

Apple Grade Inspection by Using Machine Vision

Sourabh Bisht
Dept. of Electronics and Communication
Graphic Era University
Dehradun, India
sourabh_bisht2002@yahoo.com

Vikas Rathi
Dept. of Electronics and Communication
Graphic Era University
Dehradun, India
vikas.rth@gmail.com

Gaurav Pant
Graphic Era Hill University
Dehradun (UK)
gauravpant444@gmail.com

Vivek Verma
Dept. Of Electronics and Communication
Graphic Era Hill University
vivekverma_22@yahoo.com

Abstract We see the world around through our eyes. Our eyes are the sensory organs that capture images and transmit to our brain at very fast rate. The image is representation of real scene either in black & white or in colour. The brain performs various processing functions and vision is perceived. In human beings we make use of vision for accomplishing majority of our tasks. Blindfolding ourselves and observing how our daily routine is seriously hampered without our vision can easily verify this fact. Although the first machine that captured image was a pinhole camera that was invented way back in 1850s, which was followed by many advances in image capturing techniques. Black & white camera gave way to coloured camera, resolution of picture captured enhanced, moving pictures were captured using monochrome T.V. Camera followed by coloured T.V. camera and now a days we have digital cameras as small as a size of button, embedded in our mobile phones, at a price, a student can afford from his pocket money.

Keywords: *Apple, machine, colours, morphology*

I. INTRODUCTION TO MACHINE VISION BASIC

However, when we talk of machine vision, these cameras alone do not “see” the way we do. With human vision the process of “seeing” i.e. capturing images takes place at a very fast rate, almost continuously for several hours. The most important aspect is equally fast processing of these images so as to accomplish necessary decisions and actions. Image processing is collection of programs and techniques that improve, simplify, enhance or otherwise alter an image. Machine vision also aims at aping this task of capturing images at fast rates and simultaneously processing and analysis of these images so that machine can “see”, the way we do. Machine Vision is an emerging area in automation and control where in images are captured by camera in real time, which are further, processed and analyzed to get important information out of them. Sometimes processing and analysis is done on the images that are already stored in record. Machine Vision (MV) is the application of computer vision to industry and manufacturing. Whereas computer vision is mainly focused on machine-based image processing, machine vision often requires digital input/output devices and computer networks to control other manufacturing equipment such as robotic arms. Machine Vision is a subfield of engineering that encompasses computer science, optics, mechanical engineering, and industrial automation.

Problem Statement and Objective of Project

The primary work of this paper is to develop inspection algorithm for the parameters of apple such that the inspection system will be able to decide the grade of apple.

For development of algorithm and grading colour and size of apple is being considered.

Fruit quality assessment from estimation of

1. Its primary and secondary colours.
2. Its shape and size using digital image processing.
3. Grading of apple based on colour model and size.

II. RELATED THEORY

Image Types

For an identical spatial resolution, a colour image occupies four times the memory space of an 8-bit gray scale image, and a complex image occupies eight times the memory of an 8-bit gray scale image.

Binary Image

A binary image is a digital image that has only two possible values for each pixel. Typically the two colours used for a binary image are black and white though any two colours can be used. The colour used for the object(s) in the image is the foreground colour while the rest of the image is the background colour. In the document scanning industry this is often referred to as bi-tonal.

Binary images are also called bi-level or two-level. This means that each pixel is stored as a single bit (0 or 1). The names black-and-white, B&W, monochrome or monochromatic are often used for this concept, but may also designate any images that have only one sample per pixel, such as gray scale images. A binary image is usually stored in memory as a bitmap, a packed array of bits. A 640×480 image requires 37.5 KiB of storage. Because of the small size of the image files, fax machines and document management solutions usually use this format.

Gray scale

In photography and computing, a gray scale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray, varying from black at the weakest intensity to white at the strongest. Gray scale images are distinct from one-bit bi-tonal black-and-white images, which in the context of computer imaging are images with only the two colours, black, and white (also called bi level or binary images). Gray scale images have many shades of gray in between. Gray scale images are also called monochromatic, denoting the absence of any chromatic variation (i.e. one colour).

Gray scale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. infrared, visible light, ultraviolet, etc.), and in such cases they are monochromatic proper when only a given frequency is captured. But also they can be synthesized from a full colour image. Conversion of a colour image to gray scale is not unique; different weighting of the colour channels effectively represents the effect of shooting black-and-white film with different-coloured photographic filters on the cameras. A common strategy is to match the luminance of the gray scale image to the luminance of the colour image.

To convert any colour to a gray scale representation of its luminance, first one must obtain the values of its red, green, and blue (RGB) primaries in linear intensity encoding, by gamma expansion. Then, add together 30% of the red value, 59% of the green value, and 11% of the blue value (these weights depend on the exact choice of the RGB primaries, but are typical). Regardless of the scale employed (0.0 to 1.0, 0 to 255, 0% to 100%, etc.), the resultant number is the desired linear luminance value; it typically needs to be gamma compressed to get back to a conventional gray scale representation. This is not the method used to obtain the luma in the Y'UV and related colour models, used in standard colour TV and video systems as PAL and NTSC, as well as in the L*a*b colour model. These systems directly compute a gamma-compressed luma as a linear combination of gamma-compressed primary intensities, rather than use linearization via gamma expansion and compression. To convert a gray intensity value to RGB, simply set all the three primary colour components red, green and blue to the gray value, correcting to a different gamma if necessary. Gray scale as single channels of multichannel colour images. Colour images are often built of several stacked colour channels, each of them representing value levels of the given channel. For example, RGB images are composed of three independent channels for red, green and blue primary colour components; CMYK images have four channels for cyan, magenta, yellow and black ink plates, etc. Here is an example of colour channel splitting of a full RGB colour image. The column at left shows the isolated colour channels in natural colours, while at right there are their gray scale equivalences:

RGB to Gray scale conversion

A Matlab function, RGB2gray converts RGB images to gray scale by eliminating the hue and saturation information while retaining the luminance. Convert RGB image or colour map to gray scale.

The following equations convert an RGB image into a gray scale image on a pixel-by-pixel basis.

$$\text{Gray scale value} = 0.299R + .587G + 0.114B$$

This equation is part of the NTSC standard for luminance. An alternative conversion from RGB to gray scale is a simple average: gray scale value = $(R + G + B) / 3$

Binary Morphology

Binary morphological operations extract and alter the structure of particles in a binary image. We can use these operations during our inspection application to improve the information in a binary image before making particle measurements, such as the area, perimeter, and orientation. A binary image is an image containing particle regions with pixel values of 1 and a background region with pixel values of 0. Binary images are the result of the thresholding process. Because thresholding is a subjective process, the resulting binary image may contain unwanted information, such as noise particles, particles touching the border of images, particles touching each other, and particles with uneven borders. By affecting the shape of particles, morphological functions can remove this unwanted information, thus improving the information in the binary image.

Structuring Elements

Morphological operators that change the shape of particles process a pixel based on its number of neighbours and the values of those neighbours. A neighbour is a pixel whose value affects the values of nearby pixels during certain image processing functions. Morphological transformations use a 2D binary mask called a structuring element to define the size and effect of the neighbourhood on each pixel, controlling the effect of the binary morphological functions on the shape and the boundary of a particle.

Gray scale Morphology

Morphological transformations extract and alter the structure of particles in an image. They fall into two categories: Binary Morphology functions, which apply to binary images Gray scale morphology functions, which apply to gray-level images In gray scale morphology, a pixel is compared to those pixels surrounding it in order to keep the pixels whose values are the smallest (in the case of erosion) or the largest (in the case of dilation). Use gray scale morphology functions to filter or smooth the pixel intensities of an image. Applications include noise filtering, uneven background correction, and gray-level feature extraction. The gray-level morphology functions apply to gray-level images. We can use these functions to alter the shape of regions by expanding bright areas at the expense of dark areas and vice versa. These functions smooth gradually varying patterns and increase the contrast in boundary areas. This section describes the following gray-level morphology functions:

Morphology

Is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. The most basic morphological operations are dilation and erosion. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbours. By choosing the size and shape of the neighbourhood, you can construct a morphological operation that is sensitive to specific shapes in the input image.

I. Morphology Fundamentals

Dilation and Erosion Dilation and erosion are two fundamental morphological operations. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the structuring element used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbours in the input image. The rule used to process the pixels defines the operation as a dilation or an erosion. This table lists the rules for both dilation and erosion.

Combining Dilation and Erosion

Dilation and erosion are often used in combination to implement image processing operations. For example, the definition of a morphological opening of an image is erosion followed by dilation, using the same structuring element for both operations. The related operation, morphological closing of an image, is the reverse: it consists of dilation followed by erosion with the same structuring element.

Morphological Opening

Morphological opening is used to remove small objects from an image while preserving the shape and size of larger objects in the image. For example, you can use the `imopen` function to remove all the circuit lines from the original circuit image, `circbw.tif`, creating an output image that contains only the rectangular shapes of the microchips.

Dilation and erosion-based Functions and their morphological definition

Bwhitmiss :

Logical AND of an image, eroded with one structuring element, and the image's complement, eroded with a second structuring element

Imbothat

Subtracts the original image from a morphologically closed version of the image. Can be used to find intensity troughs in an image.

Imclose

Erodes an image and then dilates the eroded image using the same structuring element for both operations.

Imtophat

Subtracts a morphologically opened image from the original image. Can be used to enhance contrast in an image.

3. Implementation Methodology

Colour Analysis of Fruits Based on Histogram Image

The histogram of a digital image with Gray levels in the range $[0, i-1]$ is a discrete function, $\tilde{n}(rk) = nk / n$, Where rk is the k th Gray level nk is the number of pixels in the image with that Gray level n is the total number of pixels in the image and $k = 0, 1, 2, \dots, i-1$ $\tilde{n}(rk)$ gives an estimate of the probability of occurrence of Gray level nk . A plot of this function for all values of k provides a global description of the appearance of an image. Although the properties are global descriptions that say nothing specific about image content, the shape of the histogram of an image gives us useful information about the possibility for contrast enhancement. For classification purposes we compare the test histogram with the reference histogram and select the best match as the one whose histogram is most similar to that of the test item. For multi spectral imagery, the histogram of each component image can be displayed.

The graphical representation of an image histogram is a plot of the percentage of image pixels at each digital intensity value. The input data for the histogram consists of a table of numbers containing the number of pixels at each intensity level. The first step in generating a graphical display of the histogram is to convert the distribution into a percentage distribution by dividing the table entries by the number of pixels within the image. The second step involves allocation of the number of pixels to be used in the annotation area for presentation of the histogram

Analysis using RGB model

In case of colour image, in the RGB colour space, every individual colour component, namely Red, Green and Blue has its histogram. Then, the percentage composition of every individual colour component, which an fruit possesses are to be evaluated. Using this percentage composition the level for a component can be set as a standard in classifying the apples based on a particular colour orientation. For an Apple the higher percentage composition of the red component was assigned the superior grade, the next lower composition the second grade & likewise the descending grades were assigned. This enabled the sorting of apples based on the colour as a parameter.

Image analysis using HSI

RGB data is first converted into HSI data. With image representation in the HSI domain, the colour analysis was based on primarily the Hue value. Hue is a colour attribute that describes a pure colour, whereas saturation gives a measure of the degree to which a pure colour is diluted by white light and finally intensity gives the effectiveness of the colour. The three dimensional RGB space is reduced to a one-dimensional 'H' Space for colour analysis. For a digitized colour image, the Hue histogram represented the colour components and the amount of that Hue in the image. Therefore, colour evaluation of fruits was achieved by analyzing Hue histograms [6,7,8,11]. The advantage of using HSI model over RGB is as under: The intensity

component is decoupled from colour information Hue and Saturation components are intimately related to the way in which human being perceive colour. Hue value is invariant to changes in light intensity. These features make the HSI model an ideal tool for developing digital image processing algorithms based on colour sensing properties of human visual system

Methods of analysis

The Hue, Saturation and Intensity (HSI) domain was chosen to quantify Apple colour by machine inspection [9], since this representation is closest to that of human perception and use of the Hue domain allowed easy segmentation of the apple from the background. However, colour image information is sensed using combination of RGB. Various colour features can be calculated from the RGB components by using linear or non-linear transformation.

Hue and Saturation are calculated from RGB[10,12] values by

$$+ (2R - G - B) / [3(G - B)] + (180) (255/360)$$
$$S = \{ [\min(R,G,B) / I] \} \times (255/360)$$

Other method

1: Colour space

The Image Processing Toolbox represents colours as RGB values, either directly (in an RGB image) or indirectly (in an indexed image, where the colour map is stored in RGB format). However, there are other models besides RGB for representing colours numerically. The various models are referred to as colour spaces because most of them can be mapped into a 2-D, 3-D, or 4-D coordinate system; thus, a colour specification is made up of coordinates in a 2-D, 3-D, or 4-D space. The various colour spaces exist because they present colour information in ways that make certain calculations more convenient or because they provide a way to identify colours that is more intuitive. For example, the RGB colour space defines a colour as the percentages of red, green, and blue hues mixed together. Other colour models describe colours by their hue (green), saturation (dark green), and luminance, or intensity. The toolbox supports these colour spaces by providing a means for converting colour data from one colour space to another through a mathematical transformation.

Types of colour space

I: YCbCr Colour Space: The YCbCr colour space is widely used for digital video. In this format, luminance information is stored as a single component (Y), and chrominance information is stored as two colour-difference components (Cb and Cr). Cb represents the difference between the blue component and a reference value. Cr represents the difference between the red component and a reference value. (YUV, another colour space widely used for digital video, is very similar to YCbCr but not identical.) YCbCr data can be double precision, but the colour space is particularly well suited to uint8 data. For uint8 images, the data range for Y is [16, 235], and the range for Cb and Cr is [16, 240]. YCbCr leaves room at the top and bottom of the full uint8 range so that additional (non image) information can be included in a video stream. The function `rgb2ycbcr` converts colour maps or RGB images to

the YCbCr colour space. `ycbcr2rgb` performs the reverse operation. For example, these commands convert an RGB image to YCbCr format. `RGB = imread('peppers.png');`
`YCBCR = rgb2ycbcr(RGB);`

II: HSV Colour Space : The HSV colour space (Hue, Saturation, Value) is often used by people who are selecting colours (e.g., of paints or inks) from a colour wheel or palette, because it corresponds better to how people experience colour than the RGB colour space does. The functions RGB to HSV and HSV to RGB convert images between the RGB and HSV colour spaces. Note MATLAB and Image Processing Toolbox do not support the HSI colour space (Hue, Saturation, Intensity). However, if you want to work with colour data in terms of hue, saturation, and intensity, the HSV colour space is very similar.

Another option is the use the LCH colour space (Luminosity, Chroma, and Hue), which is a polar transformation of the CIE L*a*b* colour space — see *Converting Between Device-Independent Colour Spaces*. As hue varies from 0 to 1.0, the corresponding colours vary from red through yellow, green, cyan, blue, magenta, and back to red, so that there are actually red values both at 0 and 1.0. As saturation varies from 0 to 1.0, the corresponding colours (hues) vary from unsaturated (shades of gray) to fully saturated (no white component). As value, or brightness, varies from 0 to 1.0, the corresponding colours become increasingly brighter.

Grading on basis of size:

Fruit shape is one of the most important quality parameters for evaluation by consumer performance. Also misshapen fruits are usually rejected according to grading standards of fruit. This case is more considered for watermelons, because of large size and sensitive skin. This study was carried out to determine “detection algorithm” for misshapen watermelons. Physical characteristics of watermelon such as mass, volume, dimensions, density, spherical coefficient and geometric mean diameter were measured. Relations and correlations coefficient obtained between above characteristics for normal and non-standard fruit shape. It was found that weight of normal watermelon could be determined by image analysis with error 2.42%. In addition fruit shape of long type watermelon (Charleston Gray cv.) in front view was well-described by an ellipsoid model with $R^2 = 0.97$. Finally, the results indicated that length to width ratio and fruit area (2D) to background area ratio can be used to determine misshapen fruit. Shape is one of the most important quality parameters of fruits. Fruit shape is affected by inheritance in addition to environmental growing conditions. In some fruits as watermelon, sensitive skin is cause of deformation in fruit shape when confronted with other object, also insufficient pollination results in misshapen melons. Recently, some researchers produced cube and pyramid watermelons by using this attribute.

Essentially, description of fruit shape is often necessary in horticulture research for a range of different purposes including cultivar descriptions in applications for plant variety rights or cultivar registers (Anonymous, 1997; Beyer *et al.*, 2002; Hasnain *et al.*, 2003), evaluation of consumer performance (Gerhard *et al.*, 2001), investigating heritability

of fruit shape traits (Currie *et al.*, 2000; White *et al.*, 2000), analysis stress distribution in the fruit skin (Considine & Brown, 1981), parameters using in packaging and shipping, or determining misshapen fruit in a cultivar, etc.

In other side official quality definitions for fruit or vegetables are hardly more than a rough ration on size and colour. Where the USDA grade standard for watermelons (USDA, 1997) specifies two shapes in three classifications (U.S. Fancy, U.S. No. 1, U.S. No. 2) based on visual comparison of fruit shape relative to reference drawing.

These drawing serve as reference in classifying cultivars and normal for fruit shape. Ratings based on visual comparison do not require any equipment. However, the method is subjective and may depend on individual performing the rating. Also, rating scores may be biased by confounding variables such as fruit size or fruit colour. Therefore this procedure run very slowly and seem not sufficient for classification fruit in distribution terminals. Alternative approaches characterize fruit shape using indices calculated from outer dimensions of fruit (e.g. tomato, Ku *et al.*, 1999; pear, White *et al.*, 2000). Since based on direct measurement, the approach is objective and expected to be reproducible. Further, if measurement were performed by caliper, no sophisticated equipment would be needed; while useful in some applications, other applications may require a continuous description of the fruit shape.

The main objective of our research is to develop a fast procedure that allows an un-biased and reproducible quantitative description of fruit shape in watermelon that is based on image analysis and utilizing of standard software for data handling and analysis. Physical properties of normal fruits. In order to obtain required parameters for detection algorithm and calibration images, physical properties e.g. three mutually perpendicular axes, major (a, longest intercept), intermediate (b, longest intercept normal to a) and minor (c, longest intercept normal to a & b) mass, volume were measured. The area is computed from the 2D image of watermelon and so an estimate of the watermelon's mass/volume is achieved. The highest degree of correlation was found between area measured by image analysis and mass measured using a scale:

$$M = 0.8638A = 0.99 \quad (1)$$

Where

A is the area (cm²) and M is the mass (g).

Using the above relationship, the mean absolute error of weighing a watermelon by determining the area using image analysis would be 2.42% (approximately 156 g for 6000 g watermelon). Thus, size and size distribution or mass and mass distribution can be recorded automatically with a high degree of accuracy by image analysis.

For mathematical describing of normal shape long type watermelon, image data was subjected to regression analysis. Non-linear regression based ellipsoid shape were carried out using the SPSS program package (SPSS), Version 8.0, SPSS Inc., Chicago, IL, USA):

$A = K \cdot \frac{a^2 \cdot b}{c}$ (2) Where Aest is estimated area (cm²), a is the longest intercept and b is the longest intercept normal to a. Parameter K was calculated with using initial value 3.1415 and after run the model K was achieved 3.3347 on R-square 0.97. A simple method of judging based on ellipsoid shape of long watermelon was used for detecting of some

misshapen fruits. Ellipsoid ratio (E) shows fruit area to background area, which is measured for all samples. This ratio for normal shape long watermelon on based of Eq. (2) is about 5.25.

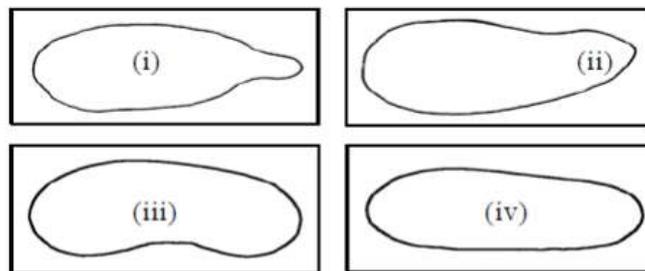


Figure 1 : Four types misshapen fruit in long watermelons (i) bottlenecks (ii) gourdnecks (iii) curvy shape and (iv) flattened

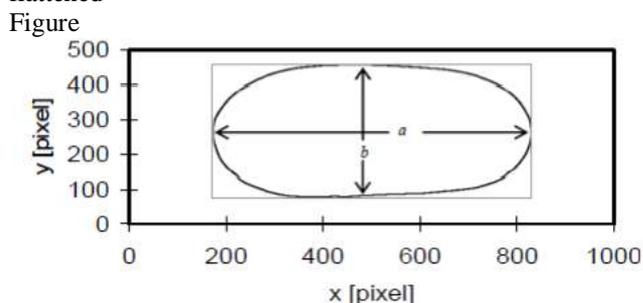


Figure 2 : Method of food acquisition of fruit shape data

3.4 Method and implementation

Classification based on histogram distribution

The probability density of the Hue pertains to surface colour is maximum at the median density pixel value. Hence the median density Hue value represents the overall distribution of surface colour of the apples. This method requires extensive training. The Meadin density of Hue histogram for a given class is taken as the basis for segregating the fruits belonging to that grade. Hence during the training process 20 Apple samples belonging to a GRADE are taken and meadian density of the combined histogram is calculated, thus obtaining a distinct median density value for each of the GRADE to build the reference table.

Inspection Block Diagram

The inspection block diagram of apple grading system is shown in figure 6. A sample image of 'A' grade apple is used to extract the feature database for the grading inspection. This feature database is compared with the features of tested image to decide the grade of test image. The sample image is proceeded with algorithm like histogram, RBG colour plane extraction etc for extracting the features of sample image. Once the feature database is created it is compared each time when new image is loaded. This inspection process is called feature based inspection process.

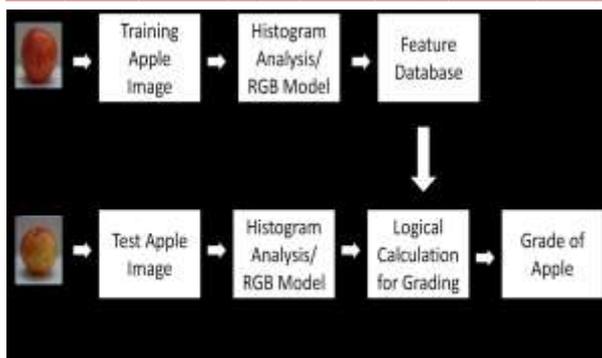


Figure 3: Block Diagram of Inspection Algorithm

Grading Flow chart

The grading flow graph for apple inspection is shown in figure 7. In the inspection process the loaded test image goes under two inspection process 1. Colour Analysis and 2. Area Analysis. The feature extracted in form of 'Area of Apple' and percentage of RED' from these two analysis are used in grading algorithm. On the basis of feature database the cut off values and grading condition can be decided. The features of test image are compared with these cut off values and grading is decided with grading algorithm.

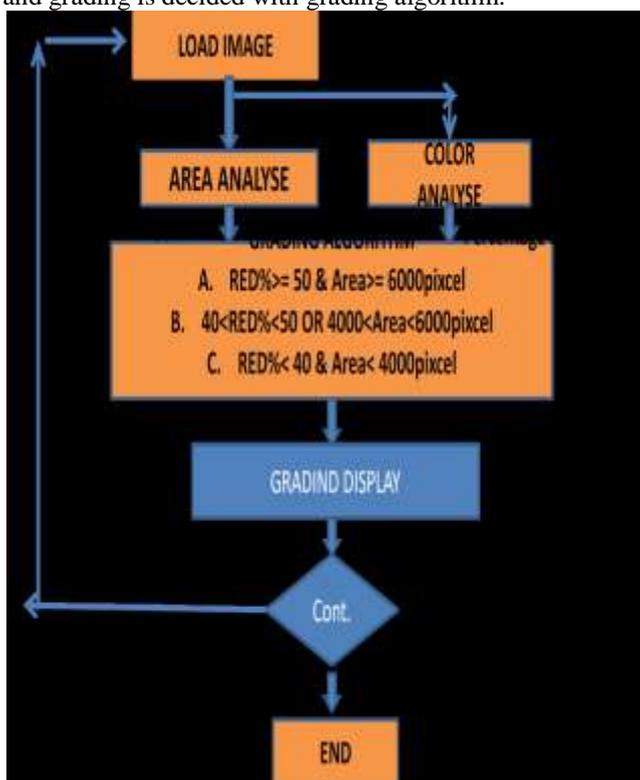


Figure 4: Grading flow graph

Function used for morphological process

1 - Strel - Create morphological structuring element. Function of strel : . SE = strel('disk', R, N)- creates a flat, disk-shaped structuring element, where R specifies the radius. R must be a nonnegative integer. N must be 0, 4, 6, or 8. When N is greater than 0, the disk-shaped structuring element is approximated by a sequence of N periodic-line structuring elements. When N equals 0, no approximation is used, and the structuring element members consist of all

pixels whose centers are no greater than R away from the origin. If N is not specified, the default value is 4.

2 - Imopen - Morphologically open image. Remove all the circuit lines from the original circuit image.

IM2 = imopen (IM,SE) - performs morphological opening on the gray scale or binary image IM with the structuring element SE. The argument SE must be a single structuring element object, as opposed to an array of objects. The morphological open operation is erosion followed by a dilation, using the same structuring element for both operations.

3 - imerode - Use to erode an image.

The imerode function accepts two primary arguments:

(i)The input image to be processed (gray scale, binary, or packed binary image)

(ii)A structuring element object, returned by the strel function, or a binary matrix defining the neighbourhood of a structuring element

IM2 = imerode(IM,SE) erodes the grayscale, binary, or packed binary image IM, returning the eroded image IM2. The argument SE is a structuring element object or array of structuring element objects returned by the strel function.

4 - imreconstruct - Morphological reconstruction can be thought of conceptually as repeated dilations of an image, called the marker image, until the contour of the marker image fits under a second image, called the mask image. In morphological reconstruction, the peaks in the marker image "spread out," or dilate.

IM = imreconstruct(marker,mask) performs morphological reconstruction of the image marker under the image mask. Marker and mask can be two intensity images or two binary images with the same size. The returned image IM is an intensity or binary image, respectively. Marker must be the same size as mask, and its elements must be less than or equal to the corresponding elements of mask.

5 - imclose - Dilates an image and then erodes the dilated image using the same structuring element for both operations

IM2 = imclose(IM,SE) performs morphological closing on the gray scale or binary image IM, returning the closed image, IM2. The structuring element, SE, must be a single structuring element object, as opposed to an array of objects. The morphological close operation is a dilation followed by an erosion, using the same structuring element for both operations.

6 - imdilate - use to dilate image.

IM2 = imdilate(IM, SE)- dilates the gray scale, binary, or packed binary image IM, returning the dilated image, IM2. The argument SE is a structuring element object, or array of structuring element objects, returned by the strel function. If IM is logical and the structuring element is flat, imdilate performs binary dilation; otherwise, it performs gray scale dilation. If SE is an array of structuring element objects, imdilate performs multiple dilations of the input image, using each structuring element in SE in succession.

7 - imcomplement - It complement the image. in the complement of a binary image, zeros become ones and ones become zeros; black and white are reversed. In the complement of an intensity or RGB image, each pixel value is subtracted from the maximum pixel value supported by the class (or 1.0 for double-precision images) and the

difference is used as the pixel value in the output image. In the output image, dark areas become lighter and light areas become darker.

$IM2 = \text{imcomplement}(IM)$ computes the complement of the image IM. IM can be a binary, grayscale, or RGB image. IM2 has the same class and size as IM.

8 – imregionalmax - Used for Regional minima. Regional minima are connected components of pixels with a constant intensity value, and whose external boundary pixels all have a higher value.

$BW = \text{imregionalmin}(I)$ - computes the regional minima of I. The output binary image BW has value 1 corresponding to the pixels of I that belong to regional minima and 0 otherwise. BW is the same size as I.

9 - bwarea - Area of objects in binary image. The bwarea function returns the area of a binary image. The area is a measure of the size of the foreground of the image. Roughly speaking, the area is the number of on pixels in the image. bwarea does not simply count the number of pixels set to on, however. Rather, bwarea weights different pixel patterns unequally when computing the area. This weighting compensates for the distortion that is inherent in representing a continuous image with discrete pixels.

$\text{total} = \text{bwarea}(BW)$ - estimates the area of the objects in binary image BW. total is a scalar whose value corresponds roughly to the total number of on pixels in the image, but might not be exactly the same because different patterns of pixels are weighted differently.

III. RESULT AND ANALYSIS:

Grading Estimation of Apple by Colour Component Analysis

Image Reading and ROI (Region of Interest) implementation

The sample image of apple is loaded to MatLab workspace by command given below. For color based inspection of grading the RED component plays significant role. If we include the background in RED component it will raise the error in the grading decision. This operation reduces amount of grading error in the inspection system. The Matlab command for creating ROI is given below.

- Reading of image from file
`aaa=imread('D:\appleimage\A0000.jpg');`
- Creating ROI and Cropping of image
`a=imcrop(aaa);`

Graphical Analysis

From the extracted color component mean and percentage of color components are calculated are given below and shown in form of bar graph in fig.4.7 and 4.8.

- Red Mean = 9.648382e+001
- Green Mean = 4.353273e+001
- Blue Mean = 2.972970e+001
- Red % = 5.684003e+001
- Green % = 2.564577e+001
- Blue % = 2.886599e+001

As we analyze the graphs it clear that for the grade 'A' apple has higher RED mean and percentage. On the basis of this analysis we can set a lower cut off value for grade 'A' in

terms of RED colour component mean and percentage. Here in our work only RED percentage is taken for simplicity of logical calculation.

Grading estimation by apple size calculation

As in manual inspection of grading the size of apple is also significant. So for grading inspection system it is required to calculate the size of apple. For size calculation it required to convert input image into binary image where the black or white region are required size. But due to lightning condition, reflection from surface and other reasons, it difficult to find black or white reason equivalent to apple area. To solve this problem we required to preprocess the input image with different morphological functions, given below

Applying image conversion and morphology

The color image is not suitable for morphological process so it is required to convert it into gray scale image. The gray scale is further processed with morphological function for calculation of size. The resultant image of apple after grayscale conversion and morphological function are shown in fig. 5(b). In fig it can be seen clearly that morphological functions reduces the bright region within the apple area. This helps in the calculation of apple size.

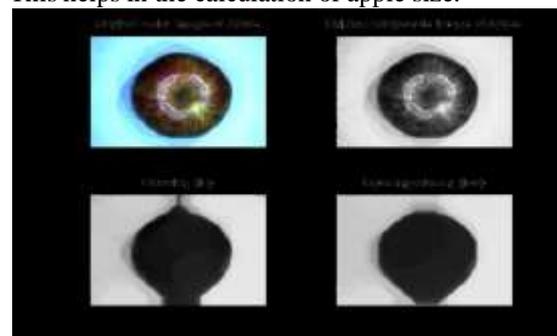
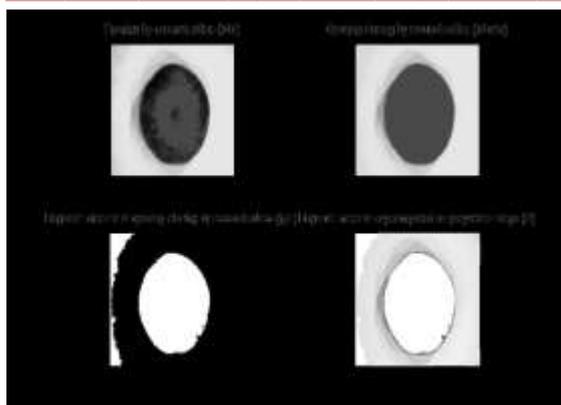


Figure 5. (a) Original image (b) After Grayscale conversion (c)&(d) After Morphological functions

Morphological operation for finding the area of apple

For calculation of apple area it is required find the region equivalent to apple size. By using morphological operations a maxima region is constructed which is superimposed with original gray scale and colour image. It is clear from resultant images (fig. 5 d & fig.6 b), the calculated region is equivalent to apple size. In this process the gray scale image goes under 'Image Opening' morphological function. But the resultant image (fig.5 (a)) the uniform black area is only at center, which will arise the error in size calculation. This image is further processed followed by 'Image closing' morphological function and significant improvement in resultant image shown in fig5 (b). From the resultant image (c) shows the calculated size of apple and from superimposed region fig. 5(d) it is clear that the calculated region is equivalent to size of apple.



6. (a) Opening of image (b) After opening and closing (c) After finding the maxima of region (d) After superimposition with original image

References

- [1] Tadhg Brosnan, Da-Wen “Inspection and grading of agricultural and food products by computer vision systems*/a review” Sun Computers and Electronics in Agriculture 36 (2002) 193_/213
- [2] Ballard, D.A., Brown, “ Computer Vision. “Prentice”-Hall, Eaglewood Cliffs, NJ. C.M .1982
- [3] Moustafa Fattal Ahmad El Turk Monah Dahrouj Ayman Badereddin ‘Automatic Apple Sorting and Grading using Machine Vision’ Department of Electrical and Computer Engineering American University of Beirut
- [4] P. Sudhakara Rao, A. Gopal, R. Revathy and K. Meenakshi “colour analysis of fruits using machine vision system for automatic sorting and grading” j. instrum. soc. india 34 (4) 284-291
- [5] S.Arivazhagan1, R.Newlin Shebiah , S.Selva Nidhyanandhan, L.Ganesan ‘ Fruit Recognition using Color and Texture Features’
- [6] Guo Feng and Cao Qixin ‘Study on Color Image Processing Based Intelligent Fruit Sorting System
- [7] Fernando Mendozaa, Petr Dejmekb, Jos’e M. Aguilera ‘Calibrated color measurements of agricultural foods using image analysis
- [8] Sonka, M, Hlavac, V, Boyle, R.,”Image Processing, Analysis, and Machine Vision.” PWS publishing, 1999 California, USA.
- [9] Baxes, Digital Image Processing Principles and Applications. Wiley, G.A. 1994, New York, USA
- [10] Hassan Sadrnia, Ali Rajabipour1, Ali Jafary,Arzhang Javadi† And Younes Mostof, “Classification and Analysis of Fruit Shapes in Long Type Watermelon Using Image Processing” international journal of agriculture & biology 1560–8530/2007/09–1–68–70