UNIVERSITY OF CALIFORNIA
RIVERSIDE

Bio-Image Analysis for Understanding Plant Development and Mosquito Behaviors

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Katya Mkrtchyan

March 2017

Dissertation Committee:

Professor Amit Roy-Chowdhury, Chairperson
Professor Eamonn Keogh
Professor Stefano Lonardi
Professor Tamar Shinar

The Dissertation of Katya Mkrtchyan is approved:

_____

_____

_____

_____

Committee Chairperson

University of California, Riverside

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor Prof. Amit Roy-Chowdhury for the continuous support of my Ph.D study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D study.

I would like to thank the rest of my thesis committee: professor Stefano Lonardi, professor Eamonn Keogh, and professor Tamar Shinar, for serving as my committee members and for their insightful comments and encouragement. I would also like to express my gratitude to my colleges in the lab for their ideas and discussions which contributed to my knowledge in the field. In particular, I would like to thank Dr. Anirban Chakraborty, Dr. Mahmudal Hasan, Dr. Nandita Nayak, Dr. Abir Das for sharing ideas and helping me in understanding concepts better. They have always been source of constant support for me.

A special thanks to my family. Words cannot express how grateful I am to my father Martin Mkrtchyan, mother Gayane Samvelyan and brother Sevak Mkrtchyan for all of their unconditional support in my career goals and guidance. They inspired me to pursue a Ph.D. and have always boosted my confidence and offered me great moral strength. I am also very grateful to my husband Robert Mikaelian, my mother-in law and father-in-law for all the sacrifices that they have made on my behalf, for their support and guidance that made my graduate studies possible. My entire family has supported and encouraged me to pursue my dreams. Finally, I want to mention my daughter Lia for making life more beautiful and everything in life meaningful.

To my family.

ABSTRACT OF THE DISSERTATION

Bio-Image Analysis for Understanding Plant Development and Mosquito Behaviors

by

Katya Mkrtchyan

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, March 2017
Professor Amit Roy-Chowdhury, Chairperson

Images play an increasingly important role in many fields of science and its applications. Biology is one of the best examples of fields that have come to depend heavily upon images for progress. Biological images contain a lot of objects and patterns, which may contain information about underlying mechanism in biology. Image analysis provides a way to extract and quantify objects and patterns in image data and obtain answers to meaningful biological questions. This dissertation focuses on image analysis for biological applications in segmentation, registration and count estimation problems.

Proper understanding of the causal relationship between cell growth patterns and gene expression dynamics is one of the major topics of interest in developmental biology. Information such as rates and patterns of cell expansion play a critical role in explaining cell growth and deformation dynamics. In our research, we focus on studying the developing plant with the goal of obtaining very accurate cell development statistics. The image processing and analysis framework for gathering the cell growth and division statistics comprises of three main parts - image registration, cell segmentation and cell tracking. Without proper segmentation and registration the subsequent parts in the image analysis system would fail. This dissertation addresses both these problems.

To provide proper segmentation of cells we propose a single framework that entails seg-

mentation and tracking of plant cell images. We show how to optimally choose the parameters in the watershed segmentation algorithm for high quality segmentation results. To register image datasets we have provided an optimization based framework to select the best image slice correspondence from consecutive image stacks by using the tissue characteristics in images. Also, we have presented a novel landmark selection method where we use characteristics of neighboring cells as unique features. To evaluate both our frameworks, cell correspondences across multiple slices and time windows are fused to obtain cell lineages and compared to recent results in this area. Experiments on multiple plant datasets show the proposed algorithms provide significantly longer, more accurate cell lineages and more comprehensive identification of cell divisions.

Another contribution of this work is in count estimation. Mosquitoes and other blood-feeding insects transmit deadly diseases to hundreds of millions of people, causing severe suffering and more than a million deaths each year. Female mosquitoes that transmit deadly diseases locate human hosts by detecting exhaled CO_2 and skin odor. Quantitative analysis of a mosquito's attraction to different odors is very important. It can lead to discoveries about mosquitoes that could have a big impact on human health. We present a method to automate mosquito counting in videos. The current manual analysis on these videos is not sufficient for quantitative analysis and available object counting approaches do not work well. We propose an automated mosquito counting technique which uses the softmax classifier with a sliding window technique to detect the mosquitoes in the separated region of interest, where the over-detections resulting from the sliding window technique are eliminated by non-maximum suppression method. The final counts of the mosquito detections represent the desired mosquito counts in images. The proposed automated method has been applied on different datasets and showed very good and consistent counts of mosquitoes.

Contents

List of Figures

List of Tables

Part I

First Part

Chapter 1

Introduction

Images play an increasingly important role in many fields of science and its applications. Biology is one of the best examples of fields that have come to depend heavily upon images for progress. There are numerous imaging techniques developed and number of biological images taken which contain spatial information or spatio-temporal information. This biological images contain a lot of objects and patterns, which may contain information about underlying mechanism in biology. Extracting this information by visual inspection and manual measurement is labor intensive, and the results are potentially inaccurate and poorly reproducible. Therefore, there is a growing need for computerized image analysis, where image analysis provides a way to extract and quantify objects and patterns in image data and obtain answers to meaningful biological questions.

In this dissertation, we are focusing on image analysis for biological applications in segmentation, registration and count estimation problems. We will present our methodology and provide experimental results to show the significance of our contribution in solving each of these problems.

1.1 Image Analysis: Segmentation and Registration

A local spatio-temporal coordination of cell growth and cell division plays a critical role in morphogenesis of both the plant and the animal tissues. The causal link between cell growth and cell division patterns and how they, in turn, affect organ formation is not well understood. Information such as rates and patterns of cell expansion play a critical role in explaining cell growth and deformation dynamics. However manual analysis is extremely tedious because of the high dimensionality and complexity of data. Therefore, the development of computational platforms that are capable of identification of cellular coordinates, automated tracking of cells and cell division events is important. Such computational platforms would facilitate the quantification of cellular parameters such as rates and patterns of cell expansion, cell division, and extraction of such information may lead to the development of growth models that can explain the causal relationships between cell deformation dynamics, cell growth and cell division patterns. This process is a computational challenge that has universal application to all developmental fields, both animals and plants.

To achieve the goal of quantification of biological parameters and observing their evolution in time, advanced microscopy techniques are used to collect time lapse videos and quantify the behavior of hundreds of cells in a tissue over multiple days. One of these techniques is the Confocal Laser Scanning Microscopy (CLSM) based Live Cell Imaging. This technique allows us to take optical cross sections of the cells in the tissue over multiple observational time points to generate spatio-temporal 4D (X-Y-Z-T) image stacks. To analyze the details of the collected image data a fully automated image processing and analysis framework has been created, which comprises of three main parts - image registration, cell segmentation and cell tracking. Cell tracking module of this framework establishes cell correspondence across multiple slides and time windows

and fuses these correspondences to obtain cell lineages, which contain cell life and division statistics. The quality of the cell lineage statistics depends heavily on cellular coordinate evaluations . Hence, without proper segmentation and registration the subsequent part in the image analysis system would fail. In this dissertation, we provide our contribution in both cell segmentation and image registration frameworks, with the goal of obtaining very accurate cell growth and division statistics.

1.1.1 Cell Segmentation

The temporal and spatial growth of the living organisms are captured into 4D (X-Y-Z-T) image stacks with Confocal laser scanning microscopy. To keep the plant alive for a long period of time, it is necessary to limit its exposure to the laser. This results in poor image quality and presents significant challenges to image analysis since the segmentation and the tracking needs to be robust to the poor image quality. We use watershed segmentation and show how to optimally choose the parameters in the watershed algorithm for high quality segmentation results. Collected experimental results are compared to recent results in this area. Quantitative analysis shows that the proposed algorithms provide significantly longer cell lineages and more comprehensive identification of cell divisions.

1.1.2 Image Registration

The procedure capturing the spatio-temporal 4D (X-Y-Z-T) image stacks causes both temporal and spatial misalignments in this live imaging stacks. The spatial shifts between images from different temporal stacks are caused from the involvement of manual work in noncontinuous imaging procedure (physically moving the specimen from one place to another). Also, because of contin-

uous growth of the living organism during the imaging procedure there is also a slice matching issue in consecutive stacks to be solved. The presence of these misalignments in image stacks causes collection of non accurate statistics. We present a fully automated 4D(X-Y-Z-T) registration method of live imaging stacks that takes care of both temporal and spatial misalignments. We present a novel landmark selection methodology where the shape features of individual cells are not of high quality and highly distinguishable. The proposed registration method finds the best image slice correspondence from consecutive image stacks to account for vertical growth in the tissue and the discrepancy in the choice of the starting focal point. Then it uses local graph-based approach to automatically find corresponding landmark pairs, and finally the registration parameters are used to register the entire image stack. The proposed registration algorithm combined with an existing tracking method is tested on multiple image stacks of tightly packed cells of Arabidopsis shoot apical meristem and the results show that it significantly improves the accuracy of cell lineages and division statistics.

1.2 Count estimation

Object counting, which is the estimation of the number of objects in a still image or video frame, is a tedious and time consuming task in image processing. It appears in a wide range of domains: biology (cell counting in microscopic images), surveillance systems (crowd monitoring), biodiversity (animal population study), agriculture (counting the number of trees in an aerial image of a forest), marine science research (fish population estimation), etc. Object counting is important for quantitative analysis that depends on estimation of certain elements. The traditional method for object counting is manual. There are many limitations related with manual counting; limited amount of data can be analyzed manually, continuous manual counting leads to eye fatigue and affects the

accuracy of results, the process of counting objects is not always straightforward or trivial, even performed manually. Therefore, manual method must be replaced by automated ones as the results of this method are erroneous and time consuming.

Mosquitoes and other blood-feeding insects are considered one of the most dangerous creatures on the planet because of their ability to spread deadly diseases. Biologists are trying to find solutions to prevent, control or treat these diseases. Mosquitoes host seeking behavior is important as it is at the core of the processes involved in the contact between a fly and a human. In laboratory conditions, it is easy to have thousands of mosquitoes under different experimental conditions. One of these experiments aims to discover the reaction of mosquitoes around different odors, for which a special environment called arm-in-cage is created and the mosquitoes behavior is captured in a video. The count of the mosquitoes is used to describe the interest of the mosquitoes to each odor. In this work, we present an automated object counting method for counting mosquitoes captured in video, recorded in arm-in-cage environment. In our automated approach the softmax classifier with a sliding window technique is used to detect mosquitoes. Next NMS (non-maxima suppression) method is used to eliminate over-detection resulting from the sliding window technique. The results of the NMS method correspond to the final mosquito counts. We demonstrate our method on multiple videos collected by biologists which have been analyzed manually. Our automated method provides very close to manually obtained ground-truth results. The results of this study would be of considerable importance to biologists by presenting a fast and accurate method for estimating mosquito counts compared to existing manual processes.

1.2.1 Organization of the Thesis

The rest of the thesis is organized as follows. Chapters 2 and 3 present correspondingly detailed cell segmentation and image registration approaches on 4D spatio-temporal image stacks [39, 40]. In Chapter 4 we present automated mosquito counting method for evaluating the number of mosquitoes in videos. We conclude the thesis in Chapter 5 with an outline of the future direction of research.

Chapter 2

Efficient Cell Segmentation and Tracking of Developing Plant Meristem

2.1 Introduction

Proper understanding of the causal relationship between cell growth patterns and gene expression dynamics is one of the major topics of interest in developmental biology. Information such as rates and patterns of cell expansion play a critical role in explaining cell growth and deformation dynamics. The need for quantification of these biological parameters is important to biologists. However manual analysis is extremely tedious because of the high dimensionality and complexity of data.

The subject of this study, the shoot apical meristems (SAMs) also referred to as the stem-cell niche, is the most important part of the plant body plan because cells for all the above ground plant parts are supplied from it. At the same time, the size of the stem-cell niche remains stable in spite of a continuous displacement and diversion of cells into a differentiation program. Under-

standing this dynamics is a major research thrust for developmental biologists. In this work, we focus on studying the developing plant meristem with the goal of obtaining very accurate segmentation and tracking. Confocal laser scanning microscopy is used to observe given set of SAMs labeled with plasma membrane localized yellow fluorescent protein (YFP), repeatedly for about 3 days by taking serial images at every 3 hour intervals. At each time point, multiple images are obtained at different depths. Each stack from all time points is registered by method of maximization of mutual information [35, 50]. As a result, the images taken from different time points but at the same slice are registered. Fig. 2.1 shows an example of the time lapse images. To keep the plant alive for a long period of time, it is necessary to limit its exposure to the laser. This results in poor image quality. This presents significant challenges to image analysis since the segmentation and the tracking needs to be robust to the poor image quality.

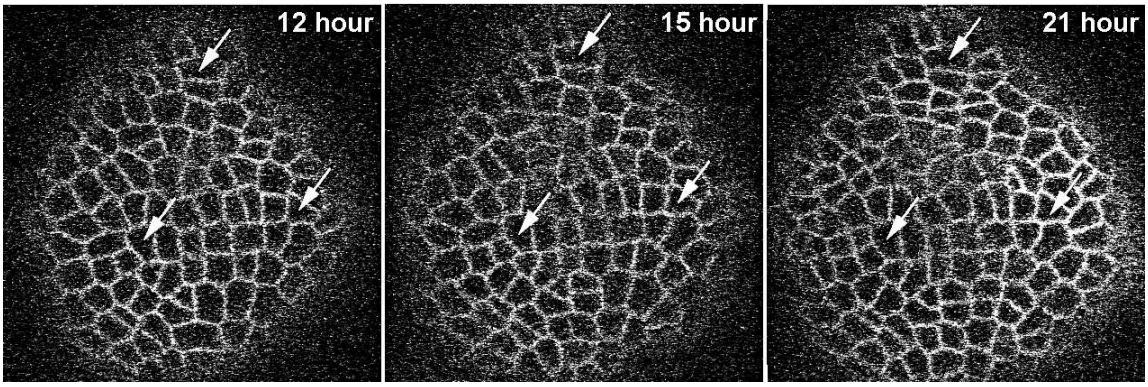


Figure 2.1: Example of three images at three time instants. Arrows show same set of cells at time points noted on each panel.

Of late, there has been some work on automated processing of such time lapse images in both plants and animals. The multiple-level-set approach is an active-contour based algorithm, which simultaneously segments cells and also tracks them [13, 30]. However, this method is not suitable for tracking of SAM cells. The method in [53] presents an approach for cell segmentation,

but has not been applied to plant cells. In [17] Softassign Procrustes algorithm was used to compute cell lineages, and to detect cell divisions, but its application to plant cells is limited. A recent work [32] addressed the problem of tracking in plant cells and especially the issue of robustness of the trackers. However, they assumed an existing level-set based segmentation method and their performance suffered due to poor quality of the segmentation in many images.

In this work, we propose a single framework that entails segmentation and tracking of plant cell images. Section 2.2 and Section 2.3 detail these two parts of the algorithm. Section 2.4 describes the experiments and results of our methods.

2.2 Segmentation

The authors in [32] used a level set algorithm for segmentation. Since the input raw images have low signal to noise ratio in the periphery and also in the central regions of the plant stem, the segmentation results of this work are of poor quality.

We used watershed transformation [42] to segment cell boundaries. Watershed treats the input image as a continuous field of basins (low intensity pixel regions) and barriers (high intensity pixel regions), and outputs the barriers, which are the cell boundaries of all the cells in the image. Prior to applying the watershed algorithm, the input image from the confocal microscope undergoes low pass filtering. This Gaussian filtered image is further processed using H-minima transformation in which all the pixels below a certain threshold percentage h are discarded. These two steps minimize the effects of noise in our segmentation. The threshold value h plays a very crucial role in the watershed algorithm. Generally, a higher value of the threshold parameter h performs under-segmentation on the image, and inversely a lower value over-segments it, as shown in Fig. 2.2.

Moreover, if the input image is very noisy, then it becomes extremely important to choose an ap-

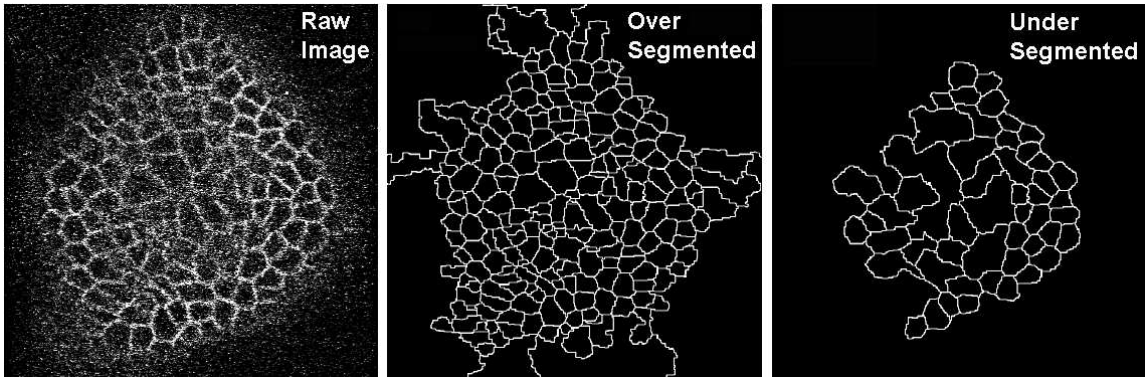


Figure 2.2: The detection of cell boundaries varies with the different values of watershed threshold h. First image is the raw input image. Over-segmentation using watershed is shown in the middle image, and under-segmentation in the right most image.

propriate threshold value such that only the correct cell boundaries are detected by watershed. One of the main contributions of this work is a quantitative metric to evaluate the "correctness" of the segmentation. We observe that the area of all the cells in a plant is almost uniform in the image. Thus watershed should ideally produce a segmented image that contains similar sized cells. We use variance in the area of the cells as a metric to measure correctness of segmentation. Thus, the value of h should be chosen such that variance of area in the segmented image is minimum. Since it is a quantitative metric, the optimal value of h is found automatically without manual intervention. The appropriateness of this metric is also verified through the successful use of these segmented images in tracking with significantly improved results.

Figure 2.3 illustrates the automatic evaluation of h for the given sample image. The segmented image obtained after watershed segmentation is a binary image.

Comparison with [32]: Since the tracking algorithm uses cell properties such as area of the cell and its centroid, it is important to retain the shape and structure of cells in the segmented images. In a noisy image, level-set method fails to preserve these cell properties, which leads to incomplete

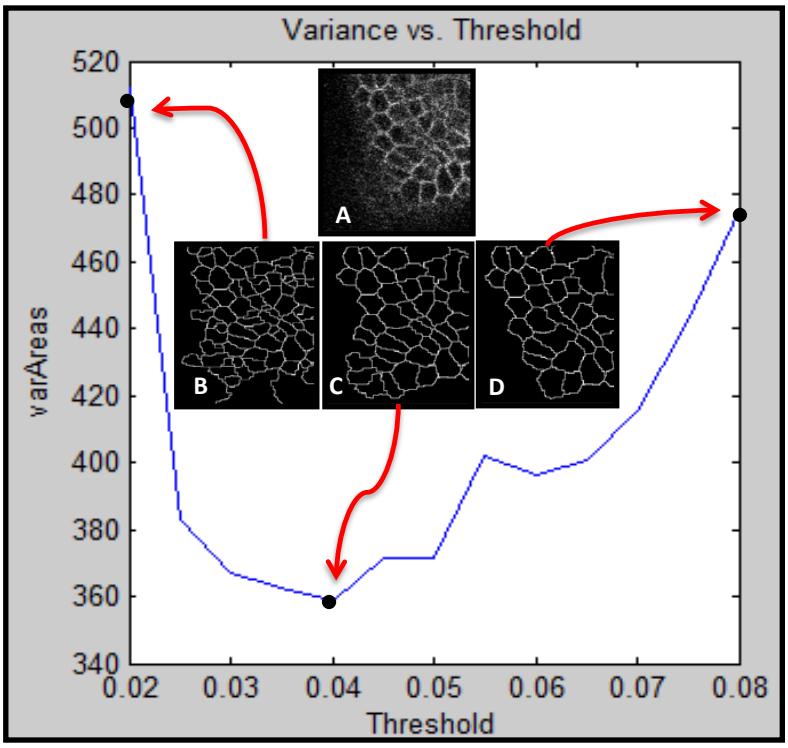


Figure 2.3: The graph shows the watershed threshold h in the X axis, and the variance in the area of the segmented cells in the Y-axis. Image A shows the input raw image, images B, C, D show the segmentation results for different values of h. Image B shows over-segmentation, image D shows under-segmentation. The correlation between the minimum variance in the areas of the cells and appropriate segmentation is clearly demonstrated in image C.

or faulty tracking of cells. On the other hand, watershed segmentation algorithm finds the geometry and other observed properties of the plant cells more accurately than the level-set, thus producing significantly better correspondence results in tracking (see Fig. 2.4).

2.3 Tracking Stem Cells

Graphical abstraction is created on a collection of cells in an image. In this process every cell is characterized by a vertex in the graph and neighboring vertices are connected by an edge. The structure of these graphs automatically contains the relative distance between two neighboring

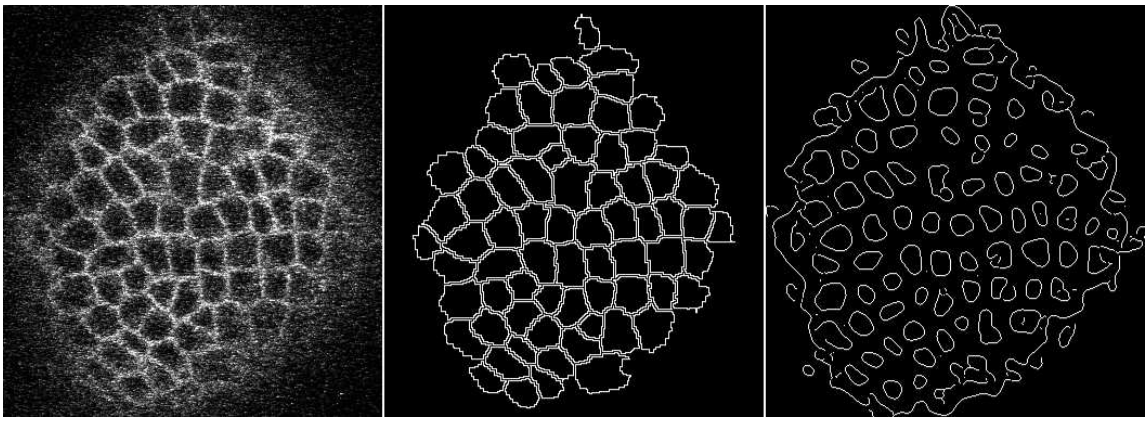


Figure 2.4: Raw image in the first column is segmented using both watershed and level-set segmentations. Second column displays watershed segmentation, while the third column is the result of level set segmentation. Qualitatively, watershed far surpasses level-set segmentation as demonstrated in this figure.

cells (the edge length) and the edge orientation. The topology and the geometry of the local graphs should not change if there is no cell division or the images are not noisy. Taking into account above conditions, the correspondences between cells are identified by matching the local graphs (Fig. 2.5 (A)). To find a correspondence between cells c_i and c_j across different time instants t and $t+1$, the

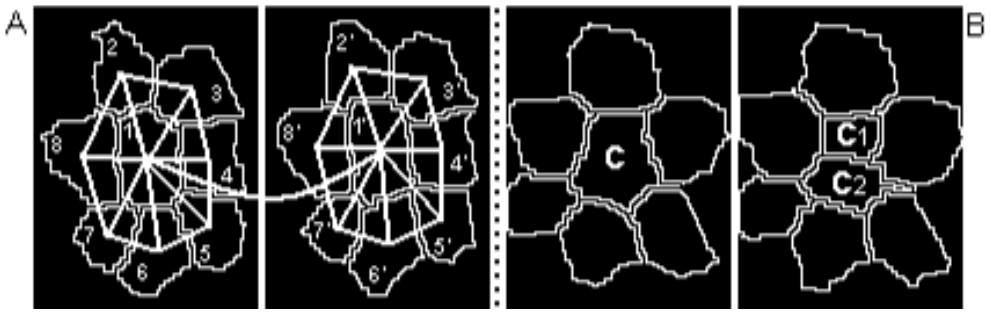


Figure 2.5: Image A: The matched local graphs G1 at t and G2 at t + 1 time points, the correspondence of the seed cell pair (1,1'), as well as the correspondences of the neighboring cells, such as (2,2') and (4,4'). Image B: A diagram to describe the relative position of the mother cell and the daughter cells.

distance measure between two local graphs is defined as follows:

$$D_L(c_i, c_j) =$$

$$\sum_{c_{k_i} \in N(c_i), c_{k_j} \in N(c_j)} \frac{(l_{c_{k_i},c_i}(t) - l_{c_{k_j},c_j}(t+1))^2}{l_{c_{k_i},c_i}(t)^2}$$

$$+\lambda \sum_{c_{k_i} \in N(c_i), c_{k_j} \in N(c_j)} (\theta_{c_{k_i},c_i}(t) - \theta_{c_{k_j},c_j}(t+1))^2$$

$$+\frac{\|P_{c_i}(t) - P_{c_j}(t+1)\|}{\Delta^2}$$

where c_{k_i} is a neighboring cell of c_i, and c_{k_j} is a neighboring cell of c_j, $N(c)$ is the neighboring cell set (set of cells that are within a certain distance around cell c), $l_{c_{k_i},c_i}(t)$ and $l_{c_{k_j},c_j}(t+1)$ are the edge lengths, $\theta_{c_{k_i},c_i}(t)$ and $\theta_{c_{k_j},c_j}(t+1)$ are the orientation angles in radians of the edges measured relative to a horizontal axis, $P_{c_i}(t)$ and $P_{c_j}(t+1)$ are the cell position vectors, and Δ is the average distance between two neighboring cells. If two local graphs match, which is when the distance measure is small, we can say that the central cells c_i and c_j are a corresponding cell pair. Overall, tracking of stem cells from any two consecutive cell image stacks using local graph matching [32] consists of the following parts: finding the seed cell pairs, finding the correspondences of the neighboring cells from the seed pairs across time (Fig. 2.6), and detecting cell divisions (Fig. 2.5 (B)). Formal description of each procedure is described in [32] in depth.

2.4 Experimental Results

Raw images segmented using watershed improves temporal tracking dramatically. Cell division largely depends on faithful detection of cell boundaries in the raw image. Because water-

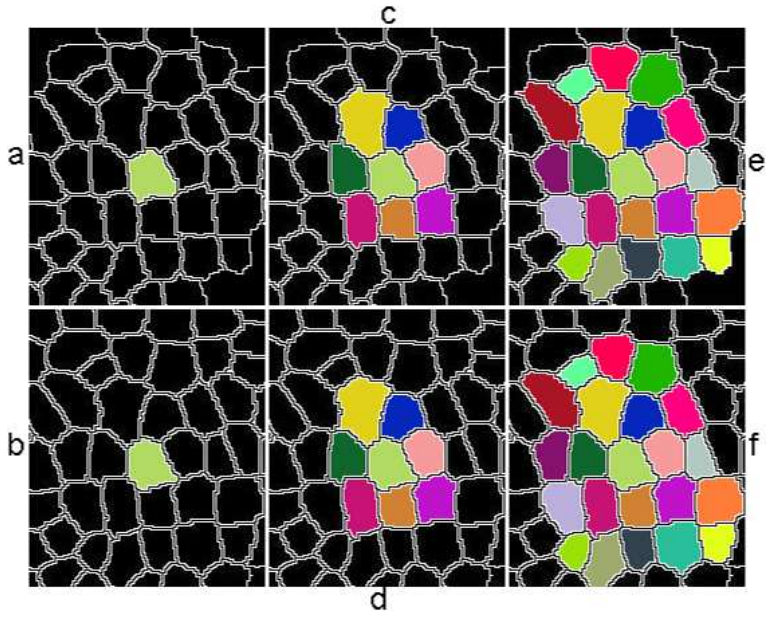


Figure 2.6: Growing cell correspondences from a seed to its neighbors. The first column denotes the seed cell pair, the middle column denotes the tracking results after the first step of the recursion of growing the correspondences, while the third column denotes the tracking results after the second step. The same color shows the matched cells.

shed produces faithful segmentation, there are less false positives as compared to level-set. We have tested watershed segmentation and tracking on different datasets of SAMs. The experimental results are shown on images obtained from plant cells and observed for 72 hours taken at every 3 hour intervals. Watershed segmentation is applied on images taken from all 24 time points at the same depth level. Then tracking is run on every 2 consecutive segmented images to get information about the cells such as cell lineage, division, area. Fig. 2.7 compares the results of temporal tracking with level-set and watershed segmentation. Biologists manually verified the accuracy of segmentation results by watershed algorithm. Also in order to verify the overall improvement of the proposed algorithm we do comparison of watershed with level-set algorithm.

The number of correctly tracked cells obtained from tracking images in consecutive time points for some period of time is compared in Table 2.1. The comparison is done on the different

datasets and we can see significant increase in the number of tracked cells obtained from watershed segmentation.

Table 2.1: Sum of the number of cells being tracked from watershed segmentation data and level-set segmentation data

Dataset	Time (h)	Watershed segmentation	Level-set segmentation
Dataset1	72	2052 (%90)	1630 (%71)
Dataset2	36	491 (%71)	215 (%31)

One of the important data to obtain is the lineage of a cell. Table 2.2 shows the comparison of average cell lineage lengths between watershed and level-set algorithms and confirms that watershed segmentation improved those results too.

Table 2.2: Average lineage length

Dataset	Watershed segmentation	Level-set segmentation
Dataset1	50	25
Dataset2	42	6

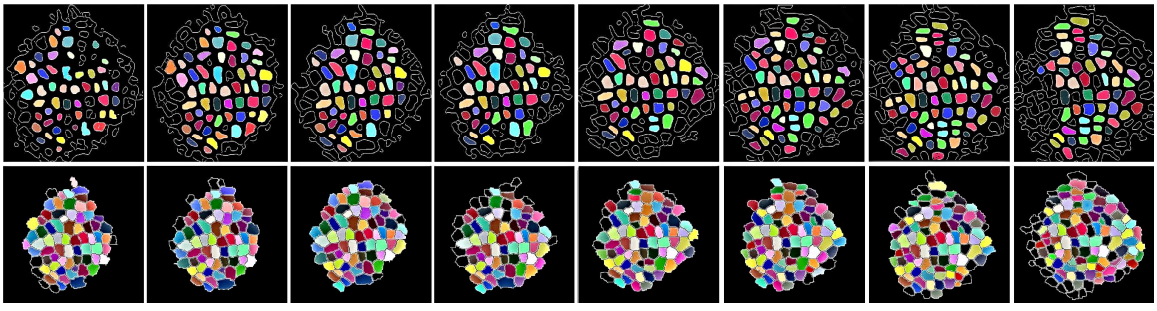


Figure 2.7: Improvements in temporal tracking using watershed segmentation. Top row shows temporal tracking using level-set segmentation, whereas the bottom row is obtained using watershed segmentation. Cells are tracked for 24 hours. The same color denotes the same cell.

2.5 Conclusion

In this chapter we proposed a novel algorithm for plant cell segmentation using watershed transformation. Specifically, we proposed a new metric for this application to find the optimal threshold parameter automatically. We show improved results of temporal tracking of cells using the proposed segmentation when compared to recent earlier results of segmentation that uses level set functions. This we believe will significantly benefit the plant biologists to find correct lineages of cell division and death for a more accurate statistical evaluation of plant growth.

Chapter 3

Optimal Landmark Selection for Registration of 4D Confocal Image Stacks in Arabidopsis

3.1 Introduction

Proper understanding of the causal relationship between cell growth patterns and gene expression dynamics is one of the major topics of interest in developmental biology. Information, such as rates and patterns of cell expansion and division, play a critical role in explaining cell growth and deformation dynamics. The need for quantification of these biological parameters and observing their evolution in time is very important. To achieve this goal, advanced microscopy techniques are used to collect time lapse videos and quantify the behavior of hundreds of cells in a tissue over multiple days. One of these techniques is the Confocal Laser Scanning Microscopy

(CLSM) based Live Cell Imaging. This technique allows us to take optical cross sections of the cells in the tissue over multiple observational time points to generate spatio-temporal 4D (X-Y-Z-T) image stacks. To analyze the details of the collected image data, it is necessary to develop a fully automated image processing and analysis framework which gives rise to many new automated visual analysis challenges.

The image processing and analysis framework for gathering the cell growth and division statistics comprises of three main parts - image registration, cell segmentation and cell tracking. Without proper registration the subsequent parts in the image analysis system would fail. The misalignments present in the live imaging stacks are both temporal and spatial. The spatial shifts between images from different temporal stacks are caused from the involvement of manual *work* in noncontinuous imaging procedure (physically moving the specimen from one place to another). Note that because of the robustness of microscopy techniques the images in one spatial (X-Y-Z) stack are almost registered. But because of continuous growth of the living organism during the imaging procedure there is also a slice matching issue in consecutive stacks to be solved. The presence of the slice mismatch between image stacks and significant shifts between images prevent getting accurate cell tracking and cell division detection, which causes collection of non accurate statistics. That is why the issue of registration is very important.

As an example let us consider this scenario; 3D confocal laser scanning microscopy is used to capture 4D (depth/time) image stacks of 'Shoot Apical Meristem' (SAM) of the plant Arabidopsis. The SAM of Arabidopsis Thaliana consists of approximately 500 cells and they are organized into multiple cell layers that are clonally distinct from one another and are tightly packed with each other. At each imaging time there is no control over which depth of the meristem the imaging starts. Also at each imaging/observational time point the plant is moved and placed under

microscope to acquire the images. Because of this replacement of the plant under the microscope and the lack of control on the imaging depth the 4D dataset contains unregistered image stacks. Figure 3.1 demonstrates three consecutive stacks imaged at three different depths where the the same depth slices do not correspond. The figure also demonstrates images with noticeable shifts between images from different time instances. Automated analysis of cell lineages, cell growth dynamics requires the 4D image stacks to be registered. We will show experimental results on this particular application setting.

3.1.1 Contribution

In this work we present a fully automated spatio-temporal registration framework for registering live imaging stacks. This framework is suitable across tissues which show the characteristic of having tightly packed cells.

We have provided an optimization based framework to select the best image slice correspondence from consecutive image stacks by using the tissue characteristics in images. Also we have presented a novel landmark selection methodology where the shape features of individual cells are not of high quality and often non-discriminative. We solved the problem by choosing the relative positions and ordered orientation of the neighboring cells as unique features. We represented the local neighborhood structures of cells as graphs and selected those landmark points that have the minimum distance between the local graphs built around them.

A preliminary version of this work was presented in [38] where we assumed that the slice correspondences between Z-stacks were given. In [38] the input comprised of any two confocal image slices from two Z-stacks. The method proposed in this work is a truly automated 4D registration method as the input to this method is a complete time lapse confocal image stack. Finding

the corresponding slices needed to register is a big step in order to scale the problem and it is a fundamental problem that we are addressing in this work. Also we are looking at how the choice of the stack is reflected in the registration problem. Our analysis, presented in (Figure 3.6), shows the fundamental effect of a proper slice selection on the registration and subsequent image analysis results. We have shown the effect of using optimal image slice correspondences in the registration module and the improvements in results are evident in Figure 3.6. As explained in the Section 3.3, there is a significant increase in number of tracked cells and lengths of cell lineages, which validates the use of the slice correspondence method in the registration framework.

Also, we have improved the theoretical side of the landmark selection method over [38] by forcing a constraint into the optimization. It was done to insure feasibility on the selected set of landmark point pairs. The reason behind this change is that the rotation is governed by the rotation of the tissue, the cell independently cannot rotate by different amounts.

To show the significance of the registration and to show that our proposed registration method provides good results we considered the CLSM image stacks of tightly packed cells of a live Arabidopsis shoot meristem (SAM) (the spatio-temporal misaligments present in the sample CLSM image stack of a Arabidopsis shoot meristem are shown in Figure 3.1). We combined our proposed registration method with *local-graph matching* based robust cell tracking algorithm [32] and show significant improvement in accuracy of cell lineages and division statistics.

3.1.2 Relation to Existing Work

Amongst the different components of the general image analysis pipeline there has been some work done on image segmentation and cell tracking in live imaging stacks. For cell segmentation, both the Watershed [47] and the Level-set method [13] have been shown to provide good

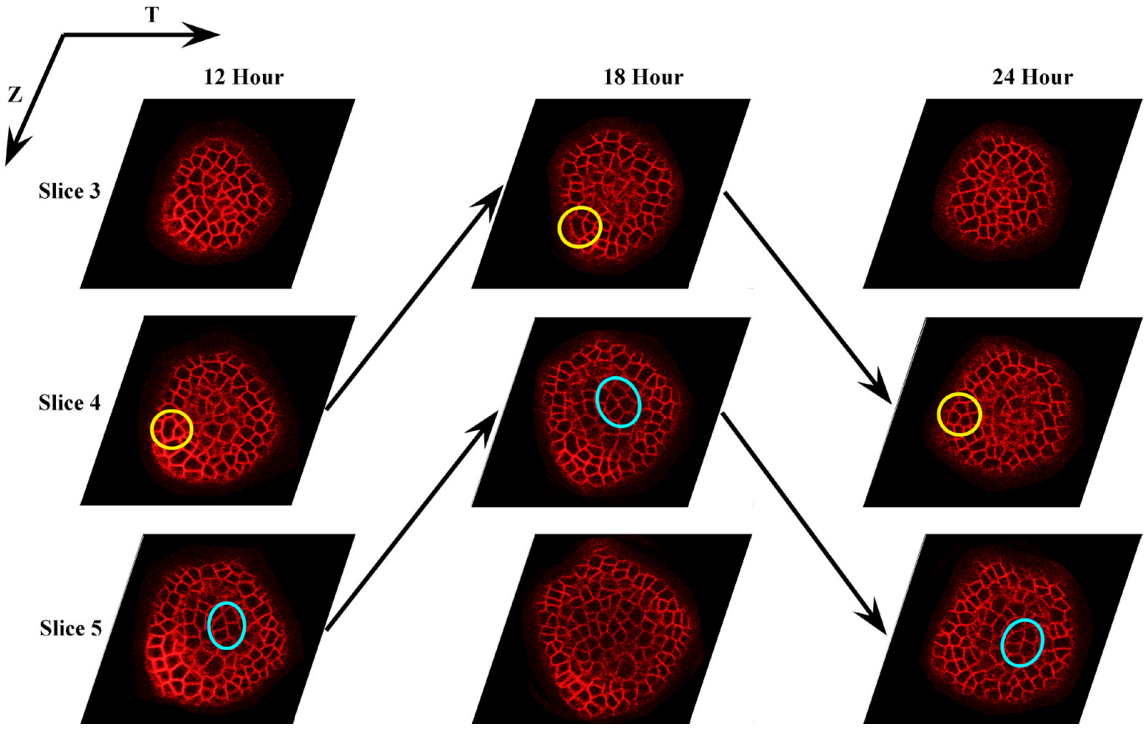


Figure 3.1: Three consecutive Z-stacks with three consecutive slices shown. The black arrows between images show the slice correspondence between Z-stacks. The same color circles show the same collection of cells through the spatio-temporal slices.

results. In general, any segmentation method, e.g. [5,31], can be used as a preprocessing step. Many tracking methods have been proposed that attempt to find correspondences between 2D segmented cells. For example, [32] exploits the local geometric structures around a cell to find correspondences between cells. [33] integrates the segmentation and tracking modules to reach an optimized segmentation and tracking result. In [15] multiangle image acquisition, three-dimensional reconstruction and cell segmentation lineage tracking approach is developed in which SAM is imaged from multiple angles. But all of these methods rely on the availability of registered image stacks of the tissue.

In many works e.g. [10, 21, 32–34, 36], the images are registered with maximization of the mutual information method [50]. But this method may not be always effective. For example,

we have looked at the confocal imagery of the Arabidopsis SAM for experiments in this work. The mutual information based tracker fails to perform satisfactorily as it uses the pixel intensities to acquire the registration. Pixel intensities in the Arabidopsis SAM images are not discriminative features. The landmark-based registration is more suitable for noisy and Z-sparse confocal images than registration based on maximization of the mutual information, which we demonstrate in the experimental results section.

A recent work [15] uses SAM images acquired from multiple angles to automate tracking and modeling. In this work, for pair of images to be registered, the user identified correspondences by pairing a few anchor points (referred as landmark points in this work). In this work, we present a fully automated method to find the corresponding temporal slices from Z-stacks of images and a fully automated landmark-based registration method that can find out correspondences between two images and utilize these correspondences to yield a better registration result. This method is suitable across tissues which show the characteristic of having tightly packed cells.

A commonly used landmark-based registration algorithm is the Iterative Closest Point algorithm [9], which is very sensitive to initialization. There are number of subsequent variations of this algorithm, e.g. Iterative Closest Point using Invariant Features [11], which uses features like eccentricity and curvature to overcome the issue. There are other registration works [18, 55] that acquire the registration using features like intensity, surfaces, and SIFT. But in tissues (e.g. Arabidopsis SAM) that are comprised of tightly packed cells with stereotypical shapes and sizes, eccentricity and other common features are not discriminative enough to be used for the selection of landmarks for registration. This is why we need to develop a novel feature to register this type of images. The novel landmark estimation method presented in this work exploits the tight spatial topology of the tissues under study and proposes a feature descriptor based on the local neighbor-

hood structure around a cell for a robust registration method.

3.1.3 Organization

The rest of the work is organized as follows. The algorithmic details of the proposed framework; finding the best image slices correspondence, feature matching in the selected images and entire Z-stack registration are provided in Section 3.2. We have shown experimental results and validation of our approach in Section 3.3 followed by concluding discussion in Section 3.4.

3.2 Detailed Registration Framework

The spatio-temporal registration of live imaging stacks comprises of three major steps: finding corresponding image slices from two consecutive image stacks, feature matching in the selected corresponding images, and aligning two stacks of images.

The next three sections present details of each step of the registration.

3.2.1 Finding Corresponding Temporal Slices

In most cases, for any two Z-stacks imaged at consecutive time intervals, the same depth image slices will not correspond. We consider all slice pairings between two consecutive Z-stacks and define distance on each pair correspondence. The distances are computed based on the tissue characteristics in images. Specifically the number of cells and the area of the tissue in the image are used to compute the distance.

Let us consider two consecutive Z-stacks $I^{(t)}$ and $I^{(t+1)}$ taken at time t and $t+1$. The image i corresponds to the confocal slice image taken at time t and at the depth h_i μm and image j, corresponds to the confocal slice image taken at time $t+1$ and at the depth h_j μm. Let $n_i^{(t)}$ and

$n_j^{(t+1)}$ be the number of cells and let $a_i^{(t)}$ and $a_j^{(t+1)}$ be the area of the tissue in images i and j, taken at time t and $t+1$ respectively. We define the distance for the image pair (i, j) to be the best pair candidate as $d(i, j)$ the following way:

$$d(i,j) = \frac{|n_i^{(t)} - n_j^{(t+1)}|}{n_i^{(t)} + n_j^{(t+1)}} + \frac{|a_i^{(t)} - a_j^{(t+1)}|}{a_i^{(t)} + a_j^{(t+1)}} \tag{3.1}$$

The constructed distance matrix d will contain distance values for all possible pairs between the two Z-stacks. We can see from the way the distances are computed that the smaller the distance corresponding to the image pair is the better image slice candidate the pair is. So the final slice correspondence between consecutive Z-stacks $I_i^{(t)}$ and $I_j^{(t+1)}$ is defined as:

$$(i,j) = \underset{i \in I^{(t)}, j \in I^{(t+1)}}{\arg\min} d(i,j) \tag{3.2}$$

Essentially, the above described problem of finding corresponding slices is analogous to the shortest path problem [12], which is to find a path between two vertices (nodes) in a graph such that the sum of the weights of its constituent edges is minimized, where in our case every slice becomes a node. So the problem of finding corresponding temporal slices from the Z-stacks can be posed as a shortest path algorithm and can be solved using Dijkstra's algorithm [51], where the shortest path is only in temporal direction.

3.2.2 Feature Matching in Selected Images

Assuming that the best image slices between two Z-stacks are chosen, we want to find the features that can be used to find corresponding landmark point pairs to register one image stack onto the other.

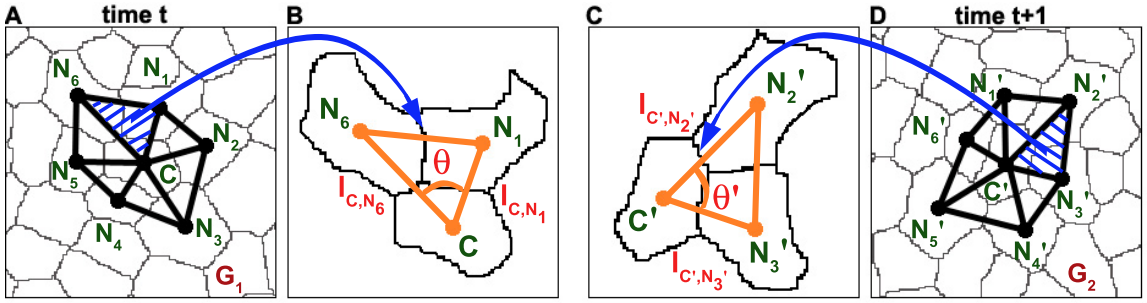
When using landmark-based registration method, the quality of the image registration result depends on the accuracy of the choice of the landmark points. Finding corresponding landmark point pairs from two images depends on the feature selection. Motivated by the idea presented in [32], we use the relative positions and ordered orientation of the neighboring cells as unique features. To exploit these properties we represent these local neighborhood structures as graphs and select the best candidate landmark points that have the minimum distance between the local graphs built around them.

Local Graphs as features

Graphical abstraction is created on the collection of cells. Vertices in the graph are the centers of the cells and neighboring vertices are connected by an edge. Neighborhood set $N(C)$ of a cell C contains the set of cells that share a boundary with C. Thus every graph consists of a cell C and a set of clockwise ordered neighboring cells (Figure 3.2 (A,D)). The ordering of the cells in $N(C)$ is important because under non-reflective similarity transformation, the absolute positions of the neighboring cells could change but the cyclic order of the cells remains invariant.

Landmark point pair estimation from local graphs

Cell divisions happen throughout the entire imaging period but at the consecutive images only a few cell divisions are present. Ideally, in the areas where there is no cell division, the local graph topology should not change (segmentation errors will circumvent this in practice). We exploit these conditions to find the corresponding landmark pairs in two images. Let $G_1^{(t)}$ and $G_2^{(t+1)}$ be two local graphs constructed around the cells C and C' in consecutive temporal slices (Figure 3.2).



(A) The local graph G_1 at time t with the central cell C and clockwise ordered neighboring cell vertices $N_1, N_2, \ldots, N_6$

(B) The local graph G_2 at time $t+1$ with the central cell C' and clockwise ordered neighboring cell vertices $N'_1, N'_2, \ldots, N'_6$

(C) Triangle subgraph of the graph G_1 with features

N_{i_1}, N_{i_2} - i_1^{th} and i_2^{th} neighboring cells of C

$\theta_{N_{i_1},C,N_{i_2}}$ – angle between $\overline{N_{i_1}C}$ and $\overline{N_{i_2}C}$

$l_{C,N_{i_1}}(t)$, $l_{C,N_{i_2}}(t)$ - neighbor edge lengths

$A_{N_{i_1}}(t), A_{N_{i_2}}(t)$ - areas of the cells N_{i_1}, N_{i_2}

(D) Triangle subgraph of the graph G_2 with features

N'_{j_1}, N'_{j_2} - j_1^{th} and j_2^{th} neighboring cells of C'

$\theta_{N'_{j_1},C',N'_{j_2}}$ – angle between $\overline{N'_{j_1}C'}$ and $\overline{N'_{j_2}C'}$

$l_{C',N'_{j_1}}(t+1)$, $l_{C',N'_{j_2}}(t+1)$ - neighbor edge lengths

$A_{N'_{j_1}}(t+1), A_{N_{j_2}}(t+1)$ - areas of the cells N'_{j_1}, N'_{j_2}

Figure 3.2: Local graphs and two enlarged triangle subgraphs with indicated features.

For each subgraph of the local graph $G(t)$, we define feature vector the following way;

$$F^C(t) = [f_1 f_2 f_3 f_4 f_5]^T, \text{where}$$

$$f_1 = \theta_{N_{i_1},C,N_{i_2}}(t),$$

$$f_2 = l_{C,N_{i_1}}(t), f_3 = l_{C,N_{i_2}}(t),$$

$$f_4 = A_{N_{i_1}}(t), f_5 = A_{N_{i_2}}(t)$$

We define the distance between two triangle subgraphs as

$$D_{TS}\left(F_i^C(t)), (F_j^{C'}(t+1))\right) = \sum_{k=1}^{5}\left(\frac{f_k - f'_k}{f_k}\right)^2, \quad \text{where } f_k \in F_i^C(t), f'_k \in F_j^{C'}(t+1). \tag{3.3}$$

To ensure that our landmark estimation method takes care of the rotation of the local area, we consider all cyclic permutations of the clockwise ordered neighbor set $\{N'_1, N'_2, \ldots, N'_m\}$ of the cell C' from the input image. The cyclic permutations of the set $\{x_1, x_2, \ldots, x_m\}$ can be written

in terms of the shift k $(k = 0, 1, \ldots, (m-1))$ as the set $\{x_{(1+k-1)_{mod(m)}+1}, x_{(2+k-1)_{mod(m)}+1}, \ldots, x_{(m+k-1)_{mod(m)}+1}\}$. As an example, if $(1, 2, 3)$ is the given sequence, then possible values of the shift $k = 0, 1, 2$ and all the cyclic permutations of the sequence $(1, 2, 3)$ will be $(1, 2, 3)$, $(2, 3, 1)$, $(3, 1, 2)$ for $k = 0, 1, 2$. We consider all cyclic permutations of the clockwise ordered neighbor set $\{N'_1, N'_2, \ldots, N'_m\}$ of the cell C' from the input image and define the distance $D(G_1, G_2^k)$ between two local graphs G_1 and G_2 based on the chosen permutation corresponding to shift k as

$$\begin{aligned} D(G_1, G_2^k) &= \sum_{\{i,j\}} D_{TS}\left(F_i^C(t)), (F_j^{C'}(t+1))\right) \\ \forall i \in \{1, 2, \ldots, m\}, &\quad j = [(i + k - 1)_{mod(m)} + 1] \end{aligned} \tag{3.4}$$

for $k \in \{0, 1, 2, \ldots, (m-1)\}$. We compute the sum of the distances between each of the ordered pairs of triangle-subgraphs for each permutation k.

It is important to notice that in the calculation of the distance $d(i, j)$ (Eqn. 3.1), which is used to find corresponding temporal slices, feature vector F, which is part of landmark point pair estimation, is not used. The reason is that the image stack slices are very stereotypical; there can be different local graph patters which can be similar in shape but be from different slices. In confocal imaging a cell is imaged in multiple Z slices along time. This may result in having all cells in a cluster of tightly packed cells being imaged at slices $z, z+1, z+2$ and at time points t and $t+1$. If we want to use this micro feature F to choose the corresponding temporal slices, essentially there is a discrimination why slice z at time point t should match with slice z at time point $t+1$. Because there is no difference in the cell cluster, it is equally possible that slice z at time point t can also match with slice $z+1$ at time point $t+1$. This feature vector F is kind of invariant across Z, it discriminates cells spatially in one slice along time but it is not a discriminative enough feature

to understand which slice should match to which slice as the same local graph (cell cluster) may be preserved along different depths. So these components of the feature vector F are micro level features and they do not capture the slice information. That is the reason why F does not capture the effect of the depth at which the cells are imaged and is not used to find the corresponding temporal slices.

The minimum distance $D^{\star}(G_1, G_2)$ between two graphs G_1 and G_2 corresponding to cells (C, C') for all permutations k is

$$D^{\star}(G_1, G_2) = \min D(G_1, G_2^k) \tag{3.5}$$

where $k \in \{0, 1, \ldots, (m-1)\}$. This guarantees that our landmark estimation method is invariant of the rotation in the local area.

For all cell pairs C_i, C'_j and corresponding graphs G_i, G_j from two consecutive images, we compute the distance $D^{\star}(G_i, G_j)$. Now, our objective is to obtain the set of best q cell pairs for which the local graphs around these cell pairs are maximally similar. However, for each of these chosen cell pairs, the optimal shift (corresponding to the minimum distances between the graphs) must result in approximately similar angles of rotation. This uniformity between the rotation of the individual graphs is obvious as all the cells are tightly packed and the rotations of individual cell clusters (the 'local graphs') are uniformly affected by the global rotation of the entire tissue. Thus, as a corollary, if any set of q cell pairs show large variations in the optimum angle of rotation, the set must contain one or more incorrectly associated local graph pairs.

Measurement of Angle of Rotation Between Pairs of Graphs

Between two graphs G_1 and G_2, the circular shift k^* corresponding to the minimum distance $D^*(G_1, G_2)$ is given as,

$$k^* \ = \ \arg\min_k D(G_1, G_2^k) \tag{3.6}$$

Assuming that the rotated graph $G_2^{k^*}$ is a candidate match for the graph G_1, the angle of rotation $\Theta^*(G_1, G_2) \ = \ \Theta(G_1, G_2^{k^*})$ could be computed as follows.

Let in graph G_1, the central cell is C and the set of neighboring cell slices around C being $\{N_1, N_2, \cdots N_m\}$. Likewise, the central cell in the graph G_2 is $C^{'}$ and the neighbors around it are $\{N_1^{'}, N_2^{'}, \cdots N_m^{'}\}$. Now, if $G_2^{k^*}$ is the transformed version of G_1, then the amount of rotation would be obtained as,

$$\Theta(G_1, G_2^{k^*}) \ = \ \frac{1}{2\pi}\left(\theta_{N_{k^*+1}^{'},C^{'},Y^{'}} \ - \theta_{N_1,C,Y}\right) \tag{3.7}$$

The angles $\theta_{N_1,C,Y}$ and $\theta_{N_{k^*+1}^{'},C^{'},Y^{'}}$ are described through Figure 3.3. The angle $\Theta(G_1, G_2^{k^*})$ represents the amount of counterclockwise rotation that G_2 must undergo with respect to $C^{'}$ to align $\overline{N_{k^*+1}^{'}C^{'}}$ with $\overline{N_1C}$ in G_1.

Optimum Set of Landmarks: Simultaneously Minimizing Dissimilarities And Variance In Angles of Rotation Between The Local Graphs

The final objective is to obtain a set of q cell pairs from the two image slices such that the individual pairs of local graphs around those slices are maximally similar as well as the variation in the angles of rotation in this set is least. This set, denoted as $s_q^* \ = \ \{(G_{i_1}, G_{j_1}^{k_1^*}), (G_{i_2}, G_{j_2}^{k_2^*}), \cdots$

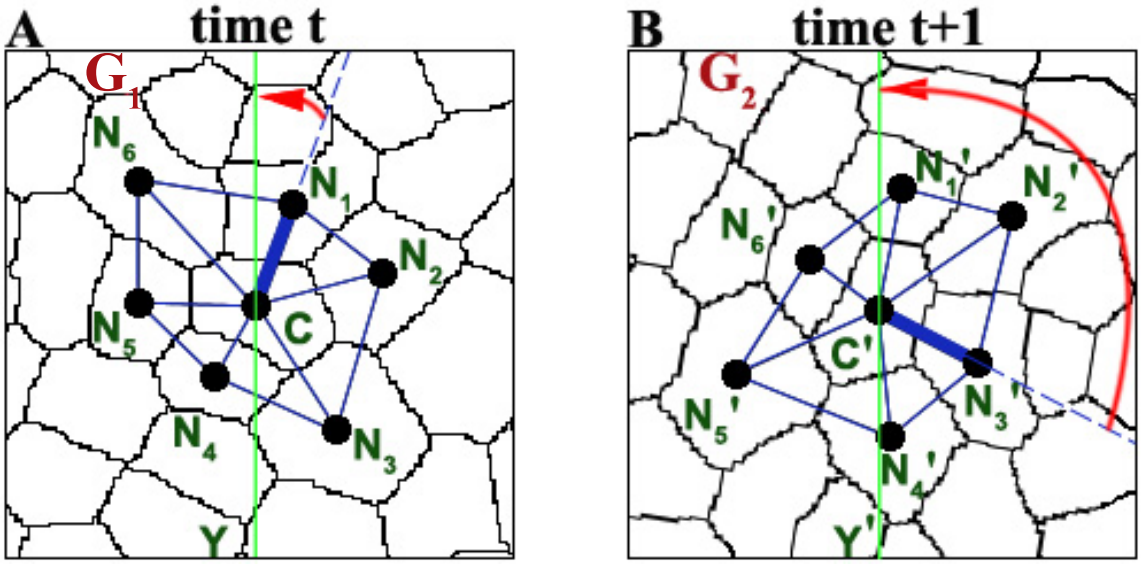


Figure 3.3: Measuring the angle of rotation of a local graph. (A) A cell cluster at any given time instant t, centered around the cell C. The neighboring cells $N_1, \cdots N_6$, along with the central cell C constitutes the nodes of the local graph G_1. (B) The same cell cluster at time $t+1$. The graph G_2 is a rotated version of G_1. The optimum shift(k^*) is estimated as 2 and hence $\overline{N_1C}$ is rotated clockwise to $\overline{N_3'C'}$ in G_2. Therefore, the relative normalized angle of rotation between G_1 and G_2 can be computed as $\Theta^*(G_1, G_2) = \frac{1}{2\pi}\left(\theta_{N_3',C',Y'} - \theta_{N_1,C,Y}\right)$.

$(G_{i_q}, G_{j_q}^{k_q^*})\}$, is the set of landmark pairs required for registration of the image pairs, where the individual landmarks are the centroids of the 2D cell slices.

Let the superset of all such possible q-pairs be $\{s_q^1, s_q^2, \cdots s_q^V\}$, and let it be denoted by $\mathcal{S}$. For any candidate set for landmarks $s_q^l \subset \mathcal{S}\ \forall l = 1, 2, \cdots V$, the average normalized dissimilarity between all pairs of local graphs in this set would be,

$$\overline{D_l^*} = \frac{1}{q} \sum_{(G_1,G_2)\in s_q^l} \frac{D^*(G_1, G_2)}{D_{max}^l} \tag{3.8}$$

where, $D_{max}^l = \max D^*(G_1, G_2)\ \forall\ (G_1, G_2) \in s_q^l$. The normalization is done to scale the average value between 0 and 1.

The variance in the angles of rotation over the set s_q^l is computed as,

$$\overline{\sigma}^2_{\Theta,l} = \frac{1}{q} \sum_{(G_1,G_2)\in s^l_q} \Theta^*(G_1,G_2)^2 \; - \left(\frac{1}{q} \sum_{(G_1,G_2)\in s^l_q} \Theta^*(G_1,G_2) \right)^2 \tag{3.9}$$

The overall cost function is defined as a weighted sum of $\overline{D}^*_l$ and $\overline{\sigma}^2_{\Theta,l}$ and the best set of q landmark pairs is estimated by minimizing this cost function.

$$s^*_q = \underset{s^l_q \in \mathcal{S}}{\arg\min} \left\{ w\, \overline{D}^*_l + (1-w)\, \overline{\sigma}^2_{\Theta,l} \right\} \tag{3.10}$$

The weight w can be user-defined. Since the feature distance is the main parameter in finding the landmark point pairs, we varied the value of w between 0.5 and 1. The registration results were not changed. For all our experiments, the w is fixed at 0.5. The choice of q is described later in the Section 3.2.3.

Once we have the landmark point pairs, the positions of the landmark pairs in the two images are used to estimate the parameters of the transformation model (affine, non-reflective similarity etc.).Then the estimated transformation function maps the rest of the points in the input image to the reference image. The Section 3.2.3 presents details on computing the spatial transformation between the images and final image registration.

3.2.3 Image Registration

After the corresponding image slices across two stacks of images taken at two consecutive time points are selected and the corresponding landmark point pairs in the selected images are acquired, we find the spatial transformation between these images. Then we use the estimated

transformation to register the entire Z-stacks.

Finding the spatial transformation between two images

Once we have the landmark point pairs corresponding to the input (the image that we wish to transform) and reference (the image against which we want to register the input) images, we find the spatial transformation between them. Transformations present in the live image stack of a multilayer, multicellular structure where the cells are tightly packed together, include a rotation and a translation. The non-reflective similarity transformation is chosen as a type of transformation to proceed with. Finding the nonreflective similarity transformation between two images is a problem of solving a set of two linear equations. For better accuracy of transformation parameters the top q landmark point pairs are used in a least square parameter estimation framework. But, as known, greater the number of landmark point pairs is, better the estimated solution of the Least Square technique is. In order to choose the best landmark point pairs we rank them according to the similarity of local graphs created on the neighboring cell structure. Choosing more landmark points will mean finding transformation between two images based on points that have less possibility to be correct corresponding points, which will eventually lead to bad registration. So there is a trade-off between these aspects. In our experiments we choose four, five or six landmark point pairs depending on the CLSM dataset image quality.

Z-stack Registration

Depending on the procedure the living tissue imaging is done, the transformations between corresponding image slices from consecutive image stacks are the same. So the transformation parameters corresponding to the selected image slices from these Z-stacks will also be the

transformation parameters between the rest of the image pairs in these two Z-stacks. So, by applying the computed transformation parameters for the selected image slices on the rest of the image pairs from the two Z-stacks, the entire Z-stack taken at time point $t + 1$ will be registered to the Z-stack taken at time point t.

3.3 Experimental Results

3.3.1 Imaging Setup

For the experiments performed in the present study, the 3D structure of the tissues are imaged using single-photon confocal laser scanning microscope and we have specially dealt with the 'Shoot Apical Meristem' (SAM) of the plant Arabidopsis. The SAM of Arabidopsis Thaliana consists of approximately 500 cells and they are organized into multiple cell layers that are clonally distinct from one another and are tightly packed with each other. By changing the depth of the focal plane, CLSM can provide infocus images from various depths of the specimen. To visualize cell boundaries of all the cells in the SAM, plasma membrane-localized Yellow Fluorescent Protein (YFP) is used. The set of images, thus obtained at each time point, constitute a 3-D stack, also known as the 'Z-stack'. Each Z-stack is imaged at a certain time interval (e.g. 3 or 6 hours between successive observations) and it is comprised of a series of optical cross sections of SAMs that are separated by approx. 1.5 μm, and a standard shoot apical meristematic cell has a diameter of about 5 - 6 μm.

At each imaging time there is no control over in which depth of the meristem the imaging starts, so in each Z-stack the first slice imaged is at random depth and the rest of the images in the stack are 1.5μm deep from each other. This means that for any two Z-stacks imaged at consecutive

time intervals, in most cases, the same depth slices of the specimen won't correspond, there will be a shift in the slice correspondence. Figure 3.1 demonstrates three consecutive stacks imaged at three different depths where the the same depth slices do not correspond.

In practice, the live cell imaging of Arabidopsis SAM comprises of several steps, where the plant has to be physically moved between different places. For normal growth of the plant, it has to be kept in a place having specific physical conditions (such as a temperature of 24^oC). The plant is moved and placed under microscope at the imaging/observational time points, before it is placed back to the aforementioned place once again. For 72 hours overall, this process is repeated every 3 hours. Because of this process of replacement of the plant under the microscope and also since the plant keeps growing during these 72 hours, various shifts can occur between two Z-stacks of images taken in consecutive time points, though images in any Z-stack are automatically registered. Figure 3.1 demonstrates images from a data set with noticeable shifts between images from different time instances.

3.3.2 Experimental Results and Analysis

We further evaluate the performance of the proposed automatic spatio-temporal registration and show numerical results. As the numerical results as direct comparison of registration results are not trivial we evaluate the tracking accuracy after and before registration and show the performance improvement.

We have tested our proposed automatic spatio-temporal registration of live imaging stacks method on different datasets that consist of 4D image stacks taken at six hour intervals for 72 hours overall. First for each two consecutive Z-stacks of every dataset, we run our method to find the corresponding image slices. Then using the image correspondence information from corresponding

Z-stacks we chose a temporal stack and applied the watershed segmentation [47] and local graph matching based tracking [32] methods and compared the obtained results. We compared tracking results of the proposed method with results obtained without registration, with semi-automated registration (the landmark pairs are chosen manually, the transformation is obtained automatically) and with Matlab registration module which is based on the maximization of the mutual information.

Pairwise Tracking - Figure 3.4 (A-E) shows cell tracking results from two consecutive images (30^{th} and 36^{th} hour), obtained with different approaches. We can see that for the chosen two consecutive images (30^{th} and 36^{th} hour) the results with maximization of the mutual information based registration and without registration show incorrect cell tracks. Whereas the proposed method and semi-automated registration correctly registered two images with 100% correct tracking results. Detailed results for the same dataset are shown in Figure 3.5. We can see that from 33 and 27 cells, present in the images at time points 5 (30^{th} hour) to 6 (36^{th} hour) respectively, none are tracked by the tracker run on the images registered with the registration based on the maximization of the mutual information and not registered images (as in Figure 3.4 (A-E)). The same result is seen for the tracking results in images at time points 6 to 7. But the tracking results obtained with proposed and semi-automated methods provided very close to manual results.

Image Slice Correspondence Figure 3.6 shows two different tracking result comparisons on a temporal stack between the case where the temporal stack is constructed from the proposed automated image slice correspondence module and the case where from each Z-stack the same slice image is chosen. Figure 3.6 A) shows the number of tracked cells across each two consecutive images in the temporal stack and Figure 3.6 B) shows the lengths of cell lineages (for how many consecutive frames the same cell was tracked). All of these results are manually verified. We can see significant increase in number of tracked cells and lengths of cell lineages in the case where the

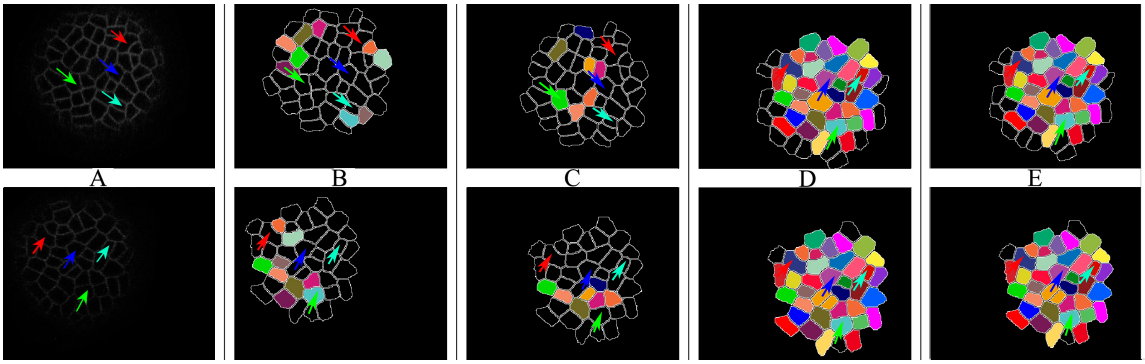


Figure 3.4: A) Raw consecutive images (the same color arrows represent the same cells). Tracking results obtained B) without registration C) with registration based on maximization of the mutual information, D) with semi-automated registration (the landmark pairs are chosen manually, the transformation is obtained automatically), E) with proposed automatic registration. The same colors represent the same cell.

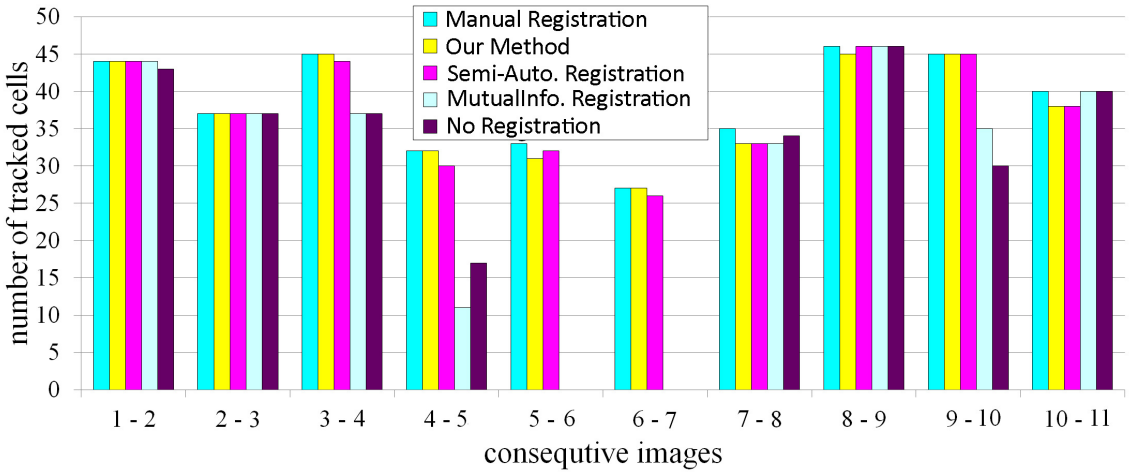


Figure 3.5: Number of tracked cells across two consecutive images.

dataset is constructed from our proposed image slice correspondence module which validates the use of the slice correspondence module in the registration framework.

Pairwise Registration - The result of the registration depends on the clearness of the images. Although, in the proposed method, we need very few landmark points we cannot control the regions of the images in which the landmark points may appear, so in the process of the selection of the landmark points we might take false positive landmarks which will result in pure registration. Figure 3.7 shows three pairwise registration results with images from three different depths, with different SNRs. As we can see the column A) in the figure shows three different images taken at hour 18 with different SNRs sorted from image with high to low SNR accordingly. Images in the column B) are registered to the images in the column A) accordingly and column C) shows the same image after registration. As the white arrows in the figure show we have registration with high tracking error in the images from the first row (high SNR), registration with medium tracking error in the images from the second row (midium SNR), and registration with low tracking error in the images from the third row (low SNR). These results can also be clearly seen in the columns D) and E) which show the fused images before registration and after registration. This shows the robustness of our algorithm.

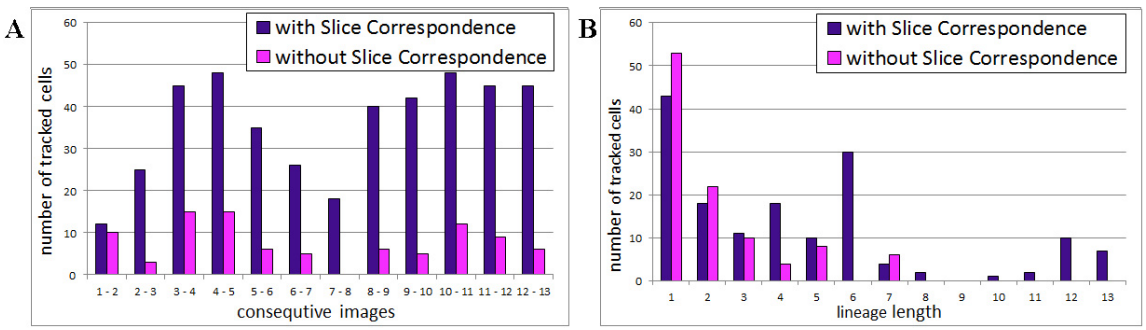


Figure 3.6: Tracking results on a temporal image stack where the images are chosen from the Z-stacks based on the proposed slice correspondence and where the same slice is chosen from all Z-stacks - A) Number of tracked cells across two consecutive images, B) Length of cell lineages.

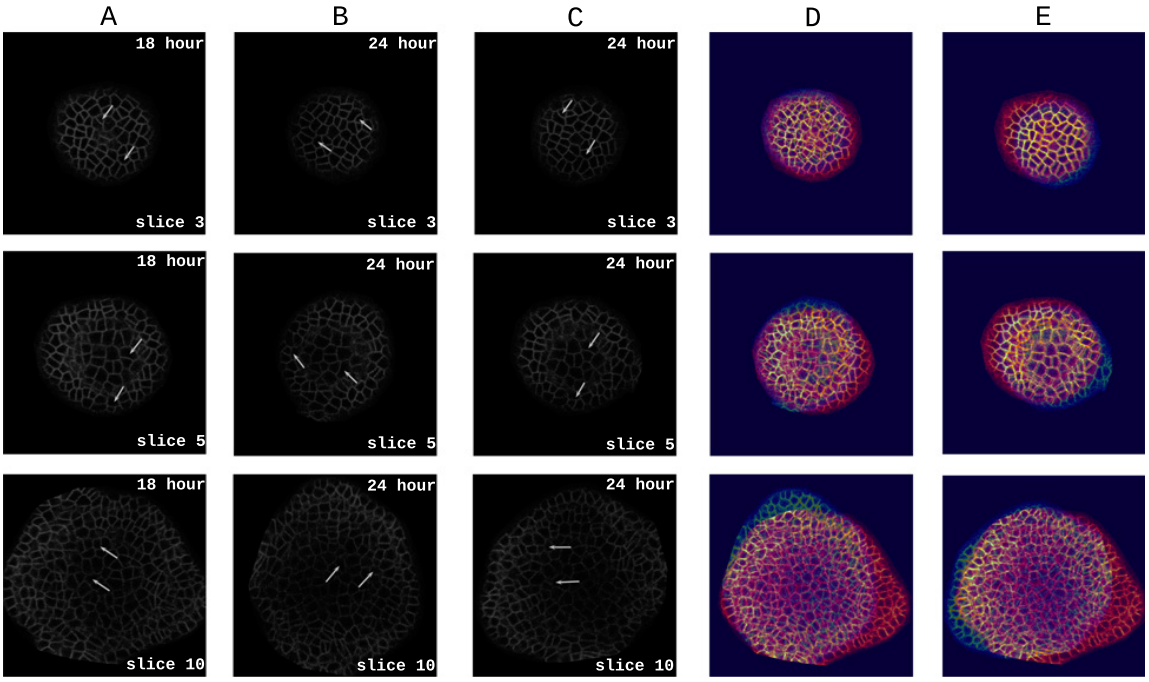


Figure 3.7: Registration results on three pairwise images from three different depths and with different SNRs. Column A), B) show the raw images from the hour 18 and 24 correspondingly, column C) shows the image in the column B) after registration, and column D) and E) show the fused images before registration and after registration (white arrows show the same cells)

Lineage Analysis - Figure 3.8 shows lengths of the cell lineages calculated with the proposed method, semi-automated registration, registration based on maximization of the mutual information and without registration. These results contain numbers from two different datasets; three hour dataset(G) and six hour dataset, where six or three indicate the interval between two successive imaging of Z-stacks. Longer the lineages are, better the results are. We can see that in tracking without registration and after registration with maximization of the mutual information, there are no cells that have lineage lengths greater then four (Figure 3.8 (A)), greater then eight (Figure 3.8 (C)) etc., as opposed to the case with the proposed and semi-automated registration, where cells have lineages for the entire 72 hours. The reason for such results is that there is a big shift between two images from consecutive time points in the middle time points. Without proper registration

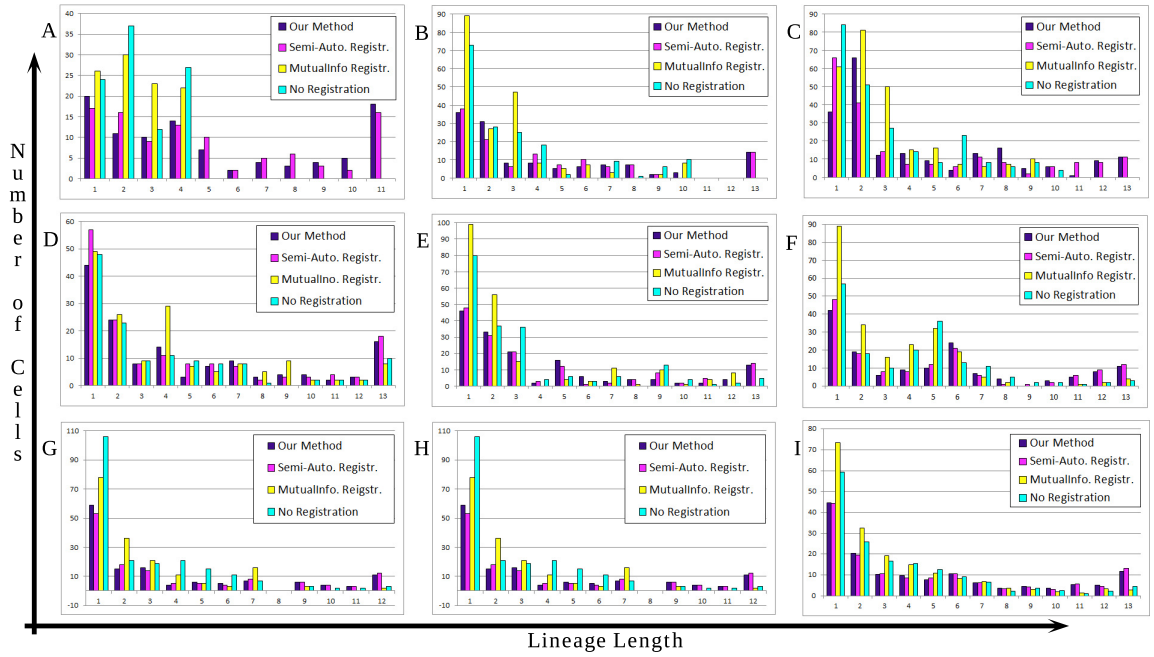


Figure 3.8: Length of cell lineages for different datasets A-H. Each bar in each graph represents how many cells (number of cells) have that lineage length. I) Average length of cell lineages across all eight datasets.

the tracking algorithm is not able to provide correct cell correspondence results, which interrupts the lineage of the cells. Figure 3.8 (I) shows the length of lineage length averaged across all eight datasets. We can see that on average we find more cells with longer lineages with our method and the semi-automated method compared to the results with registration with mutual information and without registration, where those long lineages are found in shorter peaces. Figure 3.8 (A) result can be also related to Figure 3.5 since they are representing statistics from the same dataset. Since no cells have been tracked in frames five to six and overall there are eleven frames, then no cell can have a lineage life with the length greater than or equal to five.

Table 3.1 shows the number of cell divisions in 72 hours. We can see that the semi-automated and the proposed registration provide results that are close to the manual results as opposed to without registration and registration based on maximization of the mutual information.

Table 3.1: Total Number of Cell Divisions/Ground Truth

Data	OurMethod	Semi-Auto.	MutualInfo.	NoRegistration
1	28/34	30/34	23/34	25/34
2	17/21	17/21	11/21	12/21
3	15/15	15/15	9/15	9.5/15
4	11.5/16	12.5/16	9.5/16	10/16
5	21/23	23/23	17/23	19/23
6	12.5/16	13.5/16	9/16	9.5/16

3.3.3 Discussion on the Limitations of the Proposed Method

The accuracy of the proposed image registration method depends on quality of the images and on the consistency of the cell neighborhood structure. As we have shown in our experiments, the proposed registration method can handle moderate deformations of the growing cells. However, if the deformation changes the topology of cells local neighborhood, it becomes more challenging and in some cases leads to failure of the registration module.

The image registration is acquired after the cell segmentation module, so the registration results rely on the cell segmentation results. In general in confocal microscopy based live cell imaging there could be situations where a part of the image is noisier than the other parts, especially the central regions, and hence the segmentation results sometimes are not as good as in the deeper layers. But in the proposed method we try to overcome this situation by providing landmark based registration, where it is not the entire image that is used to do the registration. We essentially find the best points from multiple images which could be used as landmark pairs. Often these corresponding points come from the better segmented regions (clearer sections of images) and it is highly unlikely that we find a very good match in the blurred section. So the proposed method is robust to noise in part of the image and segmentation errors to a certain extent; however for a very noisy image where the segmentation is not satisfactory over the whole image the registration method is quit affected by

such cases.

As shown in Figure 7, for well segmented or clearer images (Fig. 7 row 1) we have a very good registration, for not so well segmented or not so clean images (Fig. 7 row 2) we also have good registration, but where the noisy region is almost everywhere (Fig. 7 row 3) (there are more chances of finding bad landmark pairs because the nosy region is larger), we can see that the registration results are not as good as the previous cases.

Future work can consider other registration approaches to improve the results, including non-linear registration methods.

3.4 Conclusion

Automated image analysis such as registration, segmentation and tracking of cells in actively developing tissues can provide high-throughput and quantitative spatiotemporal measurements of a range of cell behaviors; cell expansion and cell-division kinetics, which will lead to a better understanding of the underlying dynamics of morphogenesis. In this work, we have described an automated spatio-temporal registration method of 4D live imaging stacks of tightly packed cells, and this method is suitable across tissues which have this spatial organization of the cells in a neighborhood. The proposed registration method first finds the best image slice correspondence from consecutive image stacks. Then our proposed landmark-based registration method uses local graph-based approach to automatically find corresponding landmark pairs in the images, and finally the registration parameters computed on the selected image pairs are used to register the entire image stack. The proposed registration algorithm combined with an existing tracking method is tested on multiple confocal image stacks of Arabidopsis shoot apical meristem and the results show that it

significantly improves the accuracy of cell lineages and division statistics.

Chapter 4

Automated count of mosquitoes

4.1 Introduction

Object counting is a very common task frequently encountered in biology, remote sensing and surveillance data analysis. The counted objects are used to do quantitative and qualitative analysis. Many methods rely on manual counting of the objects, which is extremely tedious and error-prone. Automatic counting of objects has been done in many different applications, such as counting cells in microscopic images [2,3,19,49], monitoring crowds in surveillance systems [6,23], counting the number of trees in an image of a forest [8, 16], fish population estimation and fish species classification in marine science research [20, 41], etc. Automatic counting becomes challenging when different objects are not easily distinguishable, and surrounded by noisy background. In addition, different lab conditions bring in new scenarios for which available counting methods are not applicable and new approaches are needed. In this work we present an automated object counting method for counting mosquitoes captured in video. We demonstrate our methods on multiple videos collected by entomologists in their labs and which have been, hitherto, analyzed manually.

Mosquitoes and other blood-feeding insects transmit deadly diseases such as malaria, West Nile fever, Yellow fever, and sleeping sickness to hundreds of millions of people, causing severe suffering and more than a million deaths each year. Biologists perform different experiments on mosquitoes to find solutions to this very important problem. One of these experiments aims to discover the reaction of mosquitoes around different odors, for which a special environment called arm-in-cage is created and the mosquitoes' behavior is captured in a video. The count of the mosquitoes is used to describe the interest of the mosquitoes to each odor. In this work we develop an automated object counting method that efficiently counts the number of mosquitoes. As shown in section 4.4 our automated method provides very close to manually obtained ground-truth results.

In our automated approach the softmax classifier with a sliding window technique is used to detect mosquitoes. The over-detection resulting from the sliding window technique is eliminated by the NMS (non-maxima suppression) method. The results of the NMS method correspond to the final mosquito counts. Currently, the counting of the mosquitoes is done manually by biologists, but the continuous counting by the human observer is time-consuming, leads to eye fatigue and affects the accuracy of results. In addition, the available experimental setup gives the opportunity to gather a lot of data whereas only a small amount is being analyzed by the biologists, resulting in slowing down the findings of important discoveries. Our automated mosquito counting method will help biologists to make inferences from their entire data, and discover new and useful findings about mosquitoes that could have a big impact on human health.

The rest of this work is organized as follows: section 4.2 reviews the related works. Section 4.3.2 describes the methodology of our work. Section 4.4 contains experimental results and analysis. Finally, section 4.5 concludes this work.

4.2 Literature Review

Object counting has huge applications in a wide variety of fields, where different types of objects (cells, people, insects, flies etc.) are counted. Object counting becomes challenging when different objects are not easily distinguishable, vary in size and surrounded by noisy background. Several methods for automatically counting the objects of interest have been proposed [29] which can be divided into the following categories; counting by detection, counting by regression, counting by similarity grouping. Counting by detection approaches depend on object detection in the image, where the number of detections represents the final count of objects. The main procedure of object detection is to detect the objects in a local area and then clear out those detections into an individual one for each object by applying non-maximum suppression [14], thresholding, or making some assumptions about the relations of the object parts [7, 28, 52]. Segmentation (level set, watershed [42], etc.) and background subtraction are another approaches to be used to detect objects in an image [6, 27, 54]. The simplest background subtraction technique is capturing the same scene with the objects present and absent, under the exact same conditions, and then subtracting the first image from the second. Background subtraction technique is very sensitive to background changes and may produce miss detections. These counting by detection methods achieve the best results with situations when there are small number of objects in images. To avoid the object detection in the counting problem regression analysis is used. In case of counting by regression, a direct mapping is learned between image characteristics and the number of objects [26]. Counting by similarity grouping is mostly acquired by self-similarity grouping [4] or motion similarity grouping [44], which is especially useful in counting tasks that are too challenging for object detection, such as crowded scene [6, 23]. However, this counting methods have very limited accuracy.

In [22], the authors present a detailed review of different kinds of tracking methods for tracking insects under different lab conditions. The conditions that we have are very different from any of these methods. *3DTracker*, *Ctrax*, *Flydra*, *idTracker* softwares from [22] require contstant and uncluttered background which is not the case with our dataset and results in need to generate a different approach. To detect the objects for further tracking, [22] uses background subtraction method which is applicable with constant background. In [43], the authors use laser systems for identification and tracking flying insects, detects the insects in the images by background subtraction method as well. The videos, that are generated with arm-in-cage experimental setup contain small mosquitoes that are moving on an odor treatment mesh that has very small openings, so any small errors in background subtraction method will lead to mosquito miscount. In this work, we provide a different approach to identify and count mosquitoes in these videos.

ImageJ sofware [1] has been used to count different objects [25, 37, 48]. We have used ImageJ to count number of mosquitoes from our videos and compared the results with our results obtained with automated method provided in this work. Our automated method performed better on different aspects: our method provided more accurate results, our method is completely automated, whereas with imageJ majority of the steps are manual.

4.3 Methods

In this section we present the special setup of our experiments, which is followed by detailed explanation of our methodology.

4.3.1 Experimental Setup

Arm-in-Cage experiments were performed with 40 mated A. aegypti female mosquitoes held in 30cm $\times$ 30cm $\times$ 30cm cages with a glass top. A human arm was inserted in a glove containing a window covered with a double-layer of netting. Attraction towards the arm was measured using video recordings which was done through the glass top. Each test compound solution was applied evenly to a white rectangular 7cm $\times$ 6cm polyester netting (mesh size 26 $\times$ 22 holes per square inch) in a glass petri-dish and suspended in the air for 30 minutes to allow solvent evaporation. A nitrile glove (Sol-vex) was modified such that a 5.8cm $\times$ 5cm window was present for skin odour exposure. A set of magnetic window frames were designed to secure the treated net $\sim$ 1.5 mm above the skin, and a second untreated netting $\sim$ 4.5 mm above the treated net in a manner so that mosquitoes were attracted to skin emanations in the open window but unable to contact treated

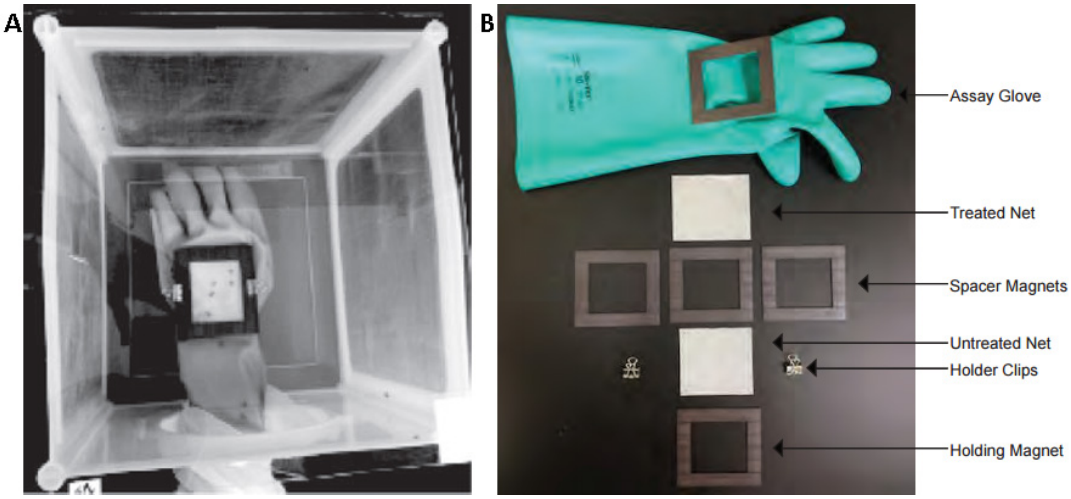


Figure 4.1: A) Arm-in-cage experimental setup, B)Assembly of the glove: a rubber glove with a window cut into the hand, a magnet glued around the cut window, a control or test odor treatment mesh, three spacer magnets that prevent mosquitoes from biting through to the hand, untreated mesh to prevent mosquitoes from touching the treated mesh, and finally a top magnet. One metal clip is then used on each side of the stack to further reinforce the arrangement of magnets and mesh.

nets with tarsi (leg segment), or contact and cut or pass through skin (Figure 4.1). Additionally the

test compound had minimal contact with skin. A clean set of gloves and magnets were used for every trial. Care was taken that the experimenter did not use cosmetics, soap, etc. on arms. For each trial the arm was inserted for 5 min and the number of mosquitoes landing or escaping the test window was recorded on video for a 5-min period. No cage was tested more than once within 1 hour of a testing session and not more than twice on any single day. The details of the experimental setup are presented in [24].

In Figure 4.2 are shown three different examples from three different videos of arm-in-cage experimental setup.

4.3.2 Methodology

We now present a detailed description of the proposed methodology. A block diagram of the proposed system is shown in Figure 4.3. First, from each frame the ROI (Region of Interest) is extracted. Next, the mosquitoes present in the ROI are detected by scanning a classifier across the image in both horizontal and vertical directions. This stage may result in generating over-detections for a mosquito. In the third stage non-maxima suppression is applied to remove over-detection of each mosquito, and finally the number of mosquitoes is counted.

4.3.3 ROI Extraction

As can be seen in Figure 4.3A, our interest is in counting the number of mosquitoes in the mesh area of the frame, inside the black magnet ends. To find that ROI from a frame we apply the Sobel edge detector. The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial frequency that correspond to edges.

The Sobel operator consists of a pair of 3×3 convolution kernels where one kernel is the

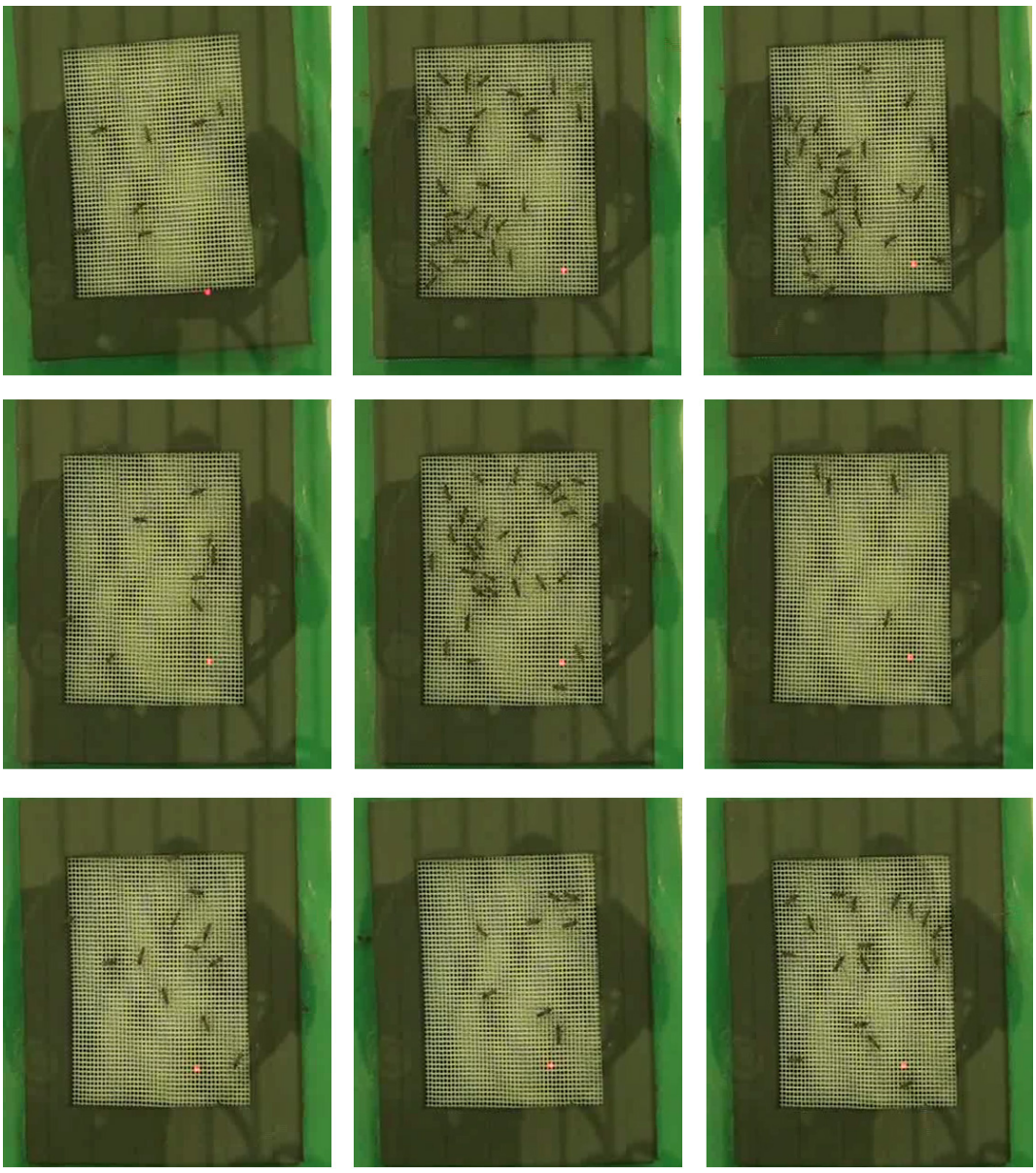


Figure 4.2: A) Different frames from three different videos. Each row is presenting frames from the video and each column is a frame captured during the first minute, third minute and fifth minute accordingly.

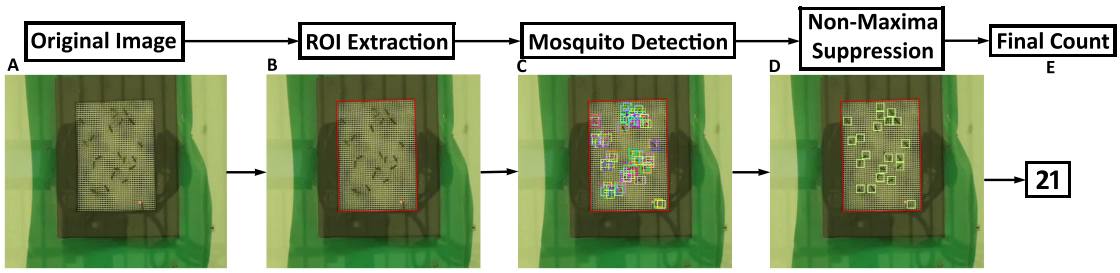


Figure 4.3: Block diagram of the proposed system: A) original image, B) result of ROI extraction from the original image, C) initial mosquito detections, D) non-overlapping detections of mosquitoes, E) final count of mosquitoes in the original image.

90 rotation of the other. Each of these kernels are applied separately to the input image and they produce measurements (G_x and G_y) of the gradient component in each orientation (horizontally and vertically). Then these two gradient measurements are combined together to find the absolute magnitude of the gradient (G) at each point:

$$G_x \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}, \qquad G_y \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix},$$

$$\mid G \mid = \sqrt{G_x^2 + G_y^2}. \tag{4.1}$$

Figure 4.4 shows in detail all the steps of the ROI extraction applied to a frame from our dataset. As can be seen in Figure 4.4B, applying of the Sobel edge detection on our dataset separates the ROI from the rest of the image, but the edge found is not a closed outer boundery edge of the ROI. Next we find the convex hull of the detected edge points (Figure 4.4C) which computes the outer boundery lines of the ROI. We finally compute the minimum bounding quadrilateral of points representing the convex hull of the ROI (Figure 4.4D) which ends up being the quadrilateral that

tightly contains the ROI. The rest of our approach will be applied on the computed ROI.

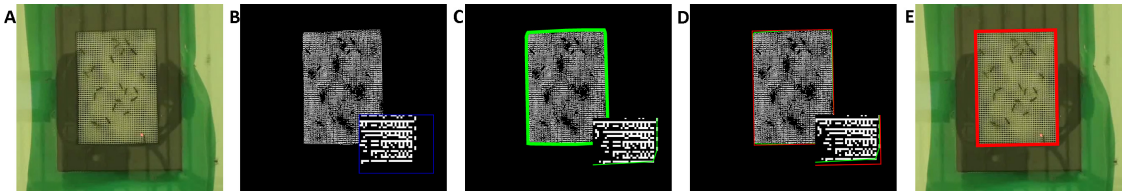


Figure 4.4: The ROI extraction procedure step-by-step: A) original image, B) the Sobel Edge detector applied on the original image, on the right bottom zoomed in portion of the image, C) the Convex hull of the Sobel edge detection points, D) the minimum bounding quadrilateral of points representing the convex hull of the ROI in red, E) the final ROI boundary plotted on the original image

4.3.4 Detection of Mosquitoes in ROI

The main challenge in object detection is the amount of variation in visual appearance. In our dataset mosquitoes vary from the background in their shape and color. To differentiate a mosquito from a background we used the HSV (Hue, Saturation and Value) and CIELab color spaces features and the HOG (histogram of oriented gradients) shape feature. On the ROI we trained a softmax classifier (multinomial logistic regression) using the specified features.

The softmax classifier has two major components: a *score function* that maps the pixel values of an image to confidence scores for each class, and a *loss function* that quantifies the agreement between the predicted scores and the ground truth labels. In other words the loss function is used to measure the quality of the predictions. Below are the details of each of the functions.

Let us assume a training dataset of N image patches, labeled $x_1, \ldots, x_N$, each having dimension D, and K distinct classes, numbered $\{1, \ldots, K\}$. Each image patch $x_i \in \mathbb{R}^D$ is assigned a label $y_i \in \{1, \ldots, K\}$ indicating the class to which the image belongs. In our case we have a training set of $N = 1000$ images, each of dimensions $D = 480 \times 720 \times 3$, and $K = 2$ classes indicating whether an image patch contains a mosquito or not.

Score function: The score function f maps each image patch $x \in \mathbb{R}^D$ to a list of K numbers which represent the confidence scores for the image to have the corresponding label (mosquito or not). The function f is defined by

$$f(x, W, b) = Wx + b. \tag{4.2}$$

For each image patch it computes the scores of the K classes as weighted sums of all of the raw image's pixel values across all 3 of its color channels. In equation 4.2 the parameter W is a $K \times D$ weight matrix where every row of W is a classifier for one of the classes and b is a $K \times 1$ column vector. The training data is used to learn the parameters W, b. These parameters should be such that the correct class of an image patch has a score that is higher than the scores of incorrect classes. The choice of these parameters are described later.

Probabilistic interpretation of the scores: There is a probabilistic interpretation of the score function in which the scores given to an image patch can be thought to represent the unnormalized log probabilities of the image belonging to the various classes. More precisely, if the image patch x has score vector $f(x, W, b) = (z_1, \ldots, z_K)^t$, then the numbers $z_1, \ldots, z_K$ can be interpreted to mean that the probability that x belongs to class j is proportional to e^{z_j} for $j \in \{1, 2, \ldots, K\}$. Normalizing to get the probabilities to add to 1, we obtain the distribution

$$P(\text{image } x \text{ belongs to class } j) = \frac{e^{z_j}}{\sum_{l=1}^{K} e^{z_l}}.$$

Let us denote this distribution P_x.

Loss function: The purpose of the loss function is to measure how good a particular choice of the parameters W and b is. For a given choice of W, b, using the probabilistic interpretation,

$f(x, W, b)$ will give the probability of the image patch x being in each possible class. However, for our training data $\{x_1, \ldots, x_N\}$ we know to which class each x_i actually belongs. The actual knowledge of the class can be specified by the distribution Q_x which assigns probability 1 to the correct class and probability 0 to the other classes. The loss function should measure how far the distribution P_x is from Q_x for our training data. A good measure for this is the Kullback Leibler divergence

$$D_{KL}(Q\|P) = \sum_j Q(j) \log \frac{Q(j)}{P(j)}$$

from information theory, which measures the amount of information gained when one revises from the prior distribution P to the posterior distribution Q.

Since in our case the entropy $\sum_j Q_x(j) \log Q_x(j)$ of Q_x is zero, this simplifies to the loss function

$$L(x) = \sum_{j=1}^{K} -Q_x(j) \log \frac{Q_x(j)}{P_x(j)} = -\log \frac{e^{z_m}}{\sum_{l=1}^{K} e^{z_l}},$$

where m is the correct class of the image patch x. This function is always non-negative, and is zero only when P_x and Q_x match. Thus it is a good measure of how far P_x is from the correct distribution Q_x.

Now the goal is to solve the optimization problem of finding parameters W and b which minimize the total loss function $\sum_{i=1}^{N} L(x_i)$ for all the image patches in our training data. To solve the optimization problem, gradient descent is applied on the loss function (the loss function is repeatedly reevaluated by moving the parameters along a gradient).

Once the softmax classifier is trained we apply a classifier to subwindows of the image, we apply the sliding window technique to classify each subwindow as a mosquito or a background. Usually sliding window based approaches produce multiple windows close to the correct location

of objects, which is a result of the visual correlation of close-by windows (Figure 4.5B). So to find the correct number of mosquitoes in images we need to obtain only one detection per mosquito.

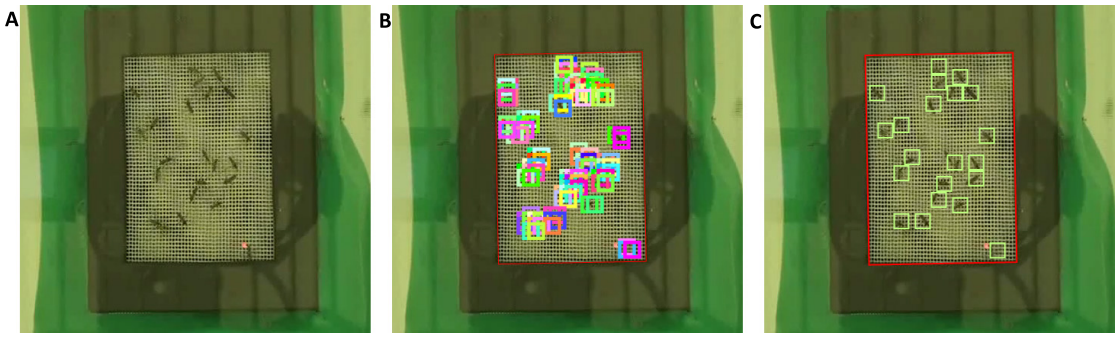


Figure 4.5: Results of the softmax classifier and the final counts: A) Original image, B) over-detections of mosquitoes obtained by the softmax Classifier, C) final counts of mosquitoes after applying non-maxima suppression on over-detections.

4.3.5 Non-maxima suppression (NMS) and final counts

NMS is a strategy for managing the over-detection. The goal of NMS is to avoid multiple bounding boxes for the same object, avoid merging two objects as one and avoid missing objects in the image. The input for the NMS method is: the list of all the bounding boxes, the confidence of each bounding box containing a mosquito and a parameter overlap. In our case the confidence of a bounding box is set to be the probability value that was computed during classification. Note that, if the probability value returned by the softmax classifier is high then there is a high possibility that the bounding box contains a mosquito, accordingly the confidence of that bounding box should be high as well. The pseudocode of the NMS method is shown below.

Figure 4.6 shows results of how NMS cleans up the over-detections of the mosquitoes.

The number of bounding boxes returned by the NMS method represents the final count of mosquitoes in the current frame (Figure 4.5C).

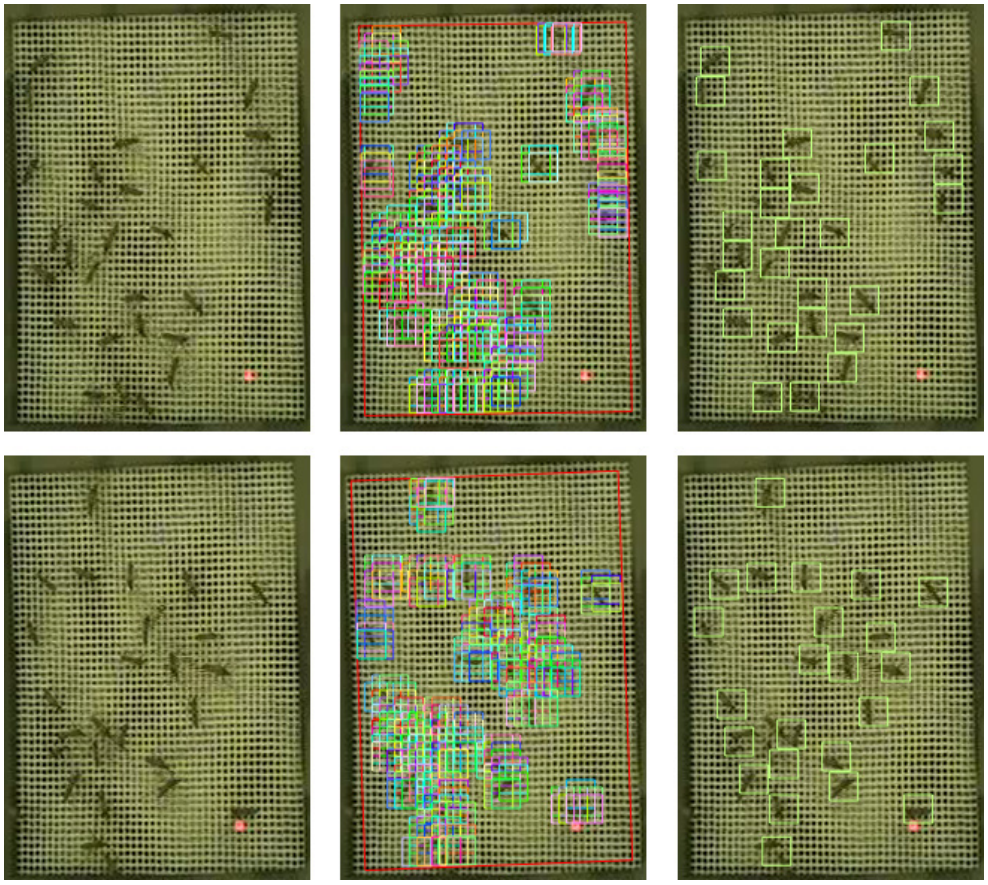


Figure 4.6: The results of the NMS technique on the over detections of the mosquitoes on two different images from different videos. First row - original frames, second row - over-detections of mosquitoes generated by softmax classifier, third row - results of NMS clean up of over-detections of mosquitoes.

4.4 Experimental Results

In this section we evaluate the performance of the proposed automatic mosquito counting method and show numerical results.

We have tested our proposed automated mosquito count method on 5 recordings of Arm-in-Cage experiments with each having about 10000 frames. There were 1000 frames chosen randomly from all of the 5 datasets and the mosquitoes were counted manually from those frames.

Instantaneous Count Results - We ran our automatic method on all of 5 datasets and

Input: list of boxes, confidence;
Result: list of picked boxes
sort the boxes in ascending order of confidence;
clear the list of picked boxes;
while *there are boxes left* **do**
    add the last box to the list of picked boxes;
    compute the intersections of the remaining boxes with the last one;
    compute the areas of the intersections;
    compute the ratios of the area of the intersection and the areas of the remaining boxes;
    remove from the list of boxes all the ones for which the ratio is larger than the allowed overlap;
end
return the picked boxes;

Algorithm 1: Pseudocode of the NMS method.

computed the number of mosquitoes in each frame of each video. We randomly selected 2 percentage of frames to manually label the mosquitoes in each frame. Figure 4.7 shows the number of mosquitoes computed from each frame of all 5 different videos, automated counted results are shown in blue and manual results are shown in red. As the graph shows, the counts of mosquitoes of our method are following the trend of counts.

Statistical Error Analysis - To evaluate the results in more detail we looked at the error (the difference of the counts of the automatic and manual methods) distribution from randomly selected 2 percentage of frames from all 5 datasets. The error distribution is provided in Figure 4.8. The numerical analysis showed that our method is on average 0.56 mosquitoes biased towards estimating under-counts compared to the real counts. Also, the root-sqare-mean error (RMS) is 1.7 and the mean percentage error (MPE) is 1.85%.

We also took manual counts biologists have had and then we compared it against ours. Those numbers are shown in Figure 4.9

We also took manual mosquito counts biologists have generated and worked with, and

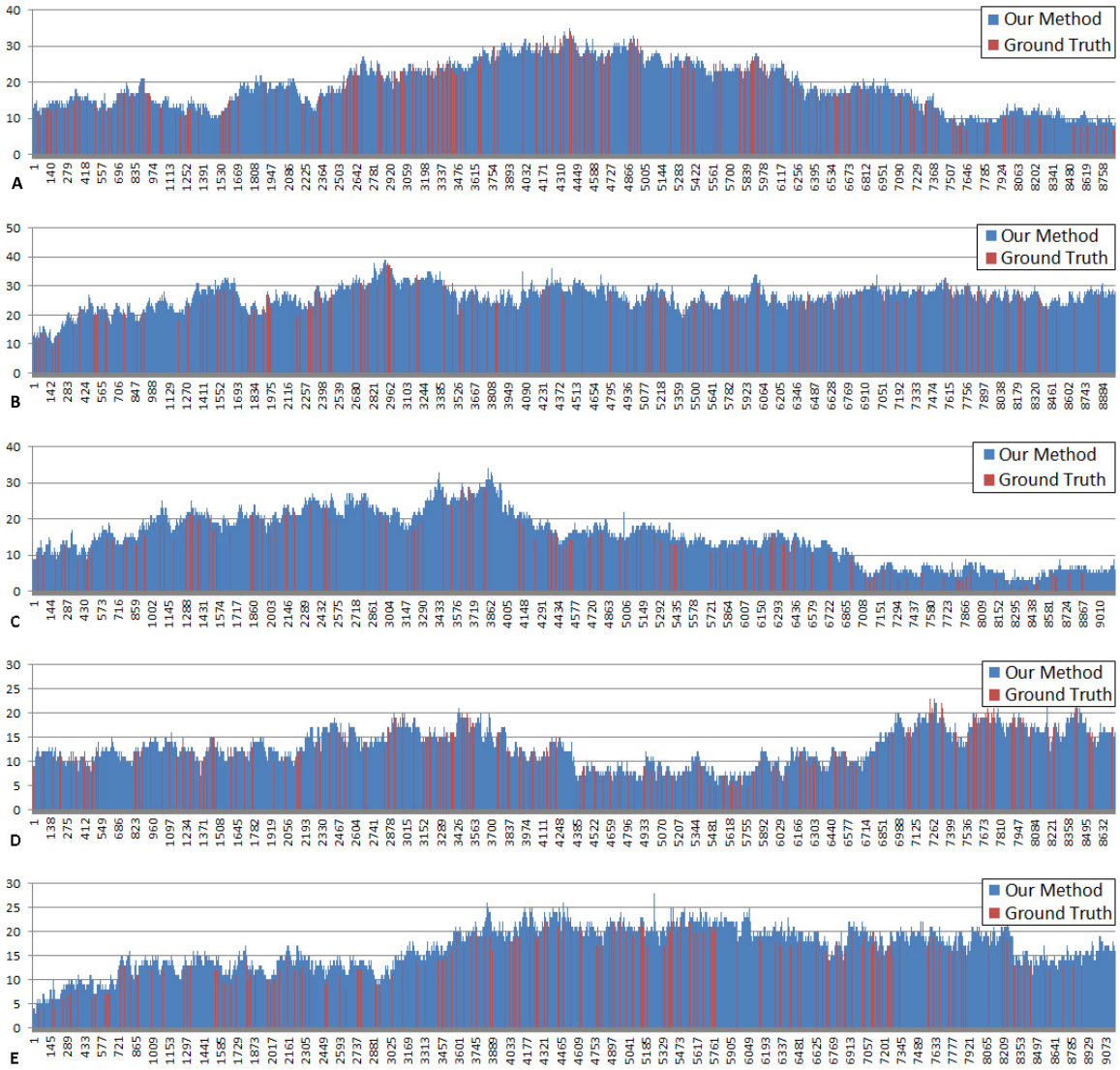


Figure 4.7: Number of mosquitoes counted on each frame of 5 different videos, with different experimental setups. In each subimage blue bars represent the mosquito counts generated by our automated method and red bars represent manual mosquito counts. It can be seen that counts generated by our automated method follow the trends of manual counts.

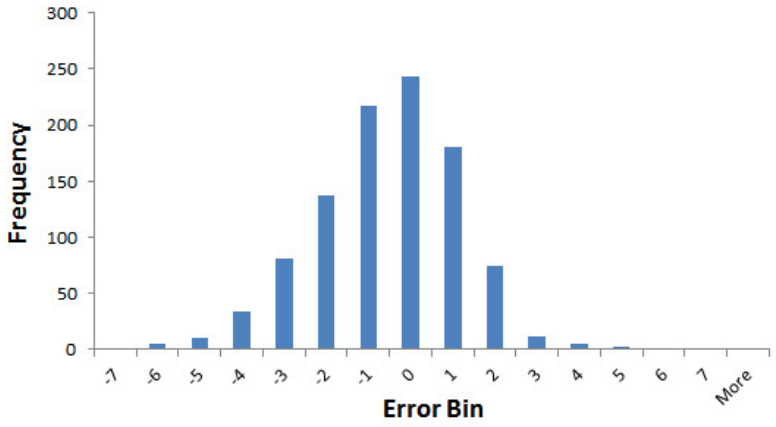


Figure 4.8: Error(the difference of the count of the automatic method and the manual count) distribution of the mosquito counts from randomly selected 2 percentage of frames from all 5 datasets. Our method is on average 0.56 mosquitoes biased towards estimating under-counts compared to the manual counts of mosquitoes.

compared those with our results. As mentioned earlier extracting information by visual inspection and manual measurement is labor intensive, hence biologists worked with limited results. We compared the average number of mosquitoes present in consecutive frames for 5 seconds with manual counts generated by biologists. Figure 4.9 shows the average results on 4 different videos. We combined the average results of all 5 datasets. The RMS for all 5 videos is 1.76. The MPE of all 5 videos is -0.45%.

Visual Examples - Figure 4.10 demonstrates the visual results of our automatic method. The figure shows examples of good, average and bad performance of our method. The count of the mosquitoes is expressed by the number of mosquitoes that have been detected by a green box. To show the performance of our method we have manually drawn red circles around those mosquitoes that were not counted by our method. As can be seen in the last row of the figure when there are very crowded areas our method tends to misdetect some mosquitoes. More specifically after the initial detection of mosquitoes using the softmax classifier is established, the NMS technique is used to have single detection of each mosquito. In case of having mosquitoes crowded together, the

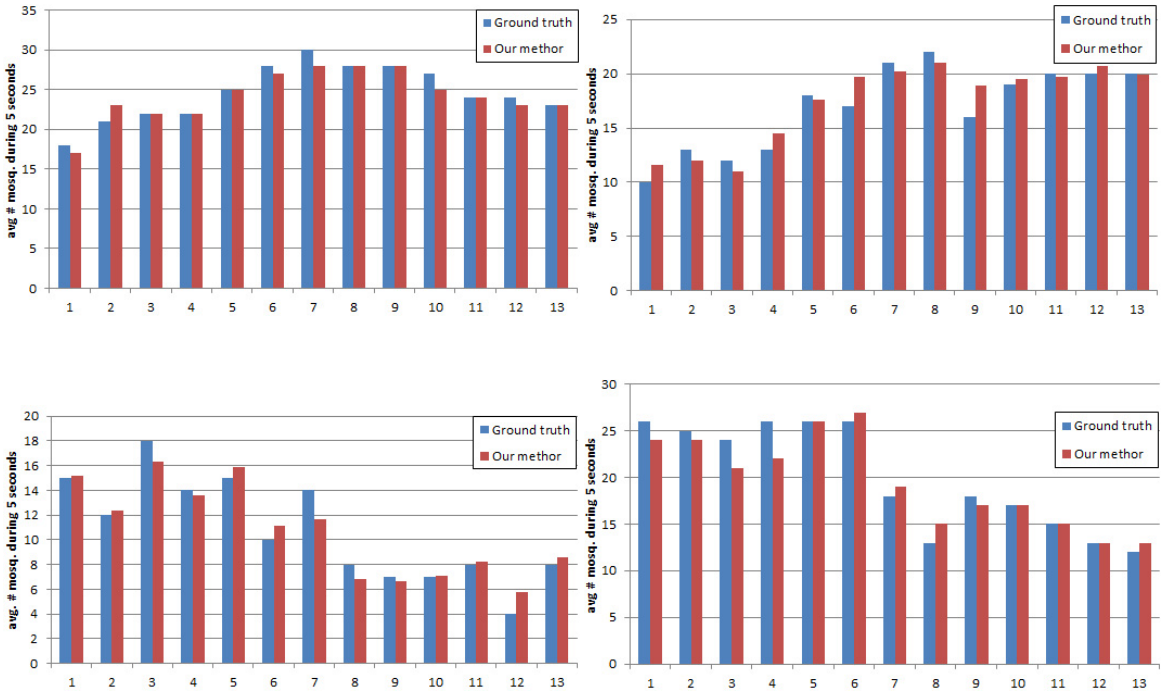


Figure 4.9: Average number of mosquitoes present in consecutive frames for 5 seconds in 13 frames for 4 different videos. In all subimages blue bars represent manual counts used by biologists and red bars represent the average counts generated by our automated method.

NMS may merge detections of several very close mosquitoes into one (Figure 4.10 last row). In our analysis of the results, this is the main source of error from the ground truth. However, this occurs rarely and has little effect on the overall statistical performance.

4.5 Conclusion

Mosquitoes and other blood-feeding insects are considered one of the most dangerous creatures on the planet because of their ability to spread deadly diseases. Quantitative analysis of a mosquito's attraction to different odors is very important. It can lead to discoveries about mosquitoes that could have a big impact on human health. In this work we present an automated

object counting method for counting mosquitoes captured in video, whereas previously the counting was done manually, which is not sufficient for quantitative analysis. We demonstrate our method on multiple videos collected by entomologists. Provided results show that our method can accurately count mosquitoes present in the video. Fast and accurate method for estimating mosquito counts would be of considerable importance in entomology.

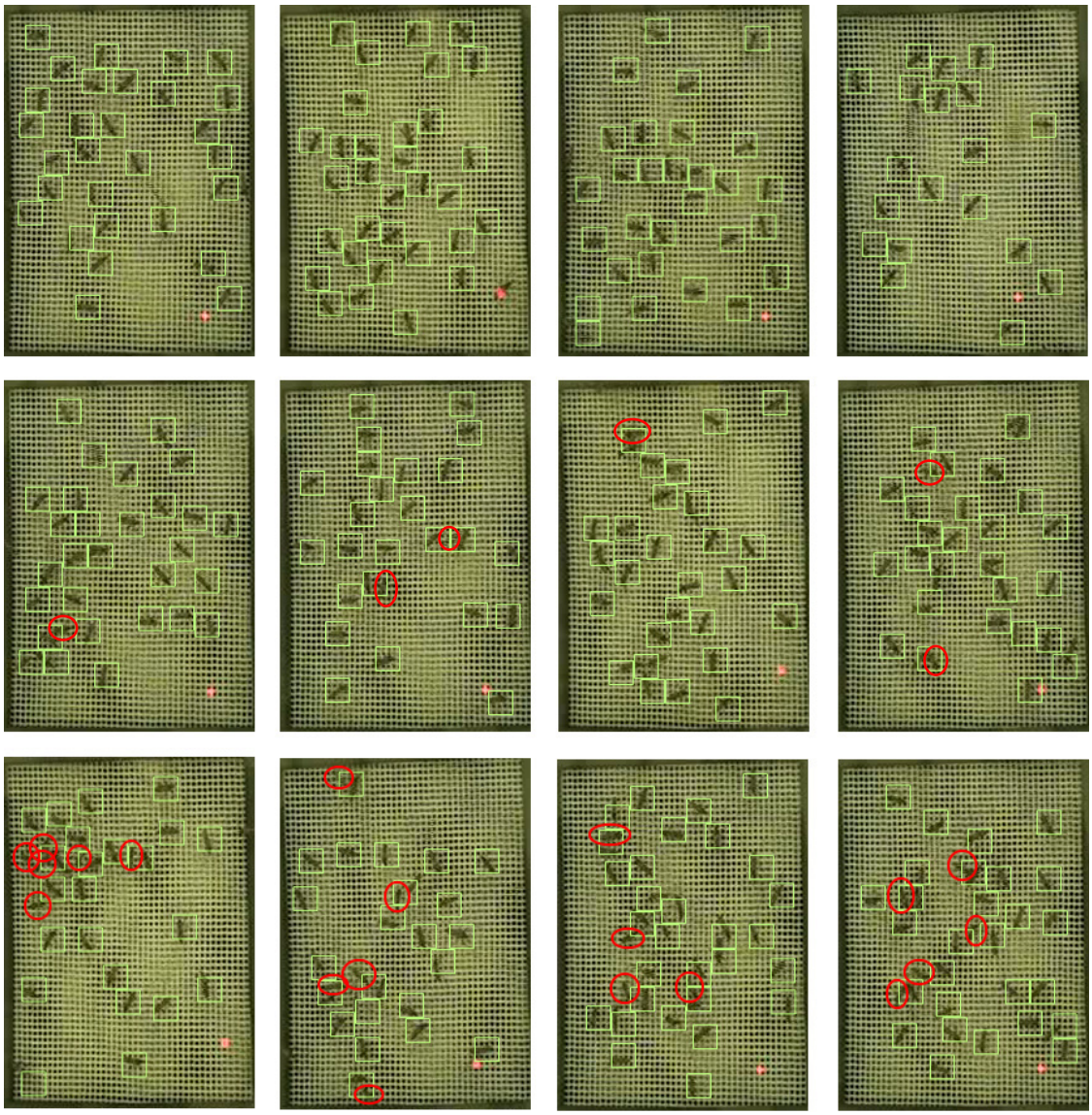


Figure 4.10: Examples showing good, average and bad performances of our method. Our method was able to correctly count the number of mosquitoes present in the first row. In the second row, the red circles show there were one or two miscounts, and the third row demonstrates frames for which our method found 4 or 5 mosquitoes less then the actual counts.

Chapter 5

Conclusion and Future Work

5.1 Thesis Summary

In this thesis, we have proposed methods on how to do image analysis in different biological applications, including gathering of cell growth and division statistics in images taken by Confocal Laser Scanning Microscopy and quantitative analysis of a mosquito's attraction to different experimental conditions from videos captured under spatial environment. The contribution of our work can be broken down in the following manner.

In Chapter 2, we proposed a single framework that entails segmentation and tracking of plant cell images. It is able to segment and find the correspondence of cells in spatio-temporal image stacks obtained using CLSM. We demonstrated a procedure to optimally choose the parameters in the watershed algorithm for high quality segmentation results. After the segmentation, we use local graph matching, which uses geometric structure and topology of the relative positions of the cells, to track cells and identify cell divisions. We tested this framework on multiple datasets and it provided significantly longer cell lineages and more comprehensive identification of cell divisions.

In Chapter 3, to further improve the cell growth and division statistics we presented a fully automated registration method of live imaging stacks that takes care of both temporal and spatial misalignments in 4D image stacks. We presented a novel landmark selection methodology. The best image slice correspondences are formed from consecutive image stacks. Then it uses local graph-based approach to automatically find corresponding landmark pairs, and finally the registration parameters are used to register the entire image stack. The proposed registration algorithm combined with an existing tracking method is tested on multiple image stacks. The results show that it significantly improves the accuracy of cell lineages and division statistics.

Finally, in Chapter 4, we proposed a method to automatically count number of mosquitoes in videos captured in special arm-in-cage environment. It uses the softmax classifier with a sliding window technique to detect the mosquitoes and eliminates the over-detections resulting from the sliding window technique by non-maximum suppression method. The proposed automated method has been applied on different datasets and showed very good and consistent counts of mosquitoes.

5.2 Future Work

Methods have shown that an adaptive framework, where the output of one module acts as an indicator of the quality for another and can be improved to obtain better results. They have shown that adaptive approach significantly improves both tracking and segmentation when compared to an open loop framework in which segmentation and tracking modules operate separately. Since in image analysis pipeline, all three modules, segmentation, registration and tracking, have very important role and depend on one another, we would like to explore generating an adaptive segmentation, registration and tracking approach to further improve cell growth and division statis-

tics. Also, our proposed methods on cell segmentation and image registration have been applied on tightly clustered cells of Arabidopsis plant's shoot apical meristem. We would like to explore the applications of these approaches to other densely clustered plant and animal tissues, to generate different cell division and growth statistics.

The proposed mosquito counting framework provides very accurate results. In future, we would like extend our work further and look into mosquito tracking problem, which will lead us to gather more statistical information about the mosquito's behavior under different experimental conditions. The results of this study would be of considerable biological importance.

Bibliography

[1] C.A. Schneider, W.S. Rasband, K.W. Eliceiri, "NIH Image to ImageJ: 25 years of image analysis". Nature Methods 9, 671-675, 2012.

[2] C. A. B. Mello, W. P. dos Santos, M. A. B. Rodrigues, A. L. B. Candeias, C. M. G. Gusmao, *Image Segmentation of Ovitraps for Automatic Counting of Aedes Aegypti Eggs*, 30th Annual International IEEE EMBS Conf. , pp. 3103-3106, 2008.

[3] H.A. Khan and G.M. Maruf, *Counting Clustered Cells using Distance Mapping*, International Conference on Informatics, Electronics and Vision (ICIEV), pp. 1-6, 2013.

[4] N. Ahuja and S. Todorovic, *Extracting texels in 2.1d natural textures*, ICCV, pp. 18, 2007.

[5] Y. Al-Kofahi, W. Lassoued, W. Lee, B. Roysam, *Improved automatic detection and segmentation of cell nuclei in histopathology images*, IEEE Transactions on Biomedical Engineering, 57(4):841-52.

[6] A.B. Chan, Z.J. Liang, N. Vasconcelos, *Privacy preserving crowd monitoring: Counting people without people models or tracking*, CVPR, 2008.

[7] O. Barinova, V. Lempitsky, P. Kohli, *On the detection of multiple object instances using Hough transforms*, CVPR, 2010.

[8] Y. Bazi, H. Al-Sharari, F. Melgani, *An automatic method for counting olive trees in very high spatial remote sensing images*, Proc. IEEE Int. Geoscience and Remote Sensing Symp., pp. 125128, 2009.

[9] P. Besl and N. McKay, *A method for registration of 3-D shapes*, IEEE Trans. Pattern Anal. Mach. Intell., 1992.

[10] M. Bhattacharya, A. Das, *Multi resolution medical image registration using maximization of mutual information and optimization by genetic algorithm*, IEEE Nuclear Science Symposium Conference Record, 2007.

[11] G.C. Sharp and S.W. Lee et al. *Invariant features and the registration of rigid bodies*, Proc. IEEE ICRA, pp. 932-937, 1999.

[12] B. Cherkassky, A. Goldberg, T. Radzik, *Shortest paths algorithms: theory and experimental evaluation*, Mathematical Programming, 1996.

[13] T.F. Chan and L.A. Vese, *Active contours without edges*, IEEE Transactions on Image Processing, Volume 10, 266277 pages, 2001.

[14] C. Desai, D. Ramanan, C. Fowlkes, *Discriminative models for multi-class object layout*, ICCV, 2009.

[15] R. Fernandez and P. Das et al., *Imaging plant growth in 4d: robust tissue reconstruction and lineaging at cell resolution*, Nat Meth, 2010.

[16] L. Fiaschi, U. Koethe, R. Nair, F. A. Hamprecht, *Learning to count with regression forest and structured labels*, Proc. ICPR, 2012.

[17] V. Gor, M. Elowitz, T. Bacarian, E. Mjolsness, *Tracking Cell Signals in Fluorescent Images*, CVPR Workshops, vol 03, pp. 142–, 2005.

[18] V. Gorbunova, S. Durrleman,Lo Pechin, X. Pennec, M. de Bruijne, *Lung CT registration combining intensity, curves and surfaces*, IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2010.

[19] X. Guo and F. Yu, *A Method of Automatic Cell Counting Based on Microscopic Image*, 5th International Conference on Intelligent Human-Machine Systems and Cybernetics, vol. 1, pp. 293-296, 2013.

[20] Y.H. Toh, T.M. Ng, B.K. Liew, *Automated Fish Counting using Image Processing*, International Conference on Computational Intelligence and Software Engineering, CiSE, pp. 1-5, 2009.

[21] X. Huang, B. Wang, R. Liu, X. Wang, Z. Wu, *CT-MR image registration in liver treatment by maximization of mutual information*, IEEE International Symposium on IT in Medicine and Education, 2008.

[22] A. I. Dell, J. A. Bender, K. Branson, I. D. Couzin, G. G. de Polavieja, L. P.J.J. Noldus, A. PrezEscudero, P. Perona, A. D. Straw, M. Wikelski, and U. Brose, *Automated image-based tracking and its application in ecology*, Trends Ecology Evolution, vol. 29, no. 7, pp. 417-428, 2014.

[23] H. Idrees, I. Saleemi, C. Seibert, M. Shah, *Multi-source multi-scale counting in extremely dense crowd images*, CVPR, 2013.

[24] P. Kain, S. M. Boyle, S. K. Tharadra, T. Guda, C. Pham, A. Dahanukar, and A. Ray, *Odour receptors and neurons for DEET and new insect repellents*, Nature 502, pp. 507512, 2013.

[25] B. Kesavaraju and S. Dickson, *New technique to count mosquito adults: using ImageJ software to estimate number of mosquito adults in a trap*, Journal of the American Mosquito Control Association, 28, 330333, 2012.

[26] D. Kong, D. Gray, H. Tao. *A viewpoint invariant approach for crowd counting*, ICPR (3), pp. 11871190, 2006.

[27] H. Kumagai, S. Morishita, K. Kuniaki, H. Asama, T. Mishima, *Accuracy improvement of counting asbestos in particles using a noise redacted background subtraction*, in Proc. IEEE Int. Conf. Multisensor Fusion and Integration for Intelligent Systems, pp. 7479, 2008.

[28] B. Leibe, A. Leonardis, B. Schiele, *Robust object detection with interleaved categorization and segmentation*, International Journal of Computer Vision, 77(1-3), pp. 259289, 2008.

[29] , V. Lempitsky and A. Zisserman,*Learning To Count Objects in Images*, Advances in Neural Information Processing Systems, 2010.

[30] K. Li and T. Kanade, *Cell Population Tracking and Lineage Construction Using Multiple-Model Dynamics Filters and Spatiotemporal Optimization*, Proceedings of the 2nd International Workshop on Microscopic Image Analysis with Applications in Biology (MIAAB), 2007.

[31] G. Li, T. Liu, A. Tarokh, J. Nie, L. Guo, A. Mara, S. Holley, ST. Wong, *3D cell nuclei segmentation based on gradient flow tracking*, BMC Cell Biology, 2007, Sep 4;8:40.

[32] M. Liu and R. Yadav et al., *Automated tracking of stem cell lineages of Arabidopsis shoot apex using local graph matching*, Plant Journal, 2010.

[33] M. Liu, A. Chakraborty, D. Singh, M. Gopi, R. Yadav, G. Reddy and A. Roy-Chowdhury, *Adaptive cell segmentation and tracking for volumetric confocal microscopy images of a developing plant meristem*, Molecular Plant, 2011.

[34] Y. Ma, J. Tian, *The Algorithm of Rapid Medical Image Registration by Using Mutual Information*, 4th International Conference on Bioinformatics and Biomedical Engineering, 2010.

[35] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, P. Suetens, *Multimodality image registration by maximization of mutual information*, IEEE Transactions on Medical Imaging, vol. 16, no. 2, pp. 187–198, 1997.

[36] F. Maes, D. Vandermelen, and P. Suetens, *Medical Image Registration Using Mutual Information*, Proceedings of the IEEE, vol. 91, no. 10, pp.1699-1722, 2003.

[37] F. Mallard, V. Le Bourlot, T. Tully, *An Automated Image Analysis System to Measure and Count Organisms in Laboratory Microcosms*, PloS one, vol. 8, no. 5, 2013.

[38] K. Mkrtchyan, A. Chakraborty, and A. Roy-Chowdhury, *Automated Registration of Live Imaging Stacks of Arabidopsis*, International Symposium on Biomedical Imaging, 2013.

[39] K. Mkrtchyan, A. Chakraborty, and A. Roy-Chowdhury, *Optimal Landmark Selection for Registration of 4D Confocal Image Stacks in Arabidopsis*, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 2016.

[40] K. Mkrtchyan, D. Singh, M. Liu, G. V. Reddy, A. Roy-Chowdhury, M. Gopi, *Efficient cell segmentation and tracking of developing plant meristem*, IEEE International Conference on Image Processing, 2011.

[41] J.N. Fabic, I.E. Turla, J.A. Capacillo, L.T. David and P.C. Naval, Jr, *Fish Population Estimation and Species Classification from Underwater Video Sequences using Blob Counting and Shape Analysis*, International Underwater Technology Symposium (UT), pp. 1-6, 2013.

[42] L. Najman and M. Schmitt, *Watershed of a continuous function*, Signal Processing Journal, vol. 38, pp. 99–112, 1994.

[43] E. R. Mullen, P. Rutschman, N .Pegram, J. M. Patt, J. J. Adamczyk, and 3. Johanson, *Laser system for identification, tracking, and control of flying insects* vol. 24, no. 11, pp. 11828-11838, 2016.

[44] V. Rabaud and S. Belongie, *Counting crowded moving objects*, CVPR, pp. 705711, 2006.

[45] A. Rangarajan, H. Chui, F.L. Bookstein, *The Softassign Procrustes Matching Algorithm*, Information Processing in Medical Imaging, 1997.

[46] G.V. Reddy, M. Heisler, D. Ehrhardt and E. Meyerowitz, *Real-time lineage analysis reveals oriented cell divisions associated with morphogenesis at the shoot apex of arabidopsis thaliana*, Development, 2004.

[47] L. Vincent and P. Soille, *Watersheds in digital spaces: An efficient algorithm based on immersion simulations*, IEEE Trans. Pattern Anal. Mach. Intell., 1991.

[48] Ivan V. Grishagin, *Automatic cell counting with ImageJ*, Analytical Biochemistry 473, pp. 63-65, 2015.

[49] B. Venkatalaksmi and K Thilagavathi, *Automatic Red Blood Cell Counting Using Hough Transform*, Proc. of 2013 IEEE Conference on Information and Communication Technology, pp. 267-271, 2013.

[50] P. Viola and M. William, *Alignment by Maximization of Mutual Information*, IJCV, 1997

[51] E.W. Dijkstra, *A note on two problems in connexion with graphs*, Numerische Mathematik, 1959.

[52] B. Wu, R. Nevatia, Y. Li, *Segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses*, CVPR, 2008.

[53] Z. Yin, R. Bise, M. Chen, T. Kanade, *Cell segmentation in microscopy imagery using a bag of local Bayesian classifiers*, Proceedings of IEEE international conference on Biomedical imaging: from nano to Macro, pp. 125–128, 2010.

[54] R. Zhang, S. Zhao, Z. Jin, N. Yang, H. Kang, *Application of SVM in the food bacteria image recognition and count*, in Proc. Int. Cong. Image and Signal Process., pp. 18191823, 2010.

[55] Y. Zhu, S. Cheng, V. Stankovi, L. Stankovi, *Image registration using BP-SIFT*, Journal of Visual Communication and Image Representation, 2013.