

Problem Set 1

Short Answer:

1. You can use convolution to perform tasks like grouping or texture matching.
2. The result is [0 1 1 1 1 1 1].
- 3.
4. This can be accomplished by increasing the initial threshold gradient or by using a high hysteresis threshold.
- 5.
6. Assuming the parts are positioned the same way on the conveyor belt you can set up a template (based on shape, color and/or texture) that the parts should follow as they go down the line. The parts should appear at regular intervals so a snapshot can be taken at the correct time to run a template match on all of the items.

Programming problems:

When coding this project, I found a couple of built-in functions to help me. In the energy function portion, I used the absolute value of the built-in gradient function to calculate the energy. When calculating the minimum cumulative energy values, I used the `ordfilt2` function to handle the majority of the work. To then go and find the seam, I coded mostly everything by hand, finding the lowest value on the bottom row and working my way up from there.

Programming Problems:

1)

Original



Here is the original image

Prague simple resize (380x640)



Here is the simple resized version. The tower buildings are noticeably shorter.

Prague smart resize (380x640)



Here is the smart resized version. While trying to take out much of the sky the tops of the buildings have been slightly clipped.



Here is the original.

Mail simple resize (769x675)



In this simple resize, everything is noticeably thinner.

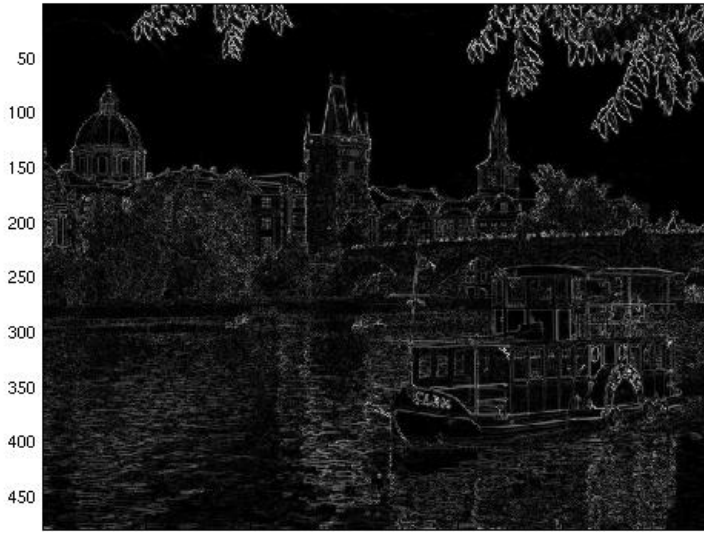
Mail smart resize (769x675)



In the smart resize version, the trees definitely show signs of being trimmed while the rest looks normal.

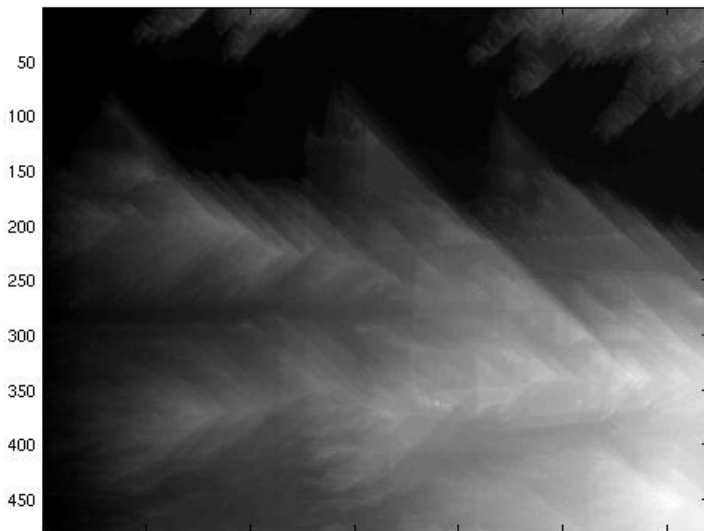
2)

Prague energy values



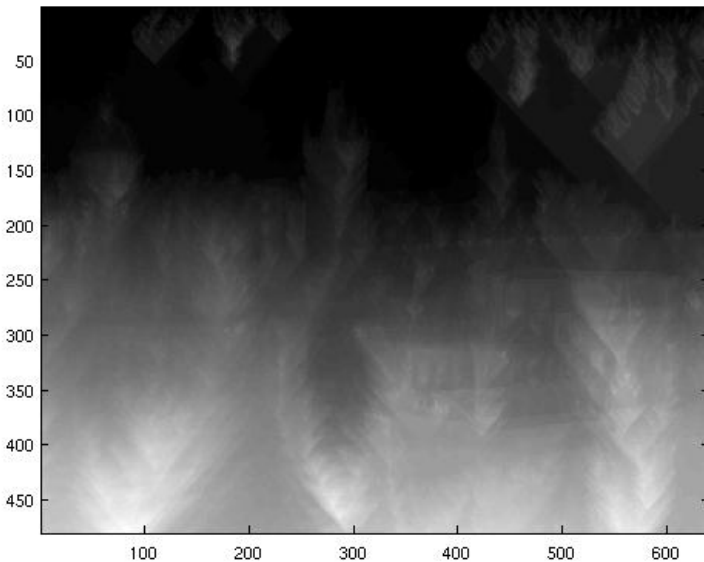
In the energy plot, the areas with high contrast are noted. These areas are mostly the edges of the buildings and the boat. The water and the tree also have high values.

Prague horizontal energy sum



Here in the horizontal energy sum plot, the tapering shadows from the buildings, water, boat, and tree can be seen heading off to the right.

Prague verticle energy sum



Here we can see the short lives shadows of the tree values above the large collection of building, water, and boat values.

3)

Prague first horizontal seam



Here the first seam found attempts to take around parts of the sky, trying to avoid the tops of the towers and the more prominent clouds.

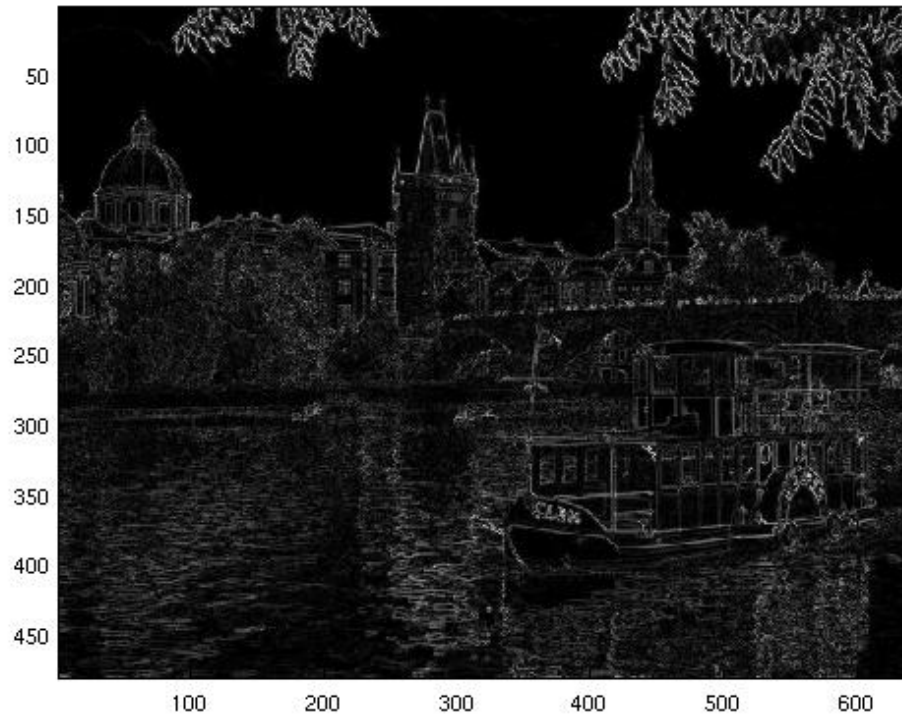
Prague first verticle seam



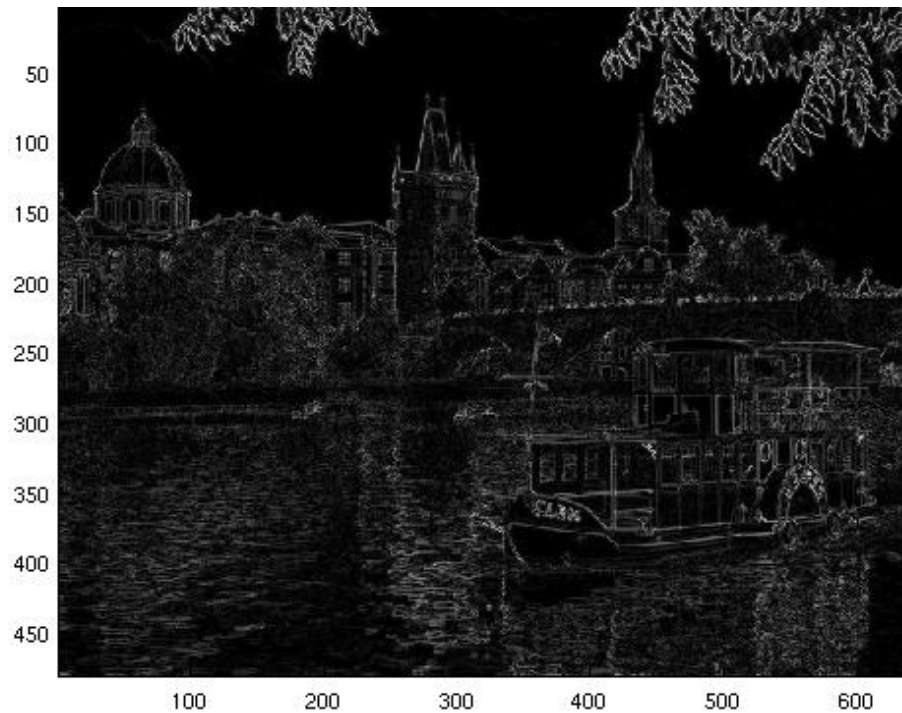
Here the vertical seam tries to remove the shadow between two buildings as well as part of the sky and water.

4)

Prague energy values



Here is the original energy function, using the gradient function in Matlab.



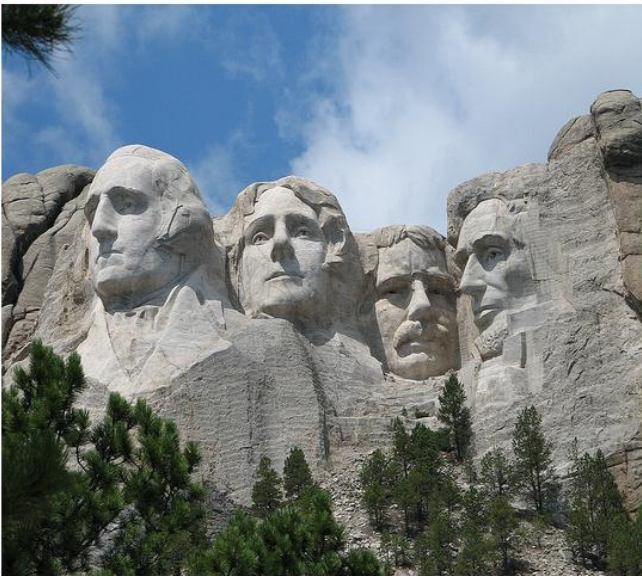
In this second version I implemented the “square root of the sum of the squared derivatives” version of the energy function. I can’t tell the difference. Did not affect seaming noticeably.

5)



I got a picture of Mt Rushmore thinking that it would be difficult of the horizontal resizing to preserve the information in the faces. As it turned out, it did ok at preserving the faces, only Washington's hair was very distorted. The vertical resize however did perfectly, cutting out the sky above the faces.

Mt. Rushmore simple resize (480x540)



Mt. Rushmore smart resize (480x540)





In this picture of a sandcastle, I was wondering how the vertical seam finder would do with the image of the rippling water. To a human, the sandcastle would be more important; however, the energy function I used did not agree and clipped the dark side of the castle. Incorporating a portion of the energy to be decided by texture would probably fix this.

Sand castle simple resize (536x540)



Sand castle smart resize (536x540)

