EECS 490 DIGITAL IMAGE PROCESSING

December 2004

SESSION 2 Inspection of Cookies

Wen-Teng Chang	Chocolate Chips Counting by Multiple Image Operation
James Eastman	Automated Tallying of Chocolate Chips Using Color Segmentation
Dmitriy Goldman	Counting the Chocolate Chips
Issac Hirt	Quality Control of Chocolate Chips through Chip Counting
Svend Johannsen	Counting Chocolate Chips in Chocolate Chip Cookies
Ruchi Kothari	Estimating Number of Chocolate Chips in Cookies
Daniel Pendergast	Feature Detection of test Images in Food Quality Control
Chris Roberts	Surface Feature Detection for Quality Assurance with Cookies
Ira Ross	Counting and Locating Chocolate Chips Using Color Segmentation and Image Morphology
Yu-Hong Yen	Counting Chocolate Chips

Chocolate Chips Counting by Multiple Image operation

Chang, Wen-Teng

Department of Electrical Engineering and Computer Science,

Case Western Reserve University, Cleveland, OH, Email: wxc31@cwru.edu

Abstract

This project uses multiple image operations to count number of doped materials on a different base. A couple of random size and shape chocolate chips are casually distributed on a cookie as an example in this project. We develop an algorithm with multi step image processing techniques, including color segmentation, filtering and clustering techniques to approach an accurate count of doped chocolate chips.

KEY WORDS

Color segmentation, nonlinear filtering, clustering, morphological closing and image dilation

INTRODUCTION

Along with mass production by automated equipment in various industries, quality control (QC) has become a part of automated chain. To simulate and replace human eyes' function, image patterning and recognition provide the role in quality control with low cost, particularly in food, medical industries and so on.

This project develops an algorithm for the solution in counting doped material, or chocolate chips distributed on cookie in this case. The solution provides an automatic quality audit in counting a distributed chocolate chips. However, an individual operator is not sufficient to develop the solution because the image analysis problems are complex and can only be solved by a combination of elementary techniques. The multiple techniques in developing the algorithm and the error types are also discussed in this project

OVERVIEW

This project will demonstrate multi steps image process techniques to count doped chocolate chips on cookies. A single operation can hardly achieve this purpose due to the following reasons: (1) a single chocolate chip may be partially occulted by cookie so single color segmentation operation cannot distinguish the error. (2) Noise from scanner or cookie and chocolate is required to be removed by filter. (3) Clustering and erosion techniques are required in order to avoid over-count or under-count chips because essentially an object appearing on an image is consisted of a cluster of pixels. Proper clustering and noise cancellation can lead to an accurate count in this case.

The sequence for the above operations is important. Basically, color segmentation can filter potential candidate of pixels counted for chocolate. Nonlinear spatial filtering is then used to filter noise caused from ambient environment. In this project, we can see vertical edge noise repeatedly appears after color selection. Followed by filtering, multiple stages of combining dilation and erosion (nonlinear filtering used instead of erosion in this case) can help to get more accurate count of chocolate chips. Finally, black and white label counts command send out the result and we paint colors back to black and white image to verify our result

TECHNIQUE DISCUSSION

1 Color Segmentation

There are several ways one can quantitatively specify a color, such as RGB format, HSI format. The images we can download are mostly RGB format. However, HSI format is designed the way humans see color [1]. The particular color such as chocolate can be easily recognized from HSI format [2].

$I1=(R+G+B)/3;$ (1)

$I2=(R-B)/2;$ (2)

$I3=(2G-R-B)/4;$ (3)

Hue, saturation and intensity (H,S and I) are derived as

$H=\tan^{-1}(I3/I2);$ (4)

$S=(I2^2+I3^2)^{1/2}$ (5)

$I=I1$ (6)

By statistical method from HSI image ("pixval" shows its value), we choose the desired ranges individually that stands for chocolate's value). Figure 1 and 2 stand for RGB and HSI format respectively, of chocolate chips on cookie.

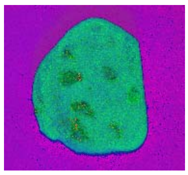


Figure 1 RGB format Figure 2 HSI format

2 Pre- filtering

Some ambient noise caused from equipment, such as scanner or camera quality can be filtered. For example, we found the noise causally appear along with vertical edge of the cookie because the vertical edges have similar value of H, S and I with selected chocolate chips. Figure 3 show the edge of the cookie has vertical 1s. The pre-filter is then chosen by a length 3 and width 9 median filter to remove vertical noise. Nonlinear of rectangular filter, such as median filter is useful in deleting certain direction of noise.

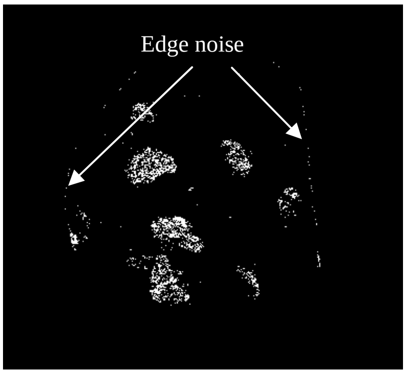


Figure 3. Vertical edge noise can be cancelled by a 1:3 median filter

3 Clustering

The purpose of clustering is to connect nearby pixels and form an integrated object so that a computer can recognize that it is a single object. Since a single chocolate chip may be centrally occulted by the cookie, which may be mistaken as two chips. Therefore, we consider a large enough morphological closing to connect nearby pixels. On the contrary, we don't want to misjudge two separate chips as one either, so we make another nonlinear filtering between the two image dilation to reduce chance of the wrong connection.

After 3 time repeated operations, the integrated objects are formed step by step and shown from figure 4 to figure 6. In figure 4, we find the first time nonlinear filtering and dilation removes some partially exposure chocolate chips but unite a cluster of 1s. The second time and the third time combining dilation and erosion are actually doing the same operation. However, we don't adopt larger dilation and larger filter for lesser operations because a larger dilation will cause misconnection while a large filter will cause over erosion.

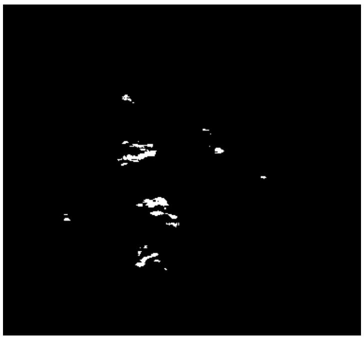


Figure 4 1st time combining dilation and filtering

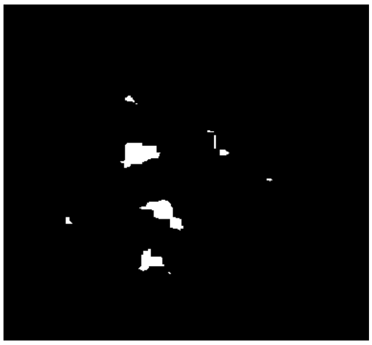


Figure 5 2nd time combing dilation and filtering

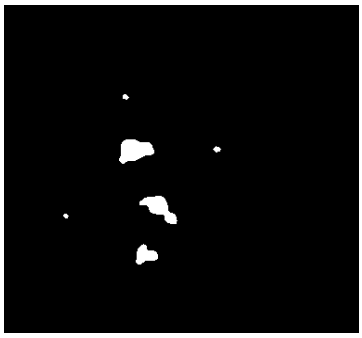


Figure 6 3rd time dilation

4 Objects count

The command “bwlabel” sends out the number of closing structure [3][4]. The bigger closings usually indicate original image has large and deep color features of chocolate chips, on the contrary, the smaller closings present the chips may have light color of a cookie due to partial exposure or merely small piece of chips over the cookie’s surface. In the figure 6, for example, this algorithm judges there are six closing and show 6 chocolate chips on the cookie. The partial exposure of chocolate chips leads to small 1s areas due to the usage of nonlinear filter.

5 Verification

To verify whether the judge is correct or not, we simply paint back the color on white area. If the 1s are painted by chocolate color, it approves the color segmentation is correct. From the results, we find the color segmentation is pretty well, however, the algorithm of the combing dilation and nonlinear filtering still have space to be improved because the results are generally undercount. Appendix B contains the results comparing the original images and judged image for the available 20 pictures. The result and discussion will be represented in the following section.

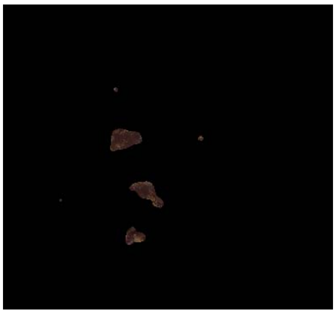


Figure 7 Verification of chocolate chips

RESULT AND EVALUATION

From the testing result of table 1, most of the chocolate chips are undercount. Some of results corresponding with the count by human eyes, even are not exactly correct, though. However, we conclude there still have two types of miscount:

Type 1 is under-count due to partial appearance of chocolate. The color area of the chocolate is so small that it is filtered out. This phenomenon can be improved via either better algorithm or better color segmentation parameter selection.

Type 2 is over-count due to occultation of chocolate in center part. This may be improved by using large closing connection. However, either type 1 or type 2 errors are inevitable because the size, distribution and shape of chocolate are unpredictable.

Figure 19 and 20 use bright color chocolate (orange and brown color) instead of black chocolate, accordingly the color segmentation is replaced by algorithm 2 instead of algorithm 1.

algorithm 1	visual count	scan count
Figure 1	6	6
Figure 2	5	3
Figure 3	6	6
Figure 4	7	6
Figure 5	5	3
Figure 6	4	3
Figure 7	5	3
Figure 8	4	3
Figure 9	3	3
Figure 10	5	5
Figure 11	5	4
Figure 12	6	4
Figure 13	5~7	3
Figure 14	4~5	3
Figure 15	6	6
Figure 16	2	1
Figure 17	6~7	4
Figure 18	6	5
algorithm 2	M & M chocolates	
Figure 19	7	5
Figure 20	4	4

Table 1. The result and comparison by human eyes for available 20 chocolate and cookies images

SUMMARY

Multi stages of image processing techniques are essential and required to achieve accurate count of randomly distributed chocolate chips. Color segmentation is applied to acquire preliminary area of chips’ location. Nonlinear filter can get rid of undesired noise. Combined dilation and erosion (or filtering) are used to cluster and integrate objects. Finally, clusters are counted and verified by painting back color.

The miscount is inevitable due to the size and shape of randomly distributed chocolate chips. The fault is contributed to over-count and under-count. Different doped materials need to modify color segmentation parameter according to this algorithm. However, from the quality control aspect, this image processing method is low cost and reliable to an automated industry with standard mass production.

REFERENCES

[1] “*Digital Image Processing*’, Prentice-Hall, 1996, K.R. Castleman

[2] “*Multi-face location in color image*” M.S. thesis, Concordia University, Shixiong Li, 2002

[3] “*Digital Image Processing, 2nd Edition”*. Prentice-Hall, 2001,Rafael C. Gonzalez and Richard E. Woods

[4] “*Digital Image Processing using MATLAB”*. Pearson education, Rafael C. Gonzalez et al.

Automated Tallying of Chocolate Chips Using Color Segmentation

James Eastman

Department of Electrical Engineering and Computer Science,

Case Western Reserve University, Cleveland, OH, Email: jxe22@cwru.edu

Abstract

This paper presents the design and implementation of algorithms to automate the process of tabulating the number of chocolate chips present in a cookie. The algorithms are the application of image processing techniques. While their accuracy may be limited in certain applications, these algorithms represent a cost-effective solution for replacing manual tabulation in large scale industrial applications.

KEYWORDS

Automated Inspection, Color Segmentation, Quality Control

INTRODUCTION

Automating visual inspection based quality control processes is highly desirable for manufacturing companies as it significantly reduces manufacturing costs and can provide for greater accuracy in the monitoring of their manufacturing processes. A wide array of image acquisition technology is available that help make this automation efficient and cost effective. This technology also greatly simplifies the image processing techniques used in the automation process by handling at the acquisition level common problems such as light reflection and color variation in the acquired images.

Images acquired in this project were scanned in with color correction to account for any extemporaneous color information from other sources. This allows the application of color segmentation techniques to, in this case, identify surface chocolate chips. Once identified, the chocolate chips can then be counted through additional processing techniques.

COLOR SEGMENTATION

Color segmentation is a process by which certain regions of an image are marked as white if the color patterns of that region fall within a specified range of red, green, blue or some combination of those three. All other portions of the image are flagged black so that the segmentation can be easily identified as white against black.

The first requirement then is to define a range of colors to segment from the rest of the image. In this case, specific shades of brown were sought out to identify chocolate chips from the much lighter tan of the cookie bread or from any other color that may exist in the background. Since brown contains elements of all three colors, the segmentation process needs to involved all three layers of the color image.

From the assorted images of cookies available, one was chosen that contained a satisfactorily large region of unobstructed chocolate chip. Within this region (Figure 1) the mean and standard deviation is calculated for all three color layers: red, green and blue.

Figure 1. Region of unobstructed chocolate chip.

Given the mean and standard deviation for red, green and blue, a range of colors is defined as within some real number of standard deviations from each mean. Each pixel of the image is then examined. If the color values of that pixel fall within the specified ranges, that pixel is set to white. Otherwise, that pixel is set to black (Figure 2).

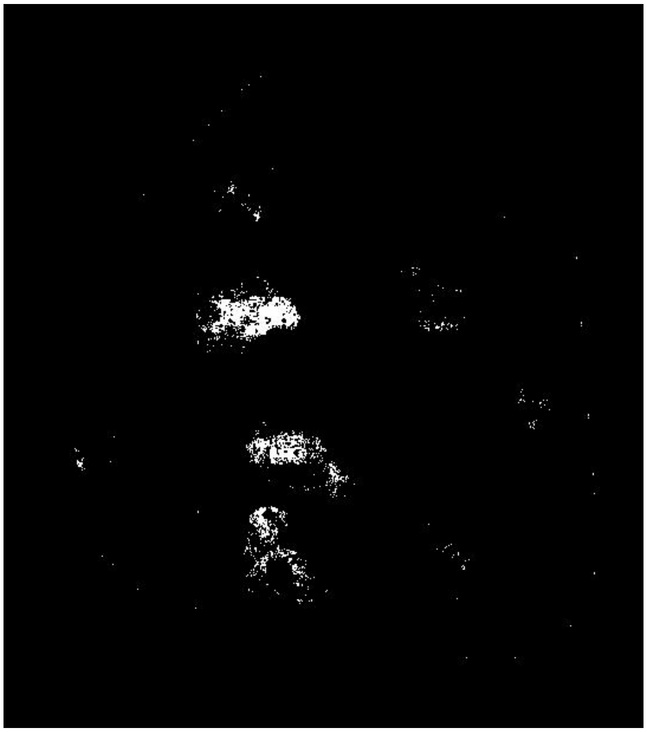


Figure 2. Color segmentation of Figure 1 using ranges of 1.5 standard deviations from each mean.

The scattered appearance in Figure 2 is caused by the fact that the chocolate chips are partially obscured by cookie bread. Since discrete regions defining chocolate chips are needed for the counting process, segmentation demonstrated in Figure 2 needs to be blocked into discrete regions of white, each identifying a chocolate chip. This is achieved by passing an averaging filter over the image and then applying a threshold value such that everything below is set to zero (black) and everything above is set to one (white) restoring the image to pure black and white as it is left in grayscale after the filtering is complete (Figure 3). Once each chocolate chip is represented by a discrete, uninterrupted block, each chip can then be counted.

COUNTING DISCRETE CHIPS

Having discrete, uninterrupted blocks is important, but a process still needs to be defined so that each block can be counted up effectively. Towards this purpose, an algorithm involving a special spatial-domain filter was devised to reduce each block of white to a single white pixel. Once each block is reduced as such it becomes trivial to count.

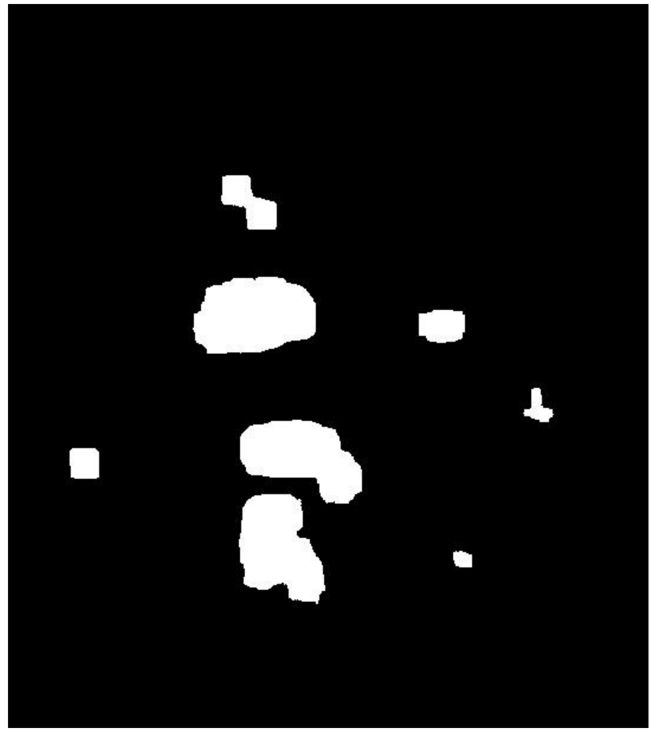


Figure 3. Color segmented image after filtering and thresholding.

The filter in question operates as follows. Examining a window of a certain size, it will count the number of white pixels. If more than one is present, it will reduce it to one pixel of white in the corner of the window corresponding with the direction of its movement across the picture. For example, if the window examined is moving across to the right and down the image, the pixel will be located in the lower right corner of the window.

In applying this algorithm, the filter described was used twice. Initially, it was applied as a 15x15 filter moving across to the right and then down. It was then applied as a 50x50 filter moving across to the left and then up. This effectively reduces each large block of white to a single pixel for easy counting.

RESULTS AND DISCUSSION

In total, twenty-four pictures of cookies were examined including twelve pictures of the tops of cookies and twelve pictures of the bottoms of cookies. Before being processed as describe above using MATLAB, each cookie was visually inspected for a manual count of the number of chocolate chips to be used as a baseline for examining the effectiveness and accuracy of the automated process. Table 1 indicates the manual and automated calculations for each cookie image and the difference between the two.

Cookie	Manual	Automated	Difference

11b	4	5	-1
11t	6	6	0
12b	7	8	-1
12t	6	4	2
13b	6	6	0
13t	5	5	0
14b	5	4	1
14t	6	4	2
21b	5	6	-1
21t	7	5	2
22b	8	10	-2
22t	5	6	-1
23b	6	8	-2
23t	5	4	1
24b	5	5	0
24t	4	3	1
31b	6	4	2
31t	6	4	2
32b	2	3	-1
32t	2	1	1
33b	1	0	1
33t	1	0	1
34b	0	0	0
34t	1	1	0

Table 1. Results of Manual vs. Automated counts.

Summing the absolute values of the differences yields 25; this is slightly more than one per cookie. Summing the regular values of difference yields -7. So while each the process of automatically tabulating the chocolate chips in a cookie generally yields an off-by-one result, over the longer range of tabulating the chips across many cookies yields a result much closer to the actual amount.

SUMMARY

These results indicate that the algorithm used in this process may not be adequate for accuracy on the scale of tabulating effectively on a cookie-by-cookie basis, but on a larger scale the algorithm proves much more effective. This algorithm would then be useful in an industrial setting where the interest is ensuring accuracy over a large number of cookies such as Chips Ahoy's promise of one thousand chips per box of cookies.

REFERENCES

Gonzalez, Rafael C and Richard E Woods "Digital Image Processing," 2nd ed. New Jersey: Prentice Hall 2002

Counting the Chocolate Chips

Dmitriy Goldman

Department of Electrical Engineering and Computer Science,
Case Western Reserve University, Cleveland, OH, Email: dxg68@cwru.edu

ABSTRACT
The objective of this project is to count the number of chocolate chips in chocolate chip cookies. This is a real life application that can be used for quality control. The number of chocolate chips in cookies affects how the cookies taste and can be a key factor in attracting potential buyers. Also in case of equipment malfunction in can detect if the amount of chocolate in cookies is above or below some acceptable tolerances.

KEYWORDS
Segmentation, boundaries, image.

INTRODUCTION
The following algorithm is based on color segmentation approach. The assumption is that the color of chocolate chip is different from surrounding background. And even if it might not be visible by an eye the color of the texture covering the chocolate chips is also slightly different from the background texture. This is the assumption that I'm going to prove. So the approach is to find visible chocolate chips (and possibly some chocolate chips slightly covered with texture) based on color difference.

IMPLIMENTATION
Fig. 1 shows the picture of the chocolate chip cookie.

Fig. 1: Picture of the Chocolate Chip Cookie

It is not an easy task to count the chocolate chips. They all have different shape and are partially covered by the cookie texture. The first step in my algorithm is to perform the color segmentation based on background texture color. The region was selected interactively from the cookie's background texture and the color segmentation was performed using available MATLAB functions. **Fig. 2** shows the result of the color segmentation.

Fig. 2: Color – Segmented Input Image

Fig. 2 is a binary image where white color corresponds to the background texture found by the color segmentation and black color inside the white area corresponds to chocolate chips. There are many small black areas inside the white as well. Those small segments are the result of some inaccuracy of the segmentation algorithm and can be considered to be the noise. To remove this noise the image is filtered with the averaging mask of size 9 x 9. After applying the averaging filter the image was converted back to binary by thresholding it at 0.5. The result of filtering is shown on **Fig. 3**.

Fig. 3: Color – Segmented Image Filtered with 9 x 9 Averaging Mask

Averaging did a very good job in removing the noise. Also the found chocolate chips look more distinct. There are still a few very small black areas that are some remaining noise and they will be removed later using different approach.

The *algorithm for counting the chocolate chips* is based on finding the boundaries of black objects inside the white area and counting them. The boundaries are found using available MATLAB functions. **Fig. 4** shows the boundary of the cookie itself.

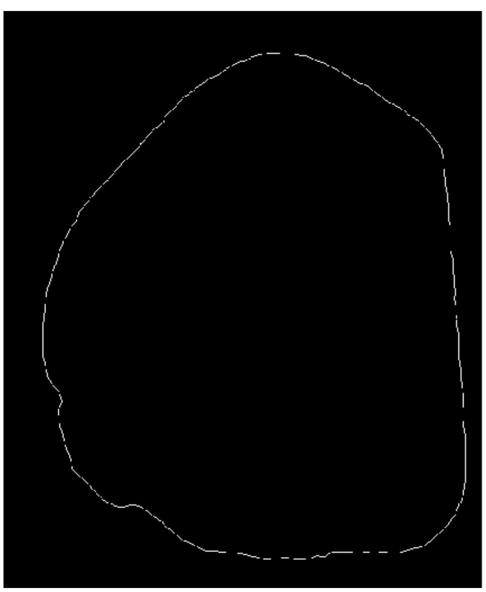


Fig. 4: Boundary of the Cookie Itself

It was not necessary to find the boundary of cookie itself to count chocolate chips. It was done for the later display with the boundaries of chocolate chips to help to visualize the locations of the chocolate chips when compared with an input image.

Next the program found the boundaries of the chocolate chips *ignoring ones with the number of pixels less than 35* to filter rest of the noise by rejecting areas that are too small. **Fig. 5** shows the boundaries of found chocolate chips.

Fig. 5: Boundaries of the Chocolate Chips

RESULTS AND DISCUSSION

The counting chocolate chips algorithm works as follows. Every time when new boundary with the number of pixels greater than 35 was found the chips counter was incremented by one. In this particular case the output of the program was 7 which is equal to the number of chocolate chips visible on the input image. **Fig. 6** shows boundaries of the found chocolate chips inside the boundary of the cookie itself. MATLAB code is in **Appendix 1**.

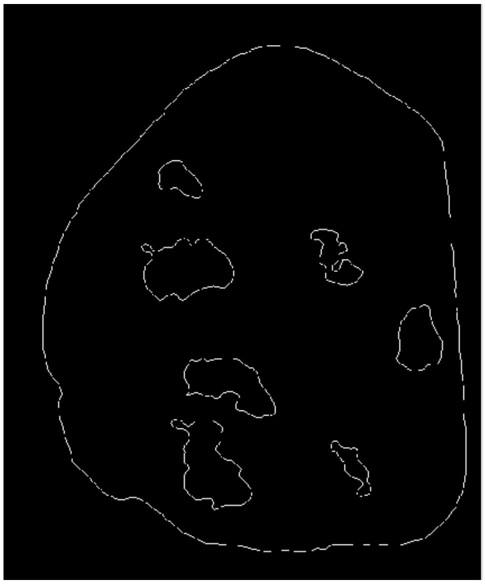


Fig. 6: Boundaries of the Chocolate Chips inside the Boundary of the Cookie itself.

SUMMARY

The above approach to count the chocolate chips inside the cookies definitely works. But its main disadvantage is that it is very sensitive to color change. From the other hand the color sensitivity is the main advantage of the above algorithm because it is able to find not only visible chocolate chips but also chocolate chips that are slightly covered by the cookie's texture because it seems that the color of the texture above the chocolate chips is slightly different than the regular texture. Also the above algorithm is not too computationally expensive, it runs pretty fast even on slower PC so it is practical for the real life inspection application.

REFERENCES

[1] Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing," Second Edition. 2002.

[2] Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, "Digital Image Processing Using MATLAB". 2004.

APPENDIX 1

```
clc
% Read input image
f = imread('chip_1.jpg');
% Covariance matrix of the vector population of the cookie's texture
% calculated using Matlab function " covmatrix ".
C = [299.1911  266.5145  217.1058;...
   266.5145  249.6469  206.4009;...
   217.1058  206.4009  182.6469];
% Mean vector of the vector population of the cookie's texture
% calculated using Matlab function " covmatrix ".
m = [140.7093; 101.4171; 65.9286];
% The main diagonal of C contains the variances of the RGB components.
d = diag(C);
% Compute standard deviation of RGB components.
sd = sqrt(d)';
% Perform color segmantation of the input image using Mahalanobis distance
% and treshold equal to 1.25 times the standard deviation
S1 = colorseg('mahalanobis', f, 1.25*max(sd), m, C);
% Find 9 x 9 averaging mask
h = fspecial('average', 9);
% Filter the segmanted image
g = imfilter(S1, h, 'replicate');
% Obtain the binary image by tresholding at 0.5
g = im2bw(g, 0.5);
figure(1), imshow(f)
title('Fig.1:  Input Image')
figure(2), imshow(S1)
title('Fig.2:  Mahalanobis Distance Was Used')
figure(3), imshow(g)
title('Fig.3:  Filtered Image')
[M, N] = size(g);
% Find and display the boundary of the cookie itself
B = boundaries(g);
d = cellfun('length', B);
[max_d, k] = max(d);
b = B{1};
G = bound2im(b, M, N, min(b(:, 1)), min(b(:, 2)));
% Invert image g to find boudaries of chocolate chips
for i = 1:M
  for j = 1:N
    if g(i,j) == 0
      g(i,j) = 1;
    else
      g(i,j) = 0;
    end
  end
end
% Find, count and display boundaries of the chocolate chips ignoring
% boundaries with length less than 35 pixels.
B = boundaries(g);
g = zeros(M, N);
Number_of_Chips = 0;
```

```
[num_ob, dummy] = size(B);
for k = 1:num_ob
   b = B{k};
   [length_of_bound, dummy] = size(b);
   if length_of_bound > 35
      g1 = bound2im(b, M, N, min(b(:, 1)), min(b(:, 2)));
      g = g + g1;
      Number_of_Chips = Number_of_Chips + 1;
   end
end
figure(4), imshow(G)
title('Fig.4:  Boundary of Cookie Itself')
figure(5), imshow(g)
title('Fig.5:  Boundaries of Chocolate Chips')
figure(6), imshow(g + G)
title('Fig.6:  Boundaries of Cookie and Chocolate Chips')
% Display number of valid boundaries, which is equal to the number of
% chocolate chips minus boundary for the cookie itself.
Number_of_Chips  = Number_of_Chips - 1
```

Quality Control of Chocolate Chips though Chip Counting

Isaac Hirt, Frank Merat
Department of Electrical Engineering and Computer Science,
Case Western Reserve University, Cleveland, OH, Email: ijh1@cwru.edu

Abstract
This paper presents an algorithm called *StatAnala* which is used to compute the number of chocolate chips which are visible in a chocolate chip cookie using statistical analysis of histograms for the image. The probability density function (PDF) for the image is used to segment the image into areas which probabilistically are a chocolate chip, then this segmented image is processed to find the number of areas which contain at a minimum ~0.5% of the number of pixels for the entire cookie.

KEYWORDS
Histogram, *StatAnala*, segment, cluster

INTRODUCTION
Quality control is an important aspect of any production process. Automated quality control is ideal if it is accurate and quick. With this in mind it is desired to create a quality control algorithm which for the chocolate chip production industry, which with a minimum amount of user input is able to determine the number of chocolate chips which are visible on both the top and bottom surfaces using digital images taken of each respective surface[1].

One use for this process is in Nabisco's Chips Ahoy!© cookie promotion claiming each bag of Chips Ahoy! contains at a minimum 1000 chips [2]. The promotion would be verified by determining the average and minimum number of visible chocolate chips for each cookie, which the quality control algorithm would then process each batch to insure the minimum standards are meet.

ALGORITHM
An algorithm called *StatAnala* was created to perform the analysis on a digital image of a chocolate chip cookie. The operator of the algorithm has only two duties to perform, selecting the image to run the algorithm on and select a regionn for histogram analysis. The cookie shown in Figure 1 will be used as a demonstration as the *StatAnala* algorithm is explained.

Figure 1: Original Cookie

The last interaction with a human is now required, the operator must select a region of the cookie to perform histogram analysis upon. This region must contain at least one chocolate chip, and cookie material, and preferably no non cookie areas. It is best of a region with few shadows are selected, and more number of pixels for the cookie to chip is best around 2:1 this is to insure the distinction between the cookie region, and the chip region is large.

Figure 2: Selected Region of the cookie

Once the region is selected shown, the red and green layers need to be separated shown in Figure 3 and 4 respectively, a local histogram must be computed for each must be calculated shown in in Figures 5 and 6 respectively.

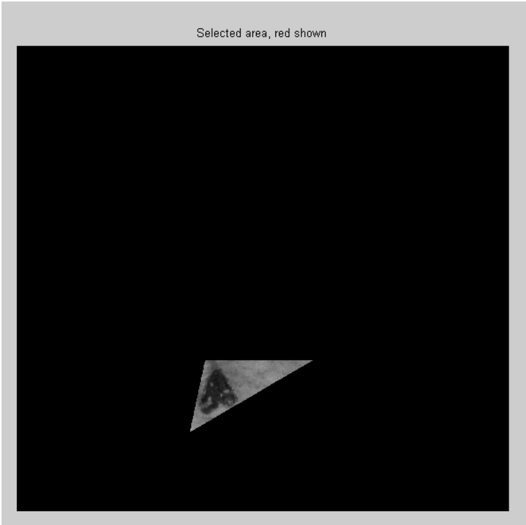


Figure 3: Red layer of selected region from Figure 2

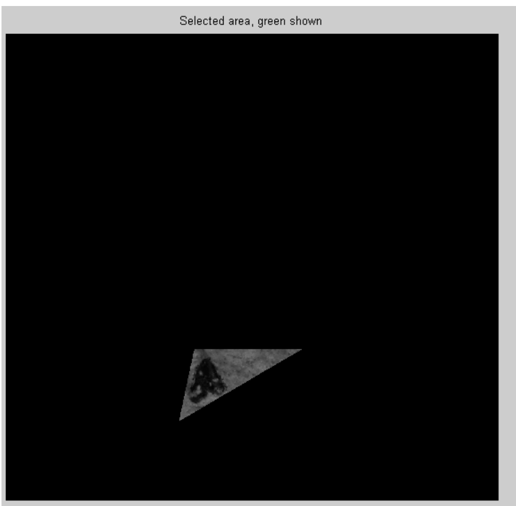


Figure 4: Green layer of selected region from Figure 2

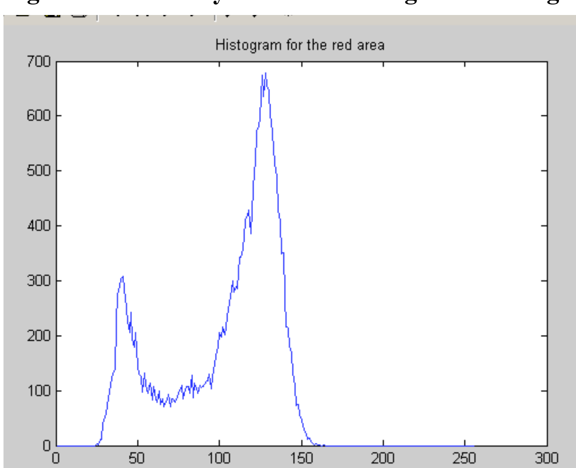


Figure 5: Histogram of the red layer of the selected region from Figure 3.

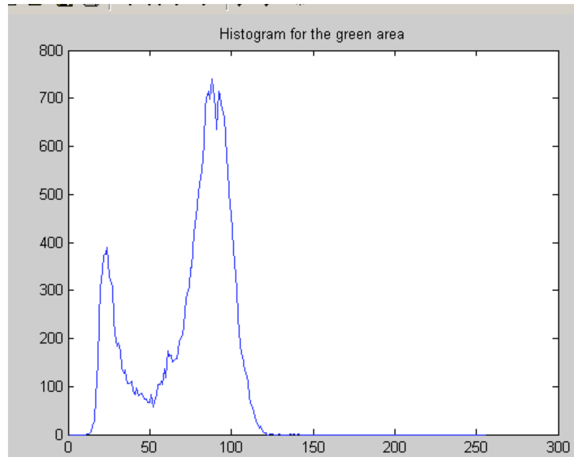


Figure 6: Histogram of the green layer of the selected region from Figure 4.

The first thing which Figures 3 and 4 shows are that in their respective layer, the selected chocolate chip(s) have a darker value then the surrounding cookie material. This leads to the two peaks in the histograms shown in Figures 5 and 6. Due to the cookie having a higher number of pixels in the selected regions, it can be inferred that the larger of the two peaks contains the information for the cookie material, and the smaller of the two peaks contains the information for the actual chocolate chip.

Due to all the pixels being part of the cookie or the chip, the histogram can be viewed as a probability density function (PDF) showing the probability that a given intensity value is either part of the cookie region, or the chip region. This will be used to determine on a pixel by pixel basis if the pixel should be part of a chocolate chip. But first the separator between the two regions must be determined, the best location is between the two peaks where the histogram is at its minimum value. Doing this minimized the error due to improperly naming the current pixel.

The by using this method, valid chip regions for both color layers are created, this is then simultaneously applied to the original image (Figure 1) to segment the image into regions of probably cookie and probably chip, this is shown in Figure 7.

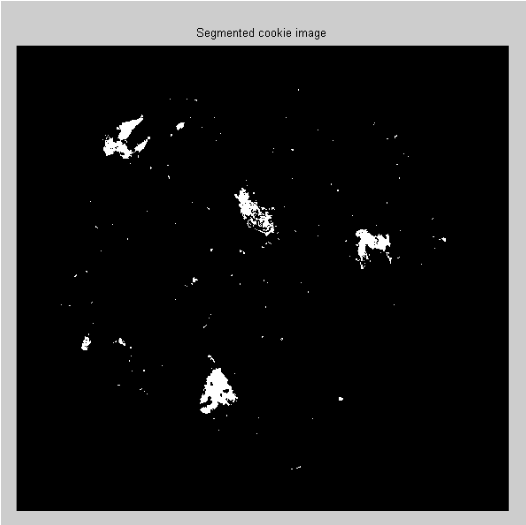


Figure 7: Segmented cookie image

Due to the noise present, and shadows when the image of the cookie was captured, regions which are not part of chips have been segmented, to remove this, the image is low pass filtered. The image is filtered so that a segmented pixel only remains valid if at least 20 of its nearest 49 pixels are also valid segmented pixels, the results of this low pass filter are shown in Figure 8.

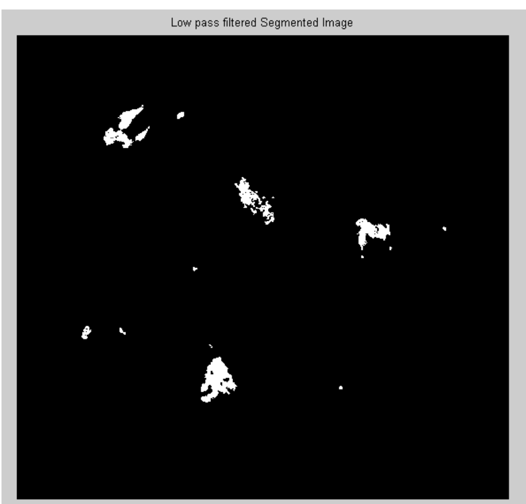


Figure 8: Low pass filtered segmented cookie image

Now that the segmented image has been low pass filtered, only a few regions remain. The next step in the *StatAnala* algorithm is 'cluster' the remaining segments. These 'clusters' consist of neighboring segmented pixels, and each cluster keeps track of the number of pixels in the cluster. These clusters are used to count the number of remaining segmented areas and to find the number of pixels in each segmented area.

Once clustering has been completed, each cluster is checked to make sure that it is at least a minimum distance from every other cluster, this attempts to insure a partially hidden chocolate chip doesn't end up getting count twice.

Now that this process is completed, the number of remaining segments are counted, and a new image of only valid chocolate chips is created, this is shown in Figure 9.

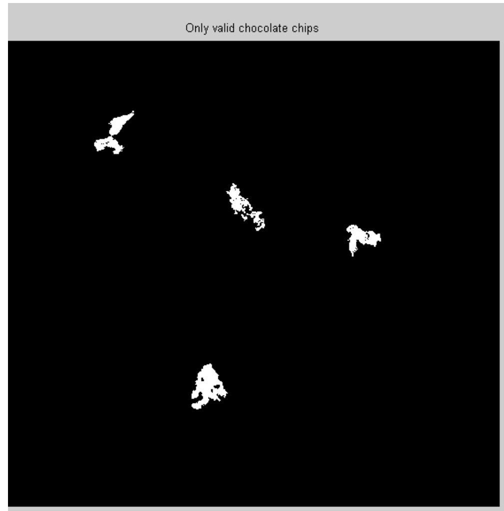


Figure 9: Valid chocolate chips region

As shown in Figure 9, there are 4 chips found. However when looking at Figure 1, it is possible that there might be another chocolate chip in the lower left side of the cookie, however this is not counted due to it's limited visability.

RESULTS

The *StatAnala* algorithm was successfully applied to all of the test cookie images. The results are shown in Table 1 along with a human counting the visible chocolate chips on the cookie. The manually counted chips are separated into two categories, light and dark. The dark chips are what *StatAnala* counts, while the light colored chocolate chips are ignored. This is intentional to speed up the algorithm for the other cookie images, the only modification which would need to be made would be to find the minimum on the high side of the PDF as opposed to the low side.

Table 1: Results of *StatAnala* on images

Picture ID	Human Counted	Program counted
1143	4+3 light	4
1144	1+2 light	1
1145	2+2 light	2
1146	2	2
1147	6	6
1148	6	3
1149	3	4
1150	5	2
1151	7	4
1152	7	7
1153	4	6
1154	9	8
1155	6	5
1156	7	4
1157	6	4
1158	6	6
1159	5	4
1160	7	7
1161	7	3
1162	5	4
1163	5	4

In general the *StatAnala* algorithm did a decent job calculating the number of chocolate chips. The inaccuracies in the algorithm are due to mostly hidden chips which the human can infer belong to a chip due to the surrounding region which *StatAnala* does not analyze.

The data in Table 1 was compiled using a good selection of the chocolate chip region, Figure 10 and Figure 11 show what happens when an invalid selection is made

Figure 10-11: Original and Invalid Segmented Image

Figures 10 and 11 illustrate the results of selecting improper regions for the local histogram calculation. An invalid minimum was selected in the local histogram which caused the shadows to merge with the chocolate chip areas creating the large blobs which are shown in Figure 11.

SUMMARY

The *StatAnala* algorithm does a fairly good job of counting the definitively visible chocolate chips and most which aren't completely visible. This is shown by accuracy in Table 1, and how the *StatAnala* algorithm was close to what a human would have counted for each cookie.

FUTURE WORK

Future work for the *StatAnala* algorithm would be wisely spent improving the speed of the algorithm due to it's current slow speed, the most efficient manner to do this would be to include morphological operators in the algorithm. Also improved chip segmentation detection could be used rather than a fixed pixel size, however nothing will allow the algorithm to detect mostly hidden pixels without erroneous detections of extra chocolate chips.

ACKNOWLEDGMENTS

This work was done for Dr. Frank Merat's EECS 490 'computer Vision' class.

REFERENCES

[1] Frank Merat, EECS 396L/490: Digital Image Processing, http://vorlon.cwru.edu/~flm/eecs490f04/home.html, 2004

[2] Nabisco Chips Ahoy! web page http://www.nabiscoworld.com/cookieguys/ 2004

APPENDIX A – CREATED FUNCTIONS

ChountChips – This implements the *StatAnala* algorithm.
getCluster – finds the cluster the current pixel belongs with.
histo – calculates the histogram of the image

Counting Chocolate Chips in Chocolate Chip Cookies

Svend Johannsen

Department of Electrical Engineering and Computer Science,

Case Western Reserve University, Cleveland, OH, Email: skj7@cwru.edu

Abstract

This paper presents a design to count chocolate chips in chocolate chip cookies using the image processing techniques color segmentation, morphological filtering and a neural net which is trained using back-propagation. The regions corresponding to the chocolate chips are extracted using the image processing techniques; these regions are referred to as segments. Characteristics of each segment are computed, and these characteristics are used as inputs to the neural net. The neural net is then trained to associate different combinations of features with the correct number of chocolate chips. Once the neural net has been trained in this way over a period of time it is capable of counting the number of chocolate chips in images never shown to the neural net before. This paper will show that the design actually is capable of counting the chocolate chips, although with a slight error.

KEYWORDS

Cookie processing, color segmentation, morphological filtering, dilation, neural net, back-propagation.

INTRODUCTION

Counting chocolate chips in chocolate chip cookies is a challenging task for various reasons. The chocolate is roughly the same color as the cookie dough which makes color segmentation [3] challenging, one has to be careful not use too narrow or too wide a filter in color space in order to extract the chocolate alone. A simple threshold filter is not an option because of shadows in the images. The task is also challenging because some of the chocolate chips are partially covered in cookie dough, effectively dividing them in two. This kind of deceitful chocolate chip would never fool a human counting the chocolate chips, but it easily fools a computer program. In order to combat this problem a morphological filter called a dilation filter [1] is applied to the color segmented image. The purpose of the dilation filter is to create large uniform regions from the very sparse and not clearly marked regions that the color segmentation produces.

The dilation filter cannot solve all our problems, chocolate chips that are divided by a wide piece of cookie dough will still appear as two segments after the dilation filter has been applied. We wish to experiment with differently sized dilation filters and study what kind of results can be achieved.

Instead of purely using image processing techniques to achieve our goal, we wish to combine the results obtained by the image processing techniques with a multi-layer feed-forward neural net using back-propagation [2]. A neural net can be trained to recognize patterns. We want to construct such a net and train it with images of chocolate chip cookies; once the training is completed the neural net should be able to count the number of chocolate chips in an unknown image, by applying the **generality** obtained through the training session. We wish to test neural nets using the back-propagation algorithm with different levels of complexity. The complexity of the neural net is defined by the number of interneurons. The performance of this neural net will be expressed both as its ability to count chocolate chips in unknown cookies as well as the squared error.

Images in general contain a very large amount of information, and are therefore completely unsuitable as input to a neural net. A major obstacle in the design is to convert the image representation of chocolate chips to a simpler form. The form chosen in the design is that each segment is represented as a center and a radius. The center and radius of each segment of the color segmented and dilated image has to be obtained.

CHOCOLATE CHIP COUNTING USING IMAGE PROCESSING AND A NEURAL NET

The images of chocolate chip cookies are similar to the one displayed in Figure 1. For a human it is a simple task to count the number of chocolate chips in this cookie, but making a computer do the job is challenging for three reasons.

1. The amount of data in an image is overwhelming. The data has to be reduced significantly in order to be useful as the input to a neural net.

2. Some of the chocolate chips in Figure 1 are partially covered in cookie dough and could easily be mistaken for 2 chocolate chips.

3. The color of the chocolate chips is very close to the color of the cookie, this makes isolating the chocolate chips hard.

Figure 1: Chocolate chip cookie.

To extract the chocolate chips we use the image processing technique called color segmentation [3]. The MatLab function `roipoly` is used to select a region in the image containing a single chocolate chip. The RGB color values of all the pixels in this region are then collected and the mean color and standard deviation is computed using the MatLab functions `mean` and `std`. This average chocolate chip color is then used to create a binary image where all pixels with a color reasonably close to the mean color are 1 and the all other pixels are 0. This criterion is given by the following equation.

$$S_{ij} = \begin{cases} 1 & \text{if } R_{ij} \in \mu_R \pm 1.25 \cdot \sigma_R \wedge G_{ij} \in \mu_G \pm 1.25 \cdot \sigma_G \wedge B_{ij} \in \mu_B \pm 1.25 \cdot \sigma_B \\ 0 & \text{otherwise} \end{cases}$$

The equation defines a cube in color space. All pixels with a color within the cube are assigned a 1 whereas pixels with a color outside the color cube are assigned a 0. The result of the color segmentation is displayed in Figure 2.

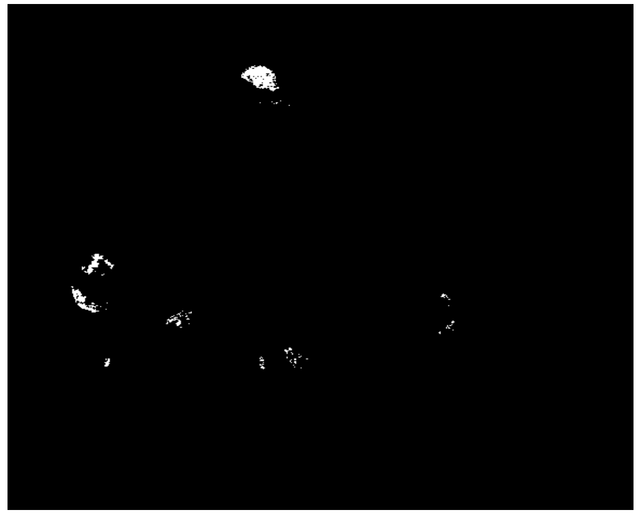


Figure 2: Extracting chocolate chips using color segmentation.

One problem with this color segmentation is that the regions representing the chocolate chips are not coherent. This flaw can be corrected using a class of image filters called morphological filters. Morphological filters operate on sets, for a complete description see [1]. The particular filter we are interested in is the dilation filter. In general a dilation filter is defined by the following equation.

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \subseteq A \right\}$$

B is the filter mask, our mask is a 31x31 square. The reflecting of the mask is obtained, which in our case means nothing since our mask is symmetric. The mask is then shifted by z. If A and B overlap by at least one element the pixel is colored white. What this really means is that a 15 pixel wide edge is added to every pixel in the image. The result of applying the dilation filter B to the object A is illustrated in Figure 3.

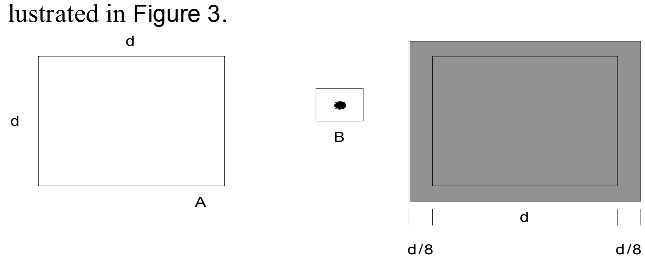


Figure 3: Dilation filter B applied to object A.

When applying the dilation filter to the image in Figure 2, we obtain the image in Figure 4.

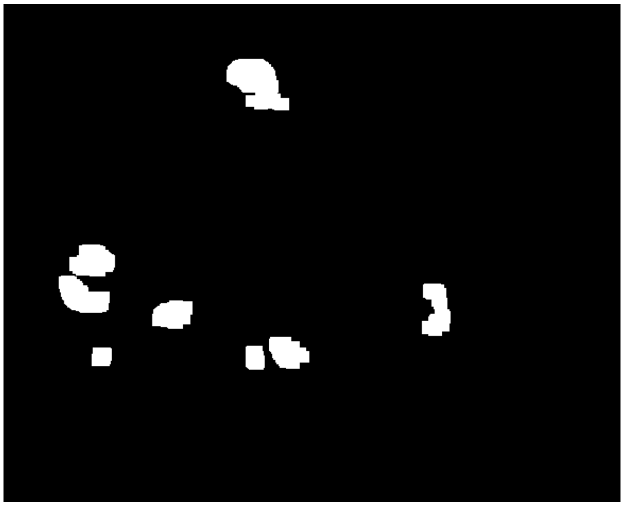


Figure 4: Enhancing the result using a dilation filter.

Looking at Figure 4 one should notice that the pixels corresponding to the chocolate chip at the top of the image is now one coherent segment, this was the primary motivation for applying the filter. One should also notice that the chocolate chip to the far left is shown as 2 segments, this is because it is partially covered in cookie dough and therefore appears as 2 chocolate chips.

The problem with the chocolate chip to the far left could be solved by increasing the size of the dilation filter. This however is a not a good idea. Although it would solve our current problem it would also create a new problem, what if two chocolate chips are very close to one another? With a larger dilation filter they would be perceived as one big chocolate chip. Another option would be to apply heavy blurring to the original image before applying the color segmentation. Experiments show that one has to apply very heavy blurring to the image to obtain the effect; this method has the same side effect as the previous. Neither option seem like a good solution, instead we will solve the problem using a different approach.

We wish to construct a neural net. This net should take the segments in Figure 4 as inputs and output the correct number of chocolate chips. Images are in general not well suited as input to a neural net, because they contain too much information. Our next task is therefore to extract key information about the segments in Figure 4. We choose to represent each segment as a circular region described by a point and a radius. The general idea is illustrated in Figure 5.

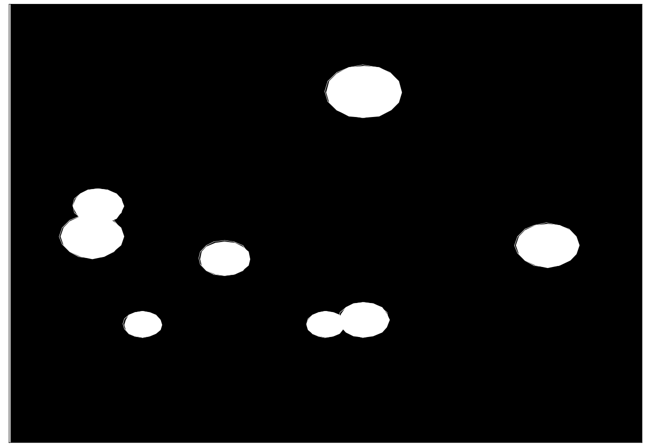


Figure 5: Desirable representation of the segments.

To obtain the center and radius of each segment in Figure 4, a simple filter is applied to the image. This filter has the following shape.

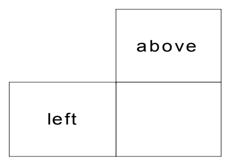


Figure 6: The filter used to assign a number to each segment.

The filter works as follows: Starting from the top left corner all pixels are visited once, one line at the time. One of 5 things can happen when a new pixel is visited.

1. *Both the pixel to the left and the one above are black.* We have run into a new segment, the segment count is increased and this number is assigned to the pixel.
2. *The pixel to the left and the one above belong to the same segment.* This pixel belongs to that same segment too.
3. *The pixel to the left has been assigned to a segment and the pixel above is black.* Clearly this pixel must belong to the segment to the left.
4. *The pixel to the left is black and the pixel above is assigned to a segment.* This pixel belongs to the segment above.
5. *The pixel to the left and the pixel above belong to two different segments.* Doh! Two segments turned out to be part of the same segment.

The fifth possibility poses a problem. When running into this problem a reference is created stating that the segment above this pixel is really the same segment as the one to the left.

Once all pixels have been visited once, all redundant segments are eliminated by using the references obtained in possibility 5 to assign the correct segment numbers to all segments. This requires another pass through all pixels. The result after the second pass of all pixels is displayed in Figure 7. The segment numbers have been scaled for illustrational purposes.

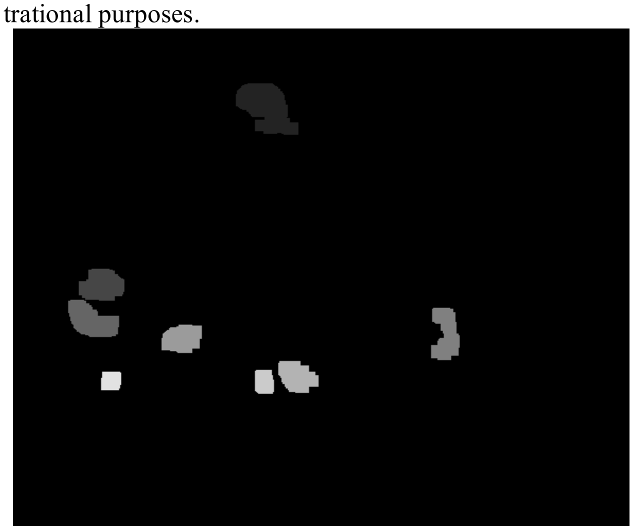


Figure 7: Each segment has received a number which is illustrated by its color.

It should be noted that the image no longer qualifies as being binary, since the first segment is made of 1's the second of 2's, etc.

With the segments numbered, it is an easy task to acquire the positions of all the pixels in a segment. The center is then calculated as the mean position, and the radius as the standard deviation. The MatLab functions `mean` and `std` are used for this.

With the center and radius of each segment obtained this data can be saved to the disc in a feature vector. The structure of this feature vector is displayed in Figure 8.

S_1 X-coord	S_1 Y-coord	S_1 radius	S_2 X-coord	S_2 Y-coord	S_2 radius	... etc.

Figure 8: Format of the feature vector.

With three values to represent each segment it would be reasonable to allow up to 10 segments i.e. 30 input values. With 30 input values the neural net would be able to count up to 10 chocolate chips. (Less if any chocolate chips are divided by cookie dough.) It turns out that a neural net with 30 inputs is very slow and not well suited for experimenting. We therefore choose to simplify the problem by creating a number of smaller images from the original chocolate chip cookie images. These smaller images will be used to train the neural net and will have the following property.

- Each image contains **zero**, **one** or **two** chocolate chips. In the case of one chocolate chip it can either be partially covered in cookie dough, and appear as two chocolate chips, or not.

Figure 9: One of the smaller images used to train the neural net.

An image with the above mentioned property is shown in Figure 9. This particular image contains one chocolate chip partially covered in cookie dough. Performing the previously described image processing on Figure 9 produces the segments shown in Figure 10.

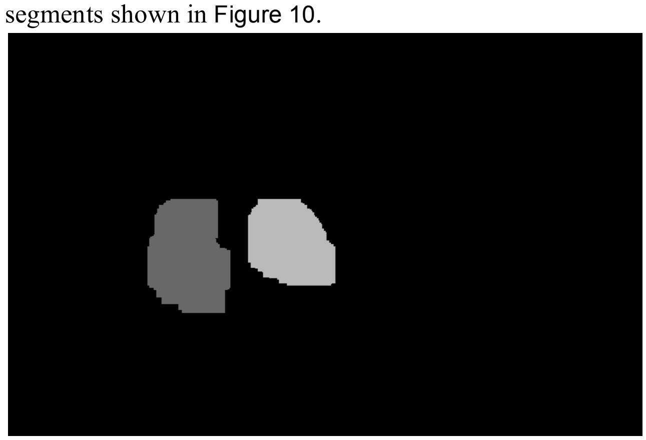


Figure 10: The corresponding segments.

All training images and their segments can be seen in

. The above property guarantees that each training image will result in at most two segments, the number of inputs to the neural net can therefore be limited to six, three from each segment.

The neural net has a single output neuron that can take on any value between 0.0 and 1.0. The interpretation of the output is summarized in Table 1.

Output value	Number of chocolate chips
0.0	0
0.5	1
1.0	2

Table 1: Interpretation of output from the neural net.

With the features of 16 training images obtained we can create training pairs by adding the target outputs to the data file. The data file can be seen in Appendix B.

The Back-propagation net used in problem set 4 is used as the basis for this project. The data file contains all the inputs and outputs we need to train the net. During the learning process the neural net is presented with the features of a single training image along with a value corresponding to the correct number of chocolate chips in the image. The neural net adjusts its weights to reduce the squared error in accordance to this input. The training images are drawn one at the time and at random. The process converges to a minimum, hopefully the global minimum.

RESULTS AND DISCUSSION

In problem set 4 we concluded that 0.1 was a good value for the learning coefficient η. This might not be the case here because a suitable value of η depends on the shape of the n-dimensional surface; n being the dimension of the input vector. The results from experimenting with different learning coefficients are summarized in Table 2.

η	Number of iterations	Squared error
0.40	500	≈0.35
0.20	1000	≈0.25
0.10	**2000**	**0.2490**
0.05	4000	0.3774

Table 2: Determining the learning coefficient η. The number of interneurons is 4 in all cases.

The two errors obtained using a η=0.40 and η=0.20 vary a lot. Learning coefficients that produce inconsistent results are not wanted, so we will stick with **η=0.10**. One would have expected the cookie data to be more tolerant with respect to a high learning coefficient than the data in project 4 was. The cookie data is of a higher dimension and the process should therefore be less likely to get stuck in local minima. This however does not seem to be the case.

We also need to determine the best number of interneurons, the number of interneurons establish the complexity of the neural net. In the following experiments the value of η=0.10 and the number of iterations was 5000. Usually increasing the number of interneurons will produce a better result, on the downside the training time is increased as well. One also has to keep in mind that the number of interneurons should be less than the number of inputs. Having as many interneurons, or more, as inputs is called *over fitting* and means the neural net will not capture the *general* behavior of the inputs. The results from experimenting with neural nets of increasing complexity are summarized Table 3.

Complexity (number of interneurons)	Squared error
4	0.1903
6	0.1529
8	**0.0855**
10	0.1020
12	0.1291
14	0.1794

Table 3: Results from neural nets with increasing complexity.

One can conclude that the neural net needs **8-10 interneurons** to produce good results, increasing the number of interneurons beyond that point does not seem to improve the result. A **total error of 8%** is a good result, it illustrates that the concept works.

To test the quality of the output from the neural net we use cookies that were not used in the training set. Four small images are created from these cookies; they are displayed in Figure 11. The features are extracted used the image processing technique, and then hard coded into a test function, the results from the test is saved in a file called result6D.dat, which can be seen in Appendix B. The results are summarized in Table 4.

Test image (Figure 11)	Output value
A	0.95
B	0.50
C	0.68
D	0.02

Table 4: Test results using an unknown cookie.

Rounding off to nearest 0.5 and using Table 1 these outputs should be interpreted as 2, 1, 1 and 0 chocolate chips, which seems reasonable when compared to the images shown in Figure 11. Test image C is close to be interpreted as having 2 chocolate chips. One can conclude that the output of the neural net is correct.

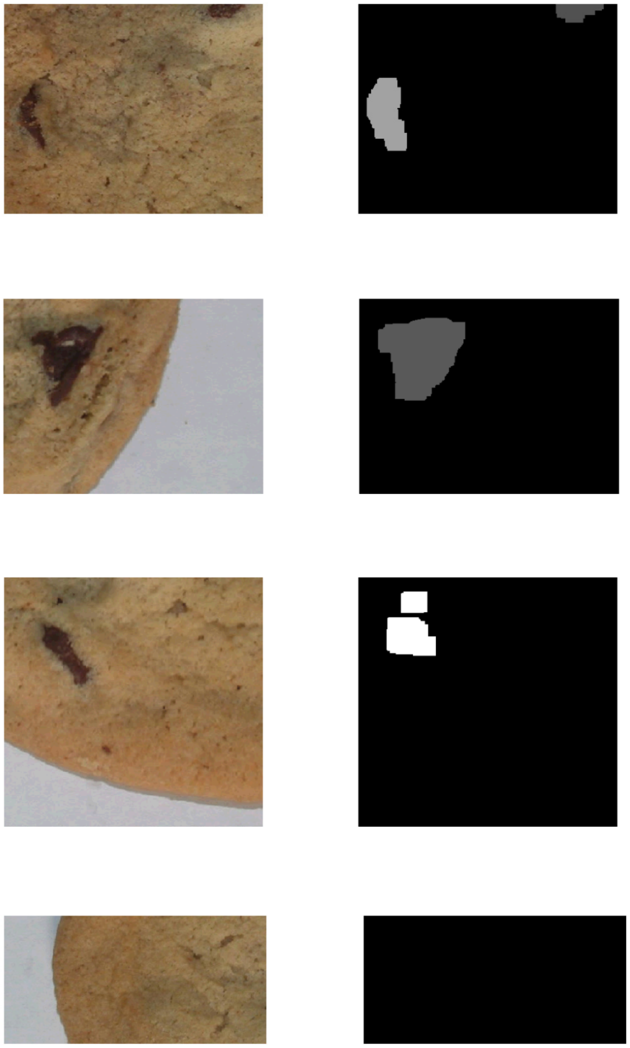


Figure 11: Four test images and their corresponding segments. Labeled A-D from top to bottom.

The more neurons a neural net has the more time consuming it become to train the net. In our test example the best performing net has a total of 15 neurons: 6 input neurons, 8 interneurons and 1 output neuron. If we were to count the chocolate chips in an entire cookie, we could either count the chocolate chips in one region at a time using our existing neural net, or we could construct a neural net with enough neurons to handle many more inputs. The last option would simply be a matter of changing a couple constants, but it would take significantly longer to train.

SUMMARY

Image processing concepts have been applied to the cookie images in order to extract key features of the chocolate chips. These features are fed to a neural net which is then trained to associate the general attributes of the features with the correct number of chocolate chips.

When the training of the neural net is complete, the net is capable of counting chocolate chips with a total error of 8%, which is a good result. The present neural net has a certain limitation: It cannot handle more than 2 chocolate chips as simultaneous inputs. This however is not a big issue because one could easily divide the cookie images into smaller images with no more than 2 segments in each. Multiple copies of the neural net could be applied in parallel, or one could construct a larger neural net capable of handling more chocolate chips by changing a few constant in the design. A larger neural net would increase the amount of time spent on training.

REFERENCES

[1] Raphael C. Gonzalez, Richard E. Woods, "Digital Image Processing" 2nd edition, pp. 519-525, 2002.

[2] fwfwd networks and back-propagation, "Ch3 BP w figs.pdf" printouts from Blackboard, 2004

[3] Raphael C. Gonzalez, Richard E. Woods, "Digital Image Processing" 2nd edition, pp 331-339, 2002

'Estimating number of chocolate chips in cookies'

Ruchi Kothari
Department of Electrical Engineering and Computer Science,
Case Western Reserve University, Cleveland, OH, Email : rxk89@cwru.edu

Abstract

Quality assurance is a major concern in the food-industry. Automated machine inspection/vision can be exploited to this end to achieve better quality products. Various image processing algorithms need to be applied to achieve high quality performance of such automated detection systems. Image processing techniques were applied to chocolate chip cookies specifically for the purpose of counting chocolate chips. This paper discusses how the image processing techniques were applied to obtain objective, quantitative and accurate results. Various cookie images were tested upon to validate the performance of the implemented algorithm.

KEYWORDS

Image processing, automated machine inspection, food inspection

INTRODUCTION

As consumer demands are rising for high quality products, manufacturing industries including the food industry are striving hard to achieve high quality at reduced costs. Manufacturing techniques can be improved upon by a considerable extent by employing automated machine inspection. Real-time, direct and fast tests can be performed on the product to be marketed to verify if it meets production standards. Also, repetitive tasks like counting chocolate chips on a cookie can get monotonous for human workers hence leading to erroneous results. The probability of errors and bias caused by the human factor in inspection of products can be reduced to a bare minimum by employing automated machine inspection [1]. Quality assessments done by employing machine detection has highly desirable features like consistency, quantization, very low error probabilities and fast analysis times.

Physicals features present on cookies can be examined by employing various image processing algorithms to extract various features of importance [1]. More specifically, cookie features like size, shape, color/texture of the baked dough, number of chips, color of the cookie, etc. can be judged on a quantitative scale and product quality can be graded in a consistent, unbiased and quantitative fashion. Higher level of image processing can be performed which integrates all the physical attributes obtained from above discussed lower level processing operations. The results of higher level information and image processing can signify the overall quality of the chocolate chip cookie as would be graded by an expert or 'connoisseur'.

Automated food inspection is one of the major applications of machine vision. It has been successfully employed to food products like almonds, corn kernels, crackers. etc. as reported by Ding and Gunasekaran [3].

Cookies and biscuits are an established food industry. Statistical analyses have indicated that consumer satisfaction and taste are influenced in part by the number of chips present in a chocolate chip cookie. Furthermore, the taste of the cookie is also influenced by the 'visible' number of chocolate chips in the cookie. An ideal number for 'perfect' taste can be found from conducting polls [1]. This number can be used within a range and cookies with number of chips far off from the desired number can be removed from the high quality lot. In machine vision/detection applications, quality of the image is an important factor. Various pre- and post-processing operations can be performed on the images to improve visibility and readability. We discuss such preprocessing operations that can be applied on a cookie image to get improved results in terms of readability by filtering the image. Thus, with better observability, easy counting of the features on the cookie image namely chocolate chips has been performed.

Various methods have been employed to examine features of the product image. Established methods include statistical methods like multiple regression methods and knowledge based methods like incorporation of parameters from expert consultation, fuzzy logic, neural networks etc. [1]. We employ an algorithm to count the number of chips in a chocolate chip cookie by extracting features from the binary image of a cookie. We use color segmentation and morphological approaches to achieve this end. A basis for the desired number can be established from the available knowledge base and operating on that basis, we can determine the number of chips in a cookie. We operate on the morphologically operated upon binary image of a cookie and use a boundary detection algorithm to find the object boundaries inside the cookie. These boundaries correspond to chocolate chips and also 'chipped' (broken) chips. We do not count very small pieces of crumbled chips but threshold according to our requirements. We obtain good and consistent results. This color segmentation based approach has also been extended to circus cookies which usually contain chocolate chips of various colors.

APPROACH

Color segmentation

Segmentation yields very good results wherein an input image can be segmented into regions based on color. Color segmentation uses RGB color vectors [2]. A set of given sample points which lie in the color range of interest are used to find an average estimate of the color to be detected. Each RGB pixel in the image is classified as having a specified color or not. A matlab function 'roipoly' was used to specify a region of interest yielding a black & white image which is usually white in the regions of interest and black elsewhere. Thus, all regions of an image having the same color can be separated from the rest of the image.

Morphological processing

Morphology is a tool for image component extraction useful for shape and boundary representation.

After the image is color segmented, we convert it to a binary image and then use various image enhancement matlab algorithms like 'bwareaopen', 'imclose', 'imfill',etc. and fill the gaps/holes in the detected regions so as to form a clustered whole of pixels for each chocolate chip. A threshold can be specified using 'bwareaopen' such that objects having number of pixels lower than the threshold can be deleted. Gaps/holes are observed in a detected region owing to the texture present in the image. We use 'imclose' and 'imfill' to fill those discontinuities in the image boundaries and make the image appear smooth. In essence, we filter the discontinuities out from the image.

Boundary detection

Finally, a boundary detection algorithm is employed to detect boundaries. Again, a threshold can be employed on the size of the bounded regions and only regions above the threshold are counted as wholesome chips. Other very small regions can be considered to be powdered or very small broken chips of chocolate. We do not count these very small regions. If two or more chips are connected together thereby appearing as a single chip, we employ an upper limit to the size of the chip and then count the bigger masses as two or more chips accordingly.

RESULTS AND DISCUSSION

We employ the algorithm to various cookie images to verify the predictability of our algorithm. Chips can be defined as whole or not depending on the threshold parameter that can be set by the user based on statistical data for the ideal size of a chip. Chips embedded in the cookie dough and also on the flip side of the cookie can obviously not be counted by just one single operation of the algorithm. The same algorithm needs to be employed on the flip side of a cookie and both the results can be added together to obtain the total number of chocolate chips in the cookie. It is almost impossible to determine the number of chips embedded in the cookie by purely binary visual methods.

Figure 1.Original cookie image

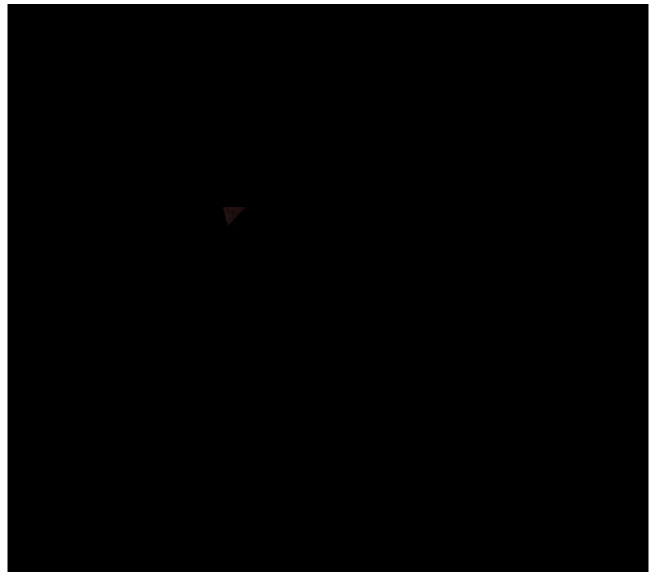


Figure 2.Color segmentation basis supplied by user

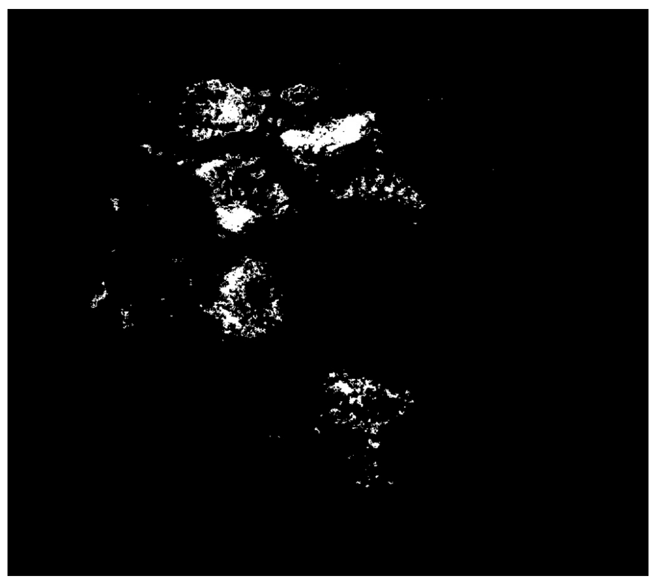


Figure 3. Color segmented image with regions matching the color selected shown in white

Figure 4. Color segmented image after filtering and morphological operations with object boundaries roughly corresponding to the visible chocolate chip boundaries

From Figure 4, it is observed that the big area in the upper most portion of the image is not one chip as seen but rather two chips abutting. We employ an upper limit on the size of the chip in our algorithm and count these chips as two correctly. All the smaller regions which correspond to embedded chips or just pieces of chips are not counted. This lack of visibility is owing to the texture base in the image. But employing visual inspection systems, we can not count those vague regions reliably. In other words, the algorithm compares equally in performance to the naked eye. So, the improvement is in the consistency (achieved by thresholding), parameterized options, fast response times and zero probability of error. Human error caused due to tediousness and cumbersome nature of the activity observed in manual inspection is totally removed by automated intelligent inspection.

Our algorithm can easily be extended for detection of more than one color in the image. This is demonstrated by application of our algorithm to circus cookies which contain various colored chips.

If the number of colors that can be encountered are known, the cookies can be consistently counted as explained above. A ‘for’ loop is employed and color segmentation is repeatedly applied to the image to detect and count variously colored chips in the cookie. The ease of implementation and consistency are attractive features of the algorithm employed.

Figure 5. Circus cookie

The algorithm presented applies color segmentation to the image recursively and then finally adds up the number of objects identified after each run of color segmentation and morphological processing.

Thus, chips on both top and bottom faces of a cookie can be counted consistently. Similar algorithmic implementation can be performed by employing intelligent devices that find the perfect taste in a cookie by solely visual image processing methods.

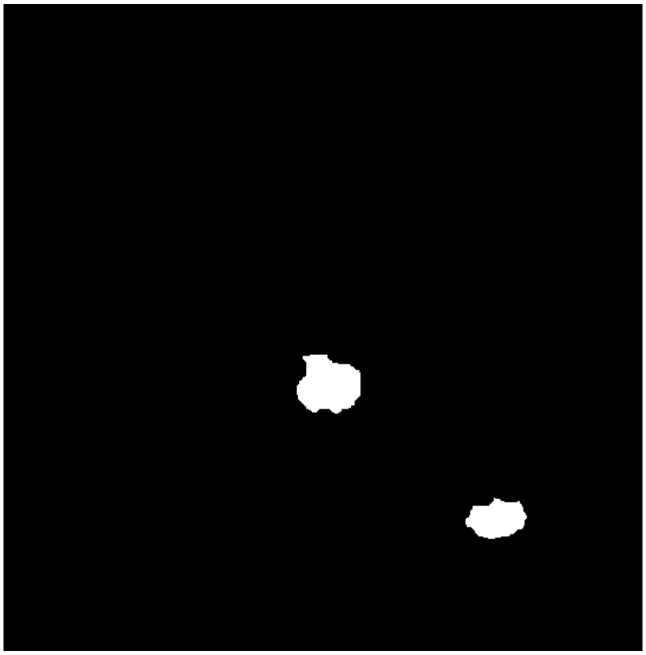


Figure 6. Color segmentation and morphological processing run1

Figure 7. Color segmentation and morphological processing run 2

As explained above, we apply our algorithm to the image in Figure 5 and obtain two images as shown in Figures 6 and 7. The algorithm adds up these two obtained numbers and provides the final result.

SUMMARY

An algorithm for counting chocolate chips in cookies has been devised on the basis of color segmentation and morphological image processing. The algorithm is simple and efficient. It yields quantitative, objective and accurate counting performance for visible chips on the cookie surface. This is of unique advantage to the food industry as the consumer ratings go up with the product quality and consistency in taste.

REFERENCES

[1]. DIGITAL IMAGE PROCESSING USING MATLAB® ,RAFAEL C. GONZALEZ, RICHARD E. WOODS AND STEVEN L. ED DINS, PRENTICE-HALL, 2004.

[2]. *Davidson, V.J.; Ryks, J.; Chu, T.,'* Fuzzy models to predict consumer ratings for biscuits based on digital image features'**,** IEEE Transactions on Fuzzy Systems, Volume: 9, Issue: 1, Year: Feb 2001, Page(s): 62-67

[3]. K. Ding and S. Gunasekaran, "Shape feature extraction and classification of food material using computer vision," *Trans. ASAE*, vol. 37, pp. 1537–1545, 1994.

Feature Detection of Test Images in Food Quality Control

Daniel Pendergast

Department of Electrical Engineering and Computer Science,
Case Western Reserve University, Cleveland, OH, Email: djp8@case.edu

Abstract

This paper presents a simple technique for feature identification within images. Specifically, the scope is of feature identification in food quality. Feature extraction was attempted on test images of Chocolate chip cookies – specifically, identifying the number of chips in each cookie imaged. Color segmentation and basic filtering were employed with a fair degree of success

KEYWORDS

Color segmentation, visual food inspection, feature identification

INTRODUCTION

Robotic quality control is the standard today in manufacturing. In quality control applications that require visual inspection, a great deal of specialized image processing is necessitated. Quality control involving visual inspection ranges from circuit board layouts, to quantity counts, to food quality measures. In terms of food quality inspection, a number of papers have been written on various automated techniques. Davidson[1] discusses the use of a neural network to judge overall quality of chocolate chip cookies based on an input space of several extracted features and consumer preference for combinations of said features. A detailed example of feature extraction for this particular example is the focus of this paper. Specifically, the number of chocolate chips cookies present in each cookie imaged. A number of image processing operations could be performed to achieve feature extraction

TECHNICAL APPROACH

There are multiple challenges in finding the chocolate chips in an imaged cookie. The most obvious is that the chips themselves may not be an undivided region in the test image due to overlay of batter. Also, there will be no consistency in the shape of the region corresponding to a chocolate chip. Finally, depending on the density of chocolate chips in a single cookie, it is possible that the distance between single chip segments, and two different chips, will be indistinguishable.

Upon initial inspection of images of chocolate chip cookies, one will note that the chips themselves are much "darker" than the cookie dough. It becomes apparent that a rough estimation of the location of chips can be had simply by converting the image to gray scale and thresholding the image at the appropriate boundary to a binary. An immediate problem presents as any dark pits or dents in the cookie's surface will be picked up by such a thresholding procedure as well.

Again, by inspection it becomes obvious that the chip regions of connected pixels are much larger than any "surface noise". As a test program for food quality inspection will be specialized for a specific application, it follows that some parameters of the test program will experimentally found. The size of the connected pixel regions that can be ignored as noise was set to <400 connected pixels. At this point the program will have isolated chip-only regions of the cookie image. There is yet an additional concern however, as a chip may be "spackled" with batter leaving the chip segments showing through small enough to appear as surface noise.

The solution to this problematic exception is color segmentation and matching. The mask found above by thresholding can be used to select pixels from the original color image to sample. From these sampled pixels, average values of RGB and corresponding standard deviations can be found. Using these values, improved identification of chip regions is possible – including "spackled" chips. A few surface variations will still be picked up by this process. They can be removed or ignored using the same observationally-based parameter of <400 connected pixels being considered noise.

A reasonable mask of chip regions free of noise and inclusive of hard to see/detect “spackled” chips can now be expected. The final step in the feature extraction of chip count is ensuring that chips that are partially buried and divided by a line or lines of dough are not counted twice. This is yet another experimentally found parameter used to set the size of an averaging filter that is employed to join segments of a single chip together. Once this step is completed, a reasonable count of chocolate chips may be confidently performed.

RESULTS

The feature extraction algorithm as implemented in this program is relatively slow. On a current mid-range laptop system the code takes around 30 seconds to execute. It is expected that a much faster, dedicated computer would be used for the quality control function on a manufacturing line.

An original test image converted to grayscale is shown in Figure 1.

Figure 1

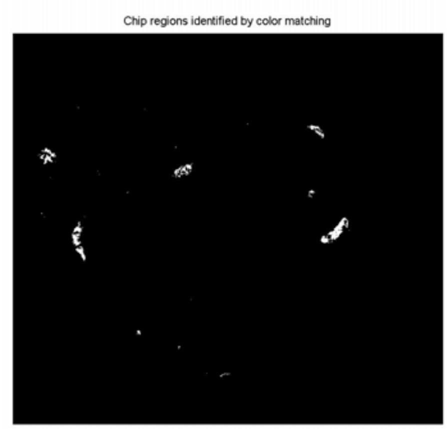
From inspection there appear to be 5 easily identifiable chips. In the upper right hand quadrant of the chip there is a bit of a chip behind a ridge in the dough and towards the very bottom there is a depression in the dough that may be due to a chip near the surface.

Upon thresholding, the preliminary chip region mask was obtained. This is shown in Figure 2

Figure 2

As noted in the technical discussion, chips that have a slight covering of batter may be missed, and indeed is in this example. Nonetheless, the mask obtained is used to sample the original color image to extract statistics on the color of the chip regions. These statistics are used to form a more sophisticated mapping as shown in Figure 3.

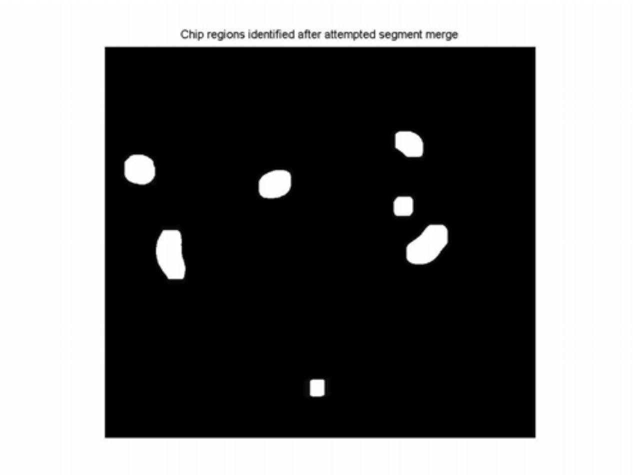
Figure 3



In Figure 3, the batter spackled chip is picked up through color matching. Also appearing is some noise that was filtered out according to the experimentally-found method described above.

Finally, an averaging filter was used to join any potential single-chip segments into one bounded region. This particular example did not have a divided chip, but the results can still be seen in Figure 5.

Figure 5



CONCLUSIONS

The program written for this application was written in MATLAB. It is conceivable that the program could be speeded up with the development specialized functions for some of the iterative steps carried out in this implementation. The program is subject to errors due to abnormalities in cookie surfaces not present in the training set of images. A suggestion taken from Davidson[1] would be the employment of a neural network to adjust the experimentally-found parameters discussed above.

ACKNOWLEDGMENTS

The author credits graduate education at Case Western Reserve University with inspiring examination of this subject.

REFERENCES

[1] V.Davidson, J. Ryks, and T. Chu, "Fuzzy Models to Predict Consumer Ratings for Biscuits Based on Digital Image Features", *IEEE Transactions on Fuzzy Systems*, Vol. 9, pp. 62-67, February 2001.

Surface Feature Detection for Quality Assurance with Cookies

Chris Roberts, Frank Merat

Department of Electrical Engineering and Computer Science,

Case Western Reserve University, Cleveland, OH, Email: Robert.Roberts@case.edu

Abstract

This paper presents a method for counting the number of visible chocolate chips or other solid mixed ingredients in baked goods such as cookies using image processing techniques. Taking such a count can be used to create an automated quality assurance method on the product assembly line to ensure the products being created meet a company's high standards. Automated quality assurance in the food industry is a rapidly growing field as businesses are driven to provide consistency in their low cost, high volume products [2]. Methods of color segmentation and image morphology are used to achieve a fairly high accuracy and high speed counting method.

KEYWORDS

Cookies, Color Segmentation, Image Morphology, Quality Assurance, Food Industry

INTRODUCTION

The food industry produces high volume low cost goods for the ever picky consumer market. The physical appearance of the goods being produced play an important role in the decisions a consumer will make about that product [1]. Therefore, automated methods for classifying goods being produced are in very high demand. Automated quality assurance in the food industry was said to be one of the fastest growing areas in the industry [2].

Baked goods are one area of the food industry where automated quality control can benefit the industry. Items that have a mixed solid ingredient, such as chocolate chip cookies, are of particular interest because the appearance of the mixed ingredient plays an important role in the appeal of the product.

The trend of consumers as seen through marketing is that more chocolate is better in a cookie. When it comes to food, consumers must then "judge a book by its cover." The surface appearance of a cookie indicates to the consumer on how much chocolate they might find in the cookie, making that product attractive. With this idea in mind, the ability to simply count how many chocolate chips are visible on the surface may be a good quality control indicator.

Other techniques, such as calculating the percentage of chocolate chip area on the cookie, and using neural networks to determine the quality of a product have been explored in industry already [1]. These methods handle a variety of products and use complex logic to compare the cookies to a standard that mimics that of consumers. For the method presented in this paper, only the physical count of the number of items, such as chocolate chips, will be explored.

Keeping the quality indicator to a simple count makes the processing fast. This would allow for more cookies in a batch to be tested, possible raising the overall standard of a product.

The surface characteristics of a cookie can also indicate more about the quality of the cookie under its surface. By taking a surface count, and then breaking apart a batch of cookies and counting the total number of chocolate chips in the cookie, a statistical confidence level could be generated about the product. With this statistical model in hand, the counter could then indicate if the cookie meets the company's desired quality simply by the count of the cookies visible on the surface.

The actual processing methods for the cookie are fairly simple. The most important factor in the detection of the cookies with this method is the color of each sample. A factory setting with and stationary camera and lighting could provide such a reliable environment to make this algorithm effective. The RGB color images are then processed using the methods of color segmentation and image morphology that are well documented in many texts [3][4].

TECHNICAL APPROACH

The first step in approaching this problem was to acquire the images to be used for testing. An accurate color imaging device with consistent lighting is important, so that identical items have the same color characteristics. For our purposes, Dr. Frank Merat was kind enough to use his color corrected flatbed scanner to digitize images of the entire sample cookie batch. The scanner is shown below.

Figure 1 - Cookies being scanned for processing **(Frank Merat)**

Once each cookie was digitized, the image was loaded into Matlab for sample processing. Matlab allowed for a quick implementation method for testing this algorithm. Resizing the images using bilinear interpolation can bring their size down for faster processing if the imaging equipment is high resolution.

The next step of the process is to perform color segmentation using the mean and standard deviation of the items to be counted of each of the 3 color components.

To find the mean and standard deviations initially, a region of an image is selected that contains the item to count, such as the chocolate chip. Multiple instances of the chip can be recorded, and the average values of the colors can be saved. Also, if there are multiple colors of items to be counted, the color values can be recorded and saved for each color.

With the color data stored, the image is segmented for each desired color, and then a binary mask of the image is created that is non-zero only in regions where the desired color is discovered.

Next, all of the binary color masks are added into a single mask. This then is non-zero for regions of all interesting colors.

The image morphology technique of erosion is then applied with a disk shaped mask of size three to erase any tiny pixels that might have occurred due to shadows. In the event that these tiny regions are items such as chocolate chips, they would be very small and hard to see unless the images are very high resolution.

Dilation is then applied to the mask using a disk shaped mask of a large size, such as 30. This ensures that any black holes on the chocolate chip from segmentation are properly bloated to act as a single region. This is important because many items may be partially covered by dough from the baking process. One downside of this is that in cases where the items to be counted are very close together, they will merge into a single item.

Next, the morphology operation to shrink each of the regions to a single pixel is performed. This then leaves a single pixel for each region, which should be each item to be counted.

Finally, the total number of non-zero pixels is counted and returned for use.

RESULTS AND DISCUSSION

The data set used in testing the algorithm consisted of one dozen cookies scanned in by Dr. Frank Merat. Of these cookies there were nine chocolate chip cookies, and three "circus" cookies that contained colored candy chocolates pressed into the dough at the time of baking. The flatbed scanner yielded 12 color corrected image each with a dimension near 1200 by 1200 pixels. This image size could most certainly be reduced, but it will be left alone for the initial testing. A sample image is shown in Figure 2 below.

Figure 2 - Captured Cookie Image

The colors of the chocolate chips and colored chocolate candies were then isolated for a single case, and the RGB mean channel values and standard deviations were recorded by the program. Because the imaging device uses constant lighting and color correction, the colors should remain constant in each of the images. Table 1 shows the color data used for the chocolate chip

Table 1 - Chocolate Chip Color

Channel	**Mean** (8 bit)	**Deviation** (8 bit)
R	51	9
G	34	8
B	28	7

The data for the chocolate chips is a very specific number without a large deviation. This makes its color fairly distinct. Using the color data the binary masks were created through the segmentation. Because the color of the chocolate chip, as well as the colored candies, are very distinct, the binary masks created are relatively noise free and readily pick up even mostly hidden chips. Figure 3 shows the resulting binary mask from segmentation.

Figure 3 - Binary Segmentation Mask

The mask shown about shows 4 very distinct cookie regions, as well as there are two smaller cookie regions where the chip just pokes through.

Image morphology is then applied, and the erosion is omitted because the images in this sample case are nearly noise free. Dilation is then applied to the images resulting in six very distinct regions which can be seen in Figure 4.

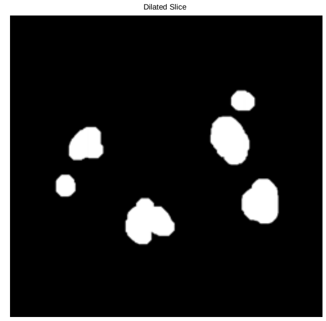


Figure 4 - Dilated Mask

This dilated mask is then eroded down to single pixels and the sum of all of the pixels in the image is taken to get a count. In the event that the dilation doe not fill in all the holes in a region, the algorithm creates a ring of that region, resulting in a much inflated cookie topping count. This was found to occur when either the chips were bunched tightly, or highly covered in dough.

This process was repeated for the remaining twelve cookies at full size, and then the data set was scaled using bilinear interpolation to 50% of their original size and processed.

At the same time, an independent volunteer was asked to examine the set of cookies and to record how many cookies he saw in each image. Observing the volunteer, he scrutinized the images, looking for any signs of a chocolate chip or other candy. Tables 2 and 3 show the results of the chocolate count based on the volunteer and the two trial runs. The trial runs were repeated several times, and returned consistent results.

Table 2 - Circus Cookie Results (# of candies)

Cookie	Volunteer	Full Sized	Half Sized
1	**7**	**7**	**7**
2	**3**	**3**	**3**
3	**4**	**4**	**4**

Table 3 - Chocolate Chip Cookie Result (# of chips)

Cookie	Volunteer 1	Full Sized	Half Sized
4	**3**	**3**	**3**
5	6	7	5
6	5	6	4
7	8	12	25
8	**6**	**6**	4
9	9	10	5
10	**6**	**6**	5
11	8	9	7
12	6	8	4

The results of the processing are very encouraging. Table 2 shows that the algorithms were very good at counting the distinct colored candies in the images, and provided correct results that matched a human inspector in 100% of the tests.

Table 3 shows the chocolate chip cookie results and they require more interpretation. Cookie 7 had a very doughy region that led to the algorithm creating a ring of pixels, inflating the value in that sample.

In the case of the full sized cookies, the algorithm was very good at picking up small cookie pieces poking through. The algorithm also picked up some of the shadow regions in the cookie of similar color. This led to the algorithms results always being equal to, or slightly greater than the actual count based on a highly scrutinized inspection count. Attempting to eliminate this with erosion eliminated the smaller chips, lowering the count slightly.

The half sized images produced results that were either on target, or slightly lower than the volunteers results. Examining the images, the bilinear interpolation process eliminates many of the small shadows and barely visible chocolate chips, so they are not picked up by the algorithm. All clearly visible chocolate chips are still counted by the algorithm, so it is highly effective.

To test the theory that all of the highly visible chocolate chips are counted, and only barely visible chips and noise are omitted, a second volunteer was solicited. Each of the twelve cookie image was printed out in color using a Hewlett Packard DeskJet 722C printer at as close to actual cookie size as could be replicated. The volunteer was then shown the cookies for a period of 15 seconds or so under bright fluorescent lighting conditions, and asked to record how many instances of the items they saw. The circus cookies once again provided 100% accurate results because the candies are so distinct, but the chocolate chip count changed, and can be seen in Table 4 below.

Table 4 – New Volunteer Results (# of chips)

Cookie	Volunteer 1	Volunteer 2	Half Sized
4	**3**	**3**	**3**
5	6	**5**	**5**
6	5	**4**	**4**
7	8	7	25
8	6	**4**	**4**
9	9	6	5
10	6	**5**	**5**
11	8	**7**	**7**
12	6	5	4

Comparing the two volunteers the data indicates that Volunteer 2 always chose the same number, or a slightly small number of chocolate chips in a cookie. Scaling the image to the proper size made the barely visible chips hard to detect under time limited conditions. If Volunteer 2s results are compared to the half-sized image results it become very apparent that the algorithm is fairly accurate because the count is the same for 66% of the test cookies. Of the remaining 33%, 22% of the images are within 1 chocolate chip in value. The only case where the count is off by much is Cookie 7, which provided problems in all cases.

In a factory setting an inspector would probably not scrutinize each cookie to the depth that the first volunteer did. The count trend of an employee would most likely match the results of the second volunteer, who was under a time constraint with the cookies being actual size. The algorithm therefore provides a fairly accurate count of the visible cookies in an image.

On a 1.4 Ghz computer with 512MB of RAM the algorithm took an average of 66 seconds to process a full sized image, and only 14 seconds to process the half sized 600 by 600 pixel image. This is a very high speed method to count the cookies, and could easily be done on the production line so that the questionable cookies could be flagged for further inspection or rejection farther down the line. (This would be the case with cookie 7).

SUMMARY

Automated quality assurance in the food industry is a fast growing region of image processing. With simple color imaging equipment under controlled lighting conditions it is possible to count the approximate number of visible chocolate chips and other toppings at very high speed. The image processing techniques of resizing, color segmentation, and image morphology allow for the fast counting of the number of instances of a colored candy or chocolate chip. The algorithm detected the highly visible items on the surface in the cookies just as a glancing inspector or consumer would in the real world. Furthermore, the algorithm showed consistent results if the trials were repeated, and could therefore provide a high success rate because the algorithm will not suffer from human factors such as fatigue. With this in mind, the algorithm could be used to successfully classify cookies in a production setting and act as an automated method to ensure a high quality product is produced. In our test setup, assuming that an error of ±1 item is acceptable, the algorithm was successful in 92% of the cases and took an average of 14 seconds to process a cookie.

ACKNOWLEDGMENTS

This work was done as a midterm project for the EECS 490 class taught by Dr. Frank Merat. Special thanks to Dr. Merat for providing guidance for this project, as well as the test images used in its implementation, and the image of his scanner in work. I would also like to thank my roommates David Rostocil and Casey Kearns for humoring me during this project and for providing their counting expertise.

REFERENCES

[1] V.J Davidson, J. Ryks, and T. Chu. "Fuzzy Methods to Predict Consumer Ratings for Biscuits Based on Digital Image Features." *IEEE Transactions on Fuzzy Systems*, Vol 1, No 1, Feb. 2001, pp. 66-67

[2] S. Gunasekaran, "Computer Vision Technology for Food Quality Assurance." Trends Food Sci. Tchnol. Vol7.. pp. 245-256, 1996.

[3] Rafael C. Gonzalez, Richard E. Woods and Steven L. Eddins, Digital Image Processing Using Matlab. Prentice-Hall, Upper Saddle River, new Jersey, 2nd Ed. 2004

[4] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing. Prentice-Hall, Upper Saddle River, new Jersey, 2nd Ed. 2004

Counting and Locating Chocolate Chips Using Color Segmentation and Image Morphology

Ira Ross

Department of Electrical Engineering and Computer Science

Case Western Reserve University, Cleveland, OH, Email: ira.ross@case.edu

ABSTRACT

This paper presents a technique for intelligently counting the number of chocolate chips in chocolate chip cookies. The counting algorithm is based on a two step process where possible chips are segmented based on color, then checked for size and shape using morphological image processing. For practical purposes, the project focuses solely on processing images of cookie bottoms. By quick comparison between images of tops and bottoms, it is evident that the bottom of the cookie has more information about chocolate chip location. In the majority of cases, this bottom processing algorithm accurately counts and locates the chocolate chips in a cookie.

KEYWORDS

Chocolate chips, color segmentation, image morphology

INTRODUCTION

Image processing is increasingly being used in food inspection, because it allows for a quicker and more detailed analysis than conventional manual inspection. Whereas a human inspector requires extensive amounts of training, an image processing setup needs only a quality algorithm to work efficiently. By decreasing human error and providing more useful information, image processing provides a better system for food inspection.

This project operates on a top-down approach to detecting and locating chocolate chips in cookies. Since in an industrial application the relative size and color of chocolate chips is known, a top-down method is appropriate for solving the problem. Information about the range of colors in chocolate chips can be found experimentally through averaging and analyzing several chip test areas. Likewise, chip size is determined using a guess and check procedure, since image resolution directly affects the chip size.

Using the experimentally determined RGB values for an average chocolate chip, color segmentation is applied to bottom images to isolate chip areas. This generates a binary mask which needs further morphological processing to remove areas that do not represent single chips. The enhanced binary mask is passed through another morphological operation that shrinks all shapes to a single point. Each point gives the coordinates of a chocolate chip, and by summing all the points, the total number of chips is found.

COLOR SEGMENTATION

The initial step for color segmenting chocolate chips is to determine the area in RGB space that encloses the proper range of brown colors. This can be experimentally done in MATLAB by selecting an area in the image with function *roipoly*, and calculating the means of the R, G, and B vectors. After repeating this process several times, take the mean of the means (Table 1) in order to get a good estimate for the RGB vector **a**.

Table 1. Mean Values for RGB Vectors

	Red	Green	Blue
	52.39	32.22	27.68
	58.49	38.57	32.20
	62.76	47.32	39.97
	63.81	38.48	30.84
	50.53	28.25	23.35
	59.40	33.67	27.29
	65.40	39.81	33.70
	57.77	34.36	29.94
	58.99	35.96	31.66
	56.47	34.75	32.48
Mean	*58.60*	*36.34*	*30.91*
Standard Dev.	*4.72*	*5.15*	*4.43*

Because the detection program will be using the bottom of cookies to locate chocolate chips, one additional step needs to be taken before color segmentation. The texture of a cookie bottom is such that it has many small holes with shadows that could be confused for a chocolate chip. The cookie image must be blurred with an averaging filter in order to eliminate this potential problem. An averaging filter of size 20 is applied to the image before any further processing (Figure 1).

Blurring the image also allows for a wider variance of the area enclosing colors in RGB space. Shown in Figure 1, the filter effectively eliminates all small divots on the cookie bottom by averaging them out. With only the chocolate chips remaining within proximity to the RGB vector **a**, a much larger standard deviation of 35 can be used for each layer to ensure that all chips are captured by segmentation.

Counting Chocolate Chips

Yu-Hong Yen

Department of Electrical Engineering and Computer Science,

Case Western Reserve University, Cleveland, OH, Email: yxy61@cwru.edu

Abstract

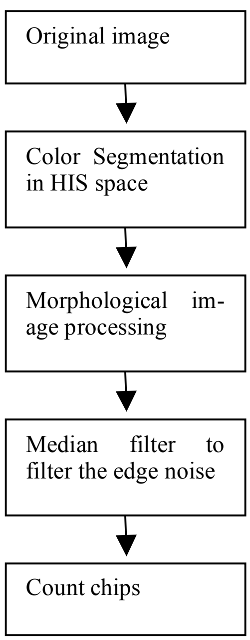
Image processing is being increasingly used in food inspection. In this project, image processing will be used to locate and count the chocolate chips visible in chocolate chip cookies. Common techniques of this project include color segmentation, filtering and Morphological image processing.

KEYWORDS

Color segmentation, Morphological image processing, Food inspection

INTRODUCE

The flowchart of this project is as following:



The first step is to choose an appropriate color space in which to operate from the wide variety of choices such as RGB, HSV, CMYK, YCbCr, etc [1]. RGB (red-green-blue) and HSV (hue-saturation-value) have been the most widely used. Figure 1 illustrates the geometries of the two spaces.

By way of example, HSV representation has certain advantages over RGB. In the RGB space, each of the three components may exhibit substantial variation under different lighting environments. In HSV space, however, the hue and saturation components are virtually unchanged. In odder to reduce the influence of illumination, I decided to use HSV space here.

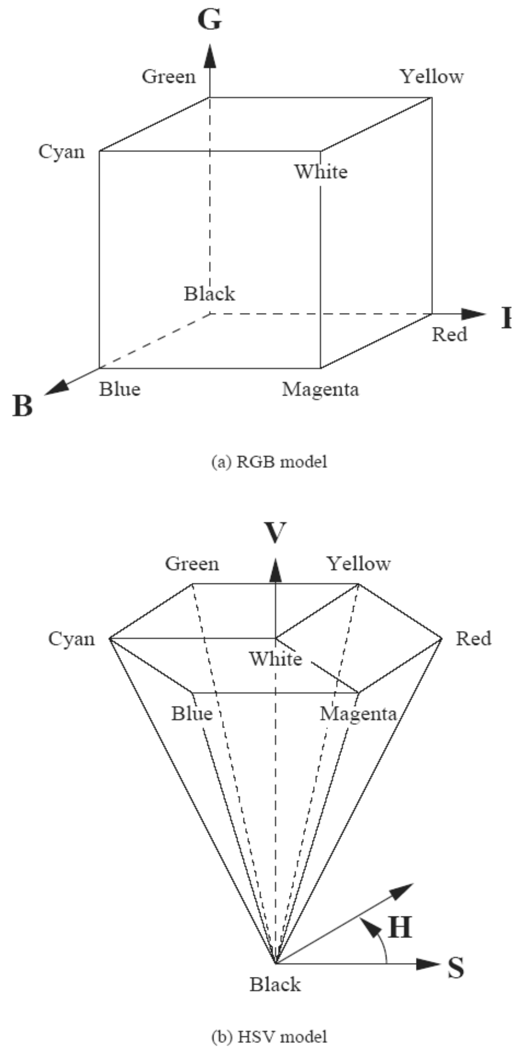


Figure 1: The RGB and HSV color models.

From the flow chart we can see that the next step after color segmentation is morphological image proc-