MONICA STETTLER

CSC481/381 Final Paper - 11/15/2018

TITLE

Automated segmentation of clothing ensembles from Pinterest

ABSTRACT

Fashion bloggers provide beautiful images of complete clothing ensembles, fully put together with accessories such as a purse, shoes, belt, earrings, scarves etc. What is lacking is the ability to buy the items in the image. In an ideal world, one would be able to click on any of the items in the outfit and be taken to a site to purchase the exact outfit or specific item. Second best would be to feed the image into an app that separates the image of the ensemble into many images– with each new image showing only one item. These single item images would then be used to search and match on an online shopping site such as Amazon.

After an extensive search, I was not able to find such an app. Some require the user to feed a perfect image to the app – one with a clean background and only one item in it. Some only let you use images that are already in their database with links to vendors with relationships with the app. One app I tried linked to shopping sites where prices for a simple blouse ranged from $400-$5,000.

My goal was to accomplish phase one of the dream app described above using MATLAB. I set out to take an image of a cute outfit with accessories from Pinterest and segment each item from the image to one item per image. For example, one image might become eight: a skirt (by itself), a blouse, a jacket, a scarf, shoes, purse, earrings and belt. The goal was also for this to be automatic, with no user intervention required.

INTRODUCTION

This project is very personal for me. I have spent most of my life wearing either a suit or yoga pants and t-shirts. Casual dress for me was my suit without the jacket. When I started the MS in Data Science program at DePaul a year ago, I realized it was time I upped my game and invest in casual/business casual wardrobe that I could wear to class and for future jobs. I discovered on Pinterest that fashion bloggers would post cute, stylish outfits that appealed to me. I began collecting the images on a Pinterest board. When I was ready to invest in a few new outfits, I found I that I couldn't but them. I had expected to click on the image and be launched to a shopping site or to the bloggers site with links so that they would get credit. Food bloggers have

this mastered. If they love something, they get you hooked and then provide the link to buy it on Amazon.

Given that the fashion industry is estimated to be $3 trillion, I figured that someone would have developed the app I need. I searched but could not find one. There is clearly a need in the marketplace for an app like this.

Without such an app, I have to take the Pinterest image and spend hours scouring Amazon and other shopping sites in order to try to find and buy the pieces. Often, I am not able to find the items. I don't enjoy shopping and I don't have time to shop. I need this app!

BACKGROUND

There has been a lot of research on segmentation. Having said that, I was not able to find anyone who had done work on my specific topic. Much of the existing work has been on isolating one item in an image.

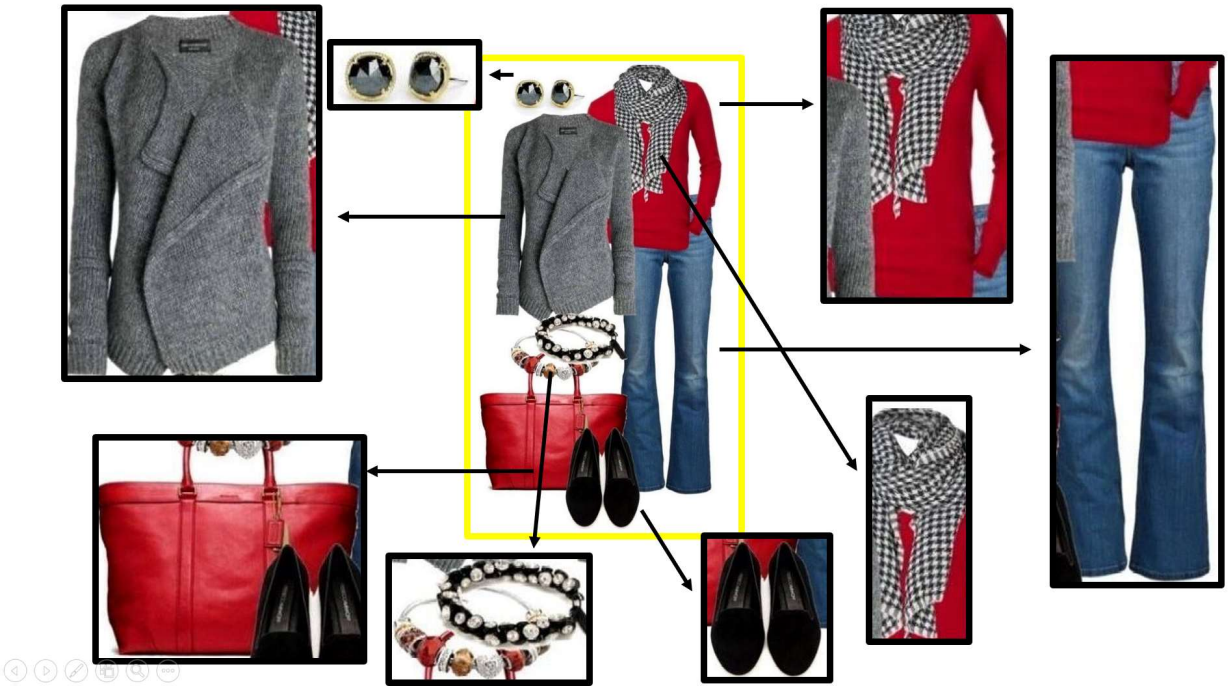
In the paper by Manfredi et al, their goal was to strip down the image to one target item of clothing and then classify its color and item type. They wanted a system that would help automatically annotate and tag the photos to replace the manual process that retailers currently use. Sujatha et al provided an analysis of edge detection methods and determined under which conditions each method performed the best. Chen et al tried to increase the speed of K-Mean clustering by using histogram quantization in HSV color space. Wang et al performed clothing segmentation using a blocking model. They worked on images of groups of people where body parts and clothing were blocked by people next to them. Their work could be interesting in future rounds of my project, especially when segmenting photos with human models. Yang et al tried to develop an integrated system of clothing co-parsing, however their work relied on tags.

METHODS

Before working on an automated process, I thought I should go through the process manually. I think I tried every process, tool and algorithm in the book. Many of the techniques we have learned in class so far are based on color or intensity. Given that my images are clothing items and accessories in an outfit, many of the pieces match. This means several items will be the same color. Using techniques like thresholding would pull out several items that shared color or intensity.

The goal of the project was to automatically segment every item in an image of a clothing ensemble. Below is an example of approximately what this would look like. The original image is the one in the middle in the yellow box. The process on this particular image would result in 8 separate, clean images of each of the individual items in the image. As I was not able to

segment all the items using image process techniques we learned in class, I used the MS Word crop tool to create the example below.



After attempting a number of different methodologies, I found a tool in MATLAB called 'bwareafilt() that would extract an object based on its size. I set it to extract the largest object in the image. The system I created automatically ran the following steps:

1) Transform image to grayscale.
2) Convert grayscale image to binary.
3) Run the 'bwareafilt' tool and extract the largest object.
4) Run the mask over the original image to return the original color to the segmented item.
5) Transform the background back to white to create the final image of that item.
6) Remove the extracted object from the starting grayscale image.
7) Repeat the process 10 times (based on assumption that no image would have more than 10 items to segment).

RESULTS

For certain images, I was able to extract a few items fairly well. I was not able to extract all items from any of the 14 images I tested. Given that bwareafilt() requires the image to be binary, the conversion process loses light or white colored objects. I tried to do a 'reverse'

binary process, but that didn't work. Stripes and plaid were a big challenge and viewed by the tool as separate objects.

challenges: matching or similar colors

9 items

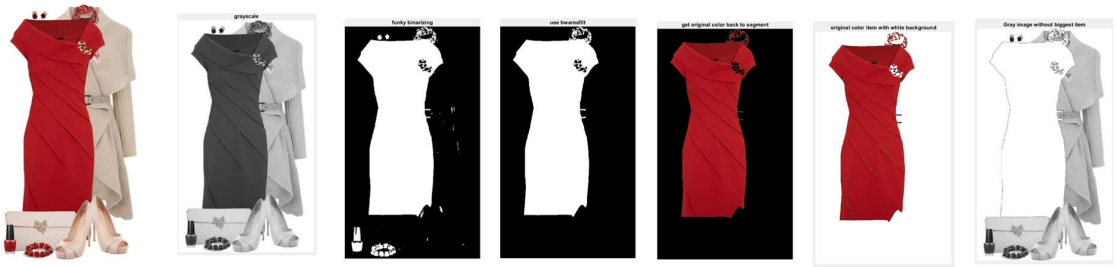
6 items

9 items

many pieces in ensemble - varies per outfit

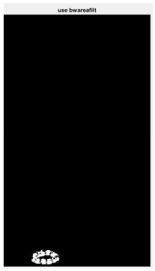
The image below shows the step by step process of my system. For the sample image, I had a 50% (ish) success rate. I was able to segment out the red dress, the red bracelet and the nail polish (although the nail polish came out as 2 images that I had to put back together manually). I was not able to extract the sweater, shoes nor purse as they effectively disappeared when the image was converted to binary.

results: pass #1



1) original 2) grayscale 3) binarized 4) 'bwareafilt'-ized 5) mask*orig 6) backgr 2 white 7) orig minus item

results: pass #2



1) grays after pass #1 2) binarized 3) 'bwareafilt'-ized 4) mask*orig 5) backgr 2 white 6) orig minus item

Note: binarizing wipes out the sweater, shoes and purse.
I tried 'reverse' binary transformation, but that didn't help. I could never get the sweater, shoes and purse.

In the next sample image, the scarf came out great and the earrings reasonably well. I think I would be able to use the segmented jacket image, but I would need to manually white out the striped dress underneath.

The tool had a very difficult time with plaid. It picked up little squares of it as “largest object”. The boots, purse and jeans came out quite well.

Not surprisingly, the tool had a difficult time with the stripes. I was surprised though that it cut the dress after the strip below the belt. I also would have expected it to grab the shoes and not cluster them with the purse, bracelet and bottom stripe. It grabbed each earring, but they look more like red finger prints than earrings.

The next sample was tricky given the tone and intensity of the colors. Everything is almost the same color. In addition, the background in this one isn't perfectly white. I didn't think I would get anything out of it and was happy to see how well the tank top and scarf came out.

The results of the next sample image surprised me the most. I thought I would get some good segmentation here. However, the tool identified nearly the entire image as one "largest" object. Color/intensity thresholding probably would have worked better than bwareafilt() on this image.

fail

1) Original 2) grayscale 3) binary 4) largest object

this is first pass… look at "largest object"

CONCLUSIONS

This project was really challenging! In the future, I would like to try some additional techniques. I attempted the quad tree decomposition, but was unable to get the code to work. I thought I could break down the image, regrow certain regions and then pass them through bwareafilt(). I am also curious to see what would happen with SVM. I read about it in a number of research papers, but didn't have enough proficiency to attempt it at this time or really see how it might apply. Future work would also include making the code more efficient.

REFERENCES

1) A complete system for garment segmentation and color classification.
Manfredi, M., Grana, C., Calderara, S., & Cucchiara, R. (2014). A complete system for garment segmentation and color classification. *Machine Vision and Applications*, *25*(4), 955-969.

2) Performance analysis of different edge detection techniques for image segmentation.
Sujatha, P., & Sudha, K. K. (2015). Performance analysis of different edge detection techniques for image segmentation. *Indian Journal of Science and Technology*, *8*(14).

3) Fast image segmentation based on K-Means clustering with histograms in HSV color space.
Chen, T. W., Chen, Y. L., & Chien, S. Y. (2008, October). Fast image segmentation based on K-Means clustering with histograms in HSV color space. In *multimedia signal processing, 2008 IEEE 10th Workshop on* (pp. 322-325). IEEE.

4) Who blocks who: Simultaneous clothing segmentation for grouping images.
Wang, N., & Ai, H. (2011, November). Who blocks who: Simultaneous clothing segmentation for grouping images. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 1535-1542). IEEE.

5) Clothing co-parsing by joint image segmentation and labeling.
Yang, W., Luo, P., & Lin, L. (2014). Clothing co-parsing by joint image segmentation and labeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3182-3189).

CODE

```
%I
=imread('https://monicastettler.files.wordpress.com/2018/10/reddressbeighjack
et.jpg');
I =
imread('https://monicastettler.files.wordpress.com/2018/10/pinkdressgrayjacke
t.jpg');
%I
=imread('https://monicastettler.files.wordpress.com/2018/10/blackdressbrownac
cess.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/whitejeansandjacke
t.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/stripedshirtwhitej
eans.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/stripeddressandjac
ket.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/purplejersey.jpg')
;
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/plaidshirt.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/jeansandjacket.jpg
');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/pencilskirt.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/stripedblackdressj
eanjacket.jpg');
%I =
imread('https://monicastettler.files.wordpress.com/2018/11/graysweater.jpg');
%I = imread('https://monicastettler.files.wordpress.com/2018/11/gray.jpg');

Ig = rgb2gray(I);

figure
imshow(Ig)
title('grayscale')

I2 = Ig < 100; %method 1 to binarize
%I2 = imbinarize(I,'global'); %method 2 to binarize

%%%% method 3 to binarize %%%%%%%%%%%%%%%%%%%%
%meanValue = mean(Ig(:))

%I2    = zeros(size(Ig));  %method 3 to binarize
%for i = 1:size(Ig,1)
    %for j = 1:size(Ig,2)
     %   if I2(i,j) < meanValue
      %      newI(i,j) = 0;
       % else
        %    newI(i,j) = 1;
```

```
        %end
   % end
%end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure
imshow(I2)
title('binarized')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')


%I want to get rid of biggest item from grayscale image by whiting it out
g2 = Ig; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
```

```
            g2(i,j) = 255;
        end
    end
end

figure
imshow(g2)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%%  ROUND 2  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
I2 = g2 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')
```

```
%I want to get rid of biggest item from grayscale image by whiting it out
g3 = g2; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g3(i,j) = 255;
        end
    end
end

figure
imshow(g3)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%  ROUND 3 %%%%%%%%%%%%%%%%%%%%%%%

I2 = g3 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
```

```
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g4 = g3; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g4(i,j) = 255;
        end
    end
end

figure
imshow(g4)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%% ROUND 4 %%%%%%%%%%%%%%%%%%%%%%%%
I2 = g4 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end
```

```
for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g5 = g4; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g5(i,j) = 255;
        end
    end
end

figure
imshow(g5)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%% ROUND 5 %%%%%%%%%%%%%%%%%%%%%%%%
I2 = g5 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
```

```
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g6 = g5; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g6(i,j) = 255;
        end
    end
end

figure
imshow(g6)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%% ROUND 6 %%%%%%%%%%%%%%%%%%%%%%%
I2 = g6 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
```

```
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g7 = g6; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g7(i,j) = 255;
        end
    end
end

figure
imshow(g7)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%% ROUND 7 %%%%%%%%%%%%%%%%%%%%%%%%
I2 = g7 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
```

```
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g8 = g7; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g8(i,j) = 255;
        end
    end
end

figure
imshow(g8)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%% ROUND 8 %%%%%%%%%%%%%%%%%%%%%%%
I2 = g8 < 100;
figure
imshow(I2)
title('funky binarizing')

I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')
```

```
red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g9 = g8; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g9(i,j) = 255;
        end
    end
end

figure
imshow(g9)
title('Gray image without biggest item')

%%%%%%%%%%%%%%%%%%% ROUND 9 %%%%%%%%%%%%%%%%%%%%%%%%%
I2 = g9 < 75;
figure
imshow(I2)
title('funky binarizing')
```

```
I3 = bwareafilt(I2,1); %pick largest object
figure
imshow(I3)
title('use bwareafilt')

red = I(:,:,1);
green = I(:,:,2);
blue = I(:,:,3);

maskr = immultiply(red, I3);
maskg = immultiply(green,I3);
maskb = immultiply(blue,I3);

for i = 1:size(maskr,1)
    for j = 1:size(maskr,2)
        if maskr(i,j) ==0
            maskr(i,j) = 255;
        end
    end
end

for i = 1:size(maskg,1)
    for j = 1:size(maskg,2)
        if maskg(i,j) ==0
            maskg(i,j) = 255;
        end
    end
end

for i = 1:size(maskb,1)
    for j = 1:size(maskb,2)
        if maskb(i,j) ==0
            maskb(i,j) = 255;
        end
    end
end

seg = cat(3, maskr,maskg,maskb);
figure
imshow(seg)
title('get original color back to segment')

%I want to get rid of biggest item from grayscale image by whiting it out
g10 = g9; %initial reduced original gray image
for i = 1:size(Ig,1)
    for j = 1:size(Ig,2)
        if I3(i,j) ==1 %if the mask (bwareafilt) has white, then make white
in grayscale
            g10(i,j) = 255;
        end
    end
end

figure
imshow(g10)
title('Gray image without biggest item')
```