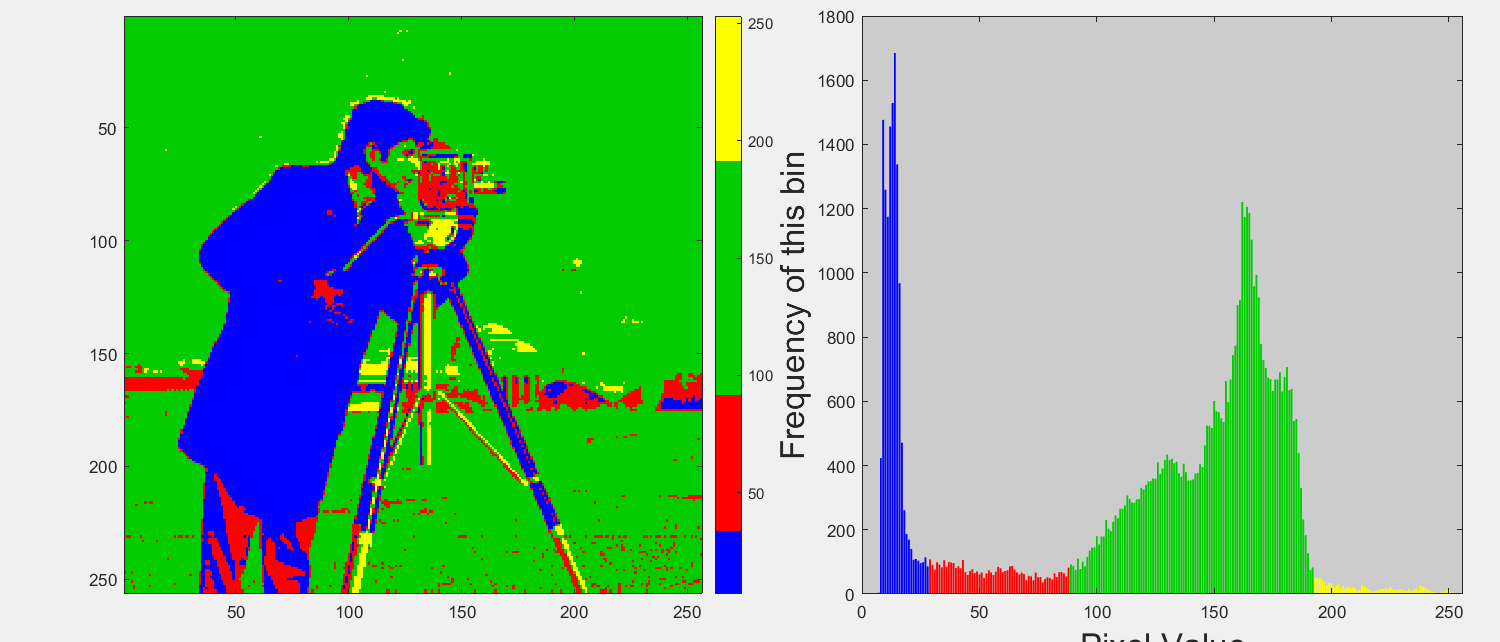
MATLAB Version 9.12 (R2022a)

Image Processing Toolbox Version 11.5 (R2022a)

***Question 2:***

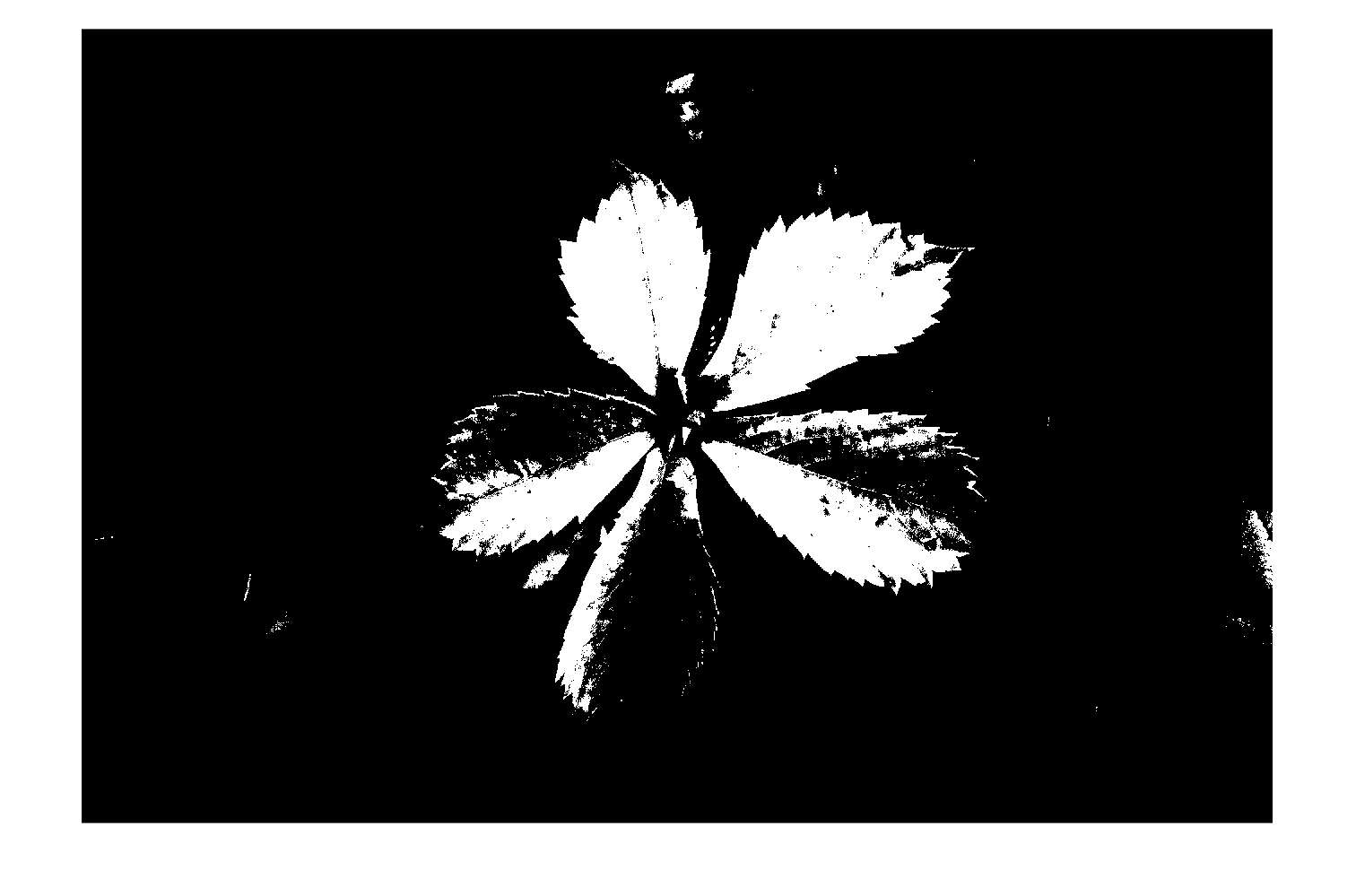
******

The main differences between the two images is that some of the details are more discernible in the cameraman’s attire and the camera. Specifically the creases and folds in the pants and jacket. In the inverted image, you can now see that the cameraman’s jacket has buttons. Some of the details in the camera are easier to see. The sky looks roughly the same, just inverted, as the white spots in the original have now turned black. However even through this inversion, the sky still has stripes throughout it.

***Question 3:***

In this diagram we learn about various “clumps” in the image. For example, the blue clump shows a majority of the cameraman’s jacket which has a low pixel value with very low variation. This means that there is a great cluster of very similarly valued pixels(i.e the man’s coat and hair mainly). We can also see that nearly all of the background is green in this image, which signifies that there exists a larger variation of pixel values. With this histogram, we can more effectively notice the pixel frequency over any particular variation to get more information about the given image

***Question 4:***

******

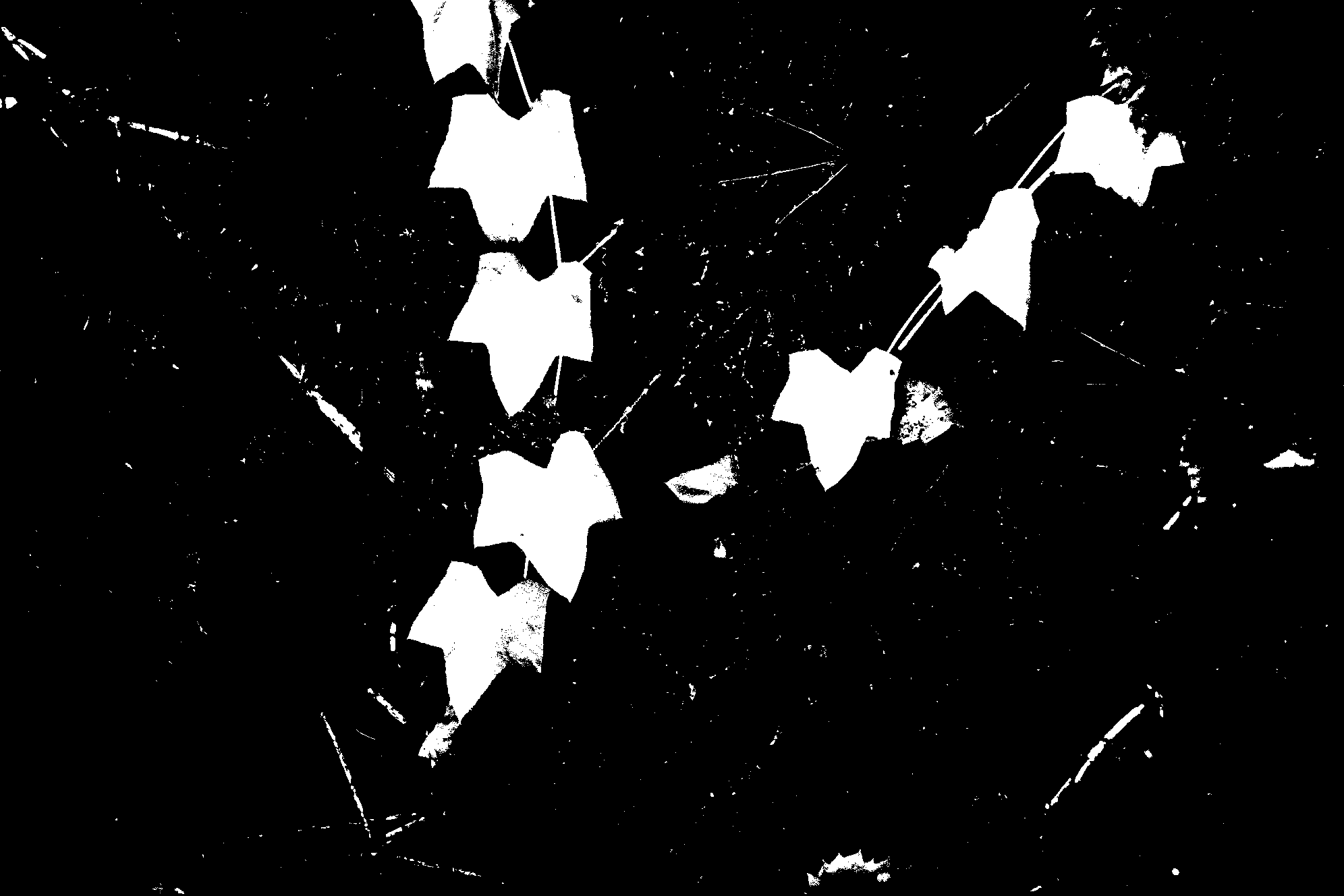
I first started out by searching for the documentation for *graythresh()*, from there I found an example on the same documentation page that essentially walked me through the rest of the process using *imbinarize()*. I first started out by using the exact gray threshold value(0.3647) for the first image, which gave a very good starting point, then looping through increasing gray threshold values to find values that maintain the black background and white foreground. Then the second image with a threshold of 0.5 which had most of the plant as white, though it had a decent chunk of the background noise. Moving ahead to the last image with a threshold of 0.6, it removed most of the background, though it invaded parts of the foreground image. Any threshold basically creates nothing but an entirely black image.

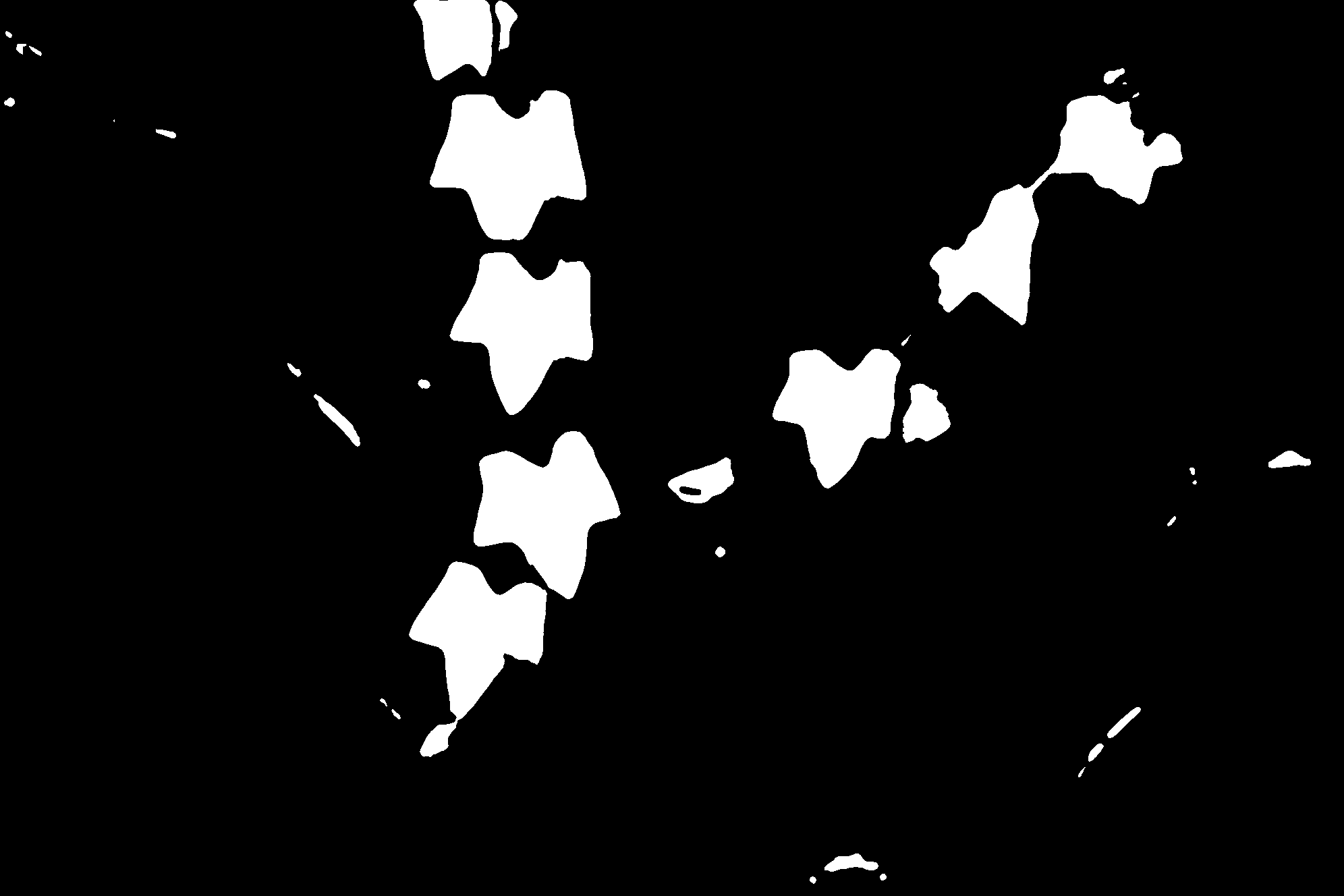
***Question 5:***



Like the previous questions, the first thing I did was look up functions about histogram equalization. The first one that showed up was *histeq()*. Following the example on that Matlab documentation, all that I needed to do was run the grayscale image though the given function and write it to a file. This leads to the image that leaves you able to see the originally hidden sign.

***Question 6:***





I decided to use *imbinarize()* for this question because I noticed that the ivy leaves were a lighter shade than the rest. I used parts of the previous questions to create the grayscale version of the image, then tested around with the gray threshold for a decent coverage of the ivy leaves. After running the binarization function(first image), I noticed that there was a lot of noise in the image, so after some quick searching around, I found the docs for *medfilt2()*. After playing around with some of the values, the image became pretty denoised (second image).

***Question 7:***

******

Out of all the methods used, the ones that made the deer stand out the most were method B(green channel) and D(grayscale). Both of these methods blurred the foliage in the foreground which showed the deer better. The worst one was part F, as it made the entire image black screen.

***Conclusion:***

Throughout this assignment I learned about more methods to better the contrast in an image for image enhancement. The first was how to use histograms to represent an image. This shows the pixel value and how many times it has shown up in the image. I then learned how to utilize binary segmentation to separate the Virginia Creeper from the background. This binarization process just utilizes a threshold to determine what pixel should turn either all white or all black. This leads to an all black and white image. However, I noticed that when the threshold is too high it turns the entire image black.

Next we moved onto using histogram equalization to enhance the contrast. This, as a concept is more difficult than it in practice, as it is just calling one single function on an image. Afterwards I tried applying the binarization again for the ivy leaves. I noticed that the ivy leaves were a lighter color than the background, so immediately gray scaling was applied then I messed around with some binarization values to get a good starting point. However I noticed that even after using the binarization method, there was a lot of noise left in the image, so after a quick google search, I learned about another method that is used to denoise images. Using this function solved the problem nearly instantly.

Although most of the functions used in this assignment were pretty straightforward after learning which ones needed to be used. The largest chunk of time and difficulty was initially figuring out what functions to use, then after looking at their docs. Afterwards, mostly everything was pretty easy. The overall consensus of this assignment was that reading docs and searching the internet are the best way to get things done. I nearly found everything I needed after 1-2 minutes of searching.