The Measurement of Mealy Bug Population Using Image Processing Techniques

Miti Ruchanurucks and Vutipong Areekul
Department of Electrical Engineering, Faculty of Engineering,
Kasetsart University,50 Paholyothin Road, Jatujak, Bangkok 10903, Thailand.
Tel. +66-2-942-8555 ext 1552, 1526, Fax.: +66-2-942-8555 ext 1550
e-mail : fengmtr@ku.ac.th, fengvpa@ ku.ac.th

Abstract: An experiment on the automatic population measurement of brown Mealy Bugs is first reported in this article. Several image processing techniques are employed along with mapping function and neural network. Though brown Mealy Bugs are difficult to detect because of their camouflage, the experimental results showed that approximately 74% of them were correctly detected.

1. Introduction

In modern agriculture, many engineering fields of technology, such as mechanics, electronics, computer and bioengineering technologies, are applied to increase agricultural productivity. Signal processing is one of those fields that it currently becomes very important in modern agriculture systems. For example, remote sensing that can be used to estimate the productivity in very large area using satellite [1].

Nowadays, insecticides are widely used all around the world. Besides, several kinds of them harmfully affect consumers and the environment. The usage of insecticide quantity is based on a mass of insects, which can be approximated by human inspection. In generally, insecticides have been used more or less than its necessity resulted in toxic residue or insect's resistance. Therefore, in modern agriculture, insecticide quantity must be used precisely and effectively for each time.

Statistics and probability are used to estimate insect population and predict the growth rates of pests. However, it is very difficult and tedious to count small insects by human beings. Thus, signal processing techniques become an important role for insect detection. There are two major approaches for insect detection: mass insect detection and individual insect detection. Several mass insect detection researches were reported using radar signal processing [2-4]. However, only a few individual insect detection researches were exist in literature. In [5], Davies used image processing technique, isotropic masks, to detect insects in grain. In [6], Phopariphat applied statistical pattern recognition and image processing for the measurement of Mealy bug population on plant. However, this method can only detect white Mealy bugs. In this research, the problem in [6] has been extended in order to detect various color of Mealy bug with camouflage.

Mealy bug (*Coccidae Homoptera*) is a small pest which destroy fruit and flowers. Its average full growth length is 2.5 mm. While it is young, its color is brown and it gets tanner as older. Later, white flour is produced in the middle of its body until it becomes totally white. With difficulty of detection, it is hardly to distinguish Mealy bug's features from the background. For example, First, while growing, its size, color, and texture vary. Second, its colors are very similar to the colors of branch and stain. Finally, some Merly bugs attach to a branch, therefore the side view can be seen in the captured images. These problem is very interesting because the recognition method of this paper is not only restricted to insect detection but can be further extend to any pattern recognition problems; i.e. automatic target detection, cell identification, object with camouflage detection etc.

In this paper, Mealy Bugs on leaf of the plant named, *Jatropa Integerina Jacq.*, are used as test input images. Images of Mealy bugs on a leaf are captured by scanner with fixing the resolution, brightness, contrast, and gain. To prove the robustness of algorithm, various types of background leaves are tested, as example: dark-bright leaves, various color leaves, leaves with stains. In the following sections, preprocessing techniques are described in section 2, various detection techniques are discussed in section 3, experimental results are in section 4, and finally conclusions are in section 5.

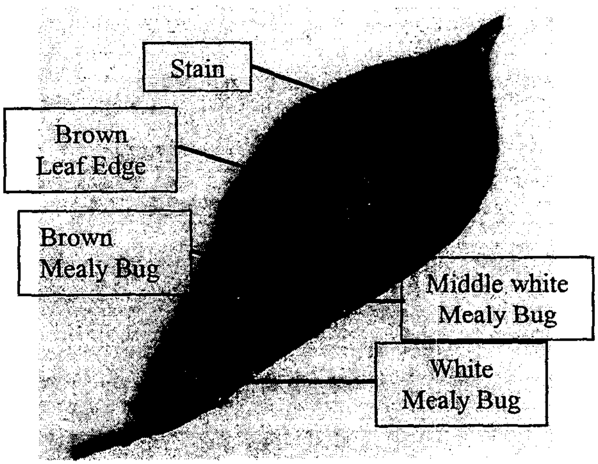
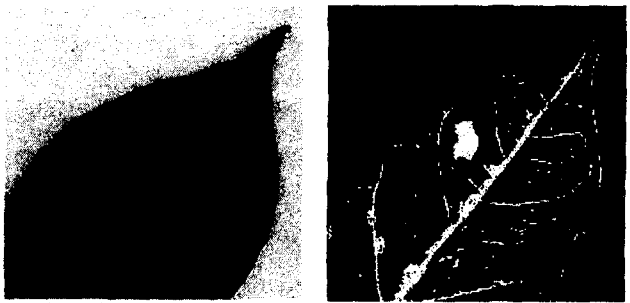


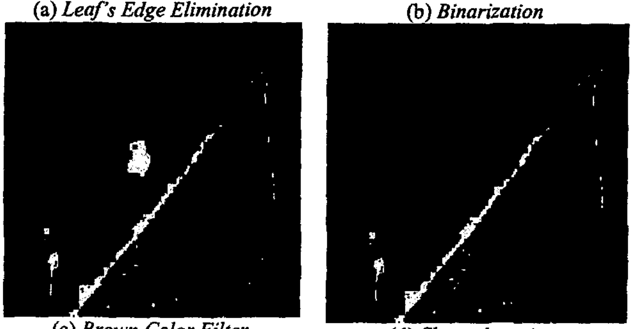
Figure 1. An Example of Input Mealy Bug on Leaf Image

2. Preprocessing Techniques

In order to detect Mealy bugs in an image, the color of Mealy bugs is used to differentiate between bugs and background. However, because of the color similarity between Mealy bugs and surrounding background such as shadow around leaf's edge, branches, and stain, various preprocessing techniques are applied. In order to get rid of shadow around leaf's edge, immediate brightness change detection can be applied since white background and green leaf are obviously distinguished. Next, leaf branches and some noises are eliminated by shape and size in which are larger and longer than Merly bug's shape and size. Finally, area and ratio of perimeter square by area can be used as shape detection to get rid of stain which may have the same color as the insect but different in size and shape.



(a) *Leaf's Edge Elimination* (b) *Binarization*



(c) *Brown Color Filter* (d) *Shape detection*

Figure 2. Preprocessing Results

2.1 Leaf's Edge Elimination

Shadow at some parts of leaf's edge creates brown color which is the same as young Mealy bug's color. To avoid fault detection in later processes, these areas must be eliminated. The algorithm takes advantage of high contrast between white background and a green leaf by detecting high contrast edges. With an experimental threshold value, the boundary of leaf can be determined. Then, "*x*" pixels connected this boundary are set to be background (black pixels) in order to remove them out of the bug detection process. This method can effectively remove shadow of leaf's edge while maintaining some part of bugs in this region. This is because the average diameter of Mealy bug is larger than "*x*"pixels and the result of this process is showed in figure 2 (a).

2.2 Binarization

Since a brown Mealy bug has its brightness different from other part of leaf and background, except branch and stain, Mealy bugs can be extracted from background by brightness thresholding into a binary image where white pixels represents expected bug's area and black pixels represents background. The brightness threshold value is set in a wide range in order to capture all parts of insect. Unfortunately, not only young Mealy bugs and part of middle-white Mealy bugs are extracted, but branches, stain, and other noise's are also combined in the bi-level image as shown in figure 2 (b).

2.3 Brown Color Filter

The 3×3 brown color filter is used as shown in Figure 3. The mask is shifted through the original color picture in aster scan fashion. If color of the center pixel is not in the brown color range, the corresponding pixel in the bi-level image from the previous process is set to be black as well as all eight pixels around. Otherwise, all pixels are left as they were.

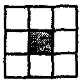


Figure 3. *3×3 brown color filter*

After filtering, the area which is not brown or brown with narrow shape i.e. small branch, are removed. Some minor part of insect may be deleted. However, because of its round shape, only bug's edge is removed resulting in figure 2 (c), Moreover, a group of connected Mealy bugs can be separated by this process.

2.4 Shape Detection

From the fact that a ratio of any perfect circle perimeter square by area always equals to a constant, 4π, as follows,

$$\frac{\rho^2}{A} = 4\pi \tag{1}$$

where ρ = circle perimeter ($2\pi r$), and A = area (πr^2). The other ratio of two-dimensional objects is always higher than 4π, i.e. ecllipse, long and thin shapes (branches). Moreover, the complex shapes, which contain some concave and convex curves, obtain much higher ratio than 4π. Therefore, this ratio, together with area of object, are used as a Mealy bug's shape extraction from brown stain, a group of connected bugs, leaf's branches, or bugs attached to branch.

This processing technique begins following the previous brown color filtered image. The median filter [8] is applied in order to smooth spike noises in the binary image. Then, the perimeter is obtained by edge tracking along the object's boundary (black and white pixels). Later, the object area is obtained by counting the number of pixel inside the object boundary. Both ratio and area threshold are experimentally test and set in order to distinguish an individual Mealy bug from noisy background. If the stain is found, it will be removed by inverting white pixels into black pixels. Then the same algorithm can be applied again. The result from this process is shown in figure 2 (d).

3. Insect Detection

After preprocessing process, the white pixels represent expected position of Mealy bugs. However, brown-color noises still exist in corresponding binary image. If the recognition process is applied, high error detection will be inevitable. Moreover, this recognition will be time-consuming process. Hence, Mapping function [9] is modified in order to apply for searching a single bug. Finally, neural network [10] is applied to recognize undetected bugs.

3.1 Bug Mapping

Mapping function is employed in this process. By using 13×13 mask with weighted coefficients, mapping is done by sliding this mask though the pre-processed binary image. The row *x* and column *y*, pixel position, are mapped into *U* plane by equation (2) inside the mask.

$$U_{xy} = \frac{K^2 - |z|^2}{1+|z|^2} = \frac{K^2 - x^2 - y^2}{1+x^2+y^2} \qquad (2)$$

where U_{xy} is the mapping coefficient, K is a constant referred to an average radius of expected bug in pixels, and $z = x + jy$ is complex number. In this case, the average radius of expected bug is obtained from experimental tests which K is set to 4 pixels.

The U-plane mapping is done inside the 13×13 mask. Then, all U-plane mapping values inside the mask are summed up to S variable, as shown in equation (3).

$$S = \sum_{x=-4}^{8} \sum_{y=-4}^{8} U_{xy} \qquad (3)$$

where S is a bug's shape feature variable.

The idea of mapping function in the mask is to locate the position of a round object such as Mealy bug as shown in figure 4. Assume that a Mealy bug is lied in the oval shape, the black pixel is the center of detection (x = 0 and y = 0). Then the shade area is represented the high values of mapping coefficient, U, which means possibility of an existing Mealy bug inside this mark window.

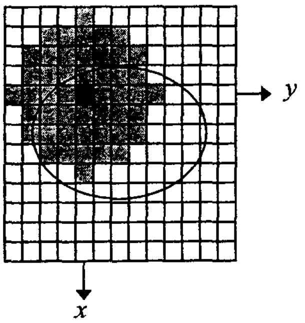


Figure 4. 13x13 mask (x-y plane), shade area represents positive mapping coefficients, oval as a Mealy bug

The relation between radius, |z|, generate by the position (x, y) of a white pixel, and mapping coefficients, U, is plotted in figure 5. This plot shows that the positive value of U represents the existence of Mealy bug in the mask window. On the other hand, the negative value of U leads to the least possibility to have a bug in the window.

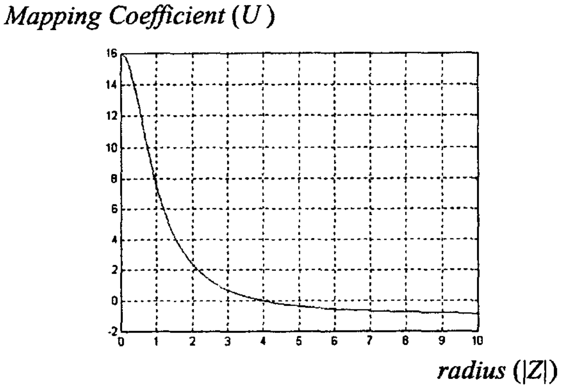


Figure 5. Mapping Coefficient v.s. Radius

By experimental tests, bug's shape feature threshold, S_{th}, is found. For each window slide in raster scan by one pixel, bug's shape feature value, S, can be calculated from equation (3). If S is higher than S_{th}, possible a bug is detected. Otherwise the window is shifted to the next pixel until the entire binary image is covered. Note that the reason why the mapping is designed to map to the top-left instead of to map to the center of the window is because there exists a middle-white Mealy bug in transition state. This mapping can efficiently capture both brown Mealy bugs and brown with middle-white Mealy bugs. Since the mask window is well defined, it is easily to find the center of a Mealy bug by simple linear shift.

3.2 Refinement Mapping

After the highly expected area is found in the previous section, another mapping function is applied to ensure the a single bug detection results. The equations of mapping function are as follows,

$$V_{xy} = \begin{cases} 1 & ;|x| \le 3, |y| \le 3 \\ -2 & ;3 < |x| \le 7, 3 < |y| \le 7 \end{cases} \qquad (4)$$

$$SV = \sum_{x=-7}^{7} \sum_{y=-7}^{7} V_{xy} \qquad (5)$$

Where V_{xy} is another mapping coefficient and SV is the summation of mapping coefficients, V_{xy}, in a new 15×15 window, as show in figure 6. The process is the same as the previous section. If the SV is greater than the threshold, SV_{th}, which obtained by experiment, the object inside the window is defined as a detected Merly bug.

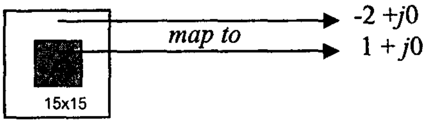


Figure 5. 15x15 mask (x-y plane), shade area as non-negative mapping coefficients

3.3 Texture Detection with Neural Network

Although the prior mapping method is fast for Mealy bug recognition, it can detect only an isolate bug. Hence, more accurate method should be applied. Since brown Mealy Bugs has some stripe texture in brown color comparing with branches, this texture can be used to distinguish a bug from the other similar objects. However, the stripes compose of light brown and dark brown color. These colors are used for camouflage. Therefore, the variation of brown color is hardly to detect. First, standard deviation is used to represent the texture, but it vastly varies when leaf brightness changes. Hence, the following function is designed in order to be independent with brightness,

$$R = \frac{\#sign\_changed\{B_{i,j} - B_{i,j+N}\}}{N} \qquad (6)$$

where R is roughness parameter. The "*# sign_changed {.}*" is a function which its range is the number of brightness sign changed between $B_{i,j}$ to $B_{i,j+N}$, where i as row, j as column, and N as the length of pixels in detection range. Clearly, if the texture is rough, the sign of brightness difference between the connected horizontal pixels changes dramatically.

However, only roughness parameter is not efficiently enough. Therefore, the other characteristics are carefully selected to feed the neural network; i.e. mean of red, mean of green, mean of blue, and size of the object. The 3-layer neural network composes of 5 input nodes, 25 hidden nodes for the first hidden layer, and 10 hidden nodes for the second hidden layer with sigmoid as transfer function. Certainly, the neural network is trained by well-known back-propagation method [9].

4. Experimental Results

In this paper, data acquisition is done by scanner with 120 dpi resolution, brightness = 125, contrast = 125, gain = 100%. Twenty images of Mealy bug on different leaf (619 insects) are tested on Pentium III 450 MHz (RAM 64 M) where the other twenty different images are used to train the neural network. All the processes together takes no longer than 3 seconds in computation time per one image. The experimental results show in table 1 and 2.

Table 1. Pre-processing Effects with neural network

Results	Without Leaf's Edge Removal	Without Brown Color Filter	All Pre-processing
Bug Found			
- Correct	74.96%	55.09%	73.18%
- Fault	25.20%	16.64%	11.79%
Bug Missing	25.04%	44.91%	26.82%

Table 1 shows the effects of various pre-processing techniques using neural network as recognition phase. All pre-processes techniques increase performance of Mealy bug detection process. However, without leaf's edge removal, number of correct detection is about 74.96% which is the highest among all processes, but the fault detection (detected something as a bug, but it is not.) is so high as well as time-consuming for recognition phase. After using all preprocessing, the fault detection is decreased.

Table 2. Different Detection Schemes

Results	Mapping Only	Neural Network Only	Mapping and Neural Network
Bug Found			
- Correct	10.50%	73.18%	73.99%
- Fault	0.65%	11.79%	11.95%
Bug Missing	89.50%	26.82%	26.01%

Table 2 shows the results of different detection schemes by using mapping only, using neural network only, and using both mapping and neural network. The last scheme achieves the highest percent of correct detection. The mapping is very helpful to locate the expected area before applying neural network. However, number of bug missing is still high, approximately a quarter of all Mealy bug can not be detected.

5. Conclusions

In this paper, various image processing techniques are reported to solve the insect counter problems. The promising results show that almost ¾ of total Mealy bugs are correctly detected. Therefore the next step is to improve the performance. Finally, future insect counter research should be able to apply with various types of insect via video sequences. This research will be very useful for the future precision farming and automatic insect monitoring system in the near future.

6. Acknowledgement

This work is supported in parts by KIP Lab (Kasetsart Image Processing Laboratory), Kasetsart University, Bangkok, and NECTEC (National Electronics and Computer Technology Center), Thailand, under the project of National Software Contest # 3.

References

[1] W. Romans, B. Poore, and J. Mutziger, "Advance Instrumentation for Agricultural Equipment", *IEEE Instrum. Meas. Magazine,* vol.3, no.1, pp. 26-29, March 2000.

[2] B. Colpitts, D. Luke, G. Boiteau, M. Doyle, "Harmonic Radar Identification Tag for Insect Tracking", *IEEE Canadian Conference,* vol.2, pp. 602-605, 1999.

[3] S.M. Sekelsky, "Multi-frequency radar observations of liquid clouds", *Geoscience and Remote Sensing Symposium, IGARSS 2000, Proceedings IEEE International,* vol. 5, pp. 1807 -1809, 2000.

[4] D.S. Zrnic and A.V. Ryzhkov, "Observations of Insects and Bird with a Polarimetic Radar", *IEEE Trans. on Geoscience and Remote Sensing,* vol.36, no. 2, pp. 661-668, March 1998.

[5] E. R. Davies, et al., "Detecting Insects and Other Dark Line Features Using Isotropic Masks", *IEE Int. Conf on Image Processing and its Application,* no.465, pp. 225-229, 1999.

[6] K. Phopariphat, K. Rerkrai, V. Areekul, "The Measurement of Mealy Bug Population on Plant", *EECON-23,* Thailand, pp. 557-560, November, 2000 (Thai Language).

[7] R. C. Gonzalez and R. E. Woods, "Digital Image Processing", Addison - Wesley, 1992.

[8] S. Thongthammachat, *Introduction to Complex Analysis,* Kasetsart University, 2000 (Thai Language).

[9] A.S. Pandya and R.B. Macy, *Pattern Recognition with Neural Network in C++,* CRC Press, 1996.