Lab: Edge Detection

Table of Contents

Overview

For this lab, We examine the method of edge detection through calculating the image gradients. We then proceed to apply this method on the image with different Gaussian kernels and thresholds. We make observations on the effect of these difference on the magnitude and orientation calculation result of this method.

A. Gradient Components

In this section we are displaying and examining the partial derivatives of an image along the horizontal and vertical directions.

original image

We calculate the partial derivative of the image along the rows.

From observation of the partial derivative image along the rows, we notice the dark lines are where the white transits into black from left to right in the original image. Meanwhile, the white part of the derivative image is where black transits to white from left to right. The gray parts are where there are insignificant color changes. This is because the brightness in the derivative graph reflects the changes in slopes of the original image. Specifically, a black to white transition results in a postive slope, which the image displays as high brightness.

We calculate the partial derivative of the image along the columns.

From observation of the partial derivative image along the columns, we observe that the dark lines are where the white transits into black from top to bottom in the original image. Meanwhile, the white part of the derivative image is where black transits to white from top to bottom. The gray parts are where there are insignificant color changes. This is because the brightness in the derivative graph reflects the changes in slopes of the original image. Specifically, a black to white transition results in a postive slope, which the image displays as high brightness. This is the same result as the previous image, however the orientation of the gradient changes from horizontal to vertical.

B. Processing Images

In this section we display and inspect the image of the gradients' magnitude.

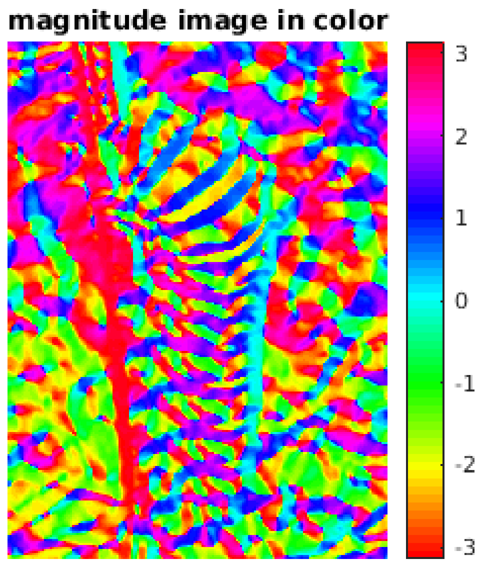
magnitude of gradience

We observe the strongest responses along the edges of the image where white is transitioned to black quickly. We see that the head of the caterpillar has higher response than its body. However, we do not know if a direct correlation between radiance and this magnitude exists. We notice that the strong responses correspond to regions with strong black and white color in our previous two gradient images. Since we are taking the magnitude, we only get information about how large the color change is, as the image does not encode information about the directions of the gradient changes.

C. Gradient Orientation

We create and inspect the images representing the orientation/direction of the gradient.

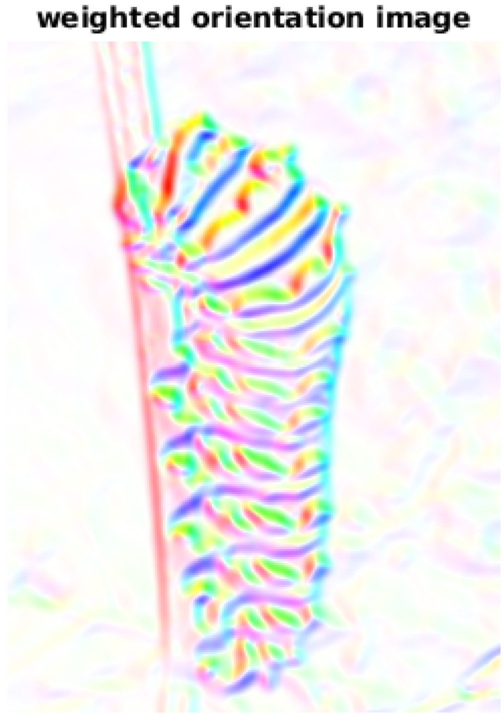
We display the magnitude image with the magnitude value of each pixels mapped to a color linearly.



As the scale of the color map is from -pi to pi, we deduct that colors represent a round angle. The color at points pi, - pi and 0 correspond to the gradient in vertical direction. These colors are bright red and cyan blue respectively. The difference of gradients of blue and red is the direction of the gradient. Similarly, the color at pi/2 and negative pi/2 correspond to gradients in horizontal direction. These colors are purple and green respectively. The purple color represents upper left to lower right, and yellow/orange represents lower left to upper right.

D. Gradient Orientation Revisited

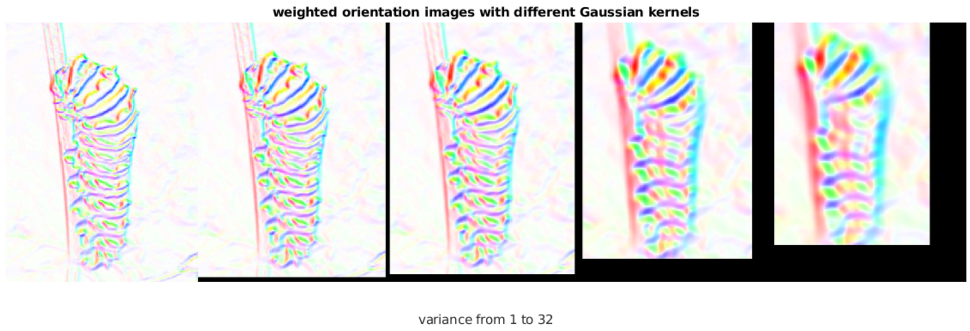
We seperate color into three components: hue, saturation, value in order to enconde orientation with hue, encode strength of the edge with saturation. Using this method, we can color only the areas of strong responses, where the edges usually are.

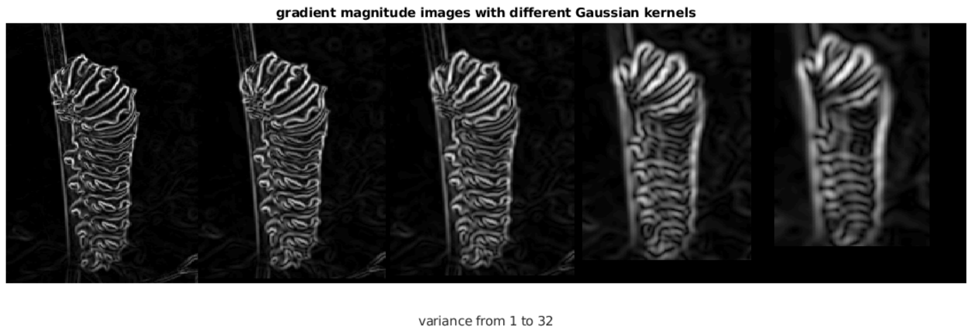


We confirmed that the edges are bright and color matches the orientation.

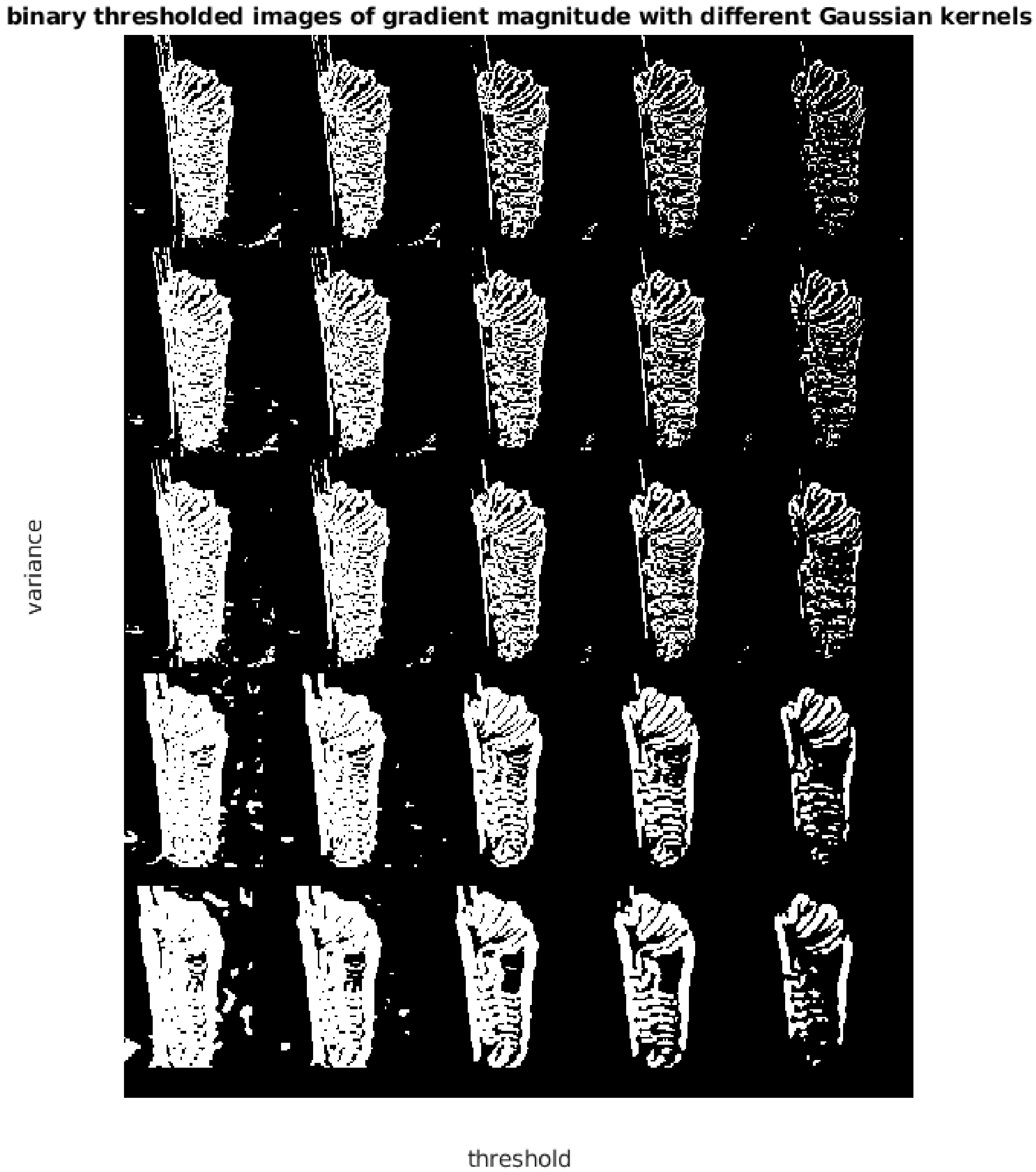
E. Edge Detection and Scale

In this part, we test the edge detection method with different Gaussian variance kernels, and different thresholds, and examine their effects. We apply the edge detection method to the image with an array [1 2 4 16 32] of different variance and display the weighted orientation images and magnitude images.





In order to have binary detection, we need to threshold our edges. We add for loop over several gradient magnitude thresholds and store the result in a 4D array.



F. Analysis

In this final section we perform analysis of the results we obtained in the previous section.

Magnitude images

As the Gaussian standard deviation increases, we immediately noticed changes in image-size and bluring. In detail, with larger Gaussian standard deviation, the black edges around the images get thicker. Since we position our images at top left, visually the lower right corners appear to get thicker black bars. Secondly, the bluring effect is also highly visible at greater scales. High-scale images have significantly softer edges, and also loose of gradient magnitude with different Gaussian kernelsfiner curvature details. We also noticed that the white details in the backgrounds of high-scale images are more visible.

Weighted orientation images

We noticed the some similar effects of high-scale images between the orientation images and the magnitude images. Namely the image-size reduction are still clear here, higher Gaussian standard deviations result in smaller images. Overall, the color hue does not change much between images. However, the blurring effect is displayed clearly here as well. The images with the most hue variation on edges is the lowest-scaled image. As the variance increases, the distinct lines of different hues adjacent to each other mix and the results are single shade dots of color. For example, the details on the caterpillar back are represented as lines of changing colors on low-scale images. On high-scale images, these details become wide dots of one color. Even color edges that do not mix with other edges still become thick and blurry with higher variance. Also, as the lines become blurrier, the saturition of the color decreases.

Binary threshold images

As the threshold decreases or the variance increases, the white lines get thicker. When the image has a high threshold and a low variance, the lines of the image are very thin, but there are some false negatives. For the image with a low threshold and a high variance, there are a lot false positives and the lines are thick. With a high variance, the edges where there are changes in high frequency (for example the back of the bug) in the original image are not detected. With a low threshold, the images include a lot of false positives in the form of white dots obscuring the relevant details.

Conclusion

Edge detection by calculating the image gradients can have significantly varying results after the application of different Gaussian kernels and thresholds. Awareness of this fact will allow us to create edge images with minimal noises and artifacts.

Acknowledgement

We referenced the houghlab.m script by Professor Jerod Weinman. We used the code provided by Professor Jerod Weinman in https://www.cs.grinnell.edu/~weinman/courses/CSC262/2019S/labs/edges.html The original image is from /home/weinman/courses/CSC262/images/bug.png

Published with MATLAB® R2018b