AN APPROACH FOR RECOGNIZING STEM-END/CALYX REGIONS IN APPLE QUALITY SORTING

[1]*Devrim Unay, Bernard Gosselin*

[1]unay@tcts.fpms.ac.be
TCTS Lab., Faculté Polytechnique de Mons,
Multitel Building, Avenue Copernic 1, Parc Initialis, B7000, Mons, Belgium

ABSTRACT

In this paper we introduce a cascaded-classifier approach to localize stem-ends and calyxes of 'Jonagold' apples. First classifier (artificial neural network) extracts candidate objects, whereas the second one (nearest neighbor) discriminates stem-ends and calyxes from others. Overall system is tested by 616 fruits from which first classifier found 414 candidate objects. Several features are extracted from these objects and these features are then selected by forward selection method. With these selected features the second classifier reached to 80% recognition rate. Classification errors of each feature is also introduced for comparison.

1. INTRODUCTION

Machine vision-based fruit grading is an important and necessary task for fruit marketing. In this area discrimination of stem-ends or calyxes from defects, which can lead to incorrect grading, is still an important and open problem being searched.

Different approaches to this problem have been introduced, some of which are mechanical. However mechanical orientation of the fruit is not applicable to our system, so the mechanical approaches are not considered here.

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.

For this specific problem we propose a cascaded-system to discriminate between stem-ends/calyxes and defects. In the first step, candidate stem-ends/calyxes are found by pixel-based search of fruit skin with artificial neural networks. Then these candidates are further analyzed by a nearest neighbor classifier to remove false stem-ends/calyxes.

2. METHODS

2.1. Image Database

Database consists of images of 819 'Jonagold' apples acquired by a ccd-camera and four bandpass filters(RGB + Infra-Red. centered at 450, 500, 750 and 800 nm with bandwidths of 80, 40, 80 and 50 nm, respectively) from one-view in a diffusely illuminated environment at the Mechanics and Construction Department of Gembloux Agricultural University of Belgium (Figure 1) [6]. Each filter image has a dimension of 430x560 pixels with 8 bits-per-pixel resolution. 280 of the images contain totally healthy skin, whereas 293 of them include stem-ends or calyxes and the rest have defects of various size and kind.

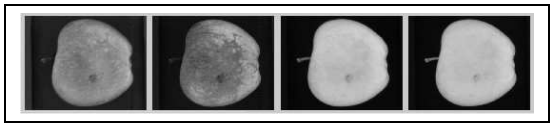

Figure 1: Images of an apple. Left to right: blue, green, red, and infra-red filters.

In order to serve as a training database, defected and stem-end/calyx regions within the database were manually segmented by O. Kleynen and D. Unay, respectively (Figure 2). They will be referred as reference images from now on in this paper.

2.2. System Architecture

Overall system for stem-end/calyx identification proposed here consists of a segmentation step and two cascaded systems (Figure 3):

- *Segmentation*: Extracts region-of-interest of fruits.

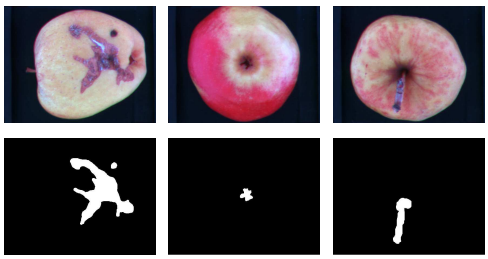

Figure 2: *Examples of original rgb images (top) and manual segmentations (bottom). Left to right: defect, calyx, stem-end.*

- *Sys-A*: Finds potential stem-end/calyx objects within the region-of-interest of a fruit.
- *Sys-B*: Decides if an object found by *Sys-A* is really a stem-end/calyx object or not.

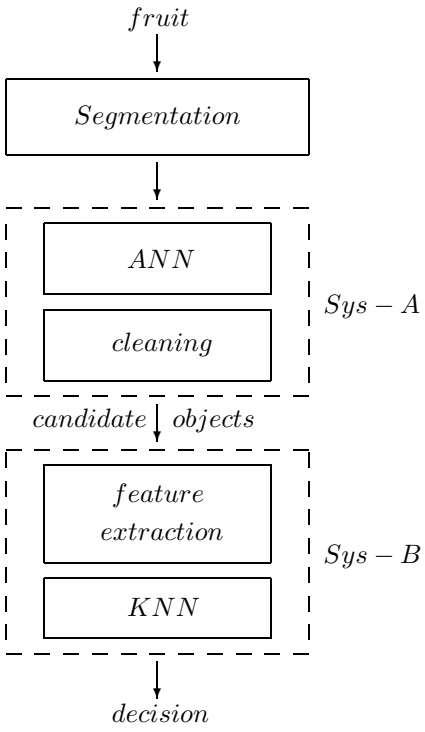

Figure 3: System architecture

In *Segmentation* step, fruit area is extracted from the background by thresholding [7]. Then morphological filling is applied to remove holes caused by previous step and lastly the image is eroded by 80x80 mask giving out the region-of-interest for *Sys-A*. Erosion step improves the accuracy of *Sys-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. Figure 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 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.

Sys-A does a pixel-based search of the fruit skin for potential stem-end/calyx objects by an artificial neural network, which is trained by 'stem-end/calyx' and 'healthy' pixels extracted from reference images. The objects smaller than 100 pixels within the candidates found by *Sys-A* are eliminated (cleaning), because a stem-end or calyx cannot be very small, a priori.

Sys-B extracts features from the candidate objects provided by *Sys-A* and classifies the object as 'stem-end/calyx' or 'defect' by a nearest neighbor classifier, which is trained by the objects of reference images.

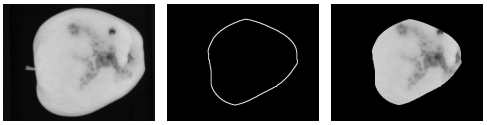

Figure 4: *An example of segmentation step: region-of-interest (roi) extraction. Left to right: original, contour of roi, roi.*

In order to eliminate the intersection of training and test sets, and to have realistic results with our small 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.

2.3. Feature Extraction

Two different feature extraction steps are used in this approach. In *Sys-A*, each sample (pixel) is represented by its intensity value plus average and standard deviation of intensities of fruit skin. As there are four filters for each fruit, we can extract 12 features for each sample. However, our initial observations on the filter images showed that the infra-red filter image, in which stem-end/calyx regions were more visible and defected areas were more suppressed, was the best within the four. So, only the features of infra-red filter were used for *Sys-A*, i.e. 3 features per sample. It should also be noted that inclusion of features of the other filter-bands did not improve the results of *Sys-A* visually.

In *Sys-B*, following features are extracted from the

bounding-box of the candidate object:

2.3.1. Statistical Features

Arithmetic mean and ranges of intensity values of the region-of-interest are used as statistical features. Region-of-interest mentioned here can be the object itself (f_1) or the bounding-box of the object (f_2). Furthermore, different filtering methods are applied on the bounding-box of the object to remove noise while preserving discriminative information. These filtering methods are anisotropic diffusion [8] (f_3), gauss filtering with 5x5-mask (f_4), median filtering with 3x3-mask (f_5), order-statistic filtering with 3x3-mask by maximum - favored (f_6) and minimum - favored (f_7) approaches, and wiener filtering with 3x3-mask (f_8).

2.3.2. Invariant Moments

Invariant moments of Hu [9], due to their descriptive characteristics invariant to rotation, translation and scaling, can be good discriminative factors for our task. Equations 1- 3 show the formulas of first three invariant moments where η_{xy} is the normalized central moment. So, the first seven invariant moments of Hu ($\phi_1 - \phi_7 : f_9 - f_{15}$) are extracted from the bounding-box of the object.

$$\phi_1 = \eta_{20} + \eta_{02} \tag{1}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{2}$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{3}$$

2.3.3. Textural Features

Textural features from Gray-Level Cooccurance Matrices (GLCM) of Haralick [10] have been widely used in textural classifications. These matrices are computed for various angular relationships and distances between neighboring resolution cell pairs on the image. Then from GLCM, Haralick's features are extracted to reveal textural information. Therefore, arithmetic mean and range values of eleven textural features from GLCM (angles=0, 45, 90, 135 and distance=1 pixel) are extracted ($f_{16} - f_{26}$: angular second moment, contrast, correlation, sum of squares, inverse difference moment, sum average, sum variance, sum entropy, entropy, difference variance, and difference entropy, respectively).

$$f_{16} = \sum_i \sum_j (p(i,j)^2) \tag{4}$$

$$f_{17} = \sum_n n^2 (\sum_{\substack{i \\ |i-j|=n}} \sum_j p(i,j)) \tag{5}$$

$$f_{18} = \frac{\sum_i \sum_j p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{6}$$

Equations 4-6 show the formulas of angular second moment, contrast and correlation features, respectively, where p refers to the GLCM and μ, σ refer to arithmetic means and standard deviations of column or row-wise summation of p.

2.3.4. Fractal Dimensions

Fractal dimension values of Kaplan [11] can also be descriptive for texture classification and segmentation, so three fractal values at dimensions 0, 1, and 2 ($f_{27} - f_{29}$) are extracted from the bounding-box of candidate objects. Equations 7 and 8 shows the general formula of the fractal values, where s defines the fractal dimension, N is the total number of pixels in the image and Δ is the difference of two pixel values.

$$H_s = \frac{1}{2} log(\frac{g_{s+1}^x(i,j) + g_{s+1}^y(i,j)}{g_s^x(i,j) + g_s^y(i,j)}) \tag{7}$$

$$g_s^{x,y} = \frac{1}{N * (N - 2^s)} \sum_i \sum_i \Delta_s^{x,y}(i,j)^2 \tag{8}$$

2.3.5. Wavelet Features

Wavelet decomposition, which is widely used in signal compression and noise removal, can be very informative for our task due to its scale-space analysis. So, wavelet decomposition is applied on the bounding-box of the object with five Daubechies wavelets (db1-db5) at 2 decomposition levels [12]. Arithmetic mean and range values of wavelet coefficients are used as features ($f_{30} - f_{69}$).

It should be noted that after extraction and selection steps, features are normalized to have mean of zero and standard deviation of one before being introduced to the classifiers.

2.4. Feature Selection

In *Sys-B*, there are 69 possible features that can be used by the classifier. However this feature set contains irrelevant features as well as relevant ones and a nearest neighbor classifier's accuracy is exponentially degraded as more irrelevant features are introduced. Forward selection method can avoid this and find a sub-optimal solution with a sub-set of features. This feature selection method initially starts with an empty set of features and adds one 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.

2.5. Classifiers

2.5.1. Artificial Neural Network

A two-layer artificial neural network is used in *Sys-A* for fruit skin classification. It is a feed-forward, error back-propagated network with adaptive learning rate. It has 3 input, 10 hidden and 2 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. Several models of artificial neural network is tested, however the one with 10 hidden nodes is found to perform better than others visually.

2.5.2. K-Nearest Neighbor

The classifier used in *Sys-B* is a k-nearest neighbor classifier with Euclidean distance metric. Different k values (neighbor number to check) are tested. Actually artificial neural networks are also tried in *Sys-B*, however the network was unable to converge due to limited number of samples for training.

3. RESULTS

As the system is designed to work automatically, all the objects found by *Sys-A* are introduced to *Sys-B* for end-decision. So, accuracy of *Sys-A* highly effects the difficulty of the problem for *Sys-B*. Especially if the candidate objects found by *Sys-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.

Images of 616 fruits of the database (140 of them have stem-end or calyx regions) are introduced to *Sys-A*. In order to speed-up the tests, the candidate objects (output of *Sys-A*) are saved and manually classified as 'defect' or 'stem-end/calyx' to serve for evaluation of the results of *Sys-B*. So, from the images of 616 fruits *Sys-A* found 414 candidate objects, where 158 of them were actual defects and 256 of them were actual stem-end/calyxes. Note that *Sys-A* did not find any healthy region as a candidate.

Some of the candidate objects found by *Sys-A* can be observed in Figure 5, where the objects are contoured in black and displayed with the fruit image. As observed in the left-top image, *Sys-A* can localize parts of a defect as candidate 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 *Sys-A*, our observations showed that it did not miss any of the stem-end or calyx regions existing within the images of 616 fruits.

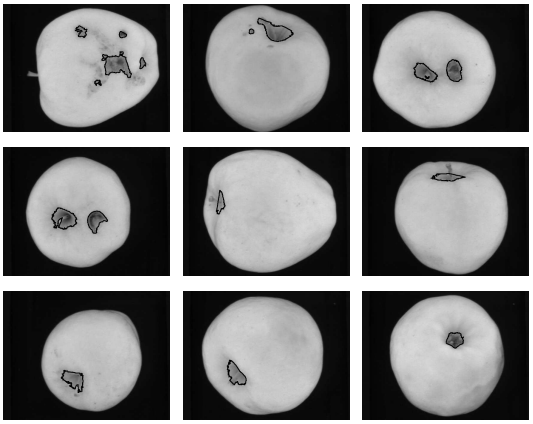

Figure 5: *Contours of candidate objects found by Sys-A.*

Accuracy of *Sys-B* is tested using *one feature at a time* by the nearest neighbor classifier with varying neighbor numbers (1-185). Table 1 shows the best classifying features from each feature group with related error percentages. Error values show the percentage of misclassifications over all the candidate objects.

feature group	feature	error
statistical	f_6, f_8	24.9
moments	f_{15}	24.9
textural	f_{16}	24.9
fractals	f_{28}	26.6
wavelets	f_{46}	23.9

Table 1: Best-discriminating features

In Table 1, the lowest classification error (23.9%) is observed with wavelet feature f_{46} (1st-level approximate coefficients of Daubechies-3), which is actually a low-pass filtered (down-sampled) version of object's bounding-box. This observation shows that wavelets can perform better noise-removal than the classical methods, like wiener filtering, in this specific task. The other features cannot perform error rates less than 25% with introducing only one feature at a time.

Introducing a combination of the above features to the classifier can improve the error rate, but selecting the best subset from these features is not only task specific but also classifier dependent. That is, a subset of features performing well with one classifier is not necessarily the best one for another classifier. Furthermore, feature selection method is also crucial to find the optimum performance of a specific classifier. Possible methods can be forward selection, backward elimination, stepwise selection, genetic approaches,... In this work forward selection method

is used for its implementation simplicity, despite its high possibility to find a sub-optimal solution than the optimal one.
Figure 6 displays the graph of error rates vs number of features selected at each selection step. Classifier used is K-Nearest Neighbor with 25 neighbors, which performed better than those with different neighbors taking into account the compromise between the error rate and the number of features selected. 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.

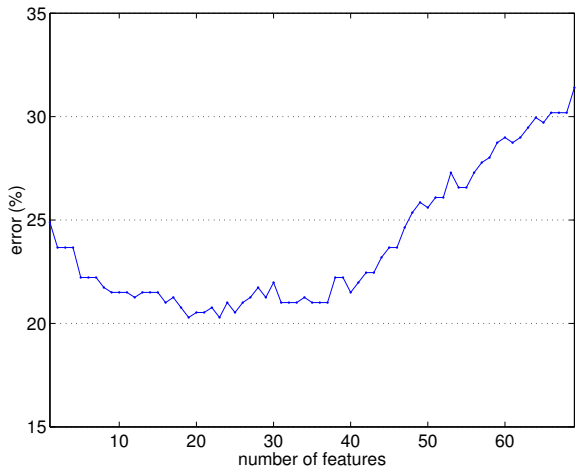

Figure 6: *Classification error of feature selection steps.*

Figure 6 reveals that our system reaches a sub-optimal solution at about 20 features selected and including further features beyond a limit (∼40 features) degrade the performance of the classifier, which explains the existence of irrelevant or redundant features.
Table 2 displays the selected features and confusion matrix of the best case among all the feature selection analysis.

features	$f_1 - f_3, f_6, f_{10} - f_{16}, f_{28}$ $f_{31} - f_{32}, f_{38}, f_{42}, f_{46}, f_{50}, f_{54}$	
	true classes	
graded in	stem-end/calyx	defect
stem-end/calyx	230	58
defect	16	100
error %	10.2	36.7
global error %	20.3	

Table 2: Confusion matrix of best-discriminating feature subset.

The lowest error rate reached by forward selection method is 20.3% with 19 features(Table 2) selected. This value is obviously lower than those of separate features(Table 1). Among the features selected, there are four statistical features ($f_1 - f_3, f_6$), six moments ($f_{10} - f_{15}$), one textural feature (f_{16}: angular second moment), one fractal feature (f_{28}: fractal dimension 2), and several wavelet features ($f_{31} - f_{32}$, f_{38}, f_{42}, f_{46}, f_{50}, f_{54}). The four statistical features selected show that bounding-box of the object and its filtered versions can reveal different discriminative information than the object itself. With six over seven of them selected, invariant moments bring important discriminative information. It should also be noted that wavelet features selected are from approximation coefficients of Daubechies 2,3,4, except the ones from horizontal and vertical detail coefficients of Daubechies 1 (Haar).
Confusion matrix displayed in Table 2 shows that the error rate of defect class is much higher than that of stem-end/calyx class, which is probably due to the highly deformed defect objects found by *Sys-A*. Keeping in mind the variability of this deformation in the candidate objects, 80% recognition rate is acceptable.

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, extraction of fractals, moments and textural features are computationally about 2, 5 and 6 times more expensive than statistical ones, respectively. Comparison of wavelet features are excluded, because they are calculated by Matlab wavelet bank, which is implemented in C language. These feature extraction algorithms are currently being implemented in C, 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.

4. CONCLUSIONS

We introduced a cascaded system for localizing potential stem-end/calyx regions and discriminating them from defects. The system uses artificial neural networks for localizing candidate stem-end/calyx regions and k-nearest neighbor classifier to discriminate them from possible defects. Different features are extracted for the discrimination part, which are evaluated one-by-one by the system and 24% error rate is reached. In order to find the best discriminating combination of features, forward selection method is used, which produced 20% error.
The system proposed here can localize stem-end/calyx regions on apples by machine vision with an accuracy of 80%, which is encouraging due to the complexity of the problem.
Artificial neural network used to find candidate stem-end

/calyx objects performs accurately and sufficiently, however a simpler approach (e.g. statistical ones) may also provide us sufficient results with less computation. So, statistical methods should be searched to replace artificial neural network in *Sys-A*.
Effect of mask-size for erosion in *Segmentation* step and the threshold for removing the very small objects in *Cleaning* step should also be searched for optimum result.
It should be noted that forward selection algorithm may suffer from its greediness by finding a sub-optimal solution rather than the optimal one, because it recursively adds features without removing any previously selected one. In reality, it is not guaranteed that a feature selected in a previous combination step should exist in the final (optimum) subset. For this reason, different methods of feature selection should be tested.
K-nearest neighbor classifier is simple but effective in certain tasks. However, effect of other classifiers (like fuzzy k-nearest neighbor, support vector machines, ...) on the accuracy of the system for this specific task should also be tested.
The system introduced in this paper will be a part of an automated fruit sorting system, so computation time is another constraint to be checked in future.

5. ACKNOWLEDGEMENTS

We would like to thank the providers of the image database, Prof. M.-F. Destain and O. Kleynen from the Mechanics and Construction Department of Gembloux Agricultural University of Belgium. This project is funded by the Ministère de la Régionne wallonne of Belgium with Convention No 9813783.

6. REFERENCES

[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.

[6] O. Kleynen and et al., "Selection of the most efficient wavelength bands for 'jonagold' apple sorting," *Postharvest Bio. and Tech.*, pp. 221–232, 2002.

[7] D. Unay and B. Gosselin, "A quality grading approach for jonagold apples," in *Proc. of the IEEE Benelux Signal Proc. Symp.*, Hilvarenbeek, Apr. 2004, pp. 93–96.

[8] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intel.*, pp. 629–639, 1990.

[9] Hu M.K., "Visual pattern recognition by moment invariants," *IRE Trans. Information Theory*, pp. 179–187, 1962.

[10] Haralick R.M. and et al., "Textural features for image classification," *IEEE Trans. Sys. Man Cyber.*, pp. 610–621, 1973.

[11] Kaplan L.M., "Extended fractal analysis for texture classification and segmetation," *IEEE Trans. Image Proc.*, pp. 1572–1585, 1999.

[12] Daubechies I., "Ten lectures on wavelets," *CBMS-NSF Lecture Notes nr. 61 SIAM*, 1992.