Blemish detection in citrus fruits

Abdul Malik Khan[1] and Andrew P. Papliński[2]

***Abstract:* - In this paper we introduce a cascaded-classifier approach to localize citrus fruit blemishes and identify the candidate blemishes for stem-ends and navel of citrus fruit oranges. All the blemish candidates are defined as 'blobs', the term is used to define non-definite shape and sizes of the candidates for blemish, stem and navel. First classifier (system - A) extracts blobs objects, whereas the second classifier (artificial neural networks) discriminates blobs as stem-ends and navel from others blemish (defect) type. Overall system-A is tested by 1400 fruits from which first classifier found 1062 blob objects. Several features are extracted from these blob objects and these features are then selected by forward selection method. With these selected features the second classifier reached to 81% recognition rate.**

***Index Terms:* - Blemish, image segmentation, Feature selection, computer vision, stems, navel, entropy, nearest-neighborhood, Range filter, k-nearest, boundaries, neural network.**

I. INTRODUCTION

Machine vision systems are largely employed for automatically controlling or analyzing processes or activities in many industries like automotive, electronics, food & beverages, pharmaceutical, textile, etc. One of the most popular applications of machine vision is to inspect qualities of produced goods based on form, color, shape, size and presence of defects. Machine vision systems benefit from specially designed image processing software to perform such particular tasks; therefore image processing plays a very crucial role in their performance. Machine vision-based fruit grading is an important and necessary task for fruit marketing. In this area discrimination of stem-ends or navel from fruit blemishes, which can lead to incorrect grading, is still an important and open problem being investigated.

Physical appearances of fresh fruits and vegetables extremely vary, causing difficulties for machine vision systems. Fruits, in particular, have numerous kinds of blemishes and highly varying skin color. Hence, they pose even more problems for machine vision-based quality inspection systems. Recent studies in machine vision-based quality inspection of fruits revealed that certain blemish are more visible at certain spectral ranges. Consequently, visible and near infrared imaging has become very popular for inspection of fruit.

In computer vision-based approaches, Yang [1] used a structured light pattern with artificial neural networks to discriminate stem-ends and calyxes. Penman[2] illuminated apples with blue linear light and used reflection patterns (light stripes) of the fruit acquired by a CCD-camera to locate stem-ends and calyxes as well as blemishes. Li et al. [3] introduced fractal dimensions with artificial neural networks to discriminate stem-end and calyxes from defects. Leemans and Destain [4] used a correlation-based pattern matching technique to localize calyxes and stem ends. Wen and Tao [5] used histogram density of an extracted object to discriminate stem-ends and calyxes from defects in a rule-based system.

Fruit is susceptible to numerous kinds of injuries caused by natural or handling factors [6]. In order to detect these injuries researchers have explored different sensing techniques like X-ray imaging [7][8], magnetic resonance imaging (MRI) [9], thermal imaging [10] and spectral reflectance based methods [11][12][13][14]. However, visible/NIR imaging techniques are the dominantly used ones for blemish detection due to their reasonable cost and relatively high processing speed.

Segmentation is the process of extracting important objects by partitioning an image into foreground and background pixels. In order to accurately perform quality inspection of fruit by machine vision, robust and precise segmentation of blemish skin is crucial. Therefore, blemish segmentation is our main concern. Du and Sun [15] has divided image segmentation techniques used for food quality evaluation into four groups:

- Thresholding-based

Majority of the works performing blemish segmentation of fruit by thresholding used simple-global techniques [16][17][18][19].

- Region-based

Region-based techniques segment images by finding coherent, homogeneous regions subject to a similarity criterion. They are computationally more costly than thresholding-based ones.

- Edge-based

These techniques segment images by interpreting gray level discontinuities using an edge detecting operator and combining these edges into contours to be used as region borders. This technique requires high computation power.

- Classification-based

These techniques attempt to partition pixels into several classes using different classification methods.

For this specific problem of detecting blemishes and identifying some of these blemishes as candidates for stem and navel we propose a cascaded solution to first identify all blemishes and then to discriminate between stem-ends and navel from regular blemish. In the first step, candidates called blobs for defect, stem-ends and navel are found by pixel-based search of fruit skin with system-A. Then these blobs are further analyzed by neural nets to segregate false stem-ends and navel-ends.

1 Faculty of Information Technology, Monash University, Clayton Campus, Wellington Road, Vic 3800, Australia, Malik.Khan@infotech.monash.edu.au

2 Faculty of Information Technology, Monash University, Clayton Campus, Wellington Road, Vic 3800, Australia, Andrew.Paplinski@infotech.monash.edu.au

II. METHODS

A. Image color model

Color contains important information about fruit status and in some cases it is decisive for fruit quality differentiation. An adequate color classification can improve system accuracy and productivity. Large-scale utilization of automatic classification system for this purpose demands a robust color classification even under different color saturation, variations of environment lighting and light reflections.

A color model is a 3D unique representation of a color. There are different color models and the use of one over the other is problem oriented. For instance, the color model RGB is used in hardware applications like PC monitors, cameras and scanners, the CMY color model is used in color printers, and the YIQ model in television broadcast. In color image manipulation the two models widely used are HSI and HSV. [20][21]

XYL model: The XYL model described by Charles Esson et al 2007 [22] uses half of the XY space to represent the chromic sensation relevant to citrus fruit sorting. The XYL transform expands the HSI color space in the range of interest and contracts the space that is not used. We have chosen the HSI model described by Allen Hanbury et al, 2005 [23] and Charles Esson et al 2007 [22] and transform this model to XYL color space. For machine vision applications it is important to encode as much color information in small number of bits for fast processing. Charles Esson et al 2007 [22] discusses the XYL color transform space, a transform that aims to meet these goals.

The source, reflection and sensor are the three functions that affect the fruit image capture. These three sensors source, reflection and sensor are not linear and are spread over the visible spectrum. The tri-value response is the result of these three functions interacting. Once a tri-value is selected to represent the resultant function, considerable information is lost. The transforms under discussion are not dealing with this issue. The discussion is limited to retaining the tri-value accuracy as we move the data through different transforms whose results are encoded in a limited number of bits. Therefore the XYL color model is represented and maintained as a simplistic HSI model and hence suites our application for the extraction of features for discriminating the blobs for defects, stem and navel.

B. Image acquisition

A multi-view camera was setup for the image acquisition that combines the traditional color information (RGB), with the near infrared information (I) of the same scene. This gives us multi-view images which are overhead and mirror picture of the fruit. The camera acquires an image composed by four bands: red (R), green (G), blue (B) and near infrared (I) of the same scene. The system has captured fruit images under laboratory conditions and the fruit used for the blemish segmentation test were citrus fruit oranges. To singularize the fruits and estimate the size, the system can use the monochromatic information provided by the NIR while for color estimation and blemish detection it is necessary to work with all four bands with the use of HSI color models as explained in earlier section. [22]

C. Fruit Image database

The fruit database consists of images of 1400 Orange citrus fruit acquired by multi-view camera in RGB and another near-Infrared camera at the research lab of color vision systems, Melbourne as shown in Fig. 1. [24] Each image has a dimension of 164x113 pixels with 24 bits-per pixel resolution. 338 of the images contain totally healthy skin, whereas 441 of them include stem-ends or navel and the rest 621 have blemishes of various size and kind.

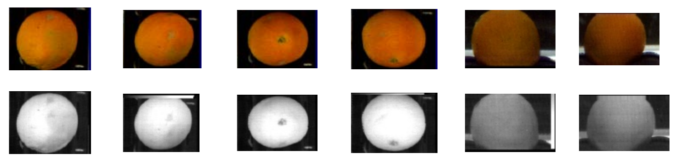


Fig. 1. Shows images of a single fruit both in visible (1st row) and NIR (2nd row) consisting in a series of strips.

A training database of defected (Blemished) and stem and navel regions within the database were marked manually. This database again consisted of 1400 fruit sets. We would refer the above training data as reference images for training and calibration purposes. We also have the unknown or the thruthing database fruit image set for performance evaluation for detecting blemished and fruit evaluation.

D. System Architecture

We use the external blemish detection of fruit using visible and NIR imaging which is dominant due to its low cost and high speed that is used by CVS. [24] Blemish detection requires accurate segmentation. In order to segment blemish on visible/NIR images we propose to use thresholding based or region based blemish classification. Detailed approach is explained in the following sections, where we propose a novel technique for citrus fruits this is based on external blemish segmentation using global and local thresholding techniques at pixel level on fruit are applied for blemish segmentation and applying pattern recognition to identify the blemish stem and navel.

This proposed method computes global image threshold. The global threshold (LEVEL) is used to convert an intensity image to a binary image and then to a normalized intensity value. The image is initially segmented into two parts using a starting threshold value and an intermediate threshold. The process is repeated, based upon the new threshold, until the threshold value does not change any more. The proposed segmentation techniques also use local Ni-black method for statistical classifiers and artificial neural networks. [25] This technique is used due to its performance in terms of speed and accuracy which is a requirement for real-time application for fruit sorting.

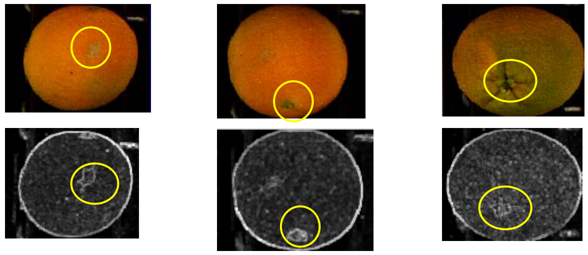


Fig. 2. Shows images of fruit with original RGB and marked segmentation of Blemish (Defect), Navel and stem-end.

Overall system for stem-end navel identification proposed here consists of a segmentation step and two cascaded systems as shown in Fig. 3:

- *Segmentation:* Extracts region-of-interest (roi) of fruits.
- System-A: Finds potential stem-end/navel objects within the roi of a fruit.
- System-B: Decides if an object found by System-A is really a stem-end/navel object or not.

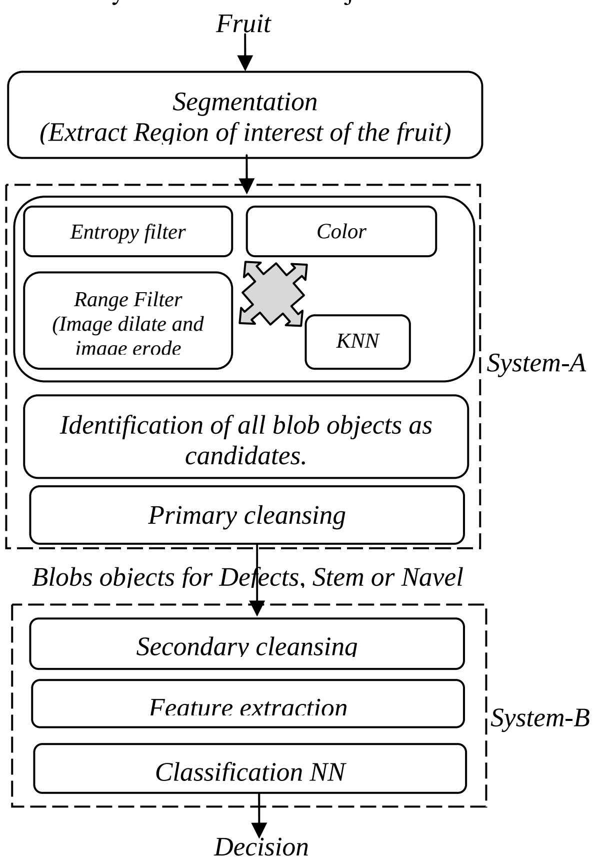


Fig. 3. System architecture

In Segmentation step, fruit area is extracted from the background by thresholding. Thresholding based on several variables where the sensor has more than one variable to characterize each pixel in an image i.e. RGB. Each pixel is characterized by 3 values, and the histogram becomes 3D. So thresholding now is concerned with finding clusters of points in 3D space. We must make a note that instead of the RGB model, the HSI model can also be used too. Equations 1 and 2 explain the multiple sub-regions r1, r2, r3 that are processed so that:

$$\bigcup_{i=1}^{n} r_i = r \qquad (1)$$

Where r_i is a connected region, $i = 1, 2, \ldots, n$

$$r_i \cap r_j = 0 \text{ for all } i \text{ and } j, i \neq j \qquad (2)$$

$$\left.\begin{array}{l} P(r_i) = TRUE \text{ for } i = 1, 2, ., n \\ P(r_i \cup r_j) = FALSE \text{ for } i \neq \end{array}\right\} \quad P(r_i): \text{logical predicate}$$

These regions grow by pixel aggregation, it starts with a set of "seed" points and from these grow regions by appending to each seed point those neighboring pixels that have similar properties. This will give us the extracted region-of-interest which is presented to the System-A. A series of steps are performed simultaneously to the roi. These are morphological filling, range filtering, and color transform of identified segmented images. Morphological filling is applied to remove holes caused by previous step, image is eroded by a large boundary of mask giving us the candidates for System-A. Erosion step is used to improves the accuracy of System-A, which otherwise tends to misclassify outer parts of a fruit probably due to illumination artifacts related to varying slope of fruit surface relative to camera and camera-angle. Fig. 4 shows an example of region-of-interest extraction of a fruit. Excluding outer regions of a fruit by erosion means leaving those regions uninspected, which is not desired.

This paper includes research with one-view images of fruits. However the future image acquisition system of this research project will be capable of acquiring multiple views of the fruit. So, for the future system if the size of mask for erosion is carefully chosen, regions of fruit excluded from the roi of one view will hopefully be included in the roi of another. System-A does a pixel-based search of the fruit skin for all potential blobs and is marked as candidates for blemish, stem-end and navel objects by producing and overlapping multiple image structures. Referring to Fig. 3 we perform entropy filtering of the fruit image in grey scale with the neighborhood of true-9, i.e. where each output pixel contains the entropy value of the 9-by-9 neighborhood around the corresponding pixel in the segmented image. In parallel create a second color transformation structure sRGB to Lab conversion by applying color transform function to the segmented image. And finally local range of this segmented image is found by using range filter morphological functions using image dilate and erode to determine the maximum and minimum values in the specified neighborhood. Mapping all the previous steps at pixel level together we now get all the candidates or blobs including the blemish stem and navel pixel points with its x and y co-ordinates. All the non-matching candidates in these steps will be cleansed as irrelevant candidates to the blobs; this is due to the fact we have a very small blob which does not satisfy the criterion for either blemish or stem or navel. This step of cleansing we call it primary-cleansing; this ends the tasks for the System-A.

Now referring in Fig. 3 System-B all the blobs are marked with pocket identifier including the actual detected stem and navel candidates as blobs until this step. The training data which is marked with reference true-positive to all stem and navel are produced to the neural nets to do the training of artificial neural network, which is trained by 'stem-end/navel' and 'healthy' pixels extracted from reference images. The blobs with smaller number of unconnected pixels points called redundant blobs within the blobs found by System-A are eliminated and we call this step as secondary cleansing, because a stem-end or navel ends including the defects cannot be very small, a priori. System-B extracts features from the blobs objects provided by System-A to classify the object as part of stem-end, navel or a blemish (defect) by a nearest neighbor classifier, which is trained by the objects of reference images.

In order to eliminate the intersection of training and test sets, and to have realistic results with our database, leave-one-out method is used throughout this work. That is, a fruit for the database is tested by excluding the reference images of that fruit from the training set.

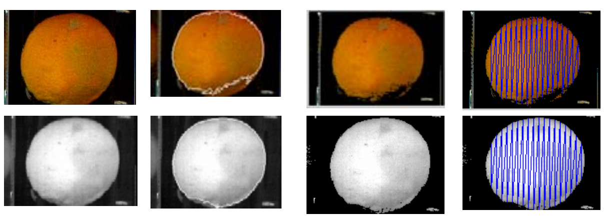


Fig. 4. Segmentation step for region-of-interest (roi) extraction. Left to right: original, contour of roi over fruit, roi

E. Inspection of fruit surface

The analysis of the complete surface of each fruit in each image gives redundant information, because there are over lapping regions of surface in consecutive images. This also would lead to an enormous consumption of time. Because the fruit rotates while translating below the camera, it is possible to create a new image, composed by the central window of each of the images of the same fruit. As shown in Fig. 1 each of these windows is called a ‘strip’ and is properly positioned in the image because we know both the geometrical features previously explained and the position of the cups. The width of the strips depends on the velocity of the cups. The resultant image obtained by joining the strips represents practically the complete surface of the fruit.

Before the on-line processing of this image, the system has to be manually trained. An operator selects different windows representing the pre-established classes (background, primary and secondary color, sound skin and blemish) on images of fruit. The independent variables are the grey levels in the four RGBI bands, although other possible combinations of 3 bands have been tested: only visible (RGB) and RGBI would be considered. The fruit under inspection forms a image strip is scanned. At the end of the scanning, the blemishes are segmented from the rest of the image and their boundaries extracted. During this process, the number of pixels belonging to blemish and fruit classification are counted. The algorithms used in system-A & B would be used for describing the geometric characteristics of the fruit for describing the blemishes. Features would be used to describe the size and distribution of blemishes in the image.

F. Definition of Region-of-interest (roi)

In order to define the region-of-interest (roi) that encloses the fruit area to be inspected, we need to successively perform background removal processes and find the edges of the fruit. Eventually, resultant roi defines the zone of inspection for each fruit.

G. Detection of all blobs for blemishes, stem and navel

The mechanical separation of fruit on the automatic sorter is not always perfect; mainly when a high through output is required. Due to bad positioning of the fruit in the cups, the boundary extractor algorithm can be hard to implement, so a boundary in the image can belong to two or more fruit. This cohesion is rare and if happens there are the three possibilities:

- When large fruit touches the adjacent fruit,
- When two or more small fruit travel in the same cup, or
- When fruit travel between two fruits that are correctly positioned in their cups.

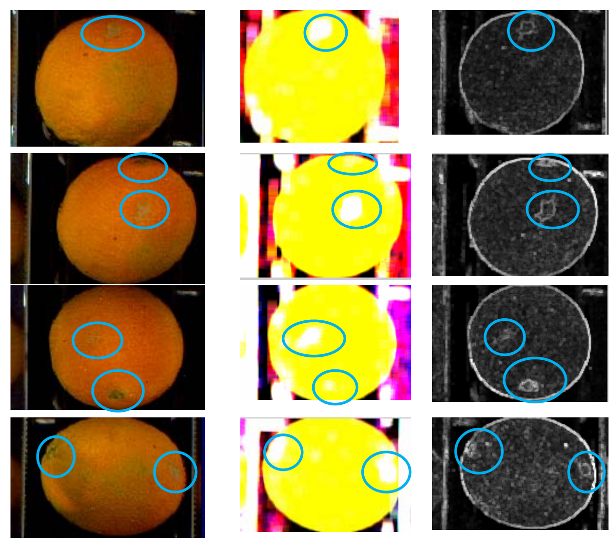


Fig. 5. The figures from left to right indicate the fruit with blemish, stem and navel blobs, and then trace region algorithm detects the region boundaries for these blobs. And finally the binary image is used to cross validate the boundaries of these blobs in a binary image.

In any of the three cases, a simple calculation of size from the previous boundaries will lead to an over-estimation of fruit size. A trace region algorithm [26] is used to distinguish between the three cases, trace region boundaries algorithm detects the positioning, and separating the individual fruit in the image. This algorithm first converts the fruit image to a binary image, and then traces of the exterior boundaries of objects, as well as boundaries of holes inside these fruit objects are identified in the binary image. Then the trace region algorithm will descend into the outermost objects (parents) and traces their children (objects completely enclosed by the parents) this is done to isolate the irregularities if the fruit and blemish does not have a clear separation due to curvature of the fruit at the edges and shades. The fruit image which is first converted to a binary image where nonzero pixels belong to an object and 0 pixels constitute the background. This is calculated using an 8-connected nearest neighborhood algorithm where the fruit image is read in and threshold of intensity image is generated and we get the two labeled objects now it will be referred as the possible blobs for blemish, stem or navel, we now display a binary image and overlay the region boundaries on the image. We get the matrix of region number (based on the label matrix) next to every boundary. The following Fig. 5 and Fig. 6 illustrate these steps and its individual components.

H. Texture analysis

Texture analysis is implemented to characterize the regions in an image by their texture content. Texture analysis attempts to quantify intuitive qualities such as rough, smooth, silky, or bumpy as a function of the spatial variation in pixel intensities. In this sense, the roughness or bumpiness refers to variations in the intensity values i.e. gray levels. Hence we use the texture analysis to find the texture boundaries, called texture segmentation.

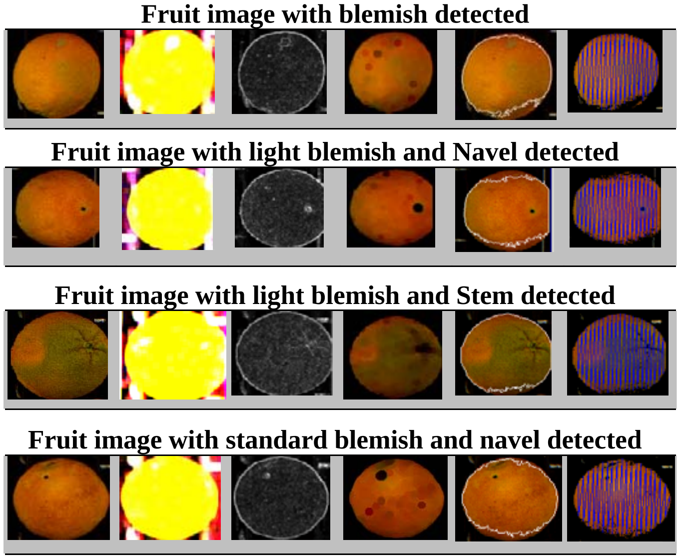


Fig. 6. System-A individual components mapped to identify and segregate blemish, navel and stem-end.

Texture analysis is also helpful when objects in an image are more characterized by their texture than by intensity, and traditional thresholding techniques cannot be used effectively on fruits such as a citrus. Texture analysis is done by filtering an image using standard statistical measures such as calculating the local range of an image, calculating the local standard deviation of an image; and then calculating the local entropy a statistical measure of randomness of a grayscale image.

The Fig. 6 illustrates these steps. These statistics can characterize the texture of an image because they provide information about the local variability of the intensity values of pixels in an image. For example, in areas with smooth texture, the range of values in the neighborhood around a pixel will be a small value; in areas of rough texture, the range will be larger. Similarly, calculating the standard deviation of pixels in a neighborhood can indicate the degree of variability of pixel values in that region i.e. neighborhood around the pixel of interest which calculates the statistic for that neighborhood, and use that value as the value of the pixel of interest in the output image. The Fig. 6 shows these steps.

I. Feature Extraction

Two different feature extraction steps are used in this approach. In System-A, after the process of combined entropy, color-transform, range filter (image erode and dilate) the extracted blobs candidates pixels are recorded for test. Then these each sample (pixel) is represented by its intensity, average and standard deviation of intensities, hue and saturation values of fruit skin. As there are four components for each fruit, we can extract a total of five features for each sample filter. However, our initial observations on the filter images showed that the infrared filter image, in which stem-end/navel regions were more prominent in terms of its intensity and defected areas were more suppressed, was the best within the four. So, both the features of infra-red and other filter were used for System-A, i.e. 5 features per filter per blob sample. It should also be noted that inclusion of features of the other filter-bands were not tested and would be a good volunteer for future research visually. In System-B, The features shown in table 1are extracted from the bounding-box of the blobs and the fruit object. The following section will explain the statistical features extraction.

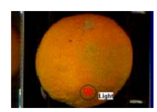
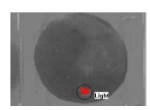
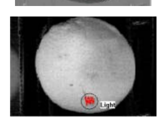
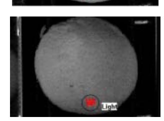
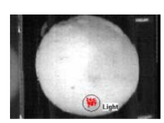
Orange fruit example	View	Color component type
		Color
		Hue/X component
• Color model HSI/XYL • The picture has top view and the mirror view		Saturation/Y-Component
		I-Intensity / Lightness component
		Near-Infrared-Component

Fig. 7. Fruit image of visible and infrared image and split components of the color model HSI/XYL.

J. Statistical Features

Statistical features are selected for the Fruit object and for all the blobs - blemishes, stem and navel. These candidate objects are referred to as blobs. The statistical features are based on the HSI color model (RGB converted) and near infrared image data. These features are basically from Hue, Saturation, intensity and near-infrared component values. Each component has minimum, maximum, range, arithmetic mean, range and standard deviation of the region-of-interest as the object (Parent) and for each blob which is the blobs for blemishes; stem or navel similar features are calculated as statistical features. Region-of-interest mentioned here can refer to the object itself (fruit) or the bounding-box of the object-blobs (blobs). Furthermore, different filtering methods are applied on the bounding-box of the object to remove noise while preserving discriminative information.

TABLE I
FEATURES FOR BLOBS (BLEMISH, STEM, & NAVEL) AND FRUIT OBJECT.

The Blobs or blob feature inputs are as follows:-	**The object feature inputs are as follows:-**
blob_x(hue)_min	object_x(hue)_min
blob_x(hue)_max	object_x(hue)_max
blob_x(hue)_range	object_x(hue)_range
blob_x(hue)_mean	object_x(hue)_mean
blob_x(hue)_std_dev	object_x(hue)_std_dev
blob_y(sat)_min	object_y(sat)_min
blob_y(sat)_max	object_y(sat)_max
blob_y(sat)_range	object_y(sat)_range
blob_y(sat)_mean	object_y(sat)_mean
blob_y(sat)_std_dev	object_y(sat)_std_dev
blob_int_min	object_int_min
blob_int_max	object_int_max
blob_int_range	object_int_range
blob_int_mean	object_int_mean
blob_int_std_dev	object_int_std_dev
blob_nir_min	object_nir_min
blob_nir_max	object_nir_max
blob_nir_range	object_nir_range
blob_nir_mean	object_nir_mean
blob_nir_std_dev	object_nir_std_dev

Each fruit has six visible images called the 'strip' coverin the whole fruit along with similar six near infrared images o the same fruit. The neural nets blemish detection algorithn with back propagation uses both with visible and NIR input The entire training and thruthing information file wer generated manually. The color models or the color transform type used were RGB, Hue-Saturation-Intensity and X-Y Lightness model. Separate functions were coded in matlab fo conversion. The picture view functions that are supported b the CVS (Color vision systems) systems were (1) Top viev parameters (pixel indent 6/8/10 pixels, rows to skip) and (2 mirror view parameters (pixel indent 6/8/10 pixels, rows to ski from the edge of the fruit object).

The Fig. 1 refers to the sample fruit captured in six different views of the same fruit. Similar six set of near infrared images are also shown in the same figure. Fig. 2: refers to images of fruit with original RGB and marked segmentation of Blemish (Defect), Navel and stem-end which form the blobs for each images of a single fruit. The training data contains the blobs pixel (x, y) location, blobs classification, blobs pocket identifier, and grouping called blob identifier. This information is required to extract the feature vector needed for modeling the neural network input data sets. The Fig. 5 and 6 shows the sample blobs data points for each detected blemish image and these data points form a blob. The blemish pixel RGB information from the corresponding blobs was extracted and the individual components i.e. Hue, Saturation and Intensity are shown in the Fig. 7 for visible and infrared images. The red spots are a sample blob pixel points in the images.

The blob feature classification is done in stage two. To become a feature the intensity, X value, Y value and NIR value must deviate from the fruit object profile. The pixels that deviate from the profile are converted to blobs. The green circle encloses all the pixel points that form the blob or blobs. Statistics features are then generated for these blobs in visible and NIR.

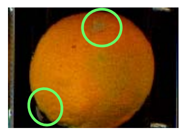
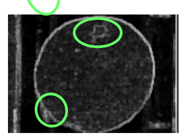
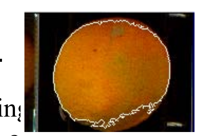
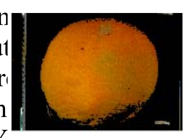
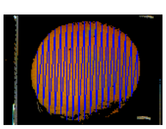
View	**Feature information**
  	Blobs
	Contour of Object information
	Profile of the fruit object
	The features of the object are calculated with an offset of 6 pixel points from the edge of the contour.

Fig. 8. Object and blob or blobs features information for blemish classification

K. Feature Selection

In System-B, there are 40 selected features that are used by the classifier. This feature set contains relevant ones and the nearest neighbor classifier's accuracy is increased as more and more relevant features are introduced. Hence forward selection method avoid irrelevancy and a sub-optimal solution with a sub-set of relevant features are used. [27]

This feature selection method initially starts with an empty set of features and adds one set of feature at a time that has the lowest classification error until a stopping criterion is met. So, forward selection method is also used in the following tests. [27]

L. Neural network for blemish classification

L.1 K-Nearest Neighbor

The classifier used in System-A is a k-nearest neighbor classifier with Euclidean distance metric. Different k values (neighbor number to check) are tested.

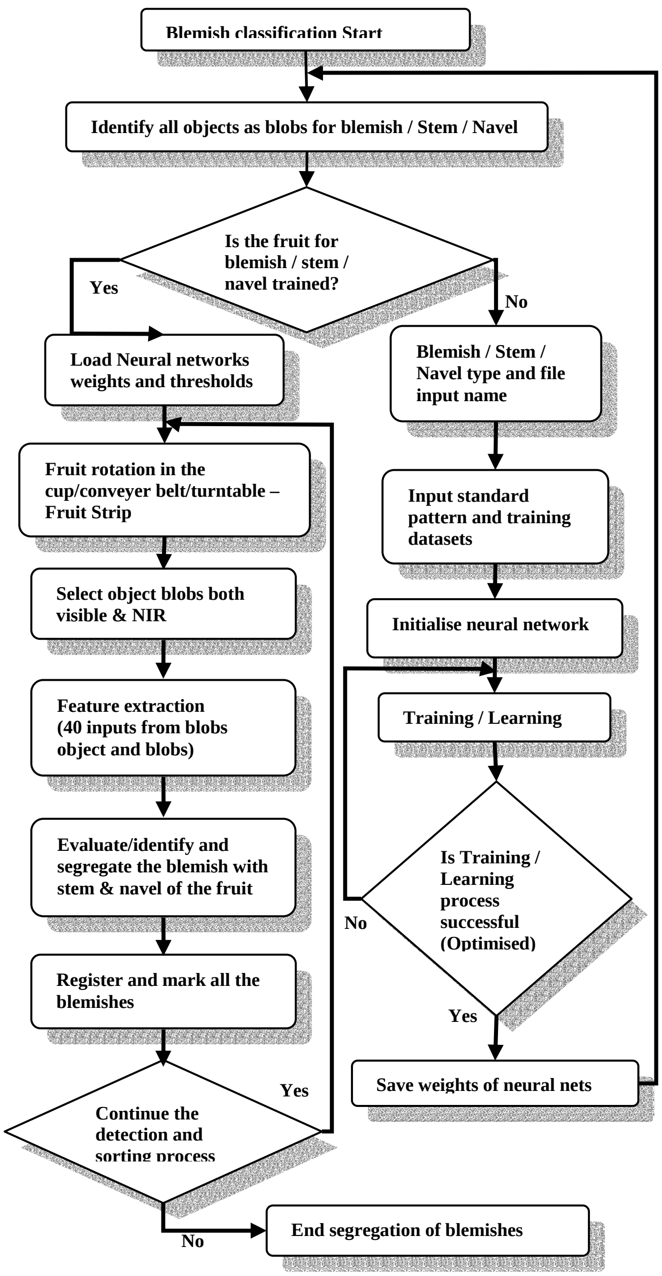


Fig. 9. Flowchart diagram for blemish detection and neural network with training/learning component

L.2 Artificial Neural Network

A two-layer feed forward preceptron is used in System-B for fruit skin classification. It is a feed-forward, error back propagated network with adaptive learning rate. It has 40 inputs, 2 hidden and 4 output nodes. It uses cross-validation technique (3/4 of the training data for training and 1/4 of it for validation) during the training step and evaluates performance on test data during the testing. Even though we have 50-50 % data sets of 1400 samples of training and thruthing data available, we decided to use 75% and 25% of training for the current test results. Several models of artificial neural network were tested, however the one we selected is with 2 hidden layers is used to mimic the behavior of the real-time model that is available at the color vision systems, This has been found to perform better than others visually. Also to keep it simple and make the real-time speed requirements.

III. RESULTS

As the system is designed to work automatically, all the objects found by System-A are introduced to System-B for end decision. So, accuracy of System-A highly effects the difficulty of the problem for System-B. Especially if the blobs objects found by System-A are actually a part of a bigger object, which will be the case if the object is on the edge of region-of-interest. The following Table-2 describes the results. Images of 1400 fruits of the database (398 of them have stem-end or navel regions) are introduced to System-A. In order to speed-up the tests, the blobs objects (output of System-A) are saved and manually classified as 'defect' or 'stem-end/navel' to serve for evaluation of the results of System-B. So, from the images of 1400 fruits System-A found 1062 blobs /candidates objects, where 621 of them were actual defects and 441 of them were actual stem-end or navel. Note that System-A did not find any healthy pixel regions as blobs.

TABLE II
THE FRUIT DATABASE IMAGES OF 1400 WERE TESTED (ACTUAL BLEMISH: 621; ACTUAL STEM OR NAVEL ENDS: 441) WERE CLASSIFIED AS POSITIVE AND NEGATIVE BY THE NEURAL NETS. TP = TRUE POSITIVE; FP = FALSE POSITIVE. A TOTAL OF 61 WERE TESTED AS FALSE POSITIVE FOR STEM/NAVEL AND 70 FOR BLEMISH.

System-A: Fruit Database images total 1400			
Total Blobs / candidate objects		Graded by System-A	
		Actual defects	Actual stem or navel ends
1062		621	441
System-B: Fruit Database images total 1400			
Graded by System-B			
		Stem / Navel	Blemish / Defects
		S / N	B / D
As stem/ navel end	S / N	TP = 362	FP = 61
As blemish / defect	B / D	FP = 70	TP = 497
Percentage error	% error	18 %	20 %
Overall Percentage error		19 %	

Some of the blobs objects found by System-A can be observed in Fig. 5, where the objects are contoured in black and displayed with the fruit image. As observed in the left-top image, System-A can localize parts of a defect as blobs objects. And also region-of-interest extraction can lead to deformed objects, like in the middle-center and middle-right images where not all but a part of stem-end objects were localized. Despite all these disadvantages of System-A, our observations showed that it did not miss any of the stem-end or navel regions existing within the images of 1400 fruits.

Feature selection covered all the combination space, i.e. algorithm started from best – discriminating feature, recursively added the next best feature until all the features are included. An a optimal selection of about 40 features selected and including further features beyond a limit (40 features) degrade the performance of the classifier, which explains the existence of relevant features only. All the feature extraction algorithms used in this work are implemented in matlab language, that's why presenting computation times of these algorithms here will not be realistic. Instead, an order of computation times can be helpful. Roughly speaking, extractions of statistical features are computationally very efficient. These feature extraction algorithms are currently being implemented in ‘forte language’, which will permit us to make a more healthy decision on which/how many features to select with a compromise in performance and computation time.

IV CONCLUSION

This paper has presented a cascaded system for localizing potential stem-end, navel regions and discriminating them from blemish or defects. The research presented in this paper is aimed directly for use in a commercial machine with stringent real-time requirements. Any envisaged solution should be cost-effective, robust and simple and suited for implementation in parallel hardware such as used in Color vision systems. The system uses mapping of multiple layers of techniques in parallel. These techniques are entropy, color transforms, thresholding, filtering algorithm for localizing candidates for stem-end, navel regions called blobs and using k-nearest neighbor classifier for cleansing redundant blobs not suitable as candidates, and finally employ artificial neural networks to discriminate them from possible blemish/defects. Appropriate use of color model to suit the application of blemish detection such as HSL/XYL is used to suite the color texture of the citrus orange fruit. Different statistical features are extracted for the discrimination part, In order to find the best discriminating combination of features, appropriate pre-selected forward selection method was used in finalizing the feature set. The system proposed here can localize stem-end and navel regions on orange by machine vision with an accuracy of 81%, which is encouraging due to the complexity of the problem. Different methods of thresholding and filtering along with artificial neural network is used to find and segregate blobs with stem-end and navel objects which has performed quite accurately and sufficiently, however this approach has been a simple approach (e.g. statistical ones). However use of features such as textural which is used in system-A for finding blobs can also provide us with sufficient information to form a good feature. So, statistical features along with textural features need to be studied and researched together to improve the performance of classification; however the speed of classification should not be compromised.

In system-A K-nearest neighbor classifier is simple but effective in certain tasks. However, effect of other classifiers (like fuzzy, support vector machines, radial-base function…..) on the accuracy of the system and comparison for this specific task would be tested in future work. The systems introduced in this paper will be a part of an automated fruit sorting system, so computation time is another constraint to be tartan in future.

V ACKNOWLEDGEMENTS

We would like to thank the Color vision systems for providing the fruit image database, Charles Esson and Gary from the Color vision systems to provide technical support for this research work. This research project is supported by color vision systems Melbourne Australia.

VI REFERENCE

Papers from Conference Proceedings (Published):

[1] Yang Q., “Apple stem and calyx identification with machine vision,” J. Agr. Eng. Res., pp. 229-236, 1996.

[2] Penman D.W., “Determination of stem and calyx location on apples using automatic visual inspection,” Comp. Electro. Agriculture, pp. 7-18, 2001.

[3] Li Q. and et al., “Computer vision based system for apple surface defect detection,” Comp. Electro. Agriculture, pp. 215-223, 2002.

[4] Leemans V. and M.-F. Destain, “A real-time grading method of apples based on features extracted from defects,” J. Food Eng., pp. 83-89, 2004.

[5] Wen Z. and Y. Tao, “Building a rule-based machine vision system for defect inspection on apple sorting and packing lines,” Expert Sys. Appl., pp. 307-313, 1999.

Technical Reports:

[6] Washington State University Tree Fruit Research and Extension Center, Market diseases of apples, pears, and quinces. [Online] http://postharvest.tfrec.wsu.edu/market diseases.html, visited: Sep 2007.

Papers from Conference Proceedings (Published):

[7] T.F. Schatzki, R.P. Haff, R. Young, I. Can, L.-C. Le, and N. Toyofuku, Defect detection in apples by means of x-ray imaging, Transactions of ASAE 40 (1997), no. 5, 1407–1415.

[8] M.A. Shahin, E.W. Tollner, M.D. Evans, and H.R. Arabnia, Watercore features for sorting red delicious apples: a statistical approach, Transactions of ASAE 42 (1999), no. 6, 1889–1896.

[9] J.J. Gonzalez, R.C. Valle, S. Bobroff, W.V. Biasi, E.J. Mitcham, and M.J. McCarthy, Detection and monitoring of internal browning development in ‘fuji’ apples using mri, Postharvest Biology and Technology 22 (2001), 179–188.

[10] L. Varith, G.M. Hyde, A.L. Baritelle, J.K. Fellman, and T. Sattabongkot, Noncontact bruise detection in apples by thermal imaging, Innovative Food Science and Emerging Technologies 4 (2003), no. 2, 211–218.

[11] C.J. Clark, V.A. McGlone, and R.B. Jordan, Detection of brownheart in ‘braeburn’ apple by transmission nir spectroscopy, Postharvest Biology and Technology 28 (2003), 87–96.

[12] Real-time defect detection in fruit part ii: An algorithm and performance of a prototype system, Transactions of ASAE 39 (1996), no. 6, 2309–2317.

[13] B.L. Upchurch, H.A. Affeldt, W.R. Hruschka, and J.A. Throop, Optical detection of bruises and early frost damage on apples, Transactions of ASAE 34 (1991), no. 3, 1004–1009, corrected.

[14] B.L. Upchurch, J.A. Throop, and D.J. Aneshansley, Detecting internal breakdown in apples using interactance measurements, Postharvest Biology and Biotechnology 10 (1997), 15–19.

[15] C J Du and D.-W. Sun, Recent developments in the applications of image processing techniques for food quality evaluation, Trends in Food Science & Technology 15 (2004), 230–249.

[16] P.M. Mehl, K. Chao, M. Kim, and Y.R. Chen, Detection of defects on selected apple cultivars using hyperspectral and multispectral image analysis, Applied Engineering in Agriculture 18 (2002), no. 2, 219–226.

[17] Cheng-Jin Du, Da-Wen Sun, Learning techniques used in computer vision for food quality evaluation: a review, Journal of Food Engineering 72 (2006) 39-55.
[18] Q. Li, M. Wang, and W. Gu, Computer vision based system for apple surface defect detection, Computers and Electronics in Agriculture 36 (2002), no. 2-3, 215–223.
[19] J.A. Throop, D.J. Aneshansley, W.C. Anger, and D.L. Peterson, Quality evaluation of apples based on surface defects: development of an automated inspection system, Postharvest Biology and Technology 36 (2005), 281–290.
[20] A.K. Jain, R.P.W. Duin, and J. Mao, Statistical pattern recognition: A review, IEEE Trans. Pattern Analysis Machine Intelligence 22 (2003), 4–37.

Books:

[21] Haim Levkowitz Kluwer (2003). Colour Theory and Modeling for computer graphics, Visualization, and Multimedia Applications. Academic Publishers Boston. 2003.
[22] Charles Esson, Kevin Azzopardi Colour vision Systems; A XY Coordination Colour representation suitable for image analysis http://www.cvs.cx,Version 1.0 9 Nov 2007
[23] Allen Hanbury and Jean Serra(2005). A 3D-polar Coordinate Colour Representation Suitable for Image Analysis.. Pattern Recognition and Image Processing Group, Institute of Computer Aider Automation, Vienna University of Technology.
[24] CVS (Color vision systems), Bacchus Marsh, Victoria, Melbourne.

Papers from Conference Proceedings (Published):

[25] W. Niblack, An Introduction to Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1986.
[26] Gang Xu, Hideki Kondo (1999) A Region-Based stereo algorithm, Vision and Robotics pp.1661 1666. 1999.
[27] KZ. Mao, “Fast orthogonal forward selection algorithm for features subset selection”, IEEE Trans. NN., vol. 13, pp.1218-1224, September 2002