AUTOMATIC ASSESSMENT OF BIOLOGICAL CONTROL EFFECTIVENESS OF THE EGG PARASITOID TRICHOGRAMMA BOURARACHAR AGAINST CADRA CAUTELLA USING MACHINE VISION

by

YUQI SONG

B.S., Jilin University, 2014

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Biological and Agricultural Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2016

Approved by:

Major Professor
Naiqian Zhang

Copyright

Abstract

The primary objective of this research is to achieve automatic evaluation of the efficiency of using *Trichogramma bourarachae* for biological control of *Cadra (=Ephestia) cautella* by calculating the rate of parasitization. *Cadra cautella* is a moth feeding as a larva on dried fruit as well as stored nuts, seeds, and other warehouse foodstuffs. It attacks dates from ripening stages while on tree, throughout storage, and until consumption. These attacks cause significant qualitative and quantitative damages, which negatively affect dates' marketability, resulting in economic losses. To achieve this research goal, tasks were accomplished by developing image processing algorithms for detecting, identifying, and differentiating between three *Cadra cautella* egg categories based on the success of *Trichogramma* parasitization against them. The egg categories were parasitized (black and dark red), fertile (unhatched yellow), and hatched (white) eggs. Color, intensity, and shape information was obtained from digital images of *Cadra* eggs after they were subjected to *Trichogramma* parasitization and used to develop detection algorithms. Two image processing methods were developed. The first method included segmentation and extractions of color and morphological features followed by watershed delineation, and is referred to as the "Watershed Method" (WT). The second method utilized the Hough Transformation to find circular objects followed by convolution filtering, and is referred to as the "Hough Transform Method" (HT). The algorithms were developed based on 2 images and then tested on more than 40 images. The WT and the HT methods achieved correct classification rates (CCRs) of parasitized eggs of 92% and 96%, respectively. Their CCRs of yellow eggs were 48% and 94%, respectively, while for white eggs the CCRs were 42% and 73%. Both methods performed satisfactorily in detecting the parasitized eggs, but the HT outperformed the WT in detecting the unparasitized eggs. The developed detection methods will enable automatic evaluation of biological control of *Cadra*

(=Ephestia) cautella using *Trichogramma bourarachae*. Moreover, with few adjustments these methods can be used in similar applications such as detecting plant diseases in terms of presence of insects or their eggs.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my sincere gratitude to Kansas State University for letting me fulfill my dream of being a student here. I would also like to thank the department of Biological and Agricultural Engineering for giving me the opportunity to write an honor thesis. To my committee, Dr. Naiqian Zhang, Dr. Daniel Flippo, and Dr. Mohammed Salih El-faki Mozib, I am extremely grateful for your assistance and suggestions through my project. To all my friends and family for helping me survive all the stress from this year and not letting me give up. Most of all, I am fully indebted to Dr. Naiqian Zhang, my advisor and director, for his understanding, wisdom, patience, enthusiasm, and encouragement and for pushing me farther than I thought I could go.

Dedication

I dedicate this thesis to my parents Lihui Song and Jiuhong Du. I hope that this achievement will complete the dream that you had for me all those many years ago when you chose to give me the best education you could. I also dedicate this work to my husband, Mr. Puxuan Li for being a great pillar of support.

Chapter 1 - Introduction

Background

The date palm (*Phoenix dactylifera L*.) is the prevalent fruit crop in the Kingdom of Saudi Arabia (KSA). As a result, the KSA is the world's second largest producer of dates, supplying 17.6% of the world market (Siddiq and Greiby 2014). Current date production in the KSA is approximately 1 million tons annually.

Dates encounter serious economic losses in both quantity and quality caused by warehouse insects. Among the insects attacking stored products is the date moth *Cadra* cautella. E. cautella (Lepidoptera: Pyralidae), which is a common cosmopolitan pest in most of the temperate world as well as in warmer areas. It causes damages to dry fruits, stored grains, and their products (Boshra, 2007; Arbogast and Chini, 2005). One biological control method includes releasing parasitoid wasps of the genus *Trichogramma* Westwood into a stored-product environment (Schoeller and Flinn, 2000; Steidle et al., 2001). Skilled technicians then measure the performance of *Trichogramma* for biological control by visually counting the numbers of parasitized (black and dark red), fertile (unhatched yellow), and hatched (white) eggs in the sample. Visual detection and counting of eggs, however, is tedious, laborious and time consuming.

One of the most popular approaches for identifying and calculating each egg category is to use image processing technologies. Images of biologically controlled *Cadra* egg samples are captured by a digital camera. Color and shape information is then extracted from the images for counting each categories' eggs. Universal methods for automatic identification of circular objects could be used in many other fields such as fruits counting, bakery products counting. Furthermore, the

accuracy for counting circular objects could be improved by modifying corresponding objects' size and color parameters for specific applications.

Objectives

The primary objective of this research was to automatically evaluate the efficiency of biological control of *Cadra* cautella using *Trichogramma*. In this study, image processing algorithms were developed to identify and differentiate egg categories based on the success of *Trichogramma* parasitization against egg categories. These egg categories are:

1) Parasitized eggs (black and dark red),
2) Fertile (unhatched) eggs (yellow),
3) Hatched eggs (white).

This research also included the following specific objectives:

1) Establish a database of digital color images of *Cadra* eggs after being subjected to *Trichogramma* parasitization for eight days.
2) Extract features that characterize the parasitized and unparasitized *Cadra* eggs based on color and shape information in egg images.
3) Design, develop, and test algorithms for segmentation, noise removal, separation of touching eggs, classification, recognition, and counting of parasitized *Cadra* eggs in digital images.

Thesis outline

The rest of the thesis is organized as follows. Chapter 2 includes the background knowledge of insect control, biological control and machine vision. Chapter 3 describes the procedure to

acquire images. Chapter 4 contains implementation details of the Hough transform (HT) algorithm, and Chapter 5 introduces implementation details of the watershed transformation (WT) approach. The performance of these two methods is listed in Chapter 6. Chapter 7 concludes the entire work and recommends future work.

Chapter 2 - Literature Review

This chapter is comprised of three parts. The first part describes the related work in general insect control methods. The next part describes the work related to the biological control methods. The last part introduces machine vision application in automatic classification, inspection, and counting.

General insect control methods

Insect control has become increasingly essential because insects post a threat to the productivity, health, and well-being of humans, livestock, companion animals, and wild life (Brogdon and McAllister 1998). Disease vectors such as ticks, lice, mosquitoes, and bugs, which directly impact public health, are of public health importance and are of increasing concern to the general population, particularly in third world countries (Gubler, 1998; Gubler, 2002). According to the American College of Allergy, Asthma, and Immunology, more than 2 million Americans are allergic to stinging insects, more than 500,000 people enter hospital emergency rooms every year suffering from insect stings, and 40-150 people die annually as a result of insect stings. Insect and mites are also a serious threat to food crops, causing estimated 20% loss of stored grain annually (Haubruge et al., 1997). Even with current insect control technology, insects destroy over 30% of the world's food crops every year (Boyer et al., 2012). Haines-Young et al. (2000) estimates at least a yearly loss of 100 billion US dollar caused by insects in the world.

Although the global population is currently 6.5 billion, the US Census Bureau projects a worldwide population of over 9.2 billion by 2050, more than a 40% increase (Miyasaka et al., 2006). The

increasing world population creates growing demands for crop production, resulting in utilization of almost all available fertile land. Improving insect control technology is one way to produce high quality of agricultural products in increasing quantities.

Most specific insect control methods can be classified as cultural control, physical control, mechanical control, biological control, chemical control or host resistance (Mahr and Ridgway, 1993).

Cultural control involves modification of standard farm practices to avoid pests or to make the environment less favorable for the pests (Mahr and Ridgway, 1993). At present, most successful cultural control projects are based on a combination of cultural control and biological control, with moderate use of chemical pesticides (Bajwa and Kogan, 2004). Sanitation ensures that the area contains no plants or materials that may harbor pests. One example is cleaning of farm equipment that may spread pests from field to field (Mahr & Ridgway, 1993). Cultural control against tomato late blight was evaluated by Tumwine, Frinking, and Jeger (2002) in six field experiments over three years. They compared tomato growth, production and the numbers of diseased leaves between standard sanitation treatment and no treatment in each experiment. They found that blight incidence and severity were greatly reduced by sanitation. In addition, field crops such as alfalfa, soybean, corn, and small grains are planted in parallel strips to create a diverse habitat that is favorable for insects' natural enemies and less favorable for most pest insects (Mahr and Ridgway, 1993). Crop rotation replaces a crop that is susceptible to a serious pest with another crop that is not susceptible, on a rotating basis. For example, Yamada (2001) studied crop rotation with sweet corn to address plant diseases caused by fungi or bacteria. He introduced sweet corn cultivation

into taro fields to control nematodes which severely damaged taro roots. A carefully considered time of planting also helps avoid some pest problems.

Physical control applies physical practices to prevent insect pests from reaching their hosts. Physical controls can be classified as passive (e.g., trenches, fences, organic mulch, particle films, inert dusts, and oils), active (e.g., mechanical, polishing, pneumatic, impact, and thermal), and miscellaneous (e.g., cold storage, heated air, flaming, hot-water immersion) (Vincent et al., 2003). Boiteau tested the control of potato virus in seed potato by combining border crops and mineral oil sprays. After three years of field tests, they found that combining border and oil provided the best reduction in potato virus and the combination worked better than using crop borders or oil sprays separately (Boiteau et al., 2009). Commercial traps can also be used for control, such as food-baited traps for ants (de Carvalho Moretti et al., 2013).

Mechanical control directly removes or kills pests. Although mechanical control is a rapid and effective method, most are only suited for small acute pest problems. However, mechanical control has minimal impact on the natural environment and can be used in conjunction with biological control in integrated pest management (Mahr and Ridgway, 1993). Hand-picking can be used for large or brightly colored pests such as Japanese beetles or slugs, and snails (Hart and Rhonda, 1994). A strong cold water spray drives mites away from gardens, green houses, and house plants, or even kills mites (Rogers, 2000).

Biological control is a bioeffector method that controls pests using living organisms (Stiling, 1992). Additional details are included in the future sections.

Chemical control utilizes chemicals to kill or inhibit insect pests' feeding, mating, or other essential behaviors. Current chemical control methods are based on the use of insecticides (Boyer et al., 2012). Chemicals used in this method are natural or synthetic materials; carbon dioxide, fumigants, organochlorines, organophosphates, and pyrethroids are commonly used for stored-product insect control (Boyer et al., 2012). Chemical control is highly effective with relatively low cost, and its effects are generally predictable and reliable. However, chemical control exhibits biological activity against many lifeforms, thereby affecting non-target organisms. Chemical controls also present various levels of hazard to humans, and most chemicals are highly toxic to beneficial insects. Another problem of chemical control is that pests often develop resistance to insecticides. Capucho et al. (2013) developed a new strategy to control coffee leaf rust through the application of triazole fungicides in soil. A contact insecticides was found to control the redbay ambrosia beetle which is a wood-boring insect and a threat to the avocado in southeastern U.S. (Peña et al., 2011). The use of chemical method should only be undertaken after careful consideration if the benefits outweigh the disadvantages.

Host resistance pertains to plants' ability to resist damaging insect invasions. Plants have physical and chemical adaptations that allow them to repel, tolerate, or kill pests. For example, the International Potato Center (CIP) have tried for several years to develop potato varieties that have high levels of resistance to potato late blight problem (Forbes, 2012). Some plants use their physical appearance as an insect deterrent, such as plants with hair that covers their leaves (Mahr and Ridgway, 1993).

Biological control methods

Biological control is the regulation of plant and animal numbers by natural enemies (Stiling, 1992). Biological control was first introduced by Harry S. Smith in 1919. He used this word to signify the use of natural enemies to control insect pests (Huffaker, 1976). Although biological control relies on predation, parasitism, herbivory, or other natural mechanisms, it also requires an active human management role. Biological control does not kill pests; instead it prevents pests from causing harm to plants. The scope of application in biological control has expanded from the use of entomophagous insects to control insect pests to the use of a whole range of organisms to control insects, mites, snails, and even plants as diverse as algae, fungi, and herbs (Huffaker, 1976). Biological control strategies can be categorized as importation (also called classical biological control), augmentation, and conservation. Importation biological control involves importation, screening, and release of natural enemies to permanently establish effective natural enemies in new areas. This method is typically applied to nonnative pests that have no natural enemies to control their population in native lands, or pests that are not adequately controlled by existing natural enemies. Augmentation biological control involves the supplemental release of natural enemies of the target pest. Conservation biological control provides resources for natural enemies and protects them from adverse conditions via farming and gardening practices that benefit all natural enemies (Mahr and Ridgway, 1993).

The use of insecticides and other chemical treatments against pests implies the risk of adverse ecological, toxicological, and economic effects (Youssef et al., 2004). Compared to chemical

control, biological control is advantageous because it does not harm humans, plants, or native wild life. Biological control also offers self-perpetuation at little or no cost following the initial effort (Huffaker, 1976). An alternative biological control method to synthetic chemical insecticides for combating *Cadra (=Ephestia) cautella* is to release parasitoid wasps of the genus *Trichogramma* Westwood into the stored-product environment (Schoeller and Flinn, 2000; Steidle et al., 2001). *Trichogramma* are extremely tiny wasps in the family Trichogrammatidae and they exist naturally in almost every terrestrial habitat and some aquatic habitats as well. These wasps lay their eggs into the *C. cautella* eggs, thus killing the eggs and preventing their development. *Trichogramma* species are the most frequently used natural enemies for the control of lepidopteran pests of fruits and cereals in the field (Li, 1994). Worldwide, egg parasitoids of the genus *Trichogramma* (Hymenoptera: Trichogrammatidae) have been successfully utilized for biological control of several Lepidopteran pests. Because of their low host specificity, *Trichogramma* can be mass reared easily in large numbers and on different natural and factitious hosts (Oezder and Kara 2010). More than 150 different species of *Trichogramma* are known from various biotopes (Pino, 1999). Nine species of *Trichogramma* are reared in private or government owned insectaries around the world and released annually on an estimated 80 million acres of agricultural crops and forests in 30 countries (Li, 1994; Olkowski and Zhang, 1990).

In Germany and Austria, the control of the Indian meal moth, *Plodia interpunctella* (Huebner) and the Mediterranean flour moth *Cadra kuehniella* (Zeller) in food processing facilities is achieved by releasing large quantities of *Trichogramma* evanescens Westwood using the inundative release strategy (Schoeller, 2001). They parasitize insect eggs, especially eggs of moths and butterflies. Some of the most important caterpillar pests of field crops, forests, and fruit and nut trees are

attacked by *Trichogramma* wasp. In Canada, *Trichogramma* species are commonly reared and used to control field and glass house insect pests (Schoeller and Fields, 2002). One of the major considerations in the design of an augmentative biological control program using egg-parasitoids is the selection of the parasitoid species or strain, which is in turn determined by its performance in the field (Hegazi, et al., 2005).

However, this method also has limitations. For example, the host develops virtually undetectable resistance to introduced natural enemies, a situation that can only be remedied by introducing new natural enemies. In addition, the results of biological control are difficult to predict and can only be known after greatly expanded research over a long period. Research seeking a biological control solution consistently demands scientific and technical staff, funds, and time, and the final solution cannot be guaranteed in advance. These disadvantages are due to complexity of the nature and lack of information about the natural relationships among the host, the pest, and the natural enemies of the pests (Mahr and Ridgway, 1993).

Evaluation of the effects of biological control methods is another difficult task. Up to date, proper evaluation methods have not yet been established. Suitable evaluation methods should identify the advantages and disadvantages of existing natural enemies, confirm the need for new enemies, and verify manipulation of the environment or natural enemies in order to make resident species more effective. Effective evaluation methods should also provide insights into the principles of population ecology relating to the interplay of biotic and abiotic factors and demonstrate the effectiveness of natural enemies to ensure continued support for biological control research and development. Biologically-controlled insects and natural enemies must also be sampled, typically

by visually searching and counting samples but this technique is time-consuming, and insects develop rapidly and change appearance. Therefore, machine vision is commonly applied to solve this problem. Another common approach is the determination of relative density from such as tissue damage, feeding sign, or frass (Huffaker, 2012).

Machine vision

Machine vision technology and methods provide imaging-based automatic inspection and analysis, robot guidance, and process control in industry. Machine vision initially obtains the image of an object and then processes it with computer for specific applications. Features such as size, color, and shape are used to discriminate between objects and non-objects. A machine vision system acquires and analyzes images, recognizes certain features or objects within the image, and exploits and imposes environmental constraints (Awcock and Thomas, 1995).

Many valuable machine vision systems have been designed to perform automatic counting of specific object types. The usual procedure for object counting starts with segmenting the objects from the background using various thresholding techniques. Bachar et al. (1997) successfully counted pollens on stigmas in 1997 by transferring the original image into a red-scale image in order to establish a good contrast between the pollen grain and the background. Pearson et al. (2002) counted the number of pink bollworm eggs on oviposition pads using histogram features of grayscale images of the pads. They utilized a thresholding image to discern dyed eggs from the background and particle areas to count eggs. Barbedo (2014) developed an imaging system for

counting white flies on soybean leaves. He applied different color models to separate objects from the image. However, changing light conditions caused difficulties in image analysis. Choi et al. (2013) designed an image processing algorithm to accurately estimate fruit count. This algorithm included normalization of intensity, citrus fruit detection by a logistic classifier, and least square circle fitting.

Chapter 3 - Materials

This chapter introduces the procedure of acquiring images.

Rearing of *Cadra cautella*

In this study *Cadra cautella* was successfully reared in laboratory cultures on a standard diet consisting of one-half part crushed date fruit, one part barley, one and one-half parts of broiler feed and one and one-half parts layer feed (by weight), as described by Al-Azab, (2007). The sterilized diet was mixed with 400 ml of glycerol. Newly emerged male and female adults were paired in a wooden cage with screened sides. A piece of white paper was put under this cage to collect eggs. Eggs that fell through the wire mesh were collected in open dishes, and then 15 mg eggs of *E. cautella* were checked under the microscope to remove any impurities or deformed eggs. The *E. cautella* eggs were transferred into one-liter glass jar containing 200 gm of the previously mentioned sterilized diet. The glass jar was covered with a layer of cheesecloth and placed in an incubator with 50-60 % relative humidity, at 29 ±1° C with 12 light photoperiod.

Rearing of *Trichogramma bourarachae*

Two species of *Trichogramma*, namely *Trichogramma bourarachae* and *T. cordubensis* (Hymenoptera: Trichogrammatidae), were reared in the lab. The first species is arrhenotokous, producing both females and males through parthenogensis reproduction while the second species produces only females (Thelytokous). These two species are known to be adapted for harsh arid and semi-arid conditions. Two cycles of rearing both *Trichogramma* and *Cadra* were carried out so far. Only *Trichogramma bourarachae* was used in all experiments and the other species was used as backup because these tiny wasps are very sensitive and require special handling during mass production. Colonies of *Trichogramma bourarachae* used in this study were brought from

Egypt. The parasitoids were reared on *Cadra cautella* (Lepidoptera: Pyralidae) eggs under laboratory conditions. The eggs of *Cadra* were obtained as described above. The method used for parasitization of *Cadra* eggs by *T. bourarachae* was that used by Roriz et al., 2006, with slight modifications. Females of *T. bourarachae* were isolated in small glass tubes (6 cm in length and 1 cm in diameter) containing cards with eggs of *Cadra*. Egg cards were exposed to the parasitoids' females for 24 h. At the end of the parasitization period, *Trichogramma* parasitoids were removed from the glass tubes and the parasitized eggs of *Cadra* on the cards were kept under the same laboratory conditions for the development of the parasitoids. The cards with parasitized *Cadra* eggs that turned black after 7 days were taken and used for image acquisition.

Image acquisition and analysis

RGB color digital images of *Cadra* eggs after subjection to parasitization by *Trichogramma* were captured using a DIGI Optika camera mounted on an Optika Tri Zoom Stereoscope with a magnification of 16.5x. The experiment was conducted under laboratory conditions at the Agriculture and Veterinary Training and Research Station, King Faisal University, Al-Ahsa, KSA. All images were acquired under the same imaging conditions (i.e. same background, resolution [3200×2400 pixels], and working distance [11.5 cm] (Figure 3.1)). The Matlab Mathworks software was used for digital image processing and analysis.

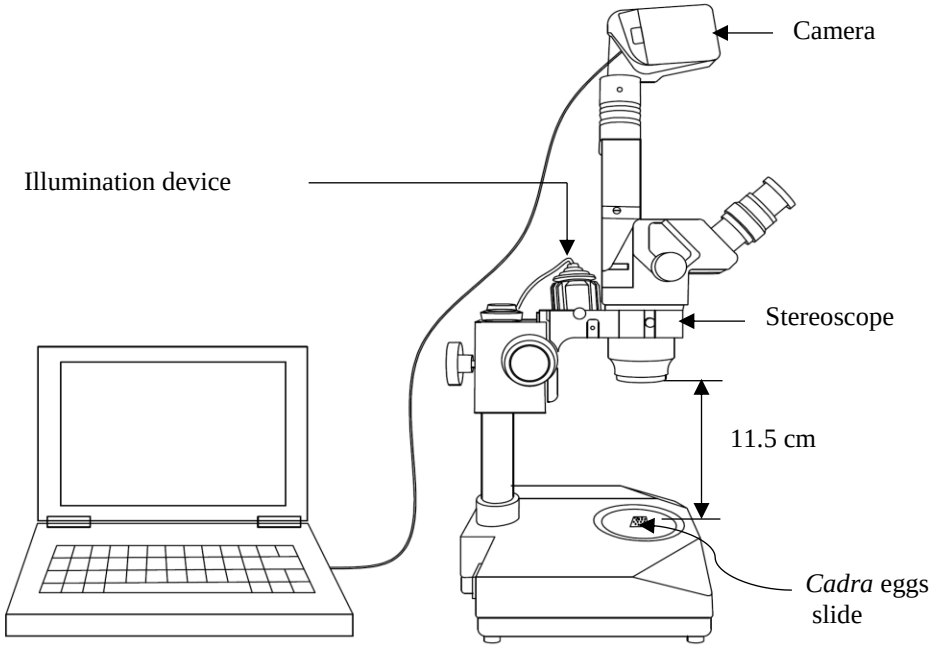


Figure 3.1 The experimental setup.

Chapter 4 - Image processing - the Hough Transformation method

This chapter explains the general principles of Hough transformation and details of the Hough transformation detection program in this study, including identification of dark-colored and light-colored cells, and the method to combine the detected cells and count the eggs in different categories.

General principle of Hough transformation

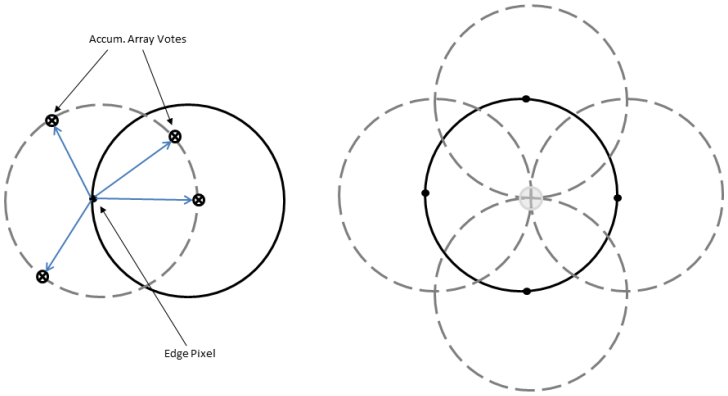
Paul Hough invented the Hough transform (HT) method in 1962 (Fokkinga, 2011). The classic HT was to identify lines in the image, although it has evolved to find imperfect instances of many other objects, such as circles or ellipses, within a certain class of shapes by a voting procedure. The HT used in this study was the circle Hough Transform (CHT), a specialization of Hough transform that detects circles. CHT is robust in the presence of noise, occlusion and varying illumination.

All HT algorithms contains the following steps:

1. Accumulator array computation.

Foreground pixels of high gradient are candidate pixels that are allowed to cast votes in the accumulator array. In the classic CHT algorithm, each candidate pixel casts votes around the candidate pixel forming a full circle with a fixed radius (The MATLAB function of "imfindcircles" – "find circles using circular Hough transform". (MATLAB, 2016)). Figure 4.1a shows one

candidate pixel lying on an actual circle (solid circle) and the classical CHT voting pattern (dashed circle) for this candidate pixel.



(a) (b)

Figure 4.1 Classic CHT voting pattern.

2. Center estimation

Votes of candidate pixels belonging to an image circle tend to accumulate at the accumulator array bin corresponding to the circle's center. Therefore, circle centers are estimated by detecting peaks in the accumulator array. Figure 4.1b shows the candidate pixels (solid dots) lying on an actual circle (solid circle), and their voting patterns (dashed circles) which coincide at the center of the actual circle.

3. Radius estimation

If one accumulator array is used for more than one radius value, as is commonly done in CHT algorithms, radii of the detected circles must be estimated as a separate step.

Hough transformation method

The HT image processing method was used in this study to find circular objects in the image. Selection of this method was based on the fact that all the eggs, regardless of their color, had a circular shape in the images, and the radii of these circles are similar. A built-in function in Matlab called *imfindcircles* was applied to find circles using circular Hough transformation. This function was used in two different ways. Dark-colored eggs could be directly detected in original RGB image by CHT while light-colored cells required a more extensive preprocessing before CHT application. After finding all circular objects in the image, detected cells were combined, and repeated cells were removed. The remaining cells were classified into different egg categories based on the color database of digital color images of *Cadra* eggs.

A flowchart of this method is shown in Figure 4.2.

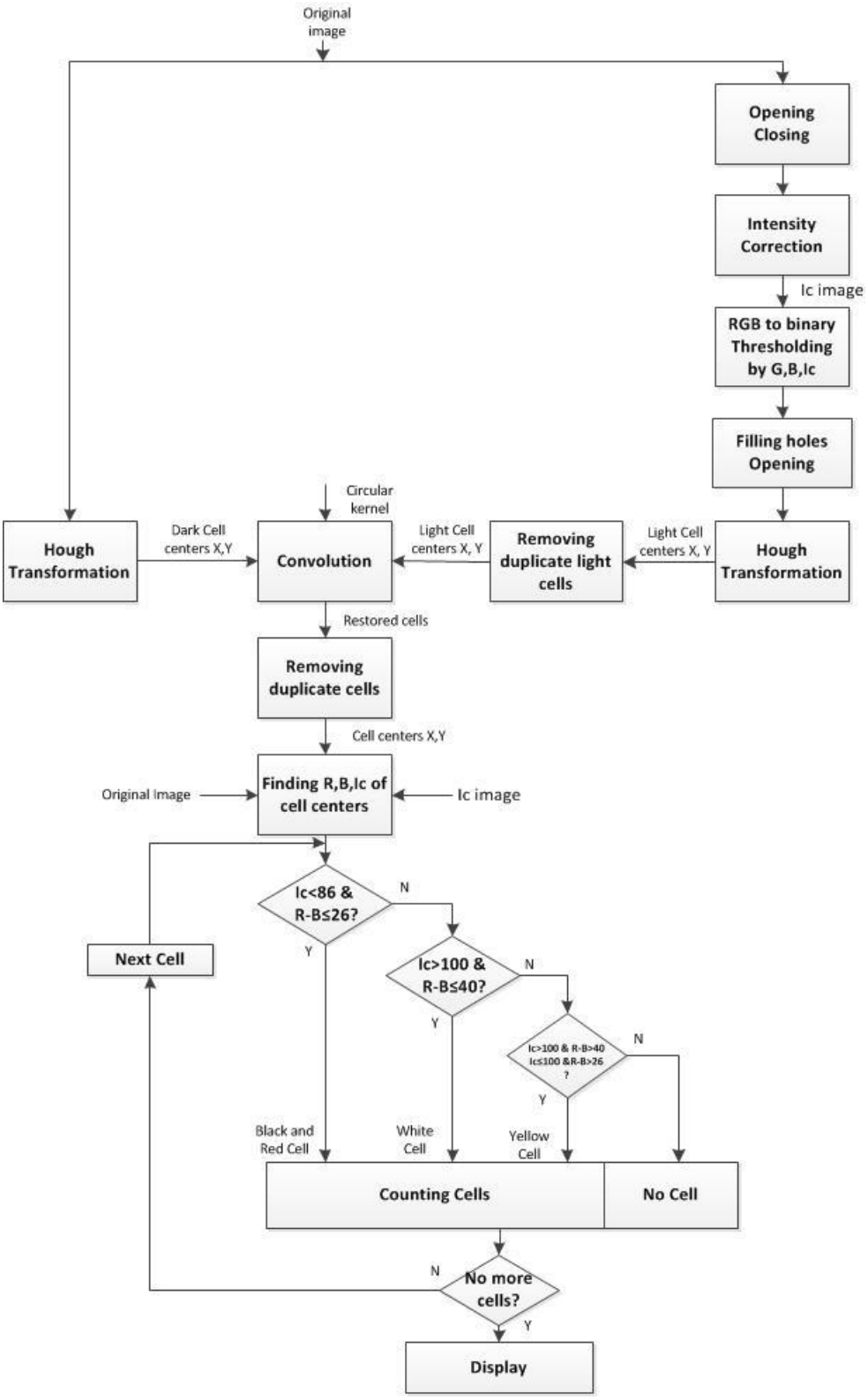


Figure 4.2 Flowchart of the HT algorithm

Finding dark-colored cells

The Hough Transformation function *imfindcircles* in Matlab was applied to identify dark-colored circular objects in the image. The function call for *imfindcircles* was

$$[centers, radii] = imfindcircles(A, radiusRange, 'ObjectPolarity', 'Sensitivity', 'Method', 'EdgeThreshold')$$

Parameters for the function call included range of radius, sensitivity, and edge threshold. Figure 4.3 shows cell diameters of the dark-colored cells in section of the original image. Because cell diameters of a majority of the dark-colored cells were between 140 - 240 pixels, the radius range used in function call was 70 - 120 pixels.

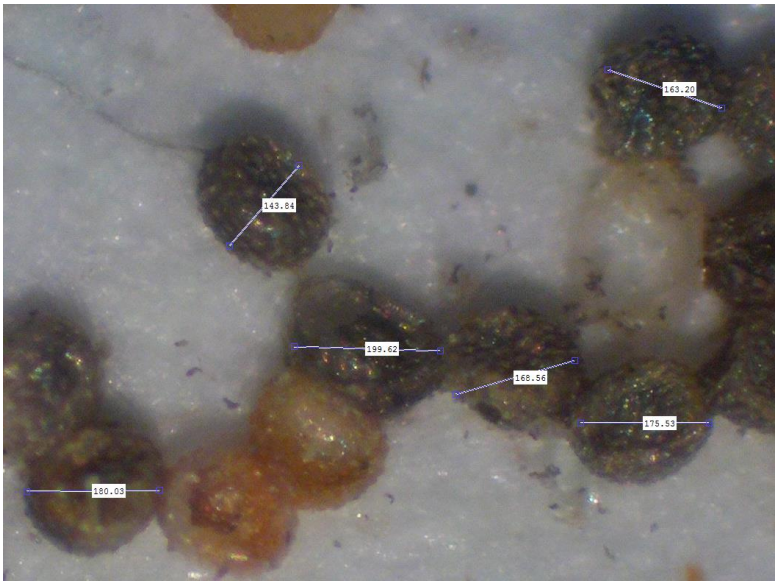


Figure 4.3 Cell diameter measurement.

The parameter "object polarity" indicates whether circular objects are brighter or darker than the background. "Dark" was specified to find circular objects that were darker than the background in the images.

The sensitivity factor refers to CHT accumulator array sensitivity, ranging from 0 to 1. A higher sensitivity value detects more circular objects. Although high sensitivity values detect weak and partially obscured circles, they also increase the risk of false detection. Sensitivity values from 0.93 to 0.98 were tested in increments 0.05 to determine the best sensitivity value that detected the most eggs with least false detections. A sensitivity value of 0.965 was then selected in our program. Figures 4.4 - 4.6 show circular objects detected with sensitivities of 0.93, 0.965, and 0.98, respectively, with identical radius range, computation method, and edge gradient.

Figure 4.4 Dark-colored cells with sensitivity of 0.93.

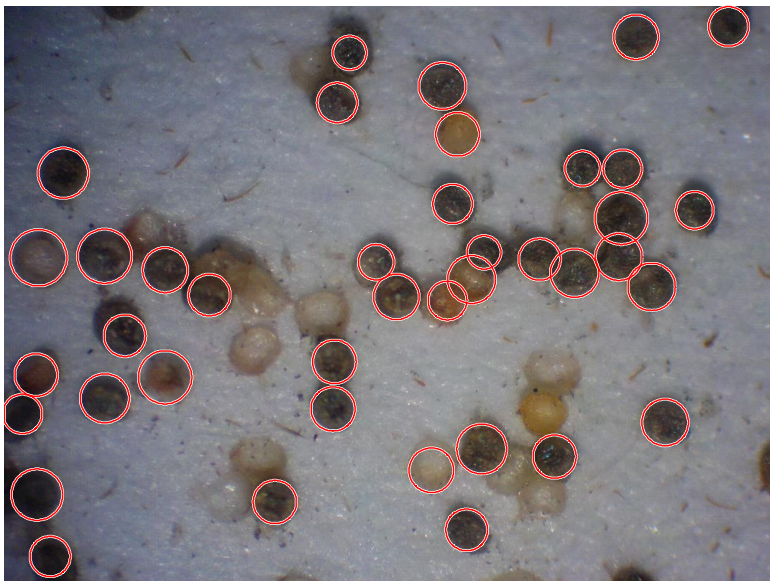


Figure 4.5 Dark-colored cells with sensitivity of 0.965.

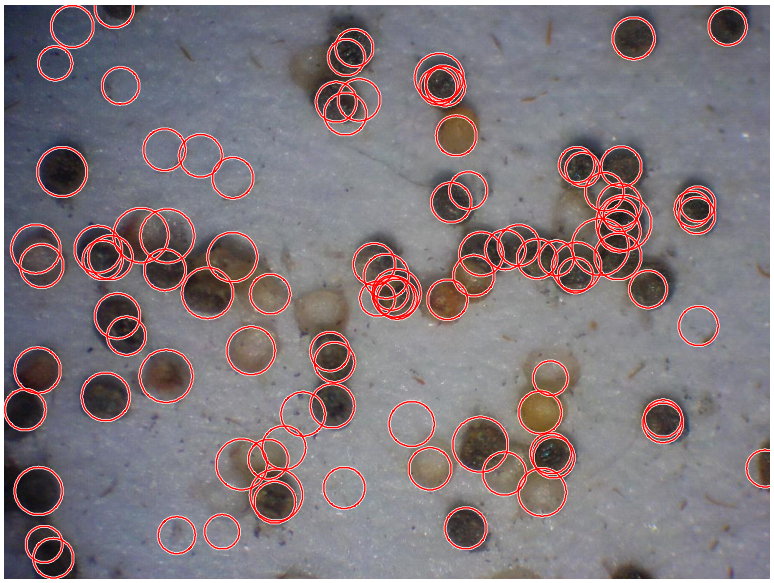


Figure 4.6 Dark-colored cells with sensitivity of 0.98.

The parameter “method” determines the method to compute the accumulator array. Two methods - phase-coding (default) and two-stage – are available. We used the two-stage method in this program. Both methods share common computational steps, but they also have unique features. The common computational steps include the following.

1. Use of two-dimensional (2-D) accumulator array:

The classic HT requires a three-dimensional (3-D) array to store votes for multiple radii, resulting in large storage requirements and long processing times. The phase-coding and two-stage methods use a single 2-D accumulator array for all radii.

2. Use of edge pixels

Memory requirements and computational speed are strongly governed by the number of candidate pixels. In order to limit the number of candidate pixels, both methods perform thresholding on the gradient magnitude of the input image so that only pixels of high gradient are included in tallying votes.

3. Use of edge orientation information:

Performance of both methods can also be optimized by restricting the number of bins available to candidate pixels. This is accomplished by utilizing locally available edge information in order to permit voting only in a limited interval along the direction of the gradient.

Despite their similarities, phase-coding and two-stage approaches differ in circle radii.

The two-stage method uses estimated circle centers and image information to explicitly estimate the radii. This technique is based on computing radial histograms. Phase-coding, on the other hand, uses complex values in the accumulator array, and the radius information is encoded in the phase of array entries. Votes cast by the edge pixels contain information about possible center locations and the radius of the circle associated with the center location. Unlike the two-stage method in

which radii must be estimated explicitly using radial histograms, the phase-coding method estimates the radii by simply decoding phase information from the estimated center location in the accumulator array (Yuen, Princen, Illingworth, & Kittler, 1990).

Another difference between these two methods is on the computational load. Although the two-stage method requires additional radius estimation, the overall computational load is typically lower than the phase-coding method, especially over a large radius range.

The "edge gradient threshold" parameter refers to the contrast between a pixel and its neighboring pixels to identify edge pixels in the image. The edge threshold ranges from 0 to 1. An edge threshold value of 0 indicates a zero gradient, and 1 the maximum gradient. A high threshold value detects fewer circles because circles with less distinct edges were ignored. Additional circular objects would be detected with a low threshold value. In this study, threshold values from 0.07 to 0.03 in decrements of 0.01 were tested, and 0.05 was finally. Figures 4.7 - 4.9 show circular objects detected with edge gradient thresholds of 0.07, 0.05, and 0.03, respectively, with identical radius range, computation method, and sensitivity settings.

Figure 4.7 Dark-colored cells detected with edge gradient threshold of 0.07.

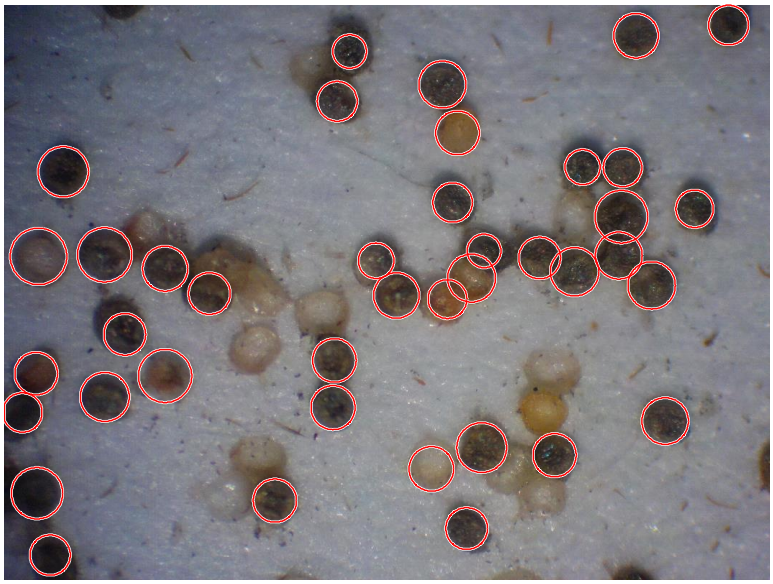


Figure 4.8 Dark-colored cells detected with edge gradient threshold of 0.05.

Figure 4.9 Dark-colored cells detected with edge gradient threshold of 0.03.

Using properly selected parameters (radius range, object polarity, sensitivity, and edge threshold), the CHT function *imfindcircles* in Matlab identified a majority of dark-colored, circular objects in the image, as shown in Figure 4.8.

Finding light-colored cells

The HT algorithm had difficulty finding circular objects with color and intensity similar to the background. Because the CHT algorithm was set to find candidate pixels on an actual circle edge based on the contrast between circular objects and the background, identifying of light-colored cells on a white background was difficult. To do this, a more extensive preprocessing was performed. The preprocessing started with a morphological opening operation (Figure 4.10). Then a complete second-order polynomial surface model was established to unify the intensities of the background pixels in the image. The second-order polynomial equation is:

$$Intensity = a + bx + cy + dx^2 + ey^2 + fxy$$

In order to derive the six coefficients of the model, six background pixels in the image need to be selected for calculation. To avoid manual selection of the six background pixels, average intensities of six 128 × 128 pixel sample areas in the image - four at the corners, one that contains the pixel with highest intensity in the image, and one more randomly selected area – were used. The 128 × 128 sample area was larger than the area of a circular cell. Thus, even if a sample area contained non-background pixels, the average intensity was still a reasonably good representation of the background intensity within the area. Figure 4.10 shows an original egg image after the morphological opening operation. Figure 4.11 shows the six sample areas in the images selected for intensity adjustment. Figure 4.12 gives 3-D surface plot displaying the intensity distribution of the image before intensity adjustment, Figure 4.13 shows the intensity-adjusted image, and Figure 4.14 gives the 3-D surface plot displaying the intensity distribution after the intensity justification.

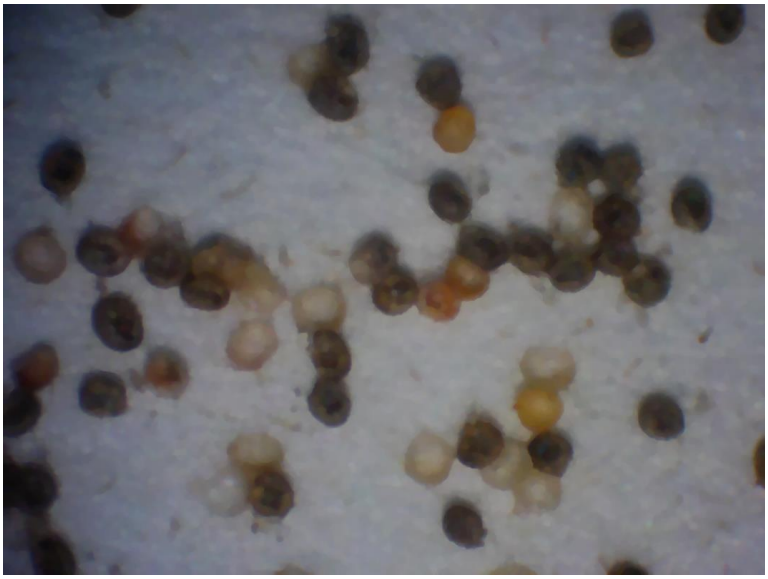


Figure 4.10 Original RGB image after image opening.

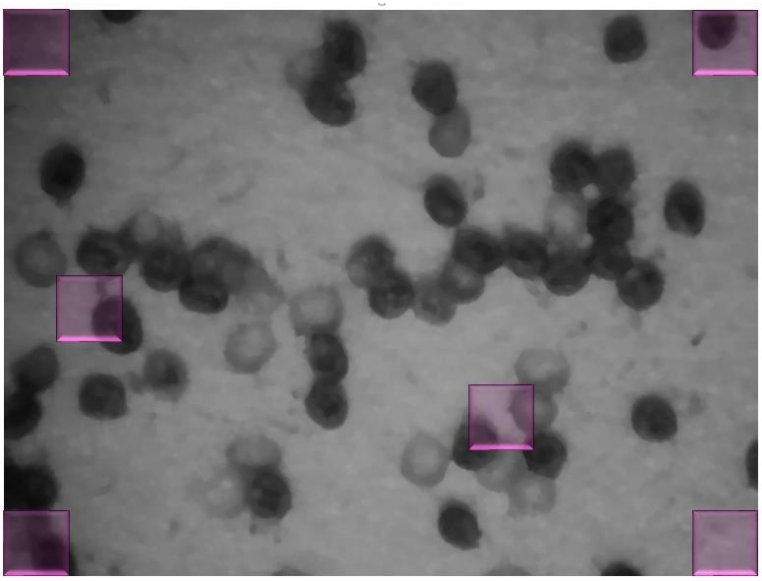


Figure 4.11 Original intensity image with six sample areas for intensity adjustment.

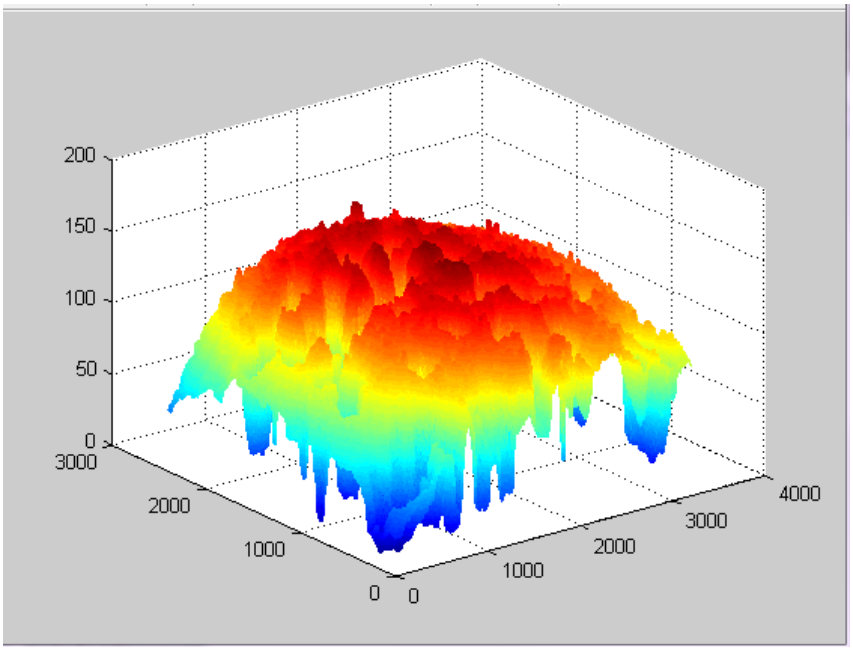


Figure 4.12 3D surface plot of original image intensity.

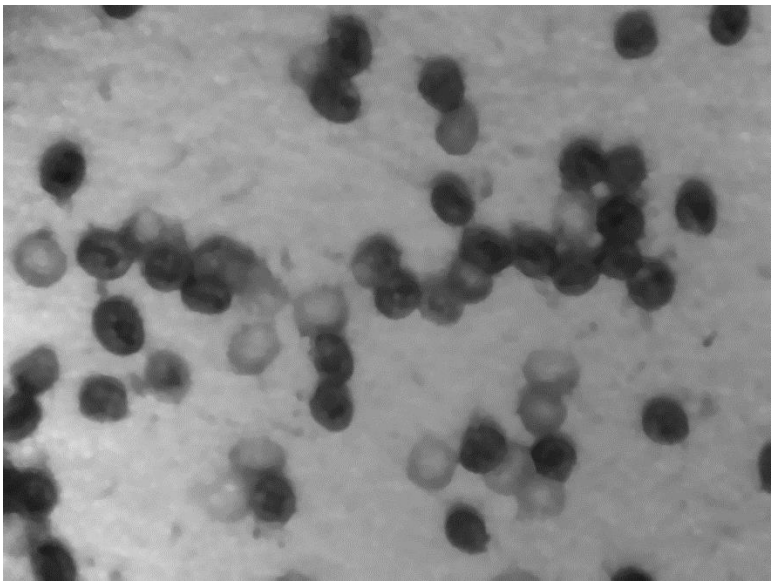


Figure 4.13 Intensity adjusted image.

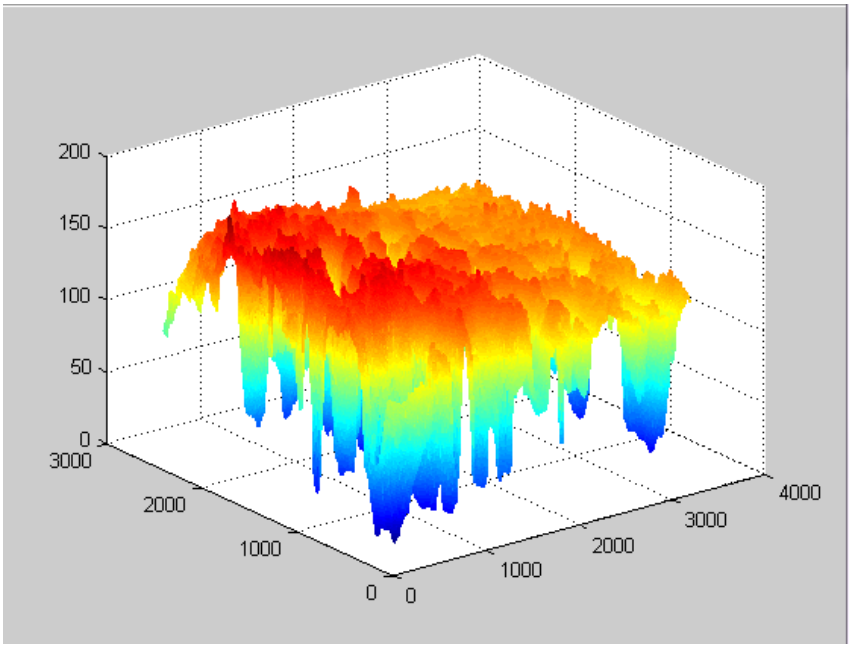


Figure 4.14 3D surface plot of corrected image intensity.

The light-colored cells parts were usually identified by intensity thresholding based on the original color image. The intensity ranges from 0 to 255. A pixel with zero intensity is displayed as black, and a pixel with 255 intensity is displayed as white. Statistical analysis indicated that intensity values of light-colored cells were typically between 80 and 150, thereby eliminating black cell

pixels, dark red cell pixels, dark yellow cell pixels, and bright background pixels. Furthermore, white cells on RGB images were rich in red color, and the background has a strong blue component. Therefore, the difference between red and blue and between red and green frames were used.

$$I_{RG} = R - G$$

$$I_{RB} = R - B$$

After intensity was adjusted, the original color image was binarized via thresholding on colors (I_{RG} and I_{RB}) and adjusted intensity. The returning binary image contained only light-colored cell pixels. Figure 4.15 shows the binary image after thresholding.

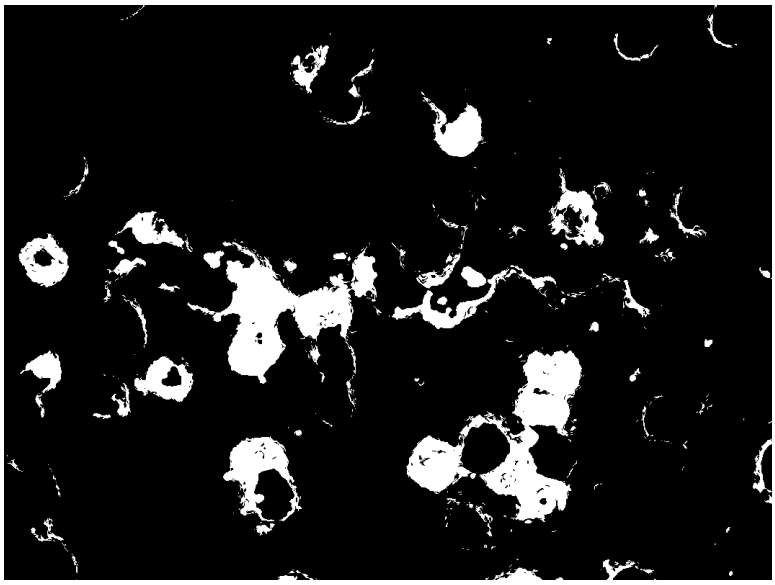


Figure 4.15 Original binary image with light-colored cells.

The thresholding operation created holes in the image (Figure 4.15). The built-in function *imfill* in Matlab was used to fill the holes, as shown in Figure 4.16. This operation was followed by a morphological opening operation (function *imopen* in Matlab) with a disk-shaped structuring element to eliminate discrete small noise parts in the binary image. These procedures separated

light-colored cells from the image background and created smooth boundaries of light-colored cells, as shown in Figure 4.17.

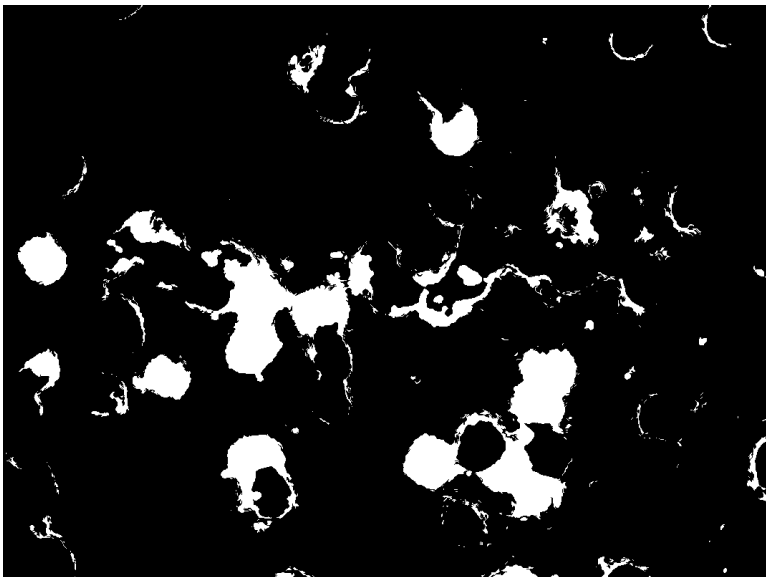


Figure 4.16 Binary image after filling holes.

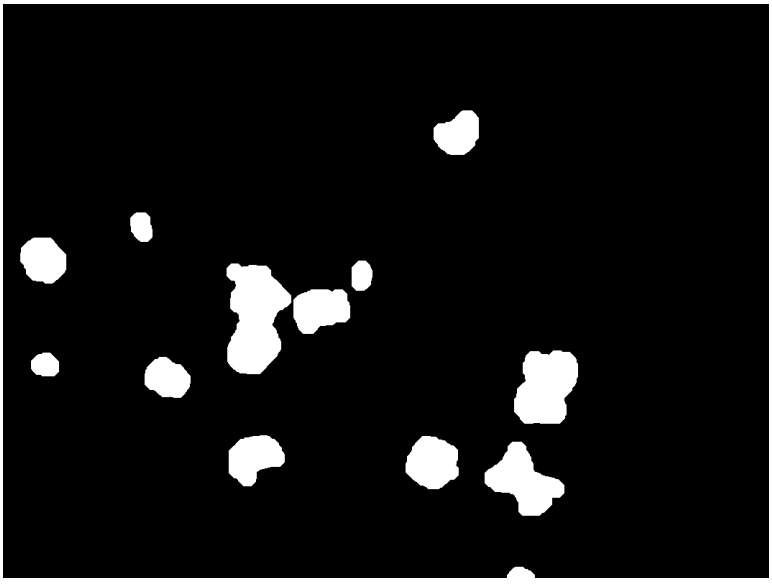


Figure 4.17 Light-colored cells detected via intensity correction, thresholding, and morphological filtering.

Following preprocessing, CHT was performed on the resulting binary image to identify circular objects. The parameters used in the function call included the cell diameter range of 70 - 140

pixels, ‘ObjectPolarity’ of ‘bright’, and sensitivity of 0.985. This high sensitivity value was used because the binary image contained a minimum amount of noise. Thus, the risk of false detection was not significant. However, the high sensitivity value increased the risk of potentially detecting one circular object several times. To avoid this type of error, center distances between detected circular objects were calculated. If the distance between the centers of two detected circular objects was shorter than the minimum cell diameter, one of the two objects was deleted. Deleted cells were removed from the array of detected cells. Figure 4.18 shows detected circular objects after applying HT on the resulting binary image. Figure 4.19 shows remaining circular objects after the duplicate cells were removed. Figure 4.20 overlaps the detected circular objects on the original RGB image.

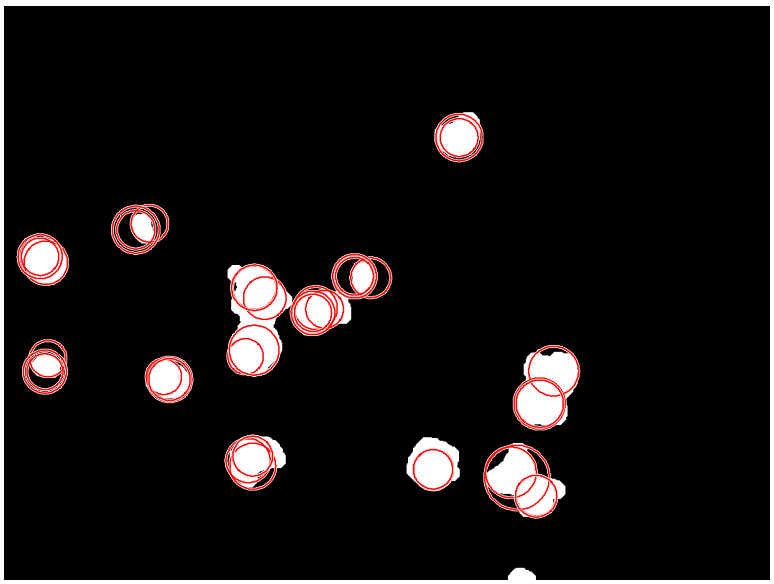


Figure 4.18 Detected circular objects after HT.

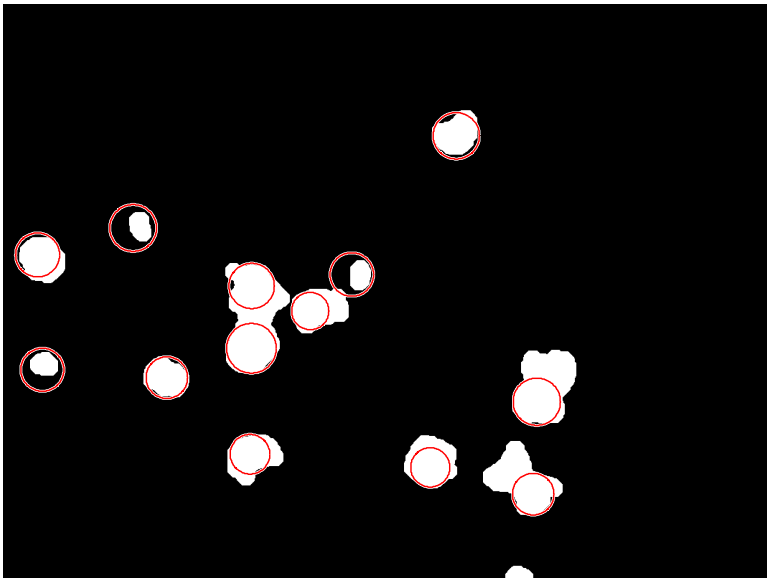


Figure 4.19 Detected circular objects after HT and after removing duplicated cells.

Figure 4.20 Detected circular objects on original RGB image.

Combining, classifying and counting detected cells

After the centers of dark- and light- colored cells were detected, they were merged into a single image (Figure 4.21). Distances between the cells were then checked. If the distance between two adjacent cells was shorter than the minimum cell diameter, one of the cells was deleted. Deleted

cells were removed from the array of detected cells (Figure 4.22), and all remaining cells were subjected to a classification procedure based on their adjusted intensity and color attributes.

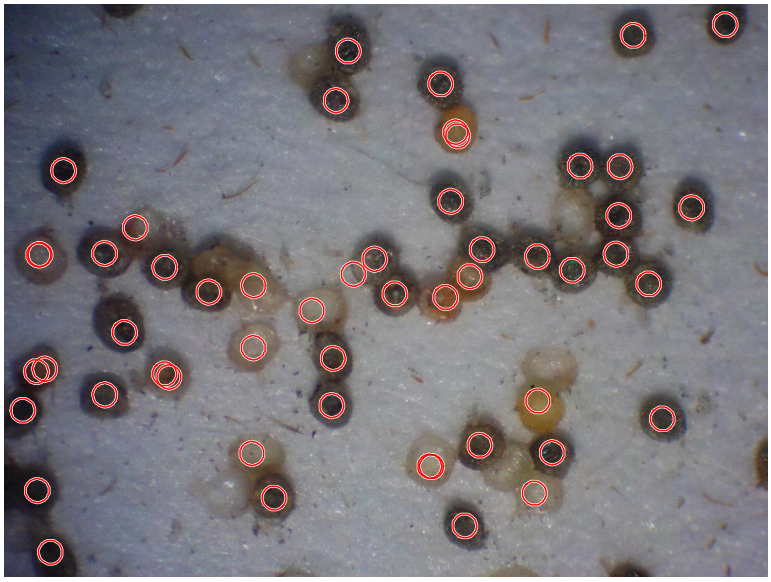


Figure 4.21 Combination of light and dark-colored circle detection results.

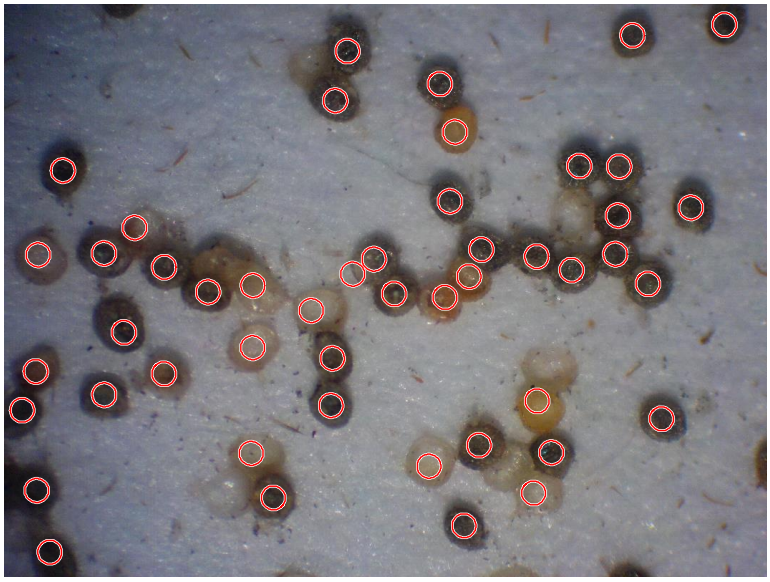


Figure 4.22 Removing duplicated cells.

After observing RGB values of cell center color, adjusted intensity (I_C) and I_{RG} were sufficient to classify cell categories. The red and blue values and adjusted intensity values of cell centers were

found from the original RGB and the corrected intensity images. Counts for all categories were updated based on classification results. To visualize the classification results, all detected eggs were plotted in their respective colors in an image with white background, as shown in Figure 4.23. In general, the image processing procedure correctly detected and classified most eggs.

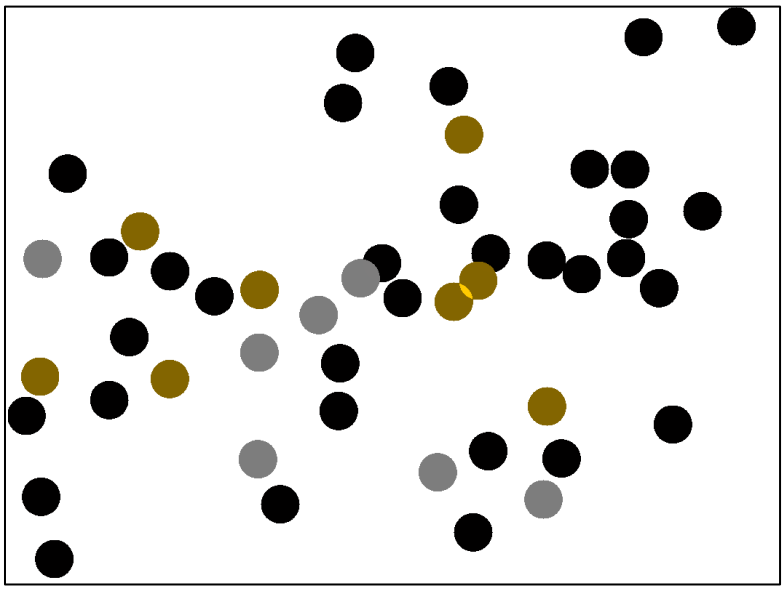


Figure 4.23 Dark-colored (black/red), yellow, and white eggs detected by the HT method.

The developed Hough transform Matlab program was tested on another image (Figure 4.24). Figure 4.25 and 4.26 give the detection results of dark and light-colored cells. Figure 4.27 displays all detected cells after they were combined. Figure 4.28 shows the detected eggs plotted in their respective colors on a white background.

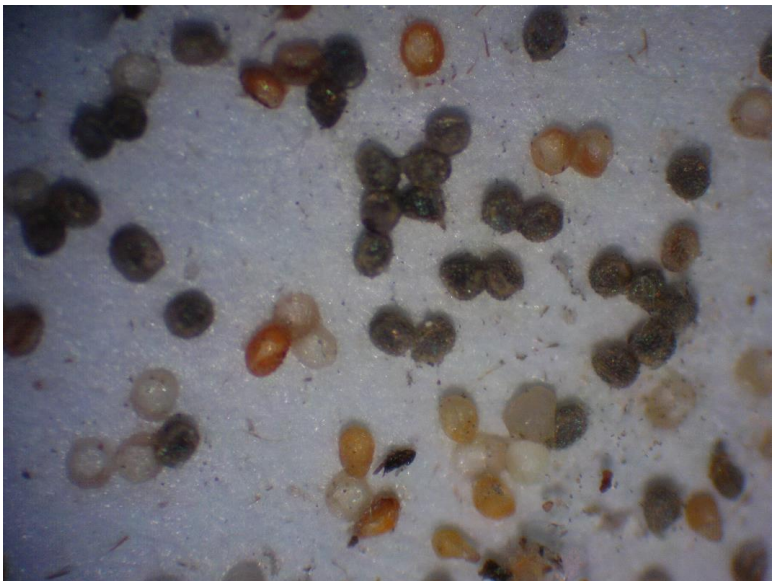


Figure 4.24 The original cell image.

Figure 4.25 Dark-colored cells detected using HT.

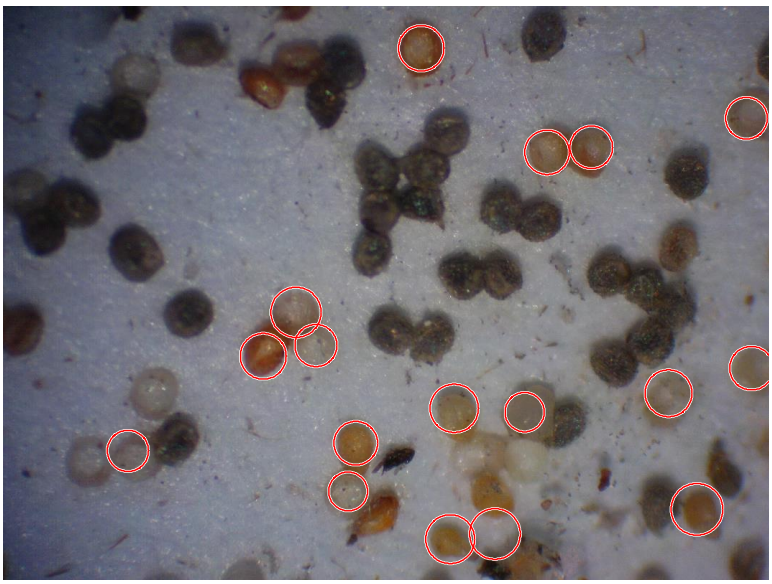


Figure 4.26 Detected light-colored cells.

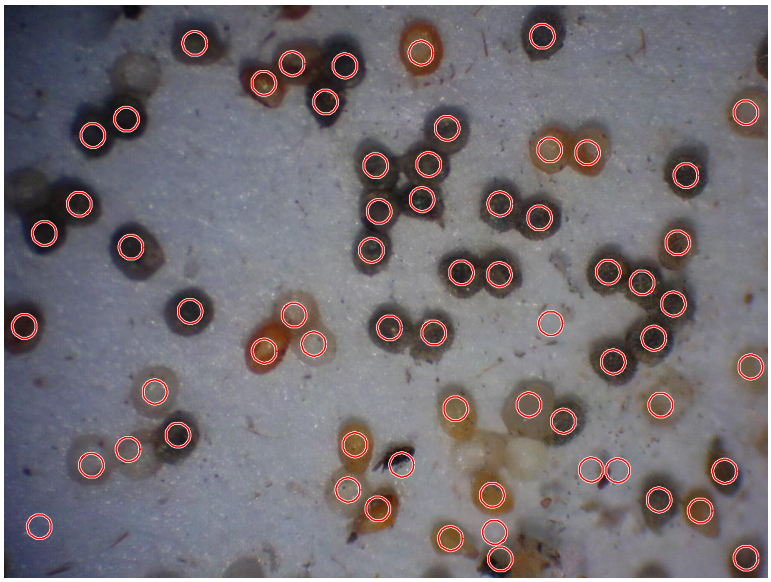


Figure 4.27 Detected cells.

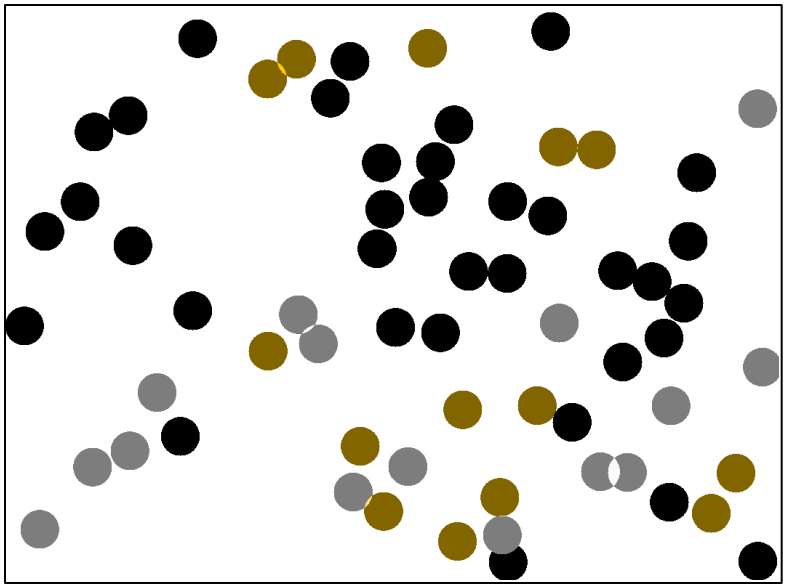


Figure 4.28 Black/red, yellow, and white eggs detected by the HT program.

Chapter 5 - The Watershed Transformation Method

In 2014, Dr. Xin Pan worked on this project using the Watershed Transformation method in image processing and wrote the Matlab program in several sections. (Pan and Zhang, 2014). The author integrated these sections and made major modifications to improve the program. This chapter describes algorithms used in the integrated and improved program.

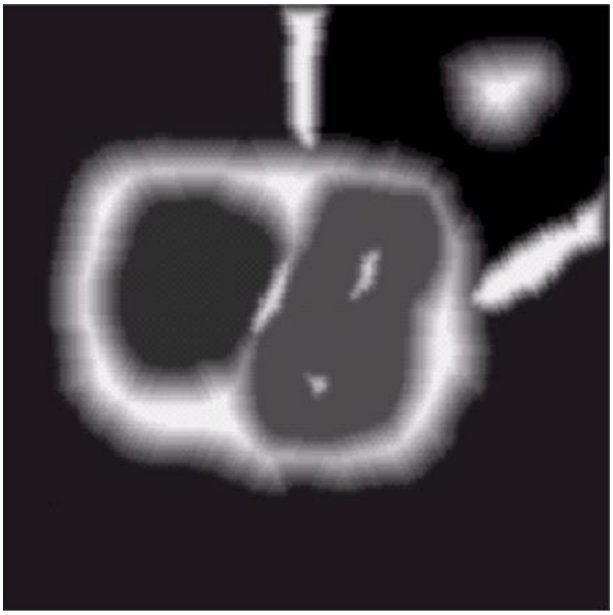
The integrated program includes three major steps: finding black/red cells, finding yellow cells, and finding white cells.

General principle of Watershed Method

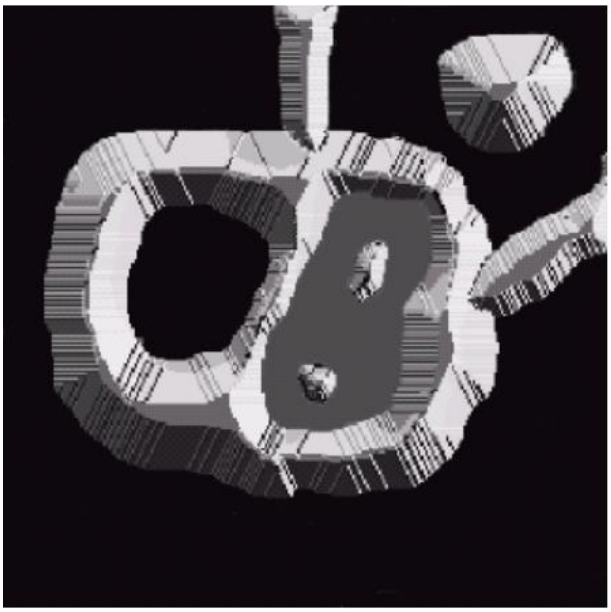
When a simple thresholding method (such as edge detection, thresholding, or region growing) is insufficient for segmentation of connected cells, extended methods, such as the watershed segmentation method, should be explored. The basic idea of the watershed segmentation method is simple, as the following analogy illustrates. Suppose that a hole is punched at the point with the minimum elevation within a catchment basin, then the entire topography in the basin would be gradually flooded from below when water rises through the holes. If dams are built to prevent the merging of rising waters from adjacent catchment basins, the flood will eventually reach a stage when only the tops of the dams are visible above the water. These visible tops can be considered the watershed divide lines. (Gonzalez and Woods, 2002).

This concept can be more clearly demonstrated through an example shown in Figure 5.1. Figure 5.1 (a) is an original grayscale image, for which we want to delineate the regions based on the

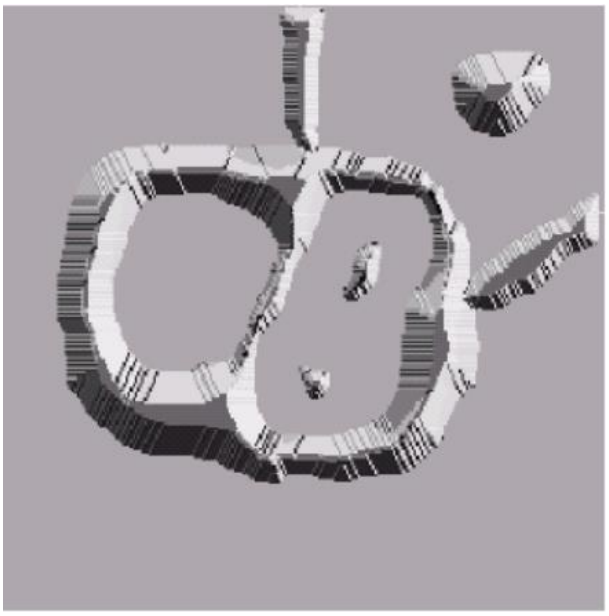
"watershed" boundaries, Figure 5.1 (b) is the topographic expression of the image, with higher intensities representing higher elevations. Obviously, two "catchment basins" can be seen in the image and they are enclosed by dams with heights greater than the highest mountains within the basins. Suppose that water is flooded from the bottom of each catchment basin and is risen at a constant rate, as shown as light gray in Figure 5.1(c). As the water continuously rises, it will eventually overflow from the left catchment basin into the right catchment basin (figure 5.1 (d)). At this time, a short dam (consisting of single pixels) is built to prevent water overflowing. As the water further rises, longer dams are built between the two catchment basins and between the basins and outside background to hold the water within the basins, as shown in Figure 5.1 (e). This process will not stop until the water level reaches the maximum height of the entire topography – the maximum intensity within the image. At that time, complete watershed lines at one-pixel width would have been built, hence completing the watershed delineation (Figure 5.1 (f)) (Gonzalez and Woods, 2002).



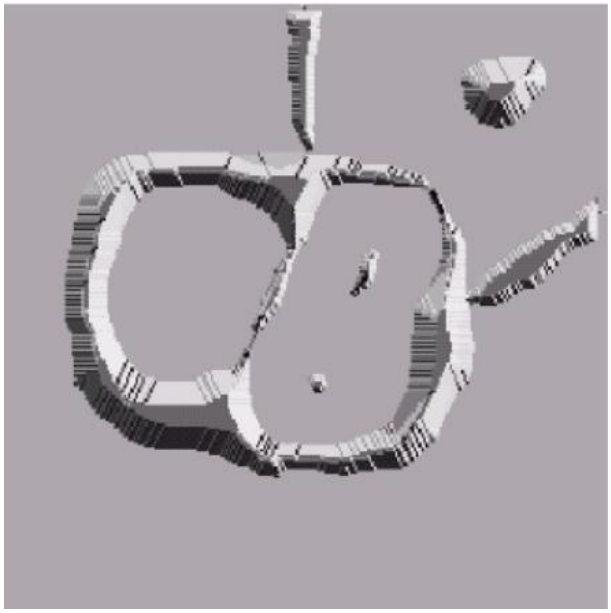
(a) Original grayscale image



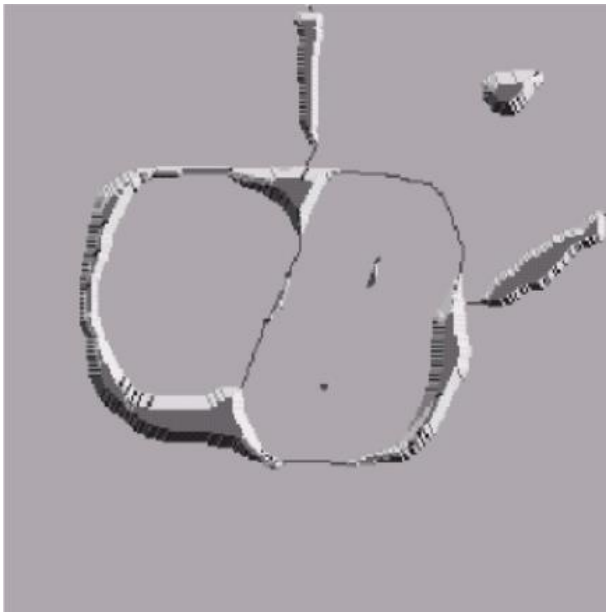
(b) Topographic surface



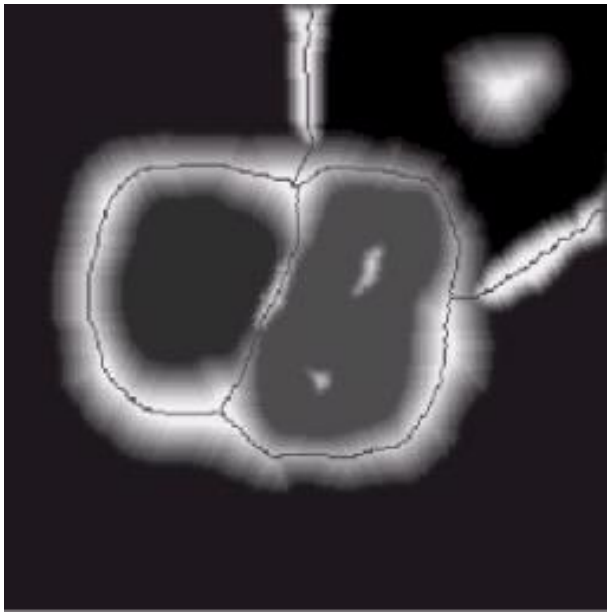
(c) Overflowing before dam built



(d) A short dam built between two catchment basin.



(e) Longer dams



(f) Final watershed lines results.

Figure 5.1 Basic steps of watershed segmentation (Gonzalez & Woods, 2002).

The watershed method

In this study, the watershed method was used to segment eggs of different categories. The overall flowchart of the WT program to detect, classify, and count egg categories is shown in Figure 5.2. The program identified and counted black and dark red cells (parasitized eggs), yellow cells (unparasitized unhatched eggs), and white cells (unparasitized hatched eggs) separately.

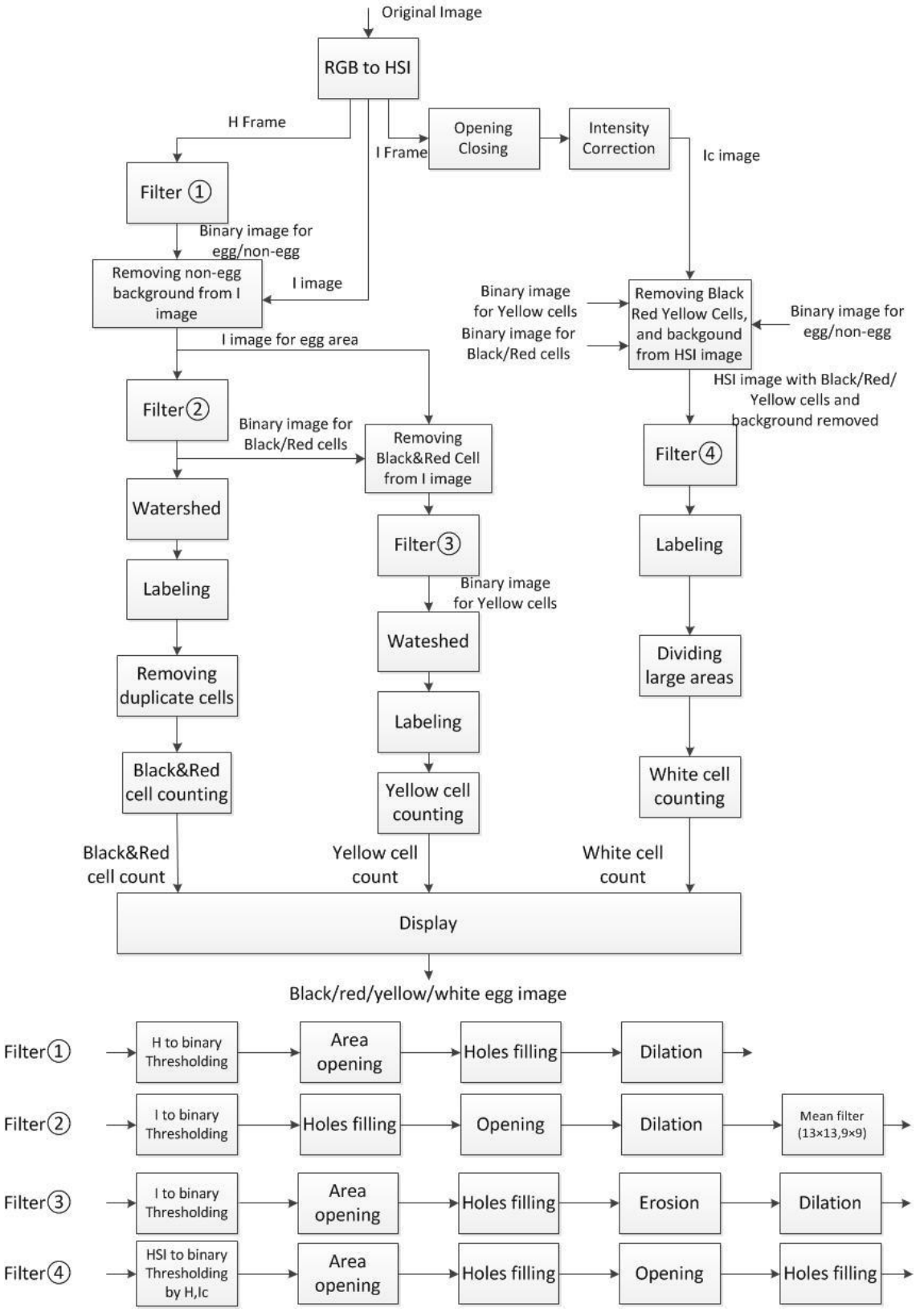


Figure 5.2 Flowchart of the WT algorithm.

Counting black/red (parasitized) eggs

In this study, the image was initially converted from RGB (Red, Green, Blue) to HIS (Hue, Saturation, Intensity) color space in the following steps.

1. Represent the RGB image in the range [0,1]
2. Find HSI components

$$\theta = \cos^{-1}\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)^{1/2}]}\}$$

3. $H(Hue) = \begin{cases}\theta & If\ B \leq G \\ 360-\theta & If\ B > G\end{cases}$
4. $S(Saturation) = 1 - \frac{3}{(R+G+B)}[\min(R,G,B)]$
5. $I(Intensity) = \frac{1}{3}(R+G+B)$

The original RGB image is shown in Figure 5.3. Figure 5.4 and 5.5 shows the Hue and the Intensity in grayscale image.

Figure 5.3 Original RGB image.

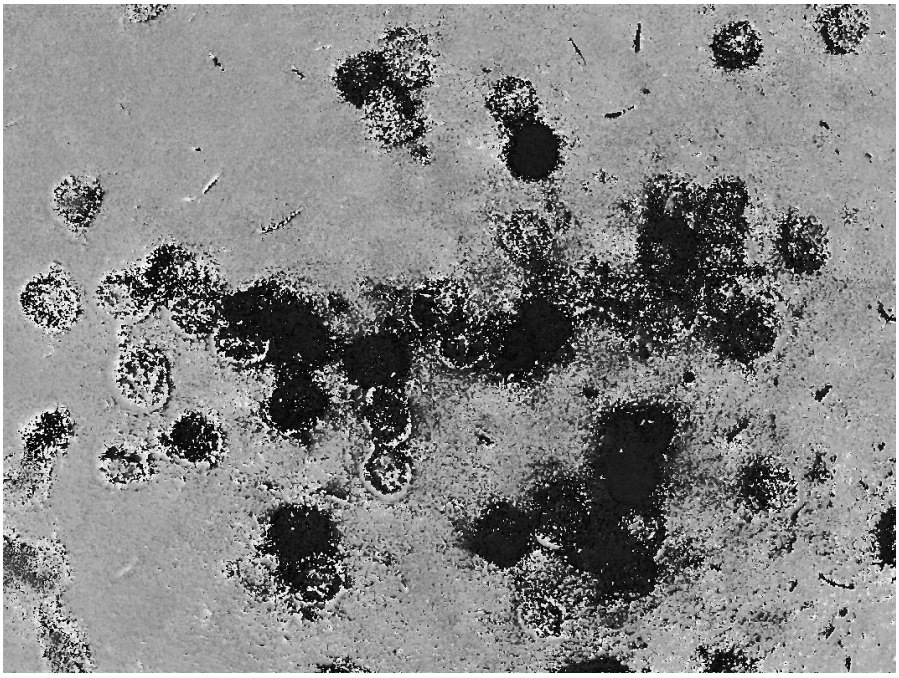


Figure 5.4 The Hue image frame in grayscale.

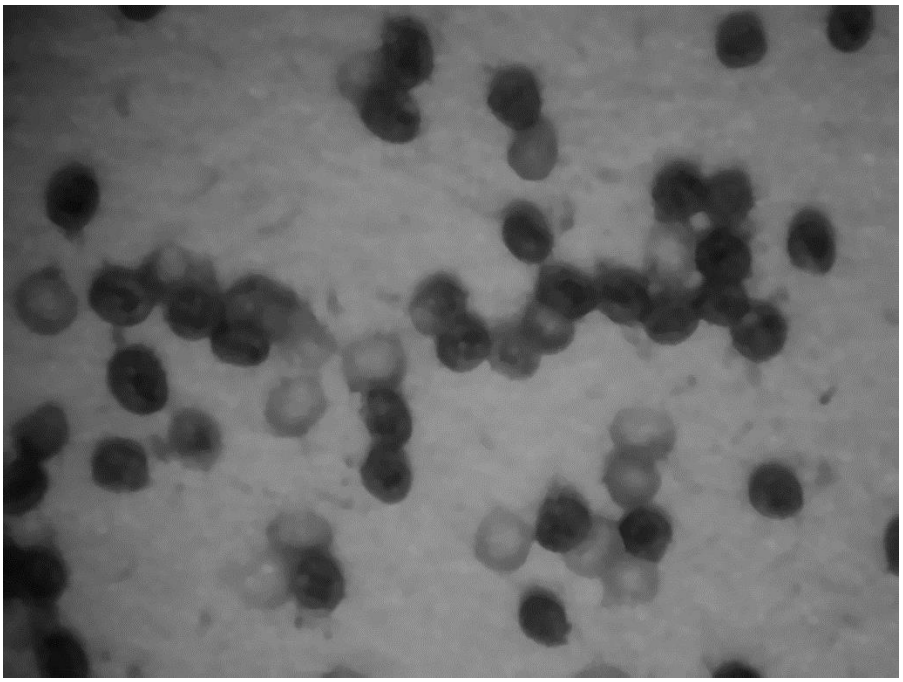


Figure 5.5 The intensity image frame in gray scale.

For dark and yellow cells, the hue image frame was first used to identify the region of interest (ROI) by removing non-egg background from the HSI image. The resulting binary image was then

filtered through “Filter 1”, which contained the following steps: (1) binarizing the hue frame using a threshold value of 0.274. This step eliminated most background areas, as shown in Figure 5.6. (2) Area opening to eliminate isolated noise. (Figure 5.7). (3) Filling holes to restore egg pixels lost during thresholding (Figure 5.8). (4) Morphological dilation using a disk shaped structuring element to further restore lost egg pixels (Figure 5.9). Figure 5.10 shows the intensity frame of the HSI image after most of the background was removed. It can be seen that, while most background was removed, areas containing the cells were retained.

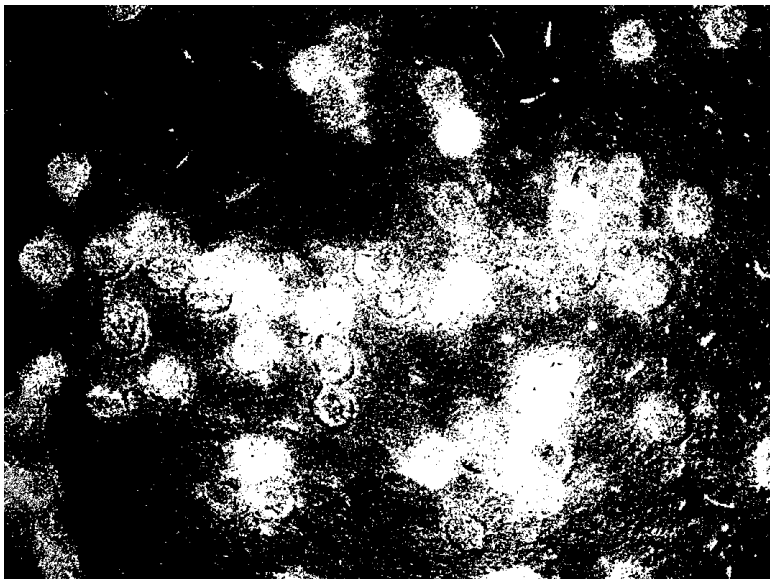


Figure 5.6 Binary image after thresholding on hue.

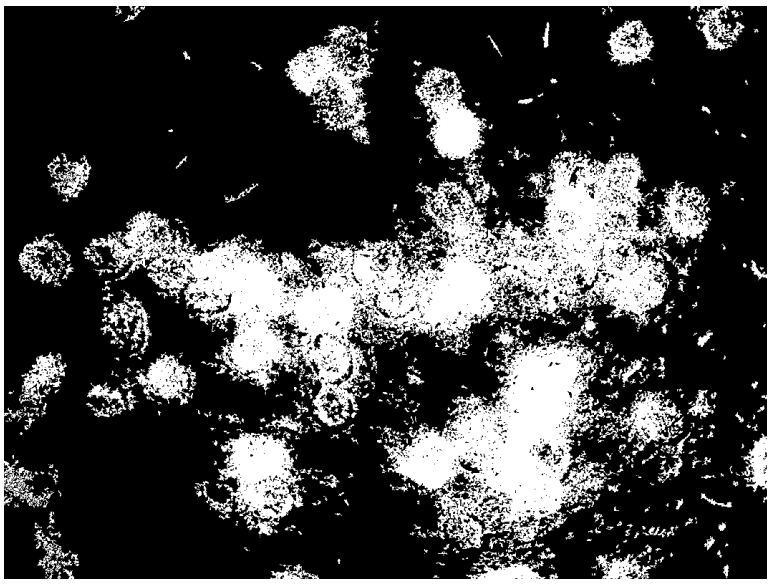


Figure 5.7 Binary image after area opening.

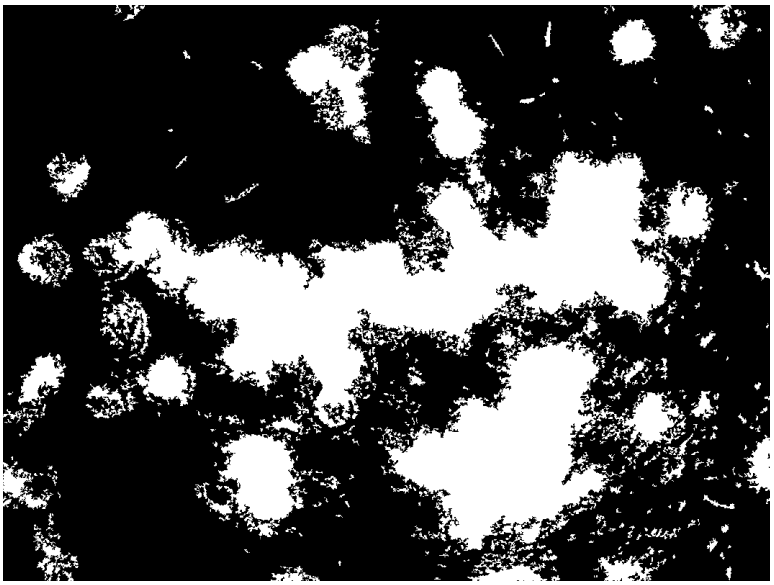


Figure 5.8 Binary image after area opening, hole filling.

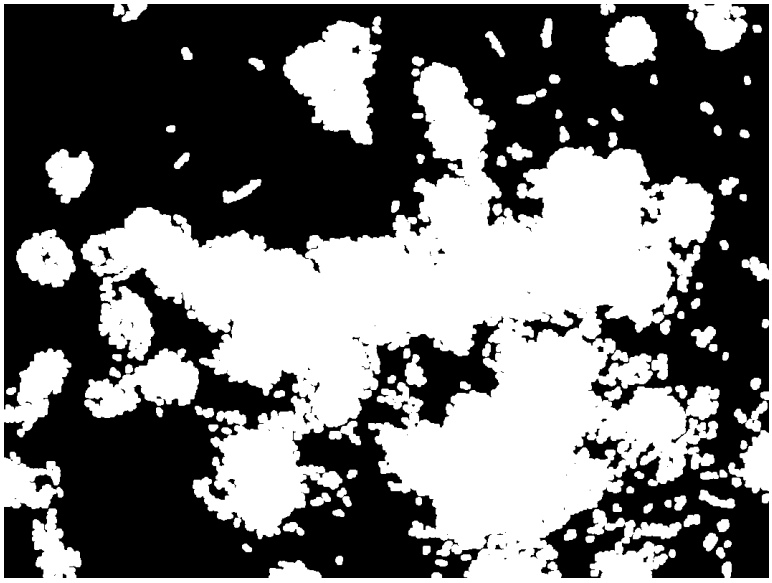


Figure 5.9 Binary image after area opening, hole filling and dilation.

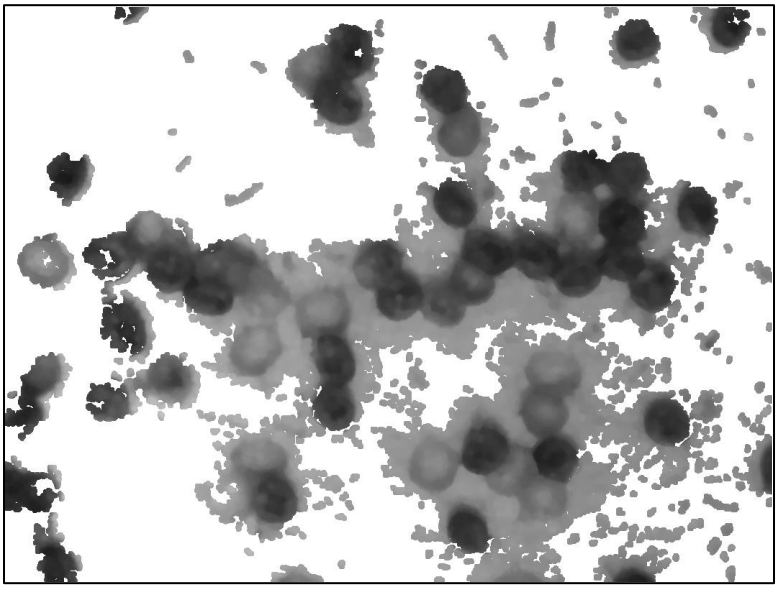


Figure 5.10 Intensity image after background removal.

The background-removed intensity image frame was then further processed using “Filter 2”. As the first step, yellow/white cells were removed through thresholding. The threshold used was 0.275 on a 0-1 scale. The remaining steps in “Filter 2” included (1) hole filling using the Matlab function of *imfill* to recover black/red cell pixels lost during the thresholding operation, (2) morphological opening to remove discrete noise, and (3) morphological dilation with a disk-shaped structuring

element. Figure 5.11 displays black/dark red cells obtained through thresholding on background-removed intensity image frame. Figure 5.12 shows the resulting binary image for the black/red cells that is ready for the watershed analysis.

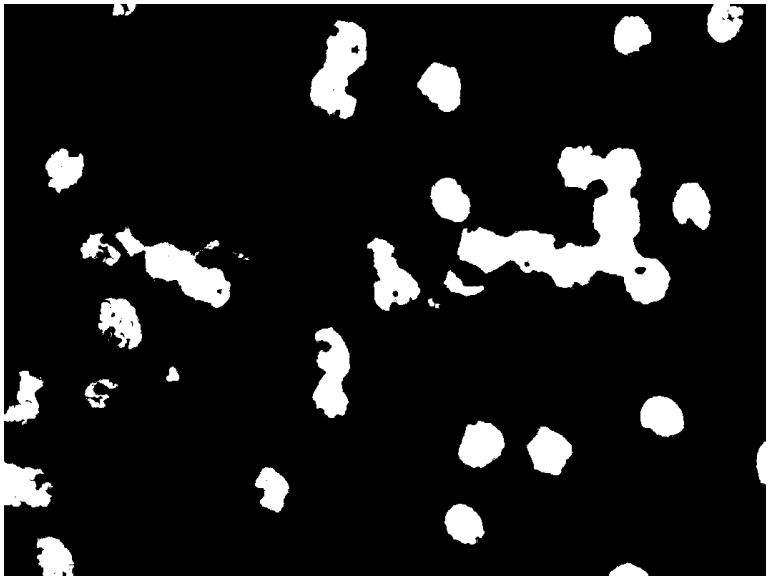


Figure 5.11 Black/dark red cells obtained through thresholding on background-removed intensity image frame.

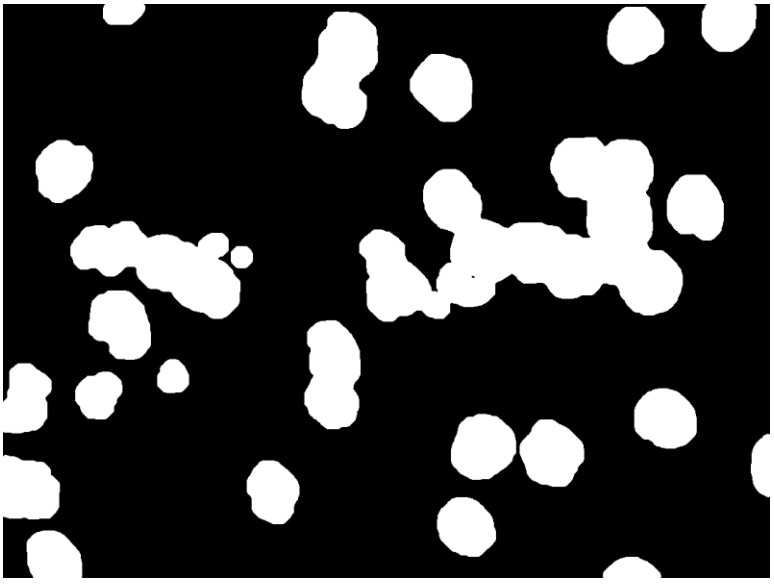


Figure 5.12 Binary image for black/red cells after thresholding and morphological filtering.

The first step of the watershed delineation was to convert the binary image for black/red cells into a distance image, in which each pixel was assigned a value that represented its distance to the nearest background pixel. The values were then scaled and inverted to gray scales, with 255 representing zero distance and 0 representing the longest distance detected, as shown in Figure 5.13. The watershed delineation was then applied to this distance image using the Matlab built-in function of *watershed* (MATLAB, 2016). The resulting watershed boundaries are shown in Figure 5.14. These boundary lines were then applied to the binary image for black/red cells to isolate the cells (Figure 5.15). A labeling procedure using Matlab function *regionprops* was then performed on the image to number the "watersheds" and measure their properties – area and center position. To remove duplicate "watersheds", distances between the "watersheds" were checked. If the distance between two adjacent "watersheds" was less than the minimum radius of the eggs, the two "watersheds" were combined and the array of the "watersheds" was updated accordingly. The final result of the labeling operation can be displayed with a useful visualization technique using a color scheme, as shown in Figure 5.16. The centers of the resulting "watersheds" were identified as the base for counting. Distances between the resulting cells were then checked. If the distance between two adjacent cells was less than the radius of the kernel, one of the cells was deleted. Deleted cells were then removed from the array of resulting objects. Figure 5.17 shows detected parasitized eggs with marked centers. This included eggs on the edge of the image with only a part visible and eggs touching each other in the image.

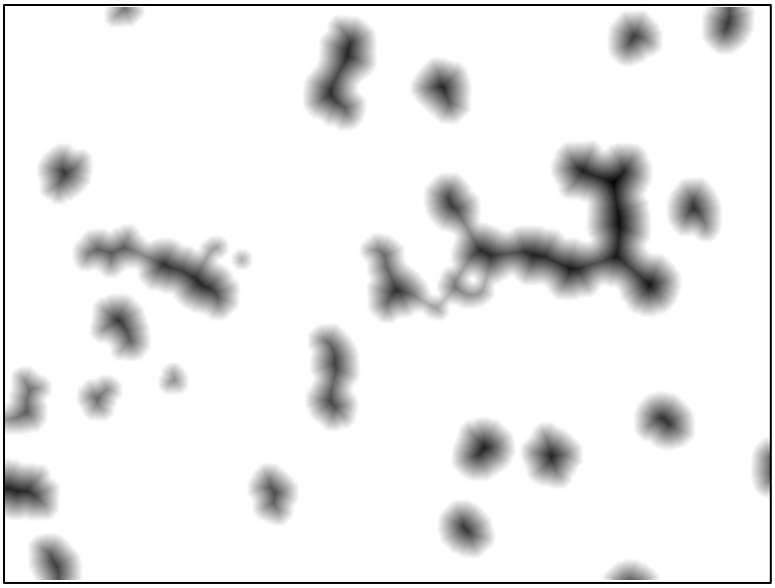


Figure 5.13 Distance transform of the complement of the binary image.

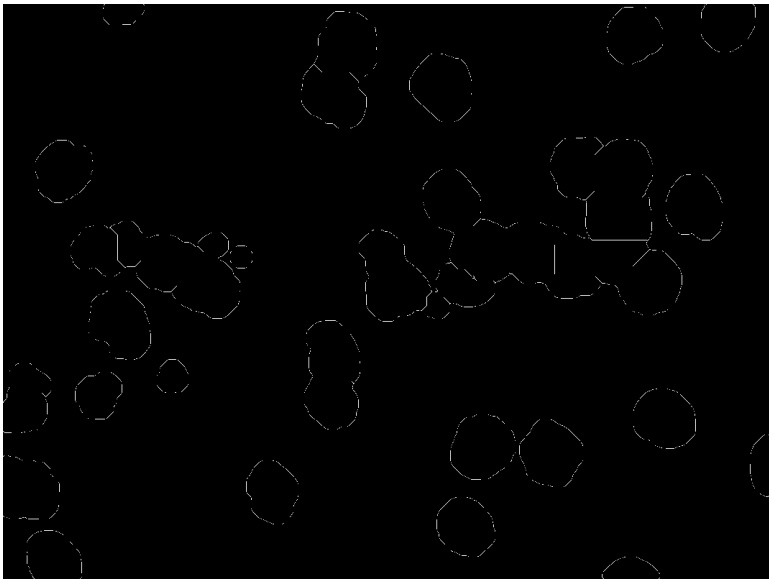


Figure 5.14 Watershed lines.

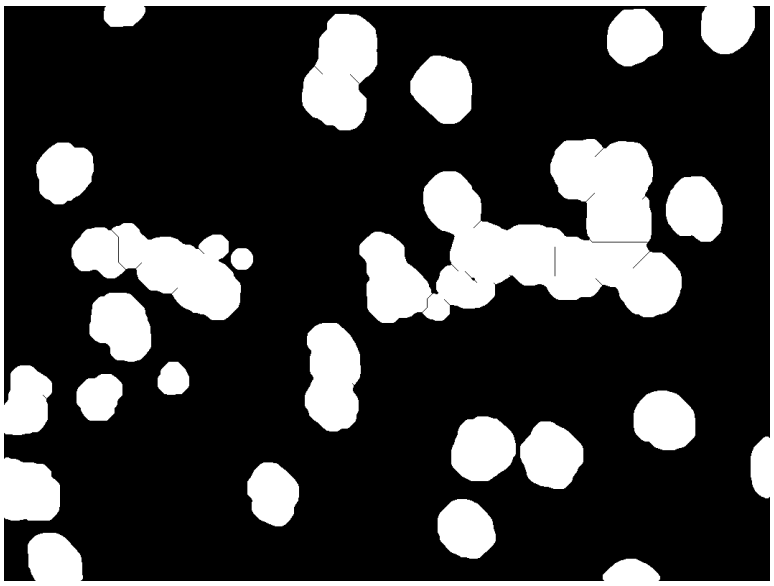


Figure 5.15 Binary image segmented by watershed lines.

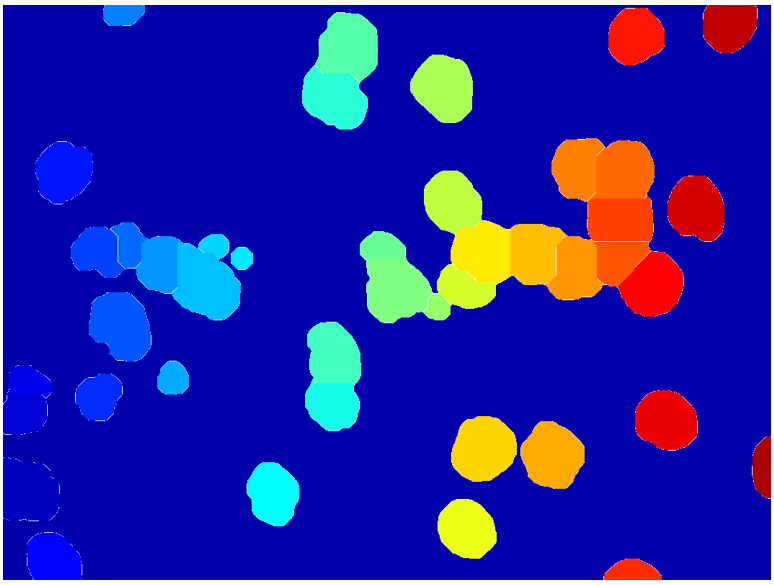


Figure 5.16 Displaying the labeled areas using a color scheme.

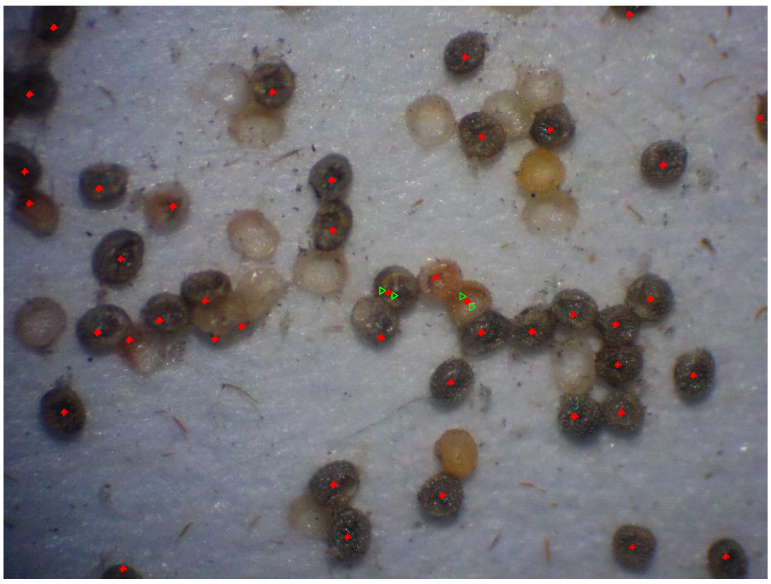


Figure 5.17 Black/red cells with centers marked via watershed delineation, labeling, and distance analysis.

Counting yellow eggs (unparasitized unhatched)

After the black/red eggs were identified, they were removed from the background-removed binary image (Figure 5.10), leaving only yellow cells, white cells, and pixels with similar colors (figure 5.18). The resulting image was then filtered using "Filter 3" to narrow down possible areas that may contain yellow cells. "Filter 3" contains the following steps: (1) Binarization by intensity. Pixels with intensity values between 70 and 100 were considered potential yellow cell pixels. Figure 5.19 shows the resulting binary image. (2) Area opening to eliminate isolated noise. . (3) Hole filling to recover pixels lost during thresholding operation. (4) Erosion and dilation with a disk-shaped element.

After "Filter 3", watershed delineation and labeling operations were performed on the image. Figure 5.20 displays the three "watersheds" that represent the yellow cells. Yellow eggs were

finally identified by examining the areas of labeled regions and distances between the centers of potential cells (Figure 5.21).

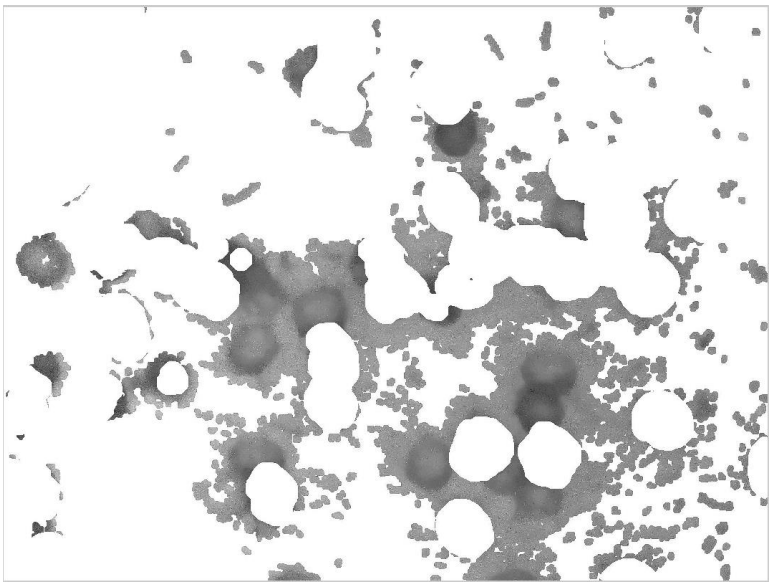


Figure 5.18 Intensity image after black/red cells and background pixels removed.

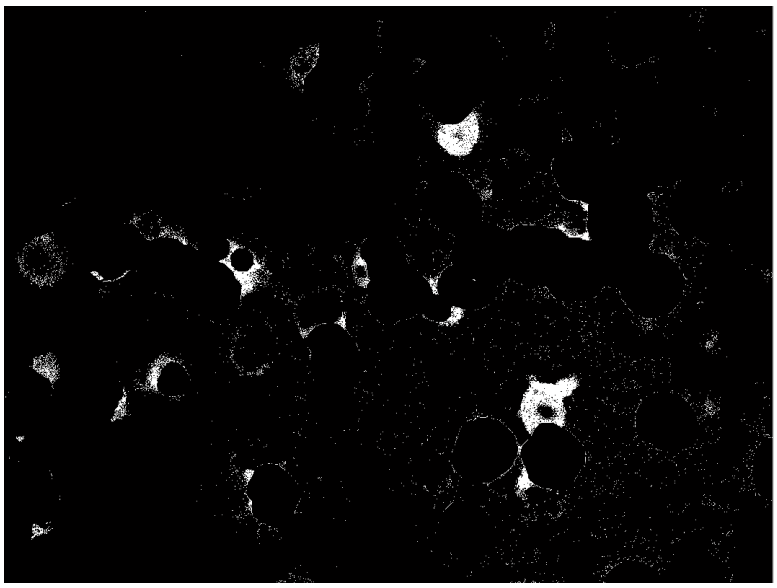


Figure 5.19 Original binary image for yellow cells.

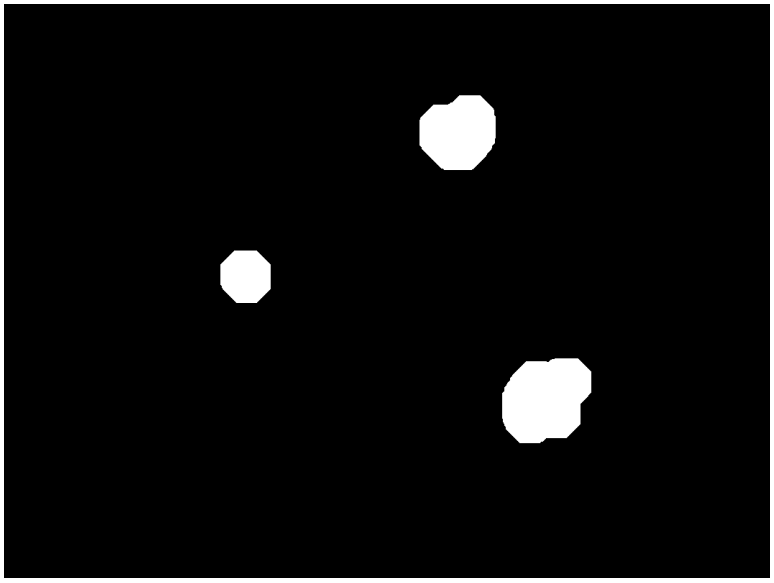


Figure 5.20 Potential yellow cell areas segmented from the image.

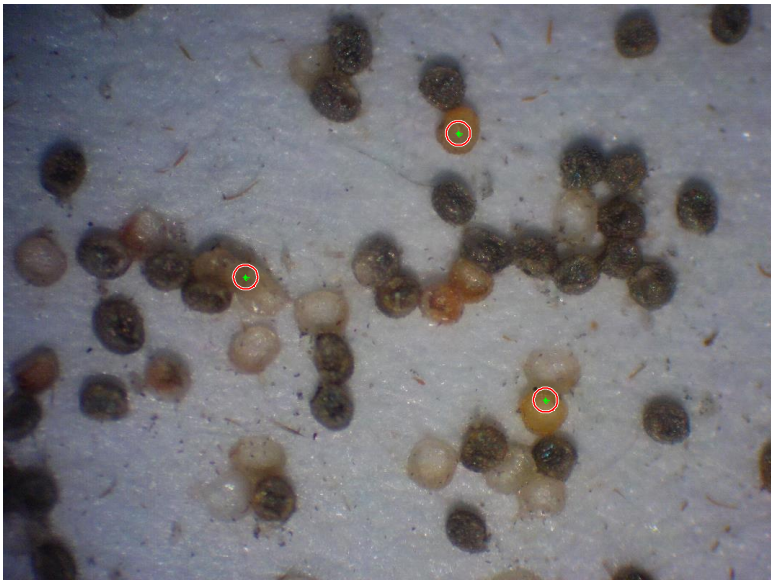


Figure 5.21 Yellow eggs extracted via watershed delineation, labeling, and area/distance analysis.

Counting white eggs (unparasitized hatched)

Among the three egg categories, the white eggs (unparasitized hatched eggs) were the most difficult objects to identify. This was mainly due to the similarity in color and intensity between white eggs and the background and the uneven illumination within the image. To solve this

problem, image intensity was adjusted using a second-order polynomial surface model as described in the section on detecting light-colored cells using the HT method. After the intensity adjustment, black/red cells, yellow cells, and background pixels detected in previous steps were removed from the adjusted intensity image, leaving only white cells and pixels with similar colors (Figure 5.22). Pixels representing the white cells were identified through thresholding using H (hue) and adjusted intensity. White eggs were finally identified after area opening, hole filling, and morphological opening, as shown in Figure 5.2 as "Filter 4". A labeling analysis returned centers of white eggs and areas of labeled region. Large labeled regions were divided into multiple eggs based on the area of the region. After a statistical analysis, we noticed that a single egg area usually contained 14,000 - 40,000 pixels. Thus, a labeled area of less than 14,000 pixels was considered no white cell and was eliminated. A labeled area with 14,000 - 40,000 pixels was considered as one white cell, whereas a labeled area with more than 40,000 pixels was assumed to contain more than one white cell. In this case, the number of pixels in the labeled are was divided by 40,000, and the quotient was then rounded to the nearest integer as the detected number of white cells in the region. Figure 5.23 shows the numbers of detected white eggs in each of the labeled region.

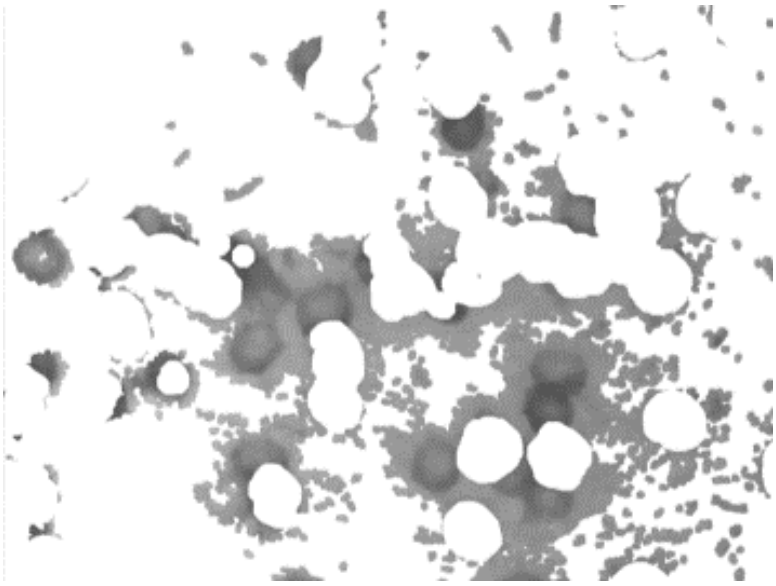


Figure 5.22 Adjusted intensity image after removal of black/red and yellow cells and background pixels.

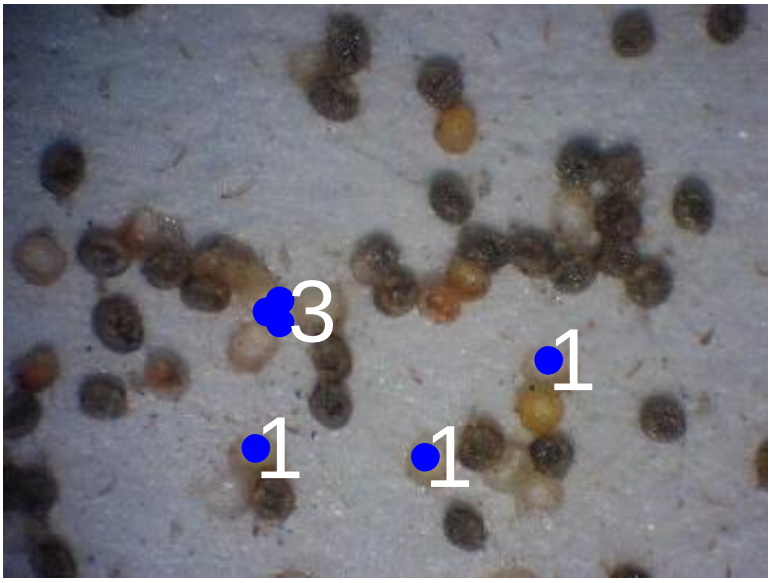


Figure 5.23 White eggs detected after labeling.

Figures 5.24 compares the original image with the complete classification results using the watershed method with detected black/red, yellow, white eggs, and the background shown in black, brown, gray, and white colors

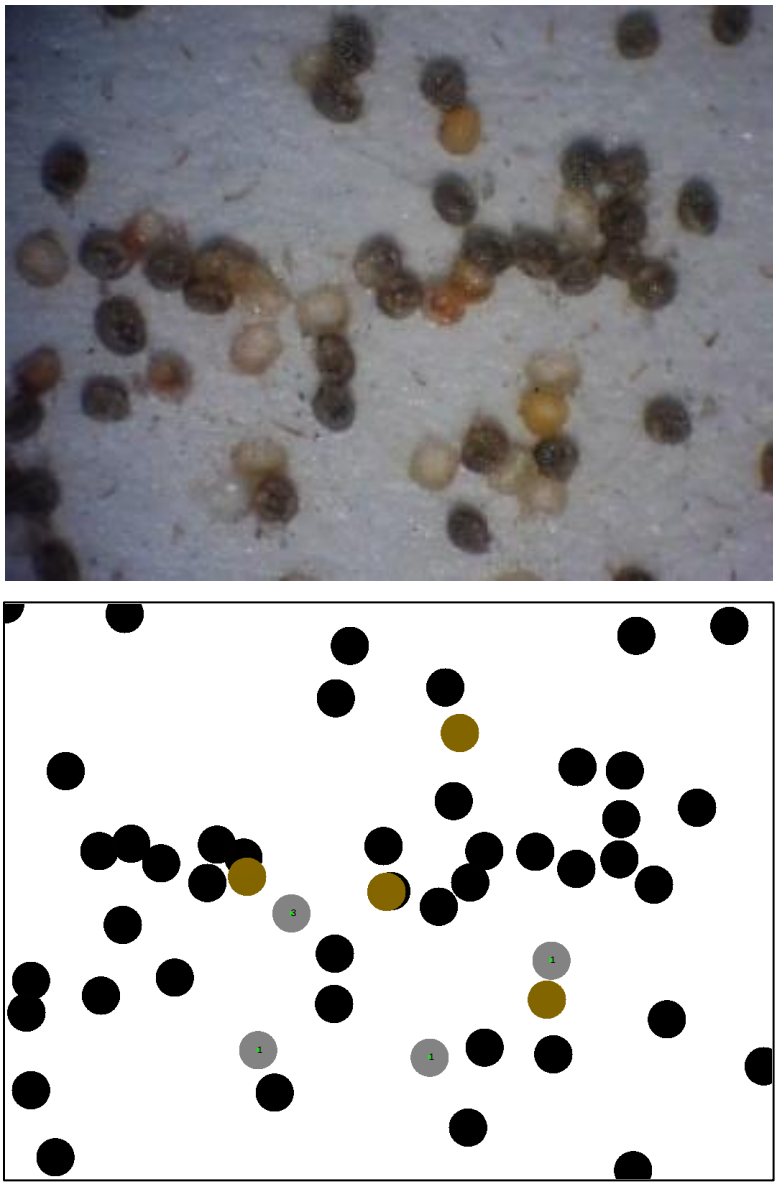


Figure 5.24 Comparison between the original image and the final classification result by the WT method.

The Watershed method was tested on another image. The results of major steps are shown in following figures: Figure 5.25 shows the original RGB image. Figure 5.26, 5.27 and 5.28 give the detection results for black/red, yellow, and white eggs. Figure 5.29 shows the final results with detected black/red, yellow, white eggs, and the background shown in black, brown, gray, and white colors, respectively

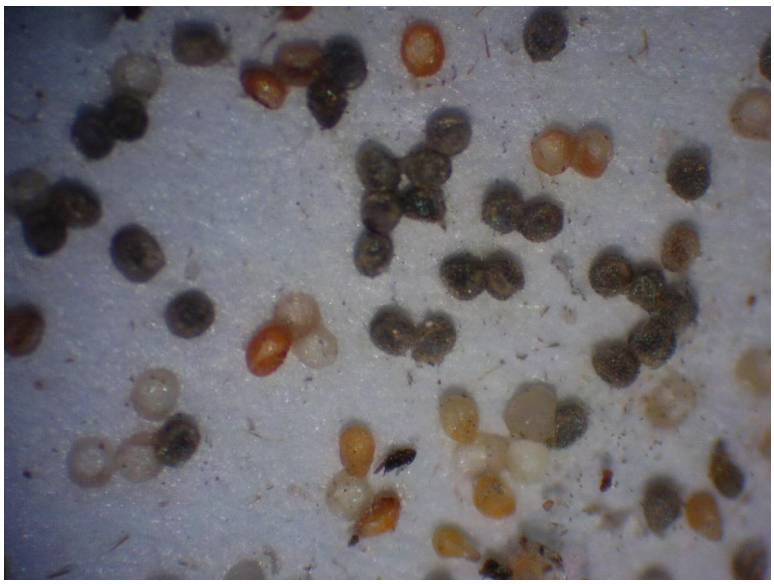


Figure 5.25 The original RGB image.

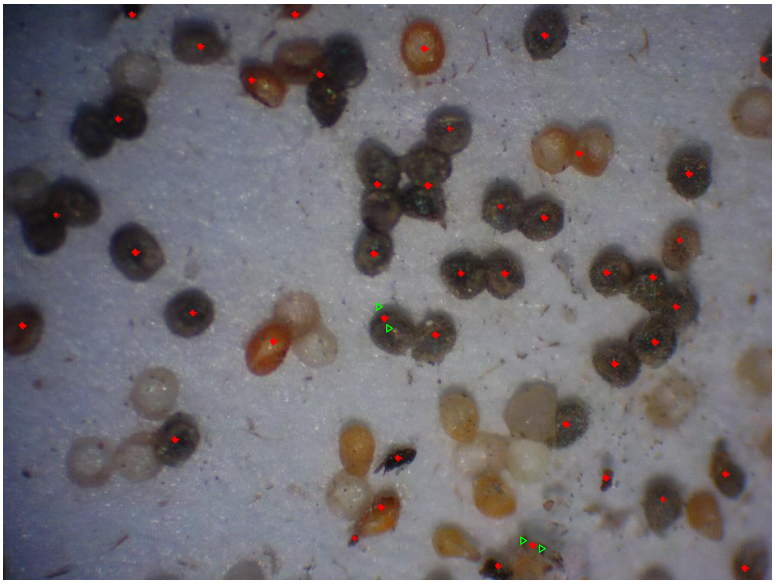


Figure 5.26 Black/red cells with centers marked through watershed delineation, labeling, and distance analysis.

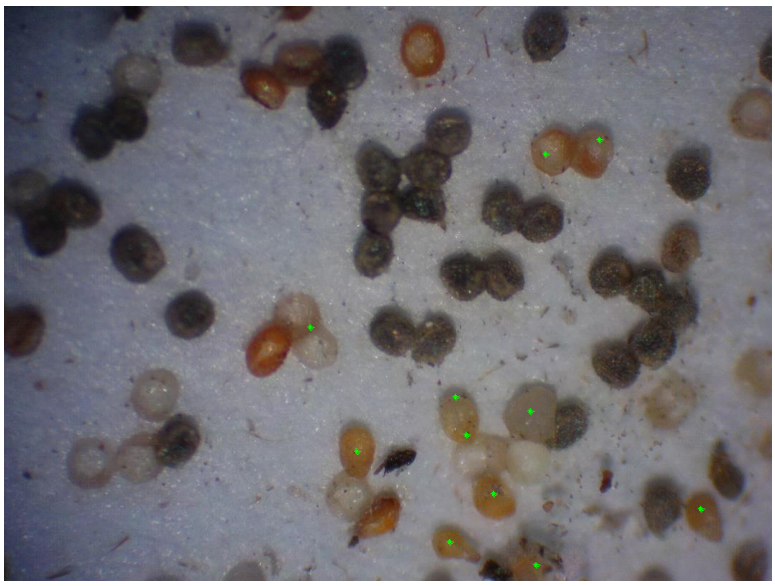


Figure 5.27 Yellow eggs extracted through watershed delineation, labeling, and area/distance analysis.

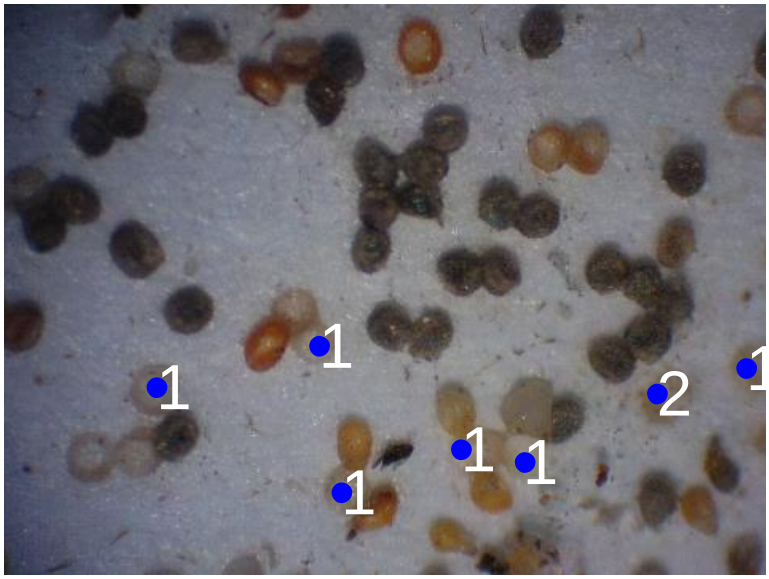


Figure 5.28 White eggs detected after labeling.

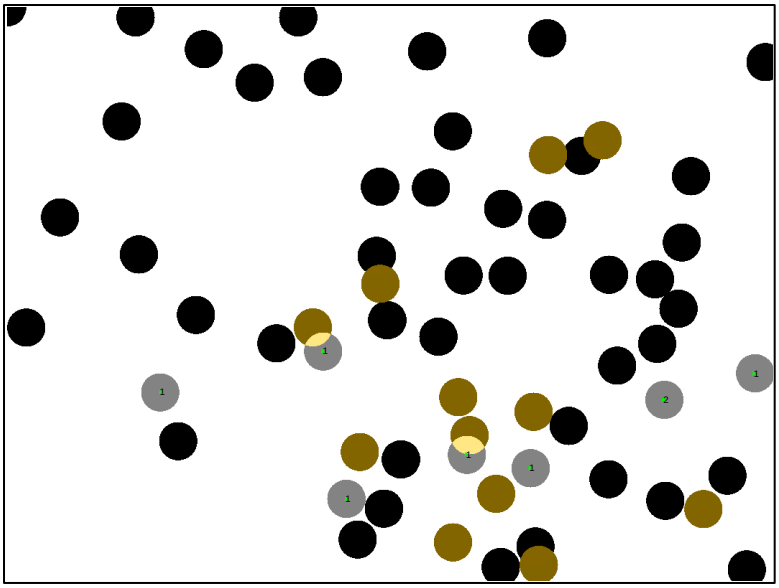


Figure 5.29 Dark-colored (black/red), yellow, and white eggs detected in another image by WT method.

Chapter 6 - Results and Discussion

The performances of Hough transformation and watershed delineation methods are evaluated and compared in this chapter.

Performance comparison between WT and HT

Because visual counting can be tedious, laborious, inaccurate, and time-consuming, this study developed algorithms to automatically count eggs in different categories for evaluation of the effectiveness of biological control. To evaluate the performance of these algorithms, correct classification rates (CCRs) for individual egg categories and misclassification rates (MCRs) were calculated.

Three classification rates are defined as (1) percentage of black/red eggs classified by the program in total number of actual black/red eggs in the image, (2) percentage of yellow eggs classified by the program in total number of actual yellow eggs in the image, and (3) percentage of white eggs classified by the program in total number of actual white eggs in the image. Accompanying these classification rates were misclassification rates across different egg categories. The actual number of three types of eggs were manually counted by the author in each image – an extremely exhausting and tedious endeavor!

Four additional types of misclassification rates were defined. They are (1) percentage of black/red eggs misclassified from background in total number of black/red eggs detected, (2) percentage of yellow eggs misclassified from background in total number of yellow eggs detected, (3) percentage

of white eggs misclassified from background in total number of white eggs detected, and (4) percentage of all eggs misclassified from background in total number of eggs detected (Choi, Lee, and Ehsani, 2013).

The HT and WT algorithms were tested in 40 images. The classification and misclassification rates are summarized in Tables 6-1 and 6-2.

Table 6-1 Classification and misclassification rates of *Cadra* (=*Ephestia*) eggs by the HT algorithm.

Actual eggs	Classified to								Total	
	Black		Yellow		White		Background			
Black	1251	96.0%	23	1.8%	1	0.0%	28	2.1%	1303	100%
Yellow	3	0.7%	395	93.6%	13	3.1%	11	2.6%	422	100%
White	1	0.2%	15	2.8%	391	72.7%	131	24.3%	538	100%
Background	38	2.9%[1]	35	7.5%[2]	99	19.6%[3]	N/A		172	7.1%[4]

[1]: Percentage of black/red eggs misclassified from background in total number of black/red eggs detected.

[2]: Percentage of yellow eggs misclassified from background in total number of yellow eggs detected.

[3]: Percentage of white eggs misclassified from background in total number of white eggs detected.

[4]. Percentage of all eggs misclassified from background in total number of eggs detected.

Table 6-2 Classification and misclassification rates of *Cadra* (=*Ephestia*) eggs by the WT algorithm.

Actual eggs	Classified to								Total	
	Black		Yellow		White		Background			
Black	1196	91.8%	10	0.8%	0	0%	97	7.4%	1303	100%
Yellow	111	26.3%	203	48.1%	3	0.7%	105	24.9%	422	100%
White	5	0.9%	33	6.1%	226	42.0%	274	50.9%	538	100%
Background	389	22.8%[1]	108	30.5%[2]	95	29.3%[3]	N/A		592	20.7%[4]

1: Percentage of black/red eggs misclassified from background in total number of black/red eggs detected.

2: Percentage of yellow eggs misclassified from background in total number of yellow eggs detected.

3: Percentage of white eggs misclassified from background in total number of white eggs detected.

4: Percentage of all eggs misclassified from background in total number of eggs detected.

From these tables, it can be seen that both the WT and HT methods performed satisfactorily in detecting the parasitized *Cadra* eggs. The CCRs of parasitized eggs for the WT and the HT were 91.8% and 96.0%, respectively. For the detection of yellow unparasitized unhatched *Cadra* eggs, the HT outperformed the WT with CCRs of 93.6% and 48.1%, respectively. The CCRs for white eggs detected by the WT and the HT were 42.0% and 72.7%, respectively. Although the HT performed better than the WT in detecting white eggs, neither method provided completely satisfactory results, probably because that the white *Cadra* eggs do not possess unique color characteristics, especially when compared with the background. Upon hatching, some unparasitized Cadea eggs became creamy white while others became more transparent and almost colorless, which appeared very similar to the background. Uneven illumination probably worsened the situation even further.

The ability of the algorithms to detect all eggs in the images without missed eggs was analyzed. This analysis was achieved by comparing the number of missed eggs to the number of eggs counted manually by humans. Missed eggs were eggs that existed in the image but were not detected by the program. In the tables, missed eggs are shown as actual eggs in the image that are classified to background by the program. The WT and HT methods performed satisfactorily in detecting parasitized *Cadra* eggs - the WT and HT missed 7.4% and 2.1% of parasitized eggs, respectively. For the yellow eggs, the HT outperformed the WT by missing 2.6% and 24.9% of the actual yellow

eggs, respectively. The rates for white eggs were 50.9% and 24.3% for the WT and HT, respectively. The HT performed better than the WT, but neither method provided completely satisfactory results. This was the consequence of the creamy or colorless appearance of the white eggs

The abilities of the algorithms to avoid false detection are also studied. False detection occurred when the program incorrectly recognized a segment of the background as eggs. The false detection rate of HT (2.9%) was much lower than that of WT (22.8%) in detecting parasitized *Cadra* eggs. The HT method also performed better than the WT in detecting yellow eggs. False detection rates of the WT and HT were 30.5% and 7.5%, respectively. Although neither algorithms performed well in false detection rate for white cells, the HT method still performed better than the WT. The false detection rates for white cells by the WT and HT were 29.3% and 19.6%, respectively.

Probably the most important measure of the performance of the algorithms is their abilities to classify eggs into correct categories. This was examined by comparing the number of wrongly classified eggs against the number of manually counted eggs. It can be seen that both methods did well for parasitized *Cadra* (black/red) eggs, but the WT performed slightly better than the HT. The HT performed much better than the WT in classifying yellow eggs. The WT misclassified 26.3% of actual yellow eggs to the black category, while the HT only misclassified 0.71%. Similarly, the WT misclassified 0.7% of actual yellow eggs to the white category, while the rate for HT was only 3.08%. Both methods performed satisfactorily in detecting white eggs.

The HT method outperformed the WT method in egg detection mainly because the WT method depends heavily on color and intensity information whereas the HT method emphasizes on shape (geometric features) information of the objects. When differences in color or intensity were not significant, using shape features may have strengthened the power of detection.

Chapter 7 - Conclusions and Future Work

This study established a database of color digital images of parasitized *Cadra* eggs; extracted features that characterize parasitized and unparasitized *Cadra* eggs; developed algorithms for segmenting objects from the background; removing image noise; identifying and separating touching eggs; recognizing, classifying, and counting parasitized and unparasitized *Cadra* eggs; and determined the rate of parasitization. Two image processing methods were developed to detect parasitized and unparasitized eggs: One method was based on the Watershed delineation algorithm and the other method used the circular Hough transformation. Although both methods satisfactorily detected and counted the number of parasitized eggs and the rate of parasitization, the HT method outperformed the WT method in the correct classification rates of all three egg categories. The rates of parasitization as detected by the WT and HT methods were 53% and 55%, respectively, whereas the actual parasitization rate was 57.6%. The CCR for recognizing parasitized eggs by the WT and HT methods were 91.8% and 96.0%, respectively. The developed detection methods will enable automatic evaluation of biological control of *Cadra* (=*Ephestia*) *cautella* using *Trichogramma bourarachae*.

Detection program performance could be improved with improved image quality. The algorithms developed in this study required many steps to resolve the problems caused by uneven illumination and the similarity in color between unparasitized hatched eggs and the background. These difficulties can be greatly lessened by proper design of the illumination and replacement of the background with colors clearly distinguishable from the three types of eggs.

References

Al-Azab, A.M.A., 2007. Alternative approaches to methyl bromide for controlling *Cadra* cautella (Walker) (Lepidoptera: Pyralidae). M. Sc. Thesis, College of Agricultural and Food Science, King Faisal University, Saudi Arabia. 129 pp.

Awcock, G. J., & Thomas, R. (1995). Applied image processing. McGraw-Hill, Inc.

Bajwa, W. I., & Kogan, M. (2004). 2 Cultural Practices: Springboard to IPM. Integrated Pest Management: Potential, Constraints and Challenges, 21.

Barbedo, J. (2014). Using digital image processing for counting whiteflies on soybean leaves. Journal of Asia-Pacific Entomology, 17(4), 685-694.

Bechar, A., Gan-Mor, S., Vaknin, Y., Shmulevich, I., Ronen, B., & Eisikowitch, D. (1997). An image-analysis technique for accurate counting of pollen on stigmas. The New Phytologist, 137(4), 639-643.

Boshra, S. A. 2007. Effect of gamma irradiation on food consumption, assimilation and digestive enzymes in *Cadra* cautella (Walker) larvae. Journal of Stored Products Research 43, 49-52.

Boyer, S., Zhang, H., & Lempérière, G. (2012). A review of control methods and resistance mechanisms in stored-product insects. Bulletin of Entomological Research, 102(02), 213–229.

Boiteau, G., Singh, M., & Lavoie, J. (2009). Crop border and mineral oil sprays used in combination as physical control methods of the aphid-transmitted potato virus Y in potato. Pest Management Science, 65(3), 255–259.

Brogdon, W. G., & McAllister, J. C. (1998). Simplification of adult mosquito bioassays through use of time-mortality determinations in glass bottles. Journal of the American Mosquito Control Association, 14(2), 159–164.

Capucho, A. S., Zambolim, L., Lopes, U. N., & Milagres, N. S. (2013). Chemical control of coffee leaf rust in Coffea canephora cv. conilon. Australasian Plant Pathology, 42(6), 667–673.

Choi, D., Lee, W., & Ehsani, R. (2013). Detecting and counting citrus fruit on the ground using machine vision. In 2013 Kansas City, Missouri, July 21-July 24, 2013 (p. 1). American Society of Agricultural and Biological Engineers.

de Carvalho Moretti, T., Solis, D. R., & Godoy, W. A. C. (2013). Ants (Hymenoptera: Formicidae) collected with carrion-baited traps in Southeast Brazil. Open Forensic Science Journal, 6, 1.

Find circles using circular Hough transform - MATLAB imfindcircles. (n.d.). Retrieved April 20, 2016, from http://www.mathworks.com/help/images/ref/imfindcircles.html

Forbes, G. A. (2012). Using host resistance to manage potato late blight with particular reference to developing countries. Potato Research, 55(3-4), 205–216.

Fokkinga, M. (2011). The Hough transform. Journal of Functional Programming,21(2), 129-133.

Gonzalez, R. C., & Woods, R. E. (2002). Digital image processing. Prentice hall Upper Saddle River.

Gubler, D. J. (1998). Dengue and dengue hemorrhagic fever. Clinical Microbiology Reviews, 11(3), 480–496.

Gubler, D. J. (2002). Epidemic dengue/dengue hemorrhagic fever as a public health, social and economic problem in the 21st century. Trends in Microbiology, 10(2), 100–103.

Haubruge, E., Arnaud, L., & Mignon, J. (1997). The impact of sperm precedence in malathion resistance transmission in populations of the red flour beetle Tribolium castaneum (Herbst)(Coleoptera: Tenebrionidae). Journal of Stored Products Research, 33(2), 143–146.

Haines-Young, R. H., Barr, C. J., Black, H. I. J., Briggs, D. J., Bunce, R. G. H., Clarke, R. T., … others. (2000). Accounting for nature: assessing habitats in the UK countryside.

Hart, Rhonda. (1994). Handpicking garden pests. (picking off pests as opposed to use chemicals to destroy them). Flower & Garden Magazine, 38(3), 8.

Huffaker, C., Messenger, P., & Adkisson, P. (1976). Theory and practice of biological control. New York: Academic Press.

Hegazi, E. M., Herz, A., Hassan, S., Agamy, E., Khafagi, Shweil, S., Zaitun, A., Mostafa, S., Hafez, M., El-Shazly, A., El-Said, S., Abo-Abdala, L., Khamis, N., and Kemny, S. 2005. Naturally occurring *Trichogramma* species in olive farms in Egypt. Insect Science 12, 185-192.

Hodges, F. M. (2003). The promised planet: Alliances and struggles of the gerontocracy in American television science fiction of the 1960s. The Aging Male, 6(3), 175-182.

James, N. E. (1988). Two sides of paradise: The Eden myth according to Kirk and Spock. In D. Palumbo (Ed.), Spectrum of the fantastic (pp. 219-223). Westport, CT: Greenwood.

Li, L.Y. 1994. Worldwide use of *Trichogramma* for biological control on different crops: A survey. In Biological Control with Eggs Parasitoids, eds. E. Wajnberg and S. A. Hassan, Oxon, U.K., CAB International. Pp. 37- 53.

Mahr, D. L., & Ridgway, N. M. (1993). Biological control of insects and mites: an introduction to beneficial natural enemies and their use in pest management. North Central Regional Publication (USA).

McCarville, M. T., & O'Neal, M. E. (2012). Measuring the benefit of biological control for single gene and pyramided host plant resistance for Aphis glycines (Hemiptera: Aphididae) management. Journal of Economic Entomology, 105(5), 1835–1843.

Miyasaka, Y., Barnes, M. E., Gersh, B. J., Cha, S. S., Bailey, K. R., Abhayaratna, W. P., … Tsang, T. S. (2006). Secular trends in incidence of atrial fibrillation in Olmsted County, Minnesota, 1980 to 2000, and implications on the projections for future prevalence. Circulation, 114(2), 119–125.

Oezder, N., and Kara, G. 2010. Comparative biology and life tables of *Trichogramma* cacoeciae, T. brassicae and T. evanescens (Hymenoptera: Trichogrammatidae) with *Cadra* kuehniella and *Cadra* cautella (Lepidoptera: Pyralidae) as hosts at three constant temperatures. Biocontrol science and Technology 20: (30), 245-255.

Olkowski, W., and Zhang, A. 1990. *Trichogramma* modern day frontier in biological control. The IPM Practitioner. 12: 1-15.

Pino, J.D. 1999. Systematics of the North American species of Trichogramma Westwood (Hymenoptera: Trichogrammatidae). Mem. Entomol. Soc. Wash. 22, 1-287.

Peña, J. E., Crane, J. H., Capinera, J. L., Duncan, R. E., Kendra, P. E., Ploetz, R. C., … Cave, R. D. (2011). Chemical control of the redbay ambrosia beetle, Xyleborus glabratus, and other Scolytinae (Coleoptera: Curculionidae). Florida Entomologist, 94(4), 882–896.

Pearson, T. C., Edwards, R. H., Mossman, A. P., Wood, D. F., Yu, P. C., & Miller, E. L. (2002). Insect egg counting on mass rearing oviposition pads by image analysis. Applied Engineering in Agriculture, 18(1), 129.

Rogers, P. (2000, July 27). Cold water spray kills mites. Telegram & Gazette, p. C11.

Roriz, V., Oliveira, I., and Garcia, P. 2006. Host suitability and preference studies of Trichogramma cordubenesis (Hymenoptera: Trichogrammatidae). Biol Control 36:331-336.

Schoeller, M., and Flinn, P. W. 2000. Parasitoids and predators. In: Subramanyam, B., hagstrum, D. W. (Eds.) Alternatives to pesticides in stored-product IPM. Kluwer Academic Publishers, Boston, pp. 229-271.

Schoeller, M., 2001. Commercial application of parasitoids to control stored-product pests in Germany and Austria. In: Zdarkova E., J. Lukas, J. (Eds.) Cost action 842 Biological control of pests insects and mites, with special reference to Entomophtorales, Proceedings of the first meeting of working group 4: Bio-control of arthropod pests in the stored products. Lisbon 6-7 September 2001, pp.29-32.

Schoeller, M.; Fields, P. 2002. Screening of North American species of Trichogramma Westwood (Hymenoptera: Trichogrammatidae) for control of the Indian meal moth, Plodia interpunctella (Huebner) (Lepidoptera: Pyralidae). Proceedings of the eighth International Working Conference of Stored-Product Protection, York, U. K., July 2002.

Stiling, P. (1992). Biological Control by Natural Enemies. Ecology, 73(4), 1520.

Steidle, J. L. M., Rees, D., Wrigth, E. J. 2001. Assessment of Australian Trichogramma species (Hymenoptera: Trichogrammatidae) as control agents of stored product moths. Journal of Stored Products Research 37, 263-275.

Siddiq, M., and Greiby, I. 2014. Overview of date fruit production, postharvest handling, processing, and nutrition. In Dates: postharvest science, processing technology and health benefits. 1st edition. Edited by M. Siddiq, S.M. Aleid, and A.A. Kader. John Wiley and Sons, Chichester, United Kingdom. Pp. 1–28.

Tumwine, J., Frinking, H. D., & Jeger, M. J. (2002). Integrating cultural control methods for tomato late blight (Phytophthora infestans) in Uganda. Annals of Applied Biology, 141(3), 225–236.

The Watershed Transform: Strategies for Image Segmentation. (n.d.). Retrieved April 20, 2016, from http://www.mathworks.com/company/newsletters/articles/the-watershed-transform-strategies-for-image-segmentation.html

Vincent, C., Hallman, G., Panneton, B., & Fleurat-Lessard, F. (2003). Management of Agricultural Inscts with Physical Control Methods*. Annual Review of Entomology, 48(1), 261–281.

YAMADA, M. (n.d.). Methods of Control of Injury Associated with Continuous Vegetable Cropping in Japan.

Youssef, A. A., Nasr, F. N., Stefanos, S. S., Abou Elkhair, S. S., Shehata, W. A., Agamy, E., Herz, A., Hassan, S. A. 2004. The side effect of plant protection products used in olive cultivation on the hymenopterous egg parasitoid Trichogramma cacoeciae Marchal. J. Appl. Ent. 128, 593-599.

Yuen, H. K., Princen, J., Illingworth, J., & Kittler, J. (1990). Comparative study of Hough transform methods for circle finding. Image and Vision Computing, 8(1), 71–77.

Zhang, P. (2014). Automatic Assessment of Biological Control Effectiveness of Trichogramma Bourarachae Against Ephestia cautella Using Machine Vision. Manhattan: No publish.

Appendix A - Hough transform program

```
clear;
rgbBF=imread('FILE0045.jpg');  %Read image
[centers,radii]=imfindcircles(rgbBF,[70
120],'ObjectPolarity','dark','Sensitivity',0.965,'Method','twostage','EdgeThr
eshold',0.05);  %Hough Transfer find circles
figure(1),imshow(rgbBF);
h=viscircles(centers,radii); %draw circles
centerdx=round(centers(:,2)); % dark cell center
centerdy=round(centers(:,1));
rgbOPEN = imopen(rgbBF,strel('disk',11));
rgbCLOSE = imclose(rgbOPEN,strel('disk',11));
RBF=rgbCLOSE(:,:,1);
GBF=rgbCLOSE(:,:,2);
BBF=rgbCLOSE(:,:,3);
r=im2double(RBF)*255;
b=im2double(BBF)*255;
g=im2double(GBF)*255;
rb=r-b;
rg=r-g;
gb=g-b;
IntensityBF=RBF/3+GBF/3+BBF/3;
IntensityBF=double(IntensityBF);
h=ones(5,5)/25;
rgbfilter=imfilter(IntensityBF,h);
px=1:1:2400;
py=1:1:3200;
[pX,pY] = meshgrid(py,px);
[maxInt] = max(rgbfilter(:));
[positionx1, positiony1] = ind2sub(size(rgbfilter),find(rgbfilter==maxInt));
x1=round(mean(positionx1));
y1=round(mean(positiony1));
x2=1200;
y2=50;
max_2=70;%IntensityBF(x2,y2);
I1=mean(mean(rgbfilter(3:150,3:150)));
I2=mean(mean(rgbfilter(2270:2398,3:170)));
I3=mean(mean(rgbfilter(2270:2398,3070:3198)));
I4=mean(mean(rgbfilter(3:170,3070:3198)));
positionx=[1 2400 2400 1 x1 x2];
positiony=[1 1 3500 3500 y1 y2];
intensity=[I1 I2 I3 I4 maxInt  max_2];
m1=[positionx;positiony];
m1=m1';
n1=intensity';
ST=regstats(n1,m1,'quadratic');
A0=ST.tstat.beta(1);
A1=ST.tstat.beta(2);
A2=ST.tstat.beta(3);
A3=ST.tstat.beta(4);
A4=ST.tstat.beta(5);
A5=ST.tstat.beta(6);
[Y,X] = meshgrid(py,px);
Z=A0+A1.*X+A2.*Y+A3.*X.*Y+A4.*X.*X+A5.*Y.*Y;
```

```
B=max(max(Z))./Z;
Intensity_rgb=B.*IntensityBF;
Intensity_show=uint8(round(Intensity_rgb));
%imtool(Intensity_show)
Gray=zeros(2400,3200);
for i=1:2400
    for j=1:3200
        if Intensity_show(i,j)>80 && Intensity_show(i,j)<150
            if rb(i,j)>5
                if rg(i,j)>5
                    if rb(i,j)>0
                Gray(i,j)=1;
                    end
                end
            end
        end
    end
end
% figure(4),imshow(Gray);
fill=imfill(Gray,'holes');
bw=imopen(fill,strel('disk',35));
% figure(5),imshow(bw);
[centersw,radiiw]=imfindcircles(bw,[70
150],'ObjectPolarity','bright','Sensitivity',0.985,'Method','twostage','EdgeT
hreshold',0.05);  %Hough Transfer find circles
figure(2),imshow(bw);
h=viscircles(centersw,radiiw);
simwx=round(centersw(:,2)); % dark cell center
simwy=round(centersw(:,1));
for i=1:size(simwx,1)
for  j=i+1:size(simwx,1)
center_distw(i,j)=sqrt((simwx(i)-simwx(j))^2+(simwy(i)-simwy(j))^2);
center_distw(j,i)=center_distw(i,j);
end
end
center_rec=center_distw;
index=1;
i=1;
j=1;
while(index==1)
            if center_distw(i,j)<150 && center_distw(i,j)>0
                simwx(j,:)=[];
                simwy(j,:)=[];
                center_distw=[];
              for m=1:size(simwx,1)
                  for  n=m:size(simwy,1)
                     center_distw(m,n)=sqrt((simwx(m)-simwx(n))^2+(simwy(m)-
simwy(n))^2);
                     center_distw(n,m)=center_distw(m,n);
                     index_1=1;
                     i=1;
                     j=1;
                  end
              end
            end
            if i+1<=size(center_distw,1)
                if j+1<=size(center_distw,2)
```

```
                    i=i;
                    j=j+1;
                else
                    j=1;
                    i=i+1;
                end
            else
                index=0;
            end


end
sim=[simwy simwx];
%figure(3),imshow(rgbOPEN);
h=viscircles(sim,radii(1:size(sim)));
x=[centerdx;simwx];
y=[centerdy;simwy];
color=rgbCLOSE(x(:),y(:),:); %put cell information together
for i=1:size(x)
    cartoon_rgb(x(i),y(i),1)=color(i,1);
    cartoon_rgb(x(i),y(i),2)=color(i,2);
    cartoon_rgb(x(i),y(i),3)=color(i,3);
end
radius = 80;
circ_mask = double(getnhood(strel('ball',radius,radius,0)));
cartoon_rgb = imfilter(cartoon_rgb,circ_mask,'conv'); % convolution
%figure(4),imshow(cartoon_rgb);
simx=x;
simy=y;
for i=1:size(simx,1)
    for  j=i+1:size(simx,1)
        center_dist(i,j)=sqrt((simx(i)-simx(j))^2+(simy(i)-simy(j))^2);
        center_dist(j,i)=center_dist(i,j);
    end
end
center_rec_all=center_dist;
index_all=1;
i=1;
j=1;
while(index_all==1)
            if center_dist(i,j)<100 && center_dist(i,j)>0
                simx(j,:)=[];
                simy(j,:)=[];
                center_dist=[];
              for m=1:size(simx,1)
                  for  n=m:size(simy,1)
                     center_dist(m,n)=sqrt((simx(m)-simx(n))^2+(simy(m)-
simy(n))^2);
                     center_dist(n,m)=center_dist(m,n);
                     index_1=1;
                     i=1;
                     j=1;
                  end
              end
            end
            if i+1<=size(center_dist,1)
```

```
                if j+1<=size(center_dist,2)
                    i=i;
                    j=j+1;
                else
                    j=1;
                    i=i+1;
                end
            else
                index_all=0;
            end


end
sim_all=[simx simy];
color_all=rgbBF(simx(:),simy(:),:);
figure(5),imshow(rgbBF);
sim_show=[simy simx];
radii_show=50*ones(size(simx));
h=viscircles(sim_show,radii_show);
M=13;
for u=1:size(color_all,1)
    r1=simx(u);
    r2=simy(u);
    if r1-M>0
        w1=r1-M;
    else
        w1=1;
    end
    if r1+M<2401
        w2=r1+M;
    else
        w2=2400;
    end
    if r2-M>0
        w3=r2-M;
    else
        w3=1;
    end
    if r2+M<3201
        w4=r2+M;
    else
        w4=3200;
    end
%     color_mean(u,:)=rgbBF(r1,r2,:);
    color_mean(u,:)=round(mean(mean(rgbBF(w1:w2,w3:w4,:))));
    Int(u,1)=round(mean(mean(Intensity_rgb(w1:w2,w3:w4))));
end
m=1;                %seperate black,red,yellow,white cells
n=1;
p=1;
k=1;
black=[];
black_color=[];
red=[];
red_color=[];
yellow=[];
```

```
yellow_color=[];
white=[];
white_color=[];
t=[];
R=im2double(color_mean(:,1));
G=im2double(color_mean(:,2));
B=im2double(color_mean(:,3));
for i=1:size(simx)
       if Int(i)<86
           if color_mean(i,1)-color_mean(i,3)<27
           black(m,1)=simx(i);
           black(m,2)=simy(i);
           black_color(m,:)=color_mean(i,:);
           black(m,3)=i;
           m=m+1;
           end
       end
      W1=and(color_mean(i,1)-color_mean(i,3)<41,Int(i)>100);

       if  W1==1
                    white(k,1)=simx(i);
                    white(k,2)=simy(i);
                    white_color(k,:)=color_mean(i,:);
                    white(k,3)=i;
                    k=k+1;
       end

                   Y1=and(color_mean(i,1)-color_mean(i,3)>40,Int(i)>100);
              Y2=and(color_mean(i,1)-color_mean(i,3)>26,Int(i)<101);
              Y=or(Y1,Y2);
              if Y==1
                       yellow(p,1)=simx(i);
                       yellow(p,2)=simy(i);
                       yellow_color(p,:)=color_mean(i,:);
                       yellow(p,3)=i;
                       p=p+1;
              end
end
countblack=size(black,1); %count
countyellow=size(yellow,1);
countwhite=size(white,1);
pic_rgb=zeros(size(rgbBF),class(rgbBF));
for i=1:size(black,1)
    pic_rgb(black(i,1),black(i,2),1)=1;    %show same group cells with same
color
    pic_rgb(black(i,1),black(i,2),2)=0;
    pic_rgb(black(i,1),black(i,2),3)=0;
end


for i=1:size(white,1)
    pic_rgb(white(i,1),white(i,2),1)=125;
    pic_rgb(white(i,1),white(i,2),2)=125;
    pic_rgb(white(i,1),white(i,2),3)=125;
end
```

```
for i=1:size(yellow,1)
    pic_rgb(yellow(i,1),yellow(i,2),1)=131;
    pic_rgb(yellow(i,1),yellow(i,2),2)=101;
    pic_rgb(yellow(i,1),yellow(i,2),3)=0;
end


radius = 80;
pic_rgb = imfilter(pic_rgb,circ_mask,'conv');%convolution
[aa,bb]=find(pic_rgb(:,:,1)==0);
pic=pic_rgb;    %change black backgroung to white background
for i=1:size(aa,1)
   pic(aa(i),bb(i),1)=255;
   pic(aa(i),bb(i),2)=255;
   pic(aa(i),bb(i),3)=255;
end
figure(6),imshow(pic);
```

Appendix B - Watershed method program

```
clear;
 rgb=imread('FILE0045.jpg');  %RGB TO HSI
 rgbBF=rgb;
 rgbOPEN = imopen(rgbBF,strel('disk',11));
rgbCLOSE = imclose(rgbOPEN,strel('disk',11));
RBF=rgbCLOSE(:,:,1);
GBF=rgbCLOSE(:,:,2);
BBF=rgbCLOSE(:,:,3);
r=im2double(RBF)*255;
b=im2double(BBF)*255;
g=im2double(GBF)*255;
rb=r-b;
rg=r-g;
gb=g-b;
IntensityBF=RBF/3+GBF/3+BBF/3;
IntensityBF=double(IntensityBF);
h=ones(5,5)/25;
rgbfilter=imfilter(IntensityBF,h);
px=1:1:2400;
py=1:1:3200;
[pX,pY] = meshgrid(py,px);
[maxInt] = max(rgbfilter(:));
[positionx1, positiony1] = ind2sub(size(rgbfilter),find(rgbfilter==maxInt));
x1=round(mean(positionx1));
y1=round(mean(positiony1));
x2=1200;
y2=50;
max_2=70;%IntensityBF(x2,y2);
I1=mean(mean(rgbfilter(3:150,3:150)));
I2=mean(mean(rgbfilter(2270:2398,3:170)));
I3=mean(mean(rgbfilter(2270:2398,3070:3198)));
I4=mean(mean(rgbfilter(3:170,3070:3198)));
positionx=[1 2400 2400 1 x1 x2];
positiony=[1 1 3500 3500 y1 y2];
intensity=[I1 I2 I3 I4 maxInt  max_2];
m1=[positionx;positiony];
m1=m1';
n1=intensity';
ST=regstats(n1,m1,'quadratic');
A0=ST.tstat.beta(1);
A1=ST.tstat.beta(2);
A2=ST.tstat.beta(3);
A3=ST.tstat.beta(4);
A4=ST.tstat.beta(5);
A5=ST.tstat.beta(6);
[Y,X] = meshgrid(py,px);
Z=A0+A1.*X+A2.*Y+A3.*X.*Y+A4.*X.*X+A5.*Y.*Y;
B=max(max(Z))./Z;
Intensity_rgb=B.*IntensityBF;
Intensity_show=uint8(round(Intensity_rgb));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
```

```
 J=imadjust(rgb,[0.10 0.10  0.10 ; 0.667 0.667 0.667],[0 0 0; .9 .9 .9 ],0.5)
;
%figure(1),imshow(J);
I=double(J)/255;
R=I(:,:,1);
G=I(:,:,2);
B=I(:,:,3);
numi=1/2*((R-G)+(R-B));
denom=((R-G).^2+((R-B).*(G-B))).^0.5;
H=acosd(numi./(denom+0.000001));
H(B>G)=360-H(B>G);
H=H/360;
S=1- (3./(sum(I,3)+0.000001)).*min(I,[],3);
I=Intensity_rgb/255;
HSI=zeros(size(rgb));
HSI(:,:,1)=H;
HSI(:,:,2)=S;
HSI(:,:,3)=I;
imwrite(HSI,'FILE0045En.jpg');
imwrite(H,'FILE0045EnH.jpg');
imwrite(S,'FILE0045EnS.jpg');
imwrite(I,'FILE0045EnI.jpg');
% Complete ROI in HSI space
hsi=imread('FILE0045En.jpg');
hsi_h=imread('FILE0045EnH.jpg');
hsi_s=imread('FILE0045EnS.jpg');
hsi_i=imread('FILE0045EnI.jpg');
[width,height]=size(hsi_h);
Ih=hsi_h;
BWH=im2bw(Ih,0.274);   % grayscale image to binary image
BWHC=~BWH;                  % reverse black and white
I3=bwareaopen(BWHC,100);  %remove samll objects from binary image
I2 = imfill(I3,'holes'); % fill image regions and holes
I3=imdilate(I2,strel('disk',10));% se=strel('disk',10);
[aa,bb]=find(I3==1);   %save white pixel position
 save ROIaabb20130424 aa bb

[width,height,dimension]=size(hsi);
ROI1=255*ones(width,height,3,class(hsi));
for i=1:size(aa,1)

   ROI1(aa(i),bb(i),1)=hsi_h(aa(i),bb(i));
   ROI1(aa(i),bb(i),2)=hsi_s(aa(i),bb(i));
   ROI1(aa(i),bb(i),3)=hsi_i(aa(i),bb(i));
end
ROI0=zeros(width,height,3,class(hsi));
for i=1:size(aa,1)

   ROI0(aa(i),bb(i),1)=hsi(aa(i),bb(i),1);
   ROI0(aa(i),bb(i),2)=hsi(aa(i),bb(i),2);
   ROI0(aa(i),bb(i),3)=hsi(aa(i),bb(i),3);
end
Ir2=ROI1(:,:,3);%Ir=filteredimage(:,:,1);
Threr=.275; %  0.59610.690.9%
 BWr=im2bw(Ir2,Threr);
  BWr=~BWr;
```

```
  BWr1=imfill(BWr,'holes');
 BW3=imerode(BWr1,strel('disk',8));
 BWr1=imdilate(BW3,strel('disk',8));
 BW5=imdilate(BWr1,strel('disk',40));
h1=ones(13,13)/169;
BW6=imfilter(BW5,h1);
h2=ones(9,9)/81;
BW2=imfilter(BW6,h2);
% figure(2),imshow(BW2);
 D       = -bwdist(~BW2,'euclidean');% 'cityblock', 'chessboard', 'quasi-
euclidean', or 'euclidean'
 D(~BW2)  =-Inf;% min(D(:)); %best value to reduce the effect of local minima
 D       = imhmin(D,2,8);% 2% is the height threshold for suppressing shallow
  L = watershed(D,8);
 figure(2), imshow(rgb), title('watershed image');
 ssL = regionprops(L, 'all');

 for i=1:size(ssL,1)
     ssLCentroid1(i)=ssL(i).Centroid(1);
     ssLCentroid2(i)=ssL(i).Centroid(2);
     ssLArea(i)=ssL(i).Area;
     ssLDiameter(i)=ssL(i).EquivDiameter;
 end

 for i=1:size(ssL,1)
    for  j=i+1:size(ssL,1)
        ssLDistance(i,j)=sqrt((ssLCentroid1(i)-
ssLCentroid1(j))^2+(ssLCentroid2(i)-ssLCentroid2(j))^2);
        ssLDistance(j,i)=ssLDistance(i,j);
    end
 end

 [ssLY,ssLI]=sort(ssLDistance);
 minssLY=100; %mean(ssLY(2,:),2);
hold on;
count=0;
 for i=1:size(ssL,1)
     if (BW2(floor(ssLCentroid2(i)),floor(ssLCentroid1(i)))==1)
             if ssLY(2,i)<minssLY

             j=ssLI(2,i);
             X=[ssLCentroid1(i),ssLCentroid1(j)];
             Y=[ssLCentroid2(i),ssLCentroid2(j)];
             plot(X,Y,'g>');

             count=count+1;
             ssLNC1(i)=mean(X);
             ssLNC2(i)=mean(Y);

             ssLNC1(j)=0;
             ssLNC2(j)=0;


           else
             ssLNC1(i)=ssLCentroid1(i);
```

```
            ssLNC2(i)=ssLCentroid2(i);
            end

          ssLNC1(j)=0;
       ssLNC2(j)=0;
     else
          ssLNC1(i)=0;
       ssLNC2(i)=0;
     end


 end

 hold on;
 for j=1:size(ssL,1)
  for i=1:4
        x20(j,i)=ssLNC1(j)+1*cos(i*0.5*pi);
        y20(j,i)=ssLNC2(j)+1*sin(i*0.5*pi);
  end
   plot(x20(j,:),y20(j,:),'r*');
 end
i=1;
countRe=0;
for j=1:size(ssLNC1,2)
    if (ssLNC1(j)==0)
        countRe=countRe+1;
tb(i,1)=j;
i=i+1;
    end
end

countBlack=size(ssLNC1,2)-countRe;

% count yellow eggs
[aa,bb]=find(BW2==1);
for i=1:size(aa,1)

     ROI1(aa(i),bb(i),1)=255;
     ROI1(aa(i),bb(i),2)=255;
     ROI1(aa(i),bb(i),3)=255;
end
RedCellImage=ROI1;
 imwrite(ROI1, 'RedCellImage.jpg');
RedCell=imread('RedCellImage.jpg');

Ib=RedCell(:,:,3);
%imtool(Ib)
Threb1=70;
[mmb,nnb]=size(Ib);
for i=1:mmb
   for j=1:nnb
       if(Ib(i,j)<Threb1)
   BWb1(i,j)=1;
       else
           BWb1(i,j)=0;
```

```
       end

   end
end
Threb2=100;
[mmb,nnb]=size(Ib);
for i=1:mmb
   for j=1:nnb
       if(Ib(i,j)<Threb2)
            BWb2(i,j)=1;
       else
           BWb2(i,j)=0;
       end

    end
end
BWMR=BWb2-BWb1;
BWMR1=bwareaopen(BWMR,100);  %remove samll objects from binary image
BWMRfill=imfill(BWMR,'holes');
Red1=imerode(BWMRfill,strel('disk',25));
 Red2=imdilate(Red1,strel('disk',100));
BWMRL=bwlabel(Red2,4);
ssLBW = regionprops(BWMRL, 'all');
D       = -bwdist(~Red2,'euclidean');% 'cityblock', 'chessboard', 'quasi-
euclidean', or 'euclidean'
 D(~Red2)  =-Inf;% min(D(:)); %best value to reduce the effect of local
minima
 D       = imhmin(D,2,8);% 2% is the height threshold for suppressing shallow
  L = watershed(D,8);
 figure(3), imshow(rgb), title('watershed image');
 ssL = regionprops(L, 'all');
 hold on
 countYellow=0;
 for i=1:size(ssL,1)%1
     ssLCentroid1y(i)=ssL(i).Centroid(1);
     ssLCentroid2y(i)=ssL(i).Centroid(2);
      ssLArea(i)=ssL(i).Area;
     ssLDiameter(i)=ssL(i).EquivDiameter;
     if (BWMRL(floor(ssLCentroid2y(i)),floor(ssLCentroid1y(i)))>=1)
          plot(ssLCentroid1y(i),ssLCentroid2y(i),'g*');
          countYellow=countYellow+1;
     end

 end
 WhiteMere=RedCell;
%  [aaR,bbR]=find(BWMRL>=1);
%   for i=1:size(aaR,1)
%        WhiteMere(aaR(i),bbR(i),1)=255;
%         WhiteMere(aaR(i),bbR(i),2)=255;
%          WhiteMere(aaR(i),bbR(i),3)=255;
%   end
  imwrite(WhiteMere, 'white20130606.jpg');
  [m,n,o]=size(WhiteMere);
  whitebinary=zeros(m,n);
for i=1:m
    for j=1:n
```

```
        if WhiteMere(i,j,1)<50
            if WhiteMere(i,j,3)>110
                whitebinary(i,j)=1;
            end
        end
    end
end
%imtool(whitebinary);
whiteopen=bwareaopen(whitebinary,100);
whitefill=imfill(whiteopen,'holes');

 BWB1=imerode(whitefill,strel('disk',20)); % to eliminate the bordering of
black and red cells.
  BWB2=imdilate(BWB1,strel('disk',20));
BWBFill=imfill(BWB2,'holes');
BWMWL=bwlabel(BWBFill);
ssLBWW = regionprops(BWMWL, 'all');
for i=1:size(ssLBWW,1)
     ssLBWWCentroid(i,1)=ssLBWW(i).Centroid(1);
     ssLBWWCentroid(i,2)=ssLBWW(i).Centroid(2);
     ssLBWWCentroid(i,3)=ssLBWW(i).Area;
end
countWhite=0;
for i=1:size(ssLBWW,1)
    if ssLBWW(i).Area>40000
         countWhite=countWhite+ceil(ssLBWW(i).Area/40000) ;
    else
        if ssLBWW(i).Area>14000
         countWhite=countWhite+1;
        end
    end
end
if (size(ssLBWWCentroid,1)~=0)
for m=size(ssLBWWCentroid,1):-1:1
    if ssLBWWCentroid(m,3)<14000
        ssLBWWCentroid(m,:)=[];
    end
end
end
for m=1:size(ssLBWWCentroid,1)
    if ssLBWWCentroid(m,3)<40000
        ssLBWWCentroid(m,4)=1;
    else
        ssLBWWCentroid(m,4)=ceil(ssLBWWCentroid(m,3)/40000);
    end
end
pic_rgb=zeros(size(rgb),class(rgb));
for i=1:size(x20,1)
    pic_rgb(round(y20(i,1))+1,round(x20(i,1))+1,1)=1;    %show same group
cells with same color
    pic_rgb(round(y20(i,1))+1,round(x20(i,1))+1,2)=0;
    pic_rgb(round(y20(i,1))+1,round(x20(i,1))+1,3)=0;
end
for i=1:size(ssLCentroid1y,2)
    pic_rgb(round(ssLCentroid2y(1,i))+1,round(ssLCentroid1y(1,i))+1,1)=131;
    pic_rgb(round(ssLCentroid2y(1,i))+1,round(ssLCentroid1y(1,i))+1,2)=101;
    pic_rgb(round(ssLCentroid2y(1,i))+1,round(ssLCentroid1y(1,i))+1,3)=0;
```

```
end
for i=1:size(ssLBWWCentroid,1)
    pic_rgb(round(ssLBWWCentroid(i,2))+1,round(ssLBWWCentroid(i,1))+1,1)=131;
    pic_rgb(round(ssLBWWCentroid(i,2))+1,round(ssLBWWCentroid(i,1))+1,2)=131;
    pic_rgb(round(ssLBWWCentroid(i,2))+1,round(ssLBWWCentroid(i,1))+1,3)=131;
end
countBlack=countBlack+1
countYellow=size(ssLCentroid1y,2)
countWhite
radius = 80;
circ_mask = double(getnhood(strel('ball',radius,radius,0)));
pic_rgb = imfilter(pic_rgb,circ_mask,'conv');%convolution
[aa,bb]=find(pic_rgb(:,:,1)==0);
pic=pic_rgb;    %change black backgroung to white background
for i=1:size(aa,1)
   pic(aa(i),bb(i),1)=255;
   pic(aa(i),bb(i),2)=255;
   pic(aa(i),bb(i),3)=255;
end
figure(4),imshow(pic);
hold on
plot(round(ssLBWWCentroid(:,1)),round(ssLBWWCentroid(:,2)),'gx');
n=num2str(ssLBWWCentroid(:,4),'%5.0f');
text(ssLBWWCentroid(:,1),ssLBWWCentroid(:,2),n);
```