Tarım Makinaları Bilimi Dergisi (Journal of Agricultural Machinery Science)
2008, 4 (2), 199 - 204

Introducing Easy to Use and Accurate Image Processing Object Detection Algorithms Suitable for Sprayer Calibration and Other Similar Purposes

Mohammad Hossein ABBASPOUR-FARD, Amin DANESHJOO
Mohammad Hossein AGHKHANI, Mahdi ARIAN
Ferdowsi University of Mashhad, Faculty of Agriculture, Department of Agricultural Machinery
Post Code: 9177948978, Mashhad, IRAN
abaspour@ferdowsi.um.ac.ir

Abstract: In this paper two image processing algorithms are presented for small object detection purpose which can be used in blob analysis of sprayer calibration and other similar applications. The algorithms named label injection (LI) and color injection (CI) respectively. In the latter an object is detected and counted by applying a sequential color code, thus at the end of processing each object (blob) will be distinguished with a unique color and in the former this is carried out by applying a unique numerical code to each object. The algorithms have been implemented and adapted in a program called Scientific Image Blob Analysis (SIBA). The program can read the scanned image of water sensitive cards, obtained from nozzle's spray. Some threshold techniques are also used for noise removal and converting the original image to a binary one (e.g. red for blob and white for background). The blob features calculated in the program are area, actual diameter, Number Median Diameter (NMD), Volume Median Diameter (VMD), blob density (Number of blobs/area), and percentage of blob's area coverage, blob count and finally the uniformity of spray (standard deviation of droplet's size).The algorithms were evaluated analytically and experimentally, comparing with the results of a leaf area meter and a planimeter. Although the results showed a more accurate and faster performance for both algorithms, a direct correlation was also found between their accuracy and the image's resolution. In other words, the higher the resolution of the image, the more accurate of the results particularly for the smaller blobs. This was mainly due to the steplike inherent of the pixels showing a blob's boundary. However, by choosing an appropriate resolution the error of measurement may reduce to %1. The program can also easily be used for any area detection and calculation of the images such as leaf, fruits,beans etc. It is a good package especially for the researchers in the developing countries with a limit access to similar resources. Moreover there might be some uncertainty about the accuracy of such commercial products.
Key words: Blob analysis, Image processing, water sensitive paper.

INTRODUCTION

One of the most important and feasible way of controlling and limiting the side effects of chemicals in agriculture can be achieved by calibration of sprayers, which leads to a more uniform and suitable size of droplets (Matthews, 1999). While a very small droplet can cause the intensification of pollution by drift, a very large droplet can pollute the soil by sliding over the plant's leaf (Kepner et al., 1987). Currently, in some developing countries such as IRAN manual methods are used for blob analysis during sprayer calibrations which are laborious, inaccurate and time consuming (Afshari and Bayat, 1992). In these methods water sensitive papers, which are smeared with Bromo phenol blue (Matthews, 2000), are used to register the droplets coming out from sprayers. In this article a soaftware is prsented which employs two innovative object detection and counting algorithems along with additional calculations required for the blob analysis of sprayers calibration. The soaftware was written in Visual Basic programming language and named SIBA (Scientific Image Blob Analysis). It is ready for use by the researchers and the manufacturers of sprayers and can be obtained from the authors of this paper.

MATERIAL and METHOD

Generally image processing technique involves creating the digital image and using a suitable programming language to modify the image and its data. In this project the digital images are created by scanning the water sensitive papers. Since the droplets sprayed from different sprayers cover a very broad range (Bindra, and Singh, 1980), the software should capable of recognizing the smallest droplet of the image. In the other words, the smallest droplet should not smaller than the zise of a pixle. Therefore, the resolution of the image defines and limits the size of the smallest droplet which can be recognize by software as follow:

$$\mathrm{dpi}_{\min} = \frac{25400}{D} \qquad (1)$$

where, $\mathrm{dpi}_{\min}$ is the minimum required resolusion of the scanned image and D is the diameter of the smallest droplet in *μm* (*micron)*.

After scanning the water sensitive paper samples, according to the conditions of the image such as the intensity of the light and the amount of noise, some preprocessing can be carried out on the image by the software. In the next step, the main processing of the image can be completed by creating a binary image and using the innovative algorithms which were provided in the software. Figure 1 shows the overall view of the software.

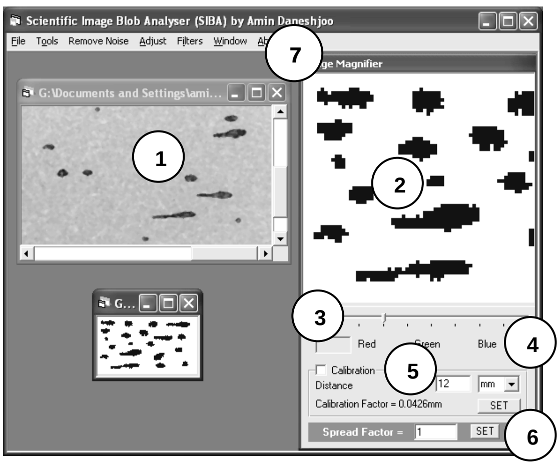

Figure 1. The overall view of the software and its menus and windows.

As it can be seen the software comprises different menus and windows including: 1) main image display; 2) magnified image display; 3) magnifier adjustment slider; 4) color display; 5) calibration setting of the image for measurement of the actual size of droplets; 6) spread factor of the droplets and 7) toolbar menu. The toolbar menu has several menus such as File, Filters (Grayscale, Invert color and Sharpen), Adjust and Remove noise. The “grayscale filter” converts the color images to grayscale one. The “Invert color” filter is used when the target objects are brighter than the background. The “Sharpen filter” is useful when for any reason the image is somewhat blur and the user wants to strengthen the details of image (Khademi, 2004).

In “adjust menu” some commands are available to adjust the color and the light intensity of the image. The “color adjust” is used when the difference between the color of the objects and the background is not sufficient for an appropriate image analysis. In “remove noise menu”, two different filters are provided; “Low pass” or “mean filter” and the “Median filter” (Pratt, 2001). While the former is appropriate for images containing relatively large objects, the latter can remove the noise form images having small objects without blurring the image (Gonzales and Woods, 1992).

The “Tools menu” has some commands to make the histogram of the image and creating the binary image. By histogram the frequency of the color level (between 0 - 255) of the main colors (Red, Green, and Blue) can be displayed. After preprocessing of the color image, it is converted to a binary image. In this process the color of all objects (blobs) converts to a unique color (in this software to red) and the color of the background also converts to another color, mainly white. To make a binary image, the software provides three different algorithms. Hence based on the image's condition one of them can be chosen and used by the user.

Water sensitive paper based binary algorithm:

Based on the investigations carried out during this project, on the texture of the images provided from scanning of the sensitive papers, it is concluded that in an image, the distinction threshold between blobs and background is level 200 of red color. In other words, if the red color level of a pixel is equal or greater than 200, the software recognizes this pixel as a part of or a blob, other wise it is part of the background.

Histogram binary algorithm:

In this method the histogram of grayscale image is used to define the distinction threshold between blobs and background. After creating the histogram, the user can move the mouse (curser) across the grayscale level of histogram and click on any value between 0 - 255 and select this value as distinction threshold. By this threshold a binary image is created. If this binary image does not satisfy the user another value can be selected by trial and error. This algorithm is used when the original image is not essentially a water sensitive paper.

Blob analysis with binary images

After preprocessing of the original image and creating a binary image, the main process of the image can be performed. This includes: detecting and counting the objects (blobs), the area of individual objects, the actual diameter of the droplets, and finally the details of calculations is displayed and saved. This software employs two innovative algorithms for detection and counting of objects which are label injection (LI) and color injection (CI) respectively. In the following sections it is shown that these algorithms have some advantages compare with convetional algorithms especially from speed and accuracy points of wiew.

***I)* Label injection:** With this algorithm each pixel is uniquely and independently allocated with a numerical code. In other words, whenever the control of program attains a “red pixel” (in a binary image) the pixel will be examined for its allocated code. If it is a non-labeled pixel, a unique code will be allocated to this pixel and also all attached red pixels to it. This means that when the program reaches the first pixel of a new blob, its all pixels will obtain an identical code hence, afterward this blob will be recognized by program with this label (code). High accuracy, independent from shape of object and suitability of small blobs are some important advantages of this algorithm.

II)* Color Injection *: With this algorithm a sequential and unique color code will be allocated to each “red pixel” and also all its attached red pixels. In other words whenever the program reaches the first pixel of a blob, which in a binary image is a red pixel, the color of this pixel and all pixels belong to this blob will obtain a new unique code color. At the end of this stage the pixels of tagged blobs which now occupy a unique code are sorted, counted and stored for further analysis. High accuracy, independent from shape of object and suitability of relatively large blobs are some important advantages of this algorithm.

Using these two algorithms and having known the number of blobs and the number of pixels for each blob, and the calibration factor of the image, which is defined by user, the area and the equivalent diameter of each blob is calculated as follow:

$$A = \frac{\pi D_e^2}{4} \Rightarrow D_e = \sqrt{\frac{4A}{\pi}} = 1.128\sqrt{A} \qquad (2)$$

According to the properties of the paper and the size of blob, a suitable spread factor (Daneshjoo, 2007) is defined and accordingly the actual size of droplets and others statistical parameters such as Number Median Diameter (NMD), Volume Median Diameter (VMD), blob density, percentage of blobs area coverage and the uniformity of spraying are calculated.

Software Validation

To evaluate the performance of program from accuracy and speed points of view, it was evaluated analytically and experimentally.

Analytical validation*:* The main objective of this section is to show the performance of the *Label injection and Color injection algorithms* in detecting and counting of the image’s objects with different shape and size. It is called analytical validation, because the employed image was created digitally in Microsoft PAINT software (Fig 2.). Therefore there is no need for any preprocessing of image such as *noise removal* etc. As a result no associated error such as unwanted removal or color change of pixels occurs.

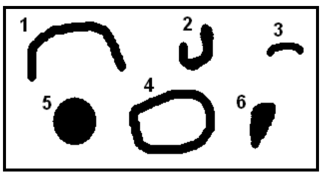

Figure 2. The image created in MS PAINT software for analytical validation of the software.

The performance of software in detecting and counting the objects is shown by numbers beside each object in Figure2. A PC computer with an AMD 750 MHz CPU was used for this evaluation.

Experimental Validation: Since all samples of water sensitive papers are first scanned and then opened in the environment of the software, some pixels may added (noise) during scanning and some pixels of objects may removed unwontedly during preprocessing of the image which can be the source of software error. For this reason a picture firstly created in MS Word with some objects of precise dimensions; then scanned and opened in SIBA software (Fig.3).

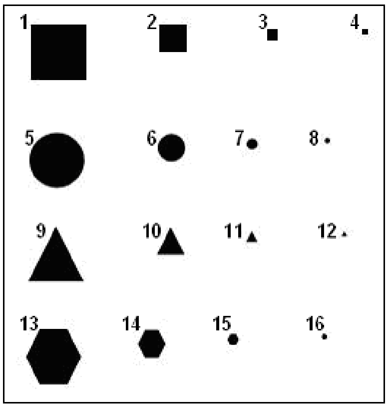

Figure 3. The actual image of a picture firstly created in MS Word software and then scanned.

The original picture had different objects in shape and size ranging from 1 to 10*mm*. **A HP Laser Jet 1500** was used for creating the hard copy of the picture. A **CanoScan 8400F** scanner was also used to create the digital images of the scanned picture with 400 and 750 dpi resolutions. These images then were opened in SIBA software for further processing and experimental validation.

Generic Performance of SIBA: Measuring the area of objects is in the interest of different areas of agricultural research such as leaf area, spot on fruits and vegetables from plant disease etc. In this section it is tried to show the ability of SIBA in measuring different area and wide range of its applications. In Figure4 the image of two leaves are show, which their area were measured in three different ways: using SIBA software, a Planimeter and a digital leaf area meter made from **Delta-T Device** Company.

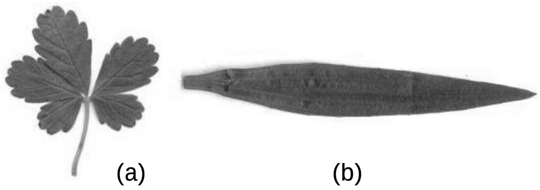

Figure 4. Leaf samples used for assessment of SIBA performance in measuring different areas.

Example of practical application of SIBA for Water Sensitive Papers: To show the performance of SIBA in treatment a real sample of sensitive paper, in this section, the analysis of a sample is presented as an example. The sample was collected from a centrifugal sprayer of containing a spinning disk of 20*cm* diameter, with 20 grooves and rotational speed of 3000*rpm*. The actual size of the paper was 69 × 30 *mm* and then scanned with 400dpi resolution. Figure5 shows the original scanned image of the paper and the binary image of it.

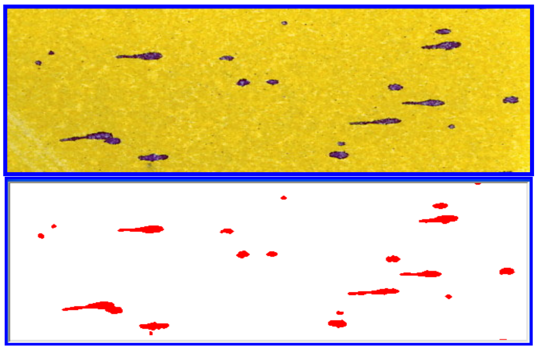

Figure 5. A Sample of water sensitive paper; Top original scanned paper and bottom its binary image created by SIBA software.

Figure6 shows the result of SIBA performance in counting the image's blobs. Each blob was tagged with a unique number by the action of Label injection algorithm and occupied a unique color due to the action of Color injection algorithm.

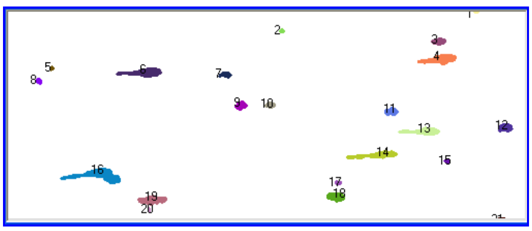

Figure 6. The image of water sensitive paper sample after processing with SIBA software and applying the Label and Color injection algorithms.

RESULTS and DISCUSSION

The results of analytical validation are shown in Table1. As also shown in Figure1, in spite of different object's shapes and sizes, both Label injection and Color injection algorithms could correctly detect and count all objects. It can be seen that regardless of object's shape and size, the program could correctly detect and count them ascending from top left corner to bottom right corner. The running time, for Label injection and Color injection algorithms were 0.57s and 0.42s respectively.

Table 1. Results of algorithms performance for analysis of the image shown in Figure1.

Object No.	Object area calculated by algorithm (in Pixel)
1	1443
2	776
3	293
4	2021
5	1607
6	805

The results of experimental validation of program in detecting and counting the objects of a real scanned image are shows as a number adjacent to the objects in Figure3. It can be seen that regardless of object's shape and size, the program could correctly detect and count them ascending from top left corner to bottom right corner. To evaluate the effect of image's resolution on the performance and accuracy of the algorithms' calculations, the image of Figure3 was scanned with two different resolutions, 75*dpi* and 400*dpi*, respectively. The results of both algorithms' analysis are shown in Table2. The real objects' area was calculated from their mathematical equation (e.g. πr^2 for circle and $\frac{bh}{2}$ for triangle) and the area calculated from program was obtained from number of pixel per objects and the size of each pixel.

Comparison of results shown in Table2 indicates that, increasing the resolution of image will improve the accuracy of program calculations. Furthermore the smaller object, the higher percentage of error. This is due to the fact that for a small object with few pixels, removing only one of its belonging pixels (e.g. by noise removal algorithms), will remove higher percentage of its area compare with a bigger object with more belonging pixels. The object's shape also is another suspicious factor that can affect the accuracy of program performance. In other words, objects with angled, diagonal and slopping boundaries will have more associated calculation's error than those with regular shapes such as rectangle, square or even a circle. This may be due to inherent characteristics of pixel base representation of digital images. In this way the boundaries of objects are approximated by the chain of outer pixels as shown in Figure7. If the boundary is vertical or horizontal its pixel representation will be as exactly as possible to its real one. However, for an inclined boundary the steplike inherent of pixel representation will remove some belonging area of the real object which is the source of additional calculation error.

Table 2. Performance of software's algorithms with different resolutions in experimental validation.

Object No.	Actual area (mm^2)	Image resolution (dpi) 75: Calculated area (mm^2)	75: Difference (%)	400: Calculated area (mm^2)	400: Difference (%)
1	100.00	99.80	0.2	100.20	0.2
2	25.00	24.95	0.2	24.95	0.2
3	4.00	3.99	0.2	3.99	0.2
4	1.00	1.2	20	0.99	0.2
5	78.50	78.10	0.5	78.90	0.5
6	19.62	19.32	1.5	19.82	1.0
7	3.14	3.32	5.9	3.21	2.3
8	0.79	0.99	27.1	0.81	3.5
9	50.00	49.56	0.9	50.40	0.8
10	12.50	13.10	4.8	12.73	1.9
11	2.00	2.10	5.3	2.06	3.0
12	0.50	0.66	33	0.54	7.8
13	75.00	74.18	1.1	75.8	1.0
14	18.75	19.29	2.9	19.12	1.9
15	3.00	2.80	6.7	3.10	3.33
16	075	0.99	33	0.81	8.0

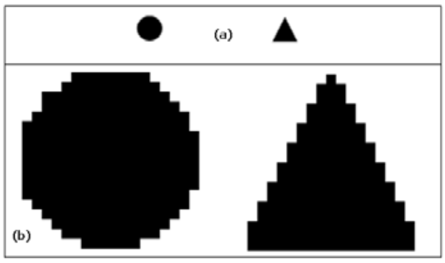

Figure 7. The effect of steplike inherent pixel representation of boundaries: a) real image of objects and b) ten times magnified image of the objects.

The results of leaf area measurements are shown in Table3. Although there is no enough information about the accuracy of the Planimeter and Leaf area meter which can compare the performance of SIBA with them, according to Table3, it can be observed that the values of SIBA measurements are interestingly within the values of Planimeter and Leaf area meter. In other words, for all measurements the Leaf area meter readings are the lower limits and the Planimeter readings are the higher limits. Therefore the performance of SIBA can be assessed based on the difference between its reading and the average of Planimeter and Leaf area meter readings. With this postulation, the average error of SIBA would be about 0.9 percent. Table3 also shows the comparative time efficiency of the software. It can be seen that the software could perform its tasks 6 and 30 times faster than the Leaf area meter and the Planimeter, respectively.

The results of practical application of SIBA on water sensitive papers are shown in Figures 5 and 6. It can be seen that the software could successfully make a binary image and detect the blobs. The details of blobs' area calculations can also be obtained from Daneshjoo, 2007. However, with a precise look at Figure6, it is seen that blob No. 16 in fact is a pair of blobs which are attached together, but the software treated them as a single blob.

Table 3. Performance of SIBA software in measuring leaf area compare with other methods.

Leaf	PLanimeter		Leaf area meter		SIBA software	
	Area (mm^2)	Consumed time (s)	Area (mm^2)	Consumed time (s)	Area (mm^2)	Consumed time (s)
a	1031	300	996	60	1023	10
b	1017	300	982	60	1008	10

CONCLUSIONS

The main objectives of this paper was introducing two innovative algorithms namely Label injection and Color injection algorithms which are suitable for small objects such as registered blobs sprayed on water sensitive papers. The algorithms were successfully implemented in software named SIBA. The software also comprises some other preprocessing algorithms, to make the images ready for use in Label injection and Color injection algorithms. Furthermore, the software was designed in a generic manner for other potential applications in different fields of research specifically for agriculture research purposes such as area meter, spots from plants' diseases etc. However, the algorithms currently treat the attached blobs as a single one, more attempts are needed to develop an algorithm to solve this problem.

REFERENCES

Afshari M. and H. A. Bayat, 1992. Water Sensitive Papers and Their Application in Sprayers' Calibration. Iranian Journal of Pests and Plant Diseases (in Farsi). 57: 71 – 75.

Bindra, O.S. and H. Singh, 1980. Pesticide-application Equipment.1st ed. New Delhi: Oxford & IBH. Pub. Co., India.

Daneshjoo, A., 2007. Design of a Suitable Software for Sprayer's Calibration Purpose (in Farsi). MSc. Thesis. Ferdowsi University of Mashhad, Mashhad, IRAN.

Gonzales, R. And R. Woods, 1992. Digital Image Processing, Prentice Hall.

Kepner R., R. Bainer, E. Barger. 1987. Principles of Farm Machinery. Cbs Publisher & Distributor.

Khademi, M., 2004. Processing of Digital Images (in Farsi). Second Edition. Ferdowsi University Publications. Mashhad, IRAN.

Matthews, G. A., 1999. Application of Pesticides to Crops. Imperial College Press. 57 Shelton Street, Covent Garden, London, WC2H 9HE.

Matthews, G. A., 2000. Changes in Application Technique Used By the Small Scale Cotton Farmer in Africa. Tropical Pest Management. 36: 166-172.

Pratt, W. K., 2001. Digital Image Processing, John Wiley & Sons.