CSC 514: Computer Vision: Homework #6

Due on Thursday, October 22, 2015

Hoover 11:00am

Julian A. Brackins

Problem 1

In class we discussed several different methods for performing filtering/template matching on images. In this homework you will create your own Matlab functions to perform both convolution and correlation on images using different convolution/correlation kernels.

Preface

Before we can answer *Where's Waldo*, we must first ask the question, *Who* is Waldo?
Where's Wally?, also known as *Where's Waldo* in the United States and Canada, is a children's book series created by English Illustrator Martin Handford. The books consist of double-paged illustrations containing hundreds of persons in a given location. In each scene, Waldo is hidden amongst the group, wearing his signature red and white striped shirt, bobble hat, and spectacles. *(source: Wikipedia)*
This assignment involves automating the process of finding Waldo in a Where's Waldo scene. This will be done using correlation to match features in two different images.

Equations

The two equations being used for this assignment are the Correlation and Convolution equations.

Correlation:

$$G\Big[i,j\Big] = \sum_{u=-k}^{k}\sum_{v=-k}^{k} H\Big[u,v\Big]F\Big[i+u,j+v\Big]$$

Where F is the image and H is the kernel.
This is referred to as cross-correlation. In Image filtering, convolution is performed by replacing each pixel with a linear combination of its neighbors. Then, the *filter* (interchangeably referred to as a *kernel* or *mask* henceforth) is used to describe the weights in the linear combination. For any given pixel in the image, the greater the intensity value after this correlation filter is applied, the closer that pixel is in relation to the kernel image. Figure 2 details the intensity map generated from performing cross correlation on the Waldo Scene and Waldo Kernel.
Convolution:

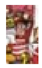
$$G\Big[i,j\Big] = \sum_{u=-k}^{k}\sum_{v=-k}^{k} H\Big[u,v\Big]F\Big[i-u,j-v\Big]$$

Where F is the image and H is the kernel.
Convolution can actually be performed using Correlation. Flip the kernel in both directions (or rotate the kernel 180°), then apply correlation.

Process

The following is the process used to find Waldo.

- Read in the Kernel and Scene Images
- Convert Kernel and Scene to Grayscale
- Find Dimensions of the Scene and Kernel



(a) Image Kernel

(b) Waldo Scene

Figure 1: Waldo Images

- Divide Kernel in Half so that we work with a smaller kernel size.
- Calculate Correlation or Convolution.
- Find Waldo by locating the highest value point in the G Matrix.
- Display Original Scene with Kernel superimposed on Waldo's location.

Correlation

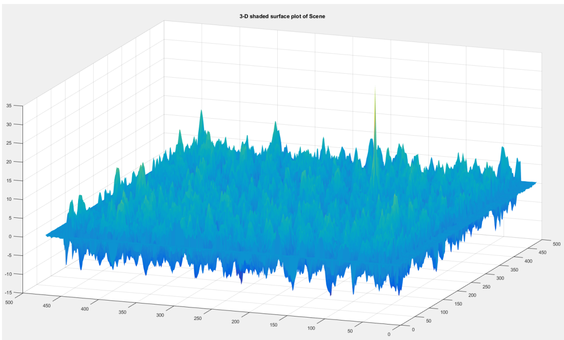
The following pseudocode is the process for calculating Correlation:

- Find the dimensions for the scene and kernel
- Set H = kernel, F = scene, G = result so that program matches formulas above
- Divide Kernel so that we work with a smaller kernel size.
- Generate Mean for Scene and Kernel
- Perform Correlation by applying the weighted kernel value to each pixel in the image.

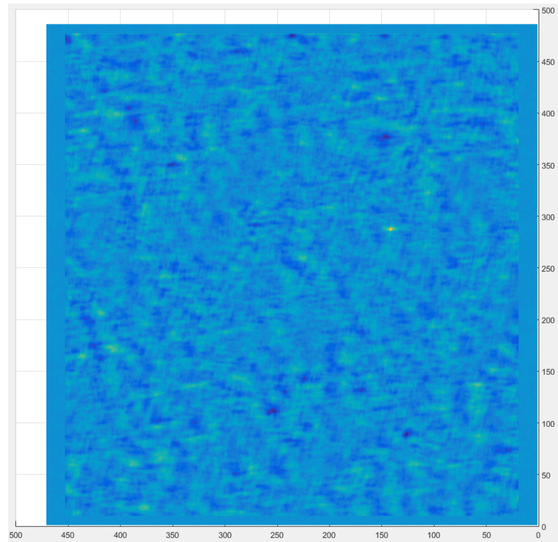
The resultant G Matrix generated by the correlation function yields the 3D surface mapping found in Figure 2.
The image displayed in Figure 3a is the scene after the correlation filter is applied. A noticeable peak is observed in the image, which will be used as a guess for Waldo's Location. Using the x,y coordinates of that peak, we can now superimpose the kernel onto the original image, as shown in Figure 3b.

Convolution

As stated earlier, Convolution is done similarly to Correlation. However, the resultant image returned from performing Convolution (Figure 4) did not produce the same accurate guess for Waldo's location that was observed in the Correlation equation.

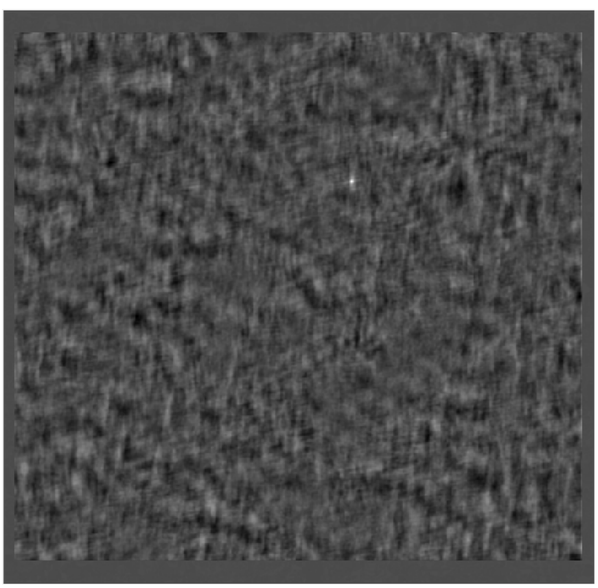


(a) 3D Surface Graph.

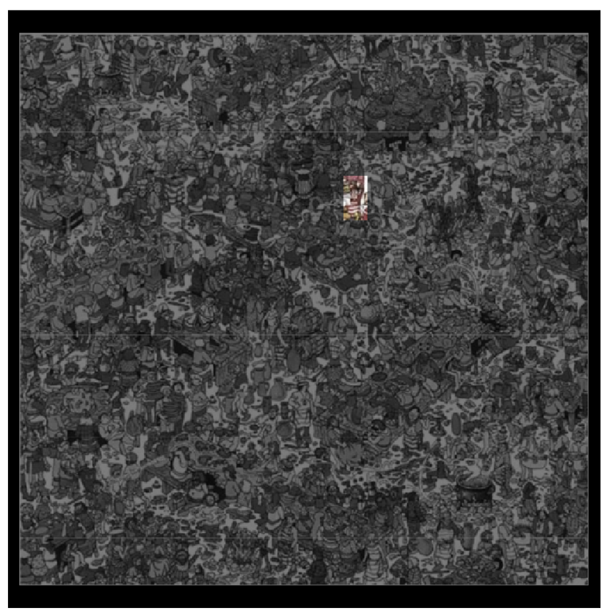


(b) Top-Down View.

Figure 2: 3D surface of image after applying correlation

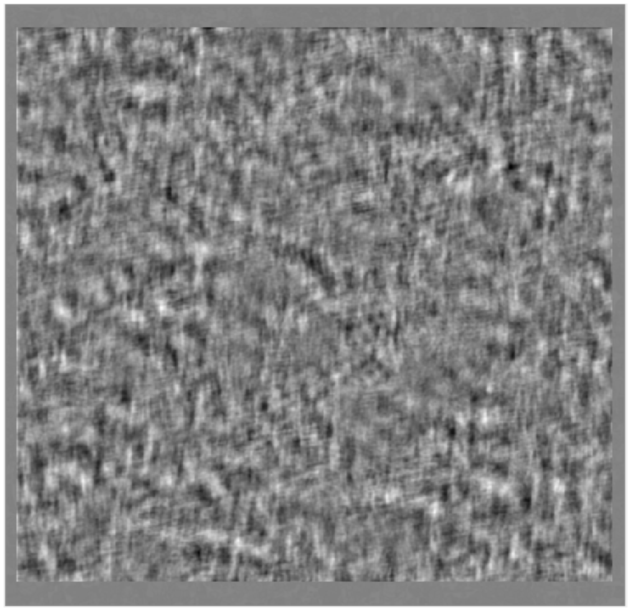


(a) Image after applying Correlation



(b) Kernel superimposed onto original scene.

Figure 3: Finding Waldo



(a) Image after applying Correlation

Figure 4: Image after applying Convolution

Matlab

The Matlab Code for the program attached after the conclusion of this Homework Document.

Conclusion

This assignment was a study in utilizing template matching to discover similar features between two images. This was done by writing Matlab functions to perform correlation and convolution.
Correlation and Convolution are two different methods for filtering discussed and examined during this homework assignment. Correlation worked well as the Waldo search algorithm because the formula focuses on finding the similarities between two matrices. On the other hand, the convolution formula did not work well for finding Waldo, as convolution is more useful in discovering how two matrices differ from one another.

Listing 1: Where's Waldo Script

```
function wheres_waldo()
    %%Read in Kernel, convert to grayscale double. Save original copy.
    kernel = imread('WaldoKernel.png');
    orig_kern = kernel;
    kernel = rgb2gray(kernel);
    kernel = im2double(kernel);

    %%Read in Scene, convert to grayscale double. Save original copy.
    scene = imread('WaldoScene.png');
    orig_scene = scene;
    scene = rgb2gray(scene);
    scene = im2double(scene);

    %%Find the dimensions for the scene and kernel
    [kH, kW] = size(kernel);
    [sH, sW] = size(scene);

     %%Divide Kernel Height and Width
    kH = kH/2;
    kW = kW/2;

    %%Calculate Correlation
    G =  correlation( scene, kernel );
    %%OR you can do convolution...
    %%G =  convolution( scene, kernel );
    %%Actually don't because convlution is trash for this program...

    %%Find Waldo by finding the highest value point
    [r,c] = find(G==max(G(:)));

    %%Pad array so that the imposed image lines up properly
    scene  = padarray(scene,[kH,kW]);
    orig_kern = padarray(orig_kern,[r,c],'pre');
    orig_kern = padarray(orig_kern,[sH-r,sW-c],'post');

    %%Show the Original Image
    figure, imshow(orig_scene,[]);
    title('Original Scene');

    %%Show the surf Image
    figure, surf(G), shading flat;
    title('Quantized Samples');

    %%Show the G matrix
    figure, imshow(G,[]);
    title('Waldo Guess location');

    %%Show the Search Result
    figure, imshowpair(scene(:,:,1),orig_kern,'blend');
    title('Kernel Superimposed on Original Image');

end
```

Listing 2: Correlation Function

```
function [ G ] = correlation( scene, kernel )
    %%Find the dimensions for the scene and kernel
    [kH, kW] = size(kernel);
    [sH, sW] = size(scene);


    %%set F, G, H matrices so that they match what's in the book.
    G = scene;
    F = scene;
    H = kernel;

    %%Divide Kernel Height and Width so we work with a smaller kern
    kH = kH/2;
    kW = kW/2;

    %%Generate Mean for Scene and Kernel
    meanS = mean(mean(scene));
    meanK = mean(mean(kernel));


    %%Perform Correlation
    for i=(kH):(sH-kH)
        for j=(kW):(sW-kW)
            G(i,j) = sum(sum((F(i-kH+1:i+kH, j-kW+1:j+kW)-meanS).*(H-meanK)));
        end
    end
end
```

Listing 3: Convolution Function

```
    %%For Convolution, just Flip the filter in both directions
    kernel = rot90(kernel,2);
    [kH, kW] = size(kernel);
    [sH, sW] = size(scene);

    G = scene;
    F = scene;
    H = kernel;

    kH = kH/2;
    kW = kW/2;

    meanS = mean(mean(scene));
    meanK = mean(mean(kernel));

    %%Perform Correlation
    for i=(kH):(sH-kH)
        for j=(kW):(sW-kW)
            G(i,j) = sum(sum((F(i-kH+1:i+kH, j-kW+1:j+kW)-meanS).*(H-meanK)));
        end
    end
end
```