Image Segmentation and Range Estimation Using a Moving-aperture Lens

Anbumani Subramanian

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

A. Lynn Abbott, Chair
Amy E. Bell
Leonard A. Ferrari

April 24, 2001
Blacksburg, Virginia

Keywords: Image segmentation, motion parallax, range estimation,
motion segmentation, optical flow.

Image Segmentation and Range Estimation Using a Moving-aperture Lens

Anbumani Subramanian

(ABSTRACT)

Given 2D images, it still remains a big challenge in the field of computer vision to group the image points into logical objects (segmentation) and to determine the locations in the scene (range estimation). Despite the decades of research, a single solution is yet to be found. Through our research we have demonstrated that a possible solution is to use moving aperture lens. This lens has the effect of introducing small, repeating movements of the camera center so that objects appear to translate in the image, by an amount that depends on distance from the plane of focus. Our novel method employs optical flow techniques to an image sequence, captured using a video camera with a moving aperture lens. For a stationary scene, optical flow magnitudes and direction are directly related to the three-dimensional object distance and location from the observer. Exploiting this information, we have successfully extracted objects at different depths and estimated the locations of objects in the scene, with respect to the plane of focus. Our work therefore demonstrates an ability for passive range estimation, without emitting any energy in an environment. Other potential applications include video compression, 3D video broadcast, teleconferencing and autonomous vehicle navigation.

Acknowledgments

I am greatly indebted to Prof. A. Lynn Abbott, for giving me this wonderful opportunity to work on a challenging problem. His highly insightful perspectives and enthusiasm, guided me throughout this work.

I like to thank Dr. Amy E. Bell for her encouragement and support throughout this work and especially for introducing me to LaTeX. I also like to thank Prof. Leonard A. Ferrari for devoting some of his valuable time in being part of my advisory committee.

Few of my friends, whom I met during my career at IBM, have influenced me in a way more than I had expected. They were the catalysts for my renewed interest in graduate studies. Saraswathi Padhmanabhan, had an unswerving determination to encourage me join graduate school in which she succeeded. I like to acknowledge Joaquim Fernandes, for all his momentous help, whose immense values only I know. Anjani Anant, during a casual talk, lighted in me the spark for graduate studies, when I was about to succumb to the corporate culture. Roshan Bangera, dispelled all the myths I had about graduate studies. E.N. Gururajan and Balaji Srinivasan inspired and helped me a lot to pursue my graduate studies.

I like to thank all my teachers and professors for imbibing in me in an education that continues to enchant me. I also like to recognize my sister for her countless good wishes and encouragement sent to my inbox, through her emails. I wish to attribute all my success to my parents and I dedicate this work to them .

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The human visual system is extremely good at differentiating objects in a scene and also excels in estimating the distances of objects. Furthermore, the visual system can perform similar tasks given two-dimensional (2D) images. But, this feat that is so easy for humans yet remains a challenge for computers.

Segmentation, explained in simple terms refers to the logical grouping of the parts of an image into individual entities. These entities, called regions may correspond to 3D objects. Range estimation is the determination of the location of an object with respect to the observer in a three-dimensional (3D) world.

Since the advent of digital computers, the past few decades witnessed many efforts in an attempt to emulate these features in human visual system on computers for various applications. But the main problem with such image analysis is that the information is available only in 2-dimensions and hence the challenge.

Our work here is an attempt to solve this long posed problem using a moving aperture

camera and optical flow techniques. Although not the ultimate solution, we are confident that the new method we propose here presents a new perspective to the problem.

1.2 Previous Work

Traditionally image segmentation has been performed using various methods. The widely used method utilizes translation of image points in video sequences, also called as optical flow. Horn and Schunck initially presented a method to calculate optical flow in images under some constraints [1]. Realizing the power of optical flow methods and its ease, many applications were later developed. Bergholm and Carlsson described the theory behind optical flow and its importance [2]. The compendium of optical flow techniques and their comparison by Barron, Fleet and Beauchemin discusses the performance of different approaches in optical flow calculations and the results obtained using these approaches for some test sequences. Wang and Adelson described a system in which images were represented in different layers using velocity maps [3, 4]. More recent work on optical flow can be found in [5]

Any motion between objects and observer can be considered as a parallax in the image sequence. This parallax information can therefore be utilized to perceive a three-dimensional view of the scene. Three-dimensional displays with parallax have been attempted using a matrix arrangement of cameras [6]. Mayhew proposed a single camera system which exploited motion in images to produce a three-dimensional perception [7]. A single lens system in which the parallax information can be controlled was presented in [8]. Although parallax in an image sequence can be controlled, there exists an upper limit above which the observer begins to experience too much motion in the sequence and begins to feel annoyed. Therefore research mainly by human psychologists looked into determining this upper limit for parallax. Loomis and Eby studied the conditions for perception of depth variation in motion

parallax [9]. Recently, Runde analyzed the conditions for a realistic image representation using motion parallax [10].

Parallax information when used with image geometry, can be used for recognition of objects in images [11]. Another application of parallax in images is to estimate the distances of objects in the scene. Some standard methods for range estimation using camera techniques are mentioned in [12]. Three-dimensional cameras in robots which estimate range for navigating is presented in [13]. Range sensors for evaluation of object volumes are described in [14].

1.3 Contributions of This Research

The following are the contributions to the field of image analysis made as a result of our research:

1. Developed and implemented a novel method of image segmentation using a moving aperture lens with optical flow techniques.

2. Demonstrated the feasibility of range estimation from 2D image sequences using optical flow methods.

3. Developed a new technique to distinguish relative depths of objects in a scene with respect to the plane of focus.

The significance of our work is emphasized by the fact that our method performs both image segmentation and range estimation with only one camera. The algorithms involved are relatively easy for implementation on hardware and hence the viability of a realizable and repeatable setup. While the traditional techniques typically employ a stereoscopic system

with two cameras, our method has the advantages of reduction in cost and ease of portability.

1.4 Outline

Chapter 2 succinctly explains the basic terminologies and techniques used in our work. Then, Chapter 3 proceeds to describe our principle, implementation and discusses in detail the methodology behind the proposed solution. The results obtained using our implementation are presented in Chapter 4. In the final chapter, we present a summary of our interpretation of the results and conclude with suggest few applications and recommendations for future work.

As we proceed further, we urge the reader not to be taken aback by the math. Wherever possible we have attempted to keep the material simple, illustrative and understandable.

Chapter 2

Fundamentals Concepts

2.1 Preface

The purpose of this chapter is to briefly outline some of the fundamental concepts in computer vision. We identified the most relevant topics that form the foundation of our work and have presented a short introduction to those topics with further references. The interested reader is suggested to consult some of the classic books [15, 16, 17] on computer vision for a detailed discussion on these topics.

2.2 Co-ordinate Geometry

In image analysis, it becomes extremely convenient to represent, interpret and analyze the three-dimensional world using co-ordinate systems often represented by X, Y and Z. Of the many possible notations, the widely used and easier one to remember is shown in Fig. 2.1. As can be seen from the figure, width is represented along the X-axis and height along the Y-axis. The positive going Z-axis extends into the world, away from the viewer. Thus if an object is in view (or in front of the observer), then its Z value is positive. Farther the object

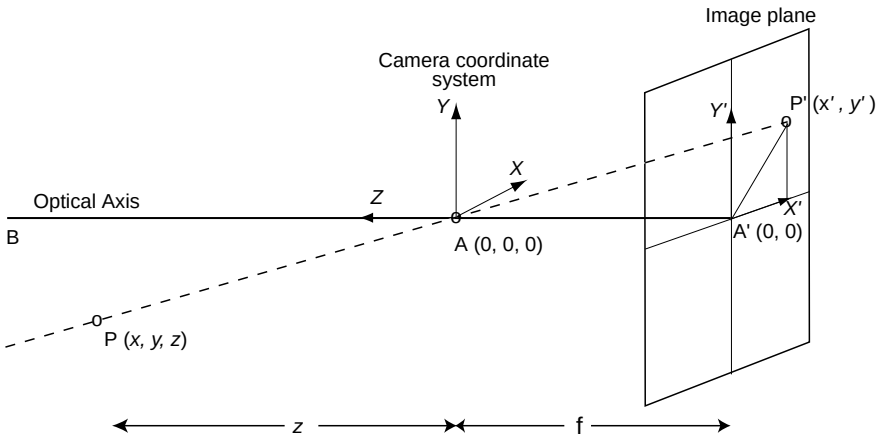


Figure 2.1: Image plane and co-ordinate geometry.

is from the viewer, higher is the value of Z. In certain cases, the object may move and be located behind the observer in which case Z is negative. Although various other conditions are possible, we treat them as special cases and hence shall skip their discussion.

In the figure, point A represents the origin which can either be the aperture of a camera or a human eye. Similarly the image plane can either be the retina of a camera or human eye. Usually the focal length f can be varied for a clear view of the object or scene. Each point in the three-dimensional scene corresponds to a point (x', y') on the two-dimensional image plane. Using the relation of similar triangles, we can write the relation [17]

$$\frac{x'}{x} = \frac{y'}{y} = \frac{f}{z} \tag{2.1}$$

or,

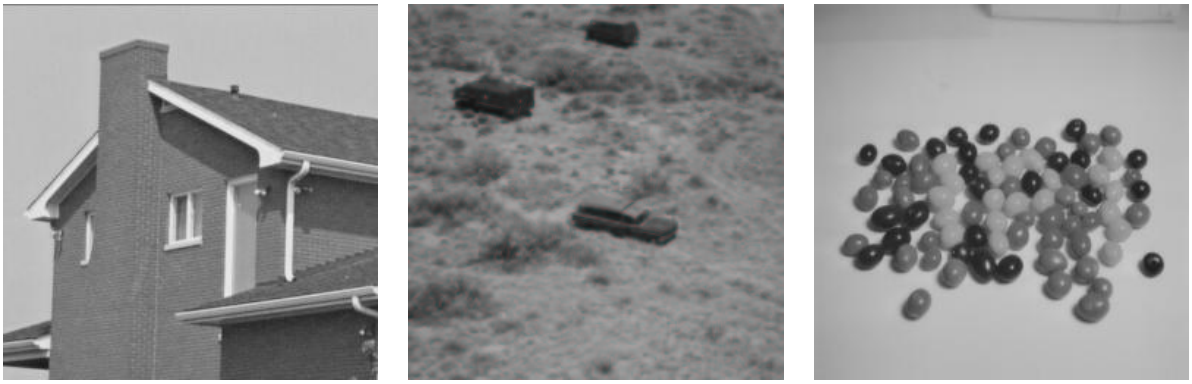
$$x' = \frac{f}{z}\,x \quad \text{and} \quad y' = \frac{f}{z}\,y \tag{2.2}$$

The mapping of the 3D world onto this 2D image plane is a many-to-one relation and hence the reverse mapping is a difficult problem.

2.3 Image Segmentation

As explained in Chapter 1, segmentation is the process of distinguishing objects or regions from its surroundings in a scene. In other words, it is the grouping of image points or finding the boundaries that correspond to an object or a region and hence being able to call such a group, an object or a region in the image.

Human visual system utilizes many cues such as color, shape, shade and shininess in an image to differentiate objects. For example, consider the images [18] shown in Fig. 2.2. In Fig. 2.2(a), we are likely to identify the edges of a roof using the grayscale (or intensity) information while in Fig. 2.2(b) the cue is probably the shape and texture. Fig. 2.2(c) is a good example where the human eye excels in identifying the many jellybeans perhaps using grayscale and shininess information. Image segmentation with computers are done using various methods some of which include motion, shading, line or edge detection and region-based matching techniques.



(a) (b) (c)

Figure 2.2: Example images where various cues are used for segmentation.

2.4 Segmentation from Motion

Consider a scene in which an object moves and let this motion be captured in an image sequence. This apparent shift or change in the location of an object between successive frames of a (video) sequence can be employed to determine the boundaries the object from its surroundings. Operations as simple as a difference of images can show the object boundaries. If the surface of the object is uniform and the shift is small, then image differencing will result in a good estimate of the boundaries (or segmentation) whereas if the shift is large, the result may not be accurate although acceptable.

2.5 Optical Flow

Inspired by the idea of flow in fluids, Horn and Schunk applied a similar idea to image sequences. They called this technique *optical flow* and thus laid the foundations of a whole new arena for research. Since their first publication on this topic [1], we can find most of its applications towards image segmentation.

Consider a sequence of images from a video where objects are moving. This object motion in images can be thought of as a flow in the image sequence and we proceed as follows. Let $I(X, Y, t_1)$ be the two-dimensional image of the scene at time t_1 and $I(X, Y, t_2)$ be the image of the same scene at time t_2. Under the assumption that the (total) image intensity remains constant, we can write

$$\frac{d}{dt} I(X, Y, t) = 0 \tag{2.3}$$

Expanding the above equation, we get

$$\frac{\partial I}{\partial X}\frac{dX}{dt} + \frac{\partial I}{\partial Y}\frac{dY}{dt} + \frac{\partial I}{\partial t} = 0 \tag{2.4}$$

which can be further simplified to

$$I_X\, U + I_Y\, V + I_t = 0 \tag{2.5}$$

by substituting

$$U = \frac{dX}{dt} \qquad \text{and} \qquad V = \frac{dY}{dt} \tag{2.6}$$

$$I_X = \frac{\partial I}{\partial X}\,, \qquad I_Y = \frac{\partial I}{\partial Y} \qquad \text{and} \qquad I_t = \frac{\partial I}{\partial t} \tag{2.7}$$

Eqn.(2.5) is the well-known *optical flow equation*. This above approach later came to be known as the *gradient-based* method [19]. Often, optical flow is also referred to as *image flow*.

The implementation of optical flow in our work is an adaptation of this method, called the *region-based matching* method [19]. In this method, we consider an arbitrary (rectangular) portion in a 2D image I_1, centered at (x, y), often called the *reference window*. Let us represent this image window by I_{w1}. We then proceed to select an image window of similar size in image I_2. The test for a good match is performed using *normalized cross-correlation*, which is defined as

$$\rho = \frac{I_{w1}\, I_{w2}}{\sqrt{I_{w1}\, I_{w2}}} \tag{2.8}$$

where $-1 \leq \rho \leq 1$. The value of $|\rho| = 1$ indicates a best match, while $\rho = 0$ indicates a poor match.

For example, let our reference window be centered at a location (x_1, y_1) in image I_1. Consider another image I_2, similar to I_1, but with small spatial displacement (or disparity) in objects. Now, the best match in I_2 for the reference window in I_1 may not be centered at (x_1, y_1). Instead, using (2.8) we may find that it is centered at a different location in the image, say $(x_1 + \Delta x,\, y_1 + \Delta y)$. We can consider this displacement, Δx and Δy occurring at (x_1, y_1) in the image, as the components of the optical flow field. This concept of optical flow

can be extended to a sequence of images with flow calculated between successive images.

The inherent problem with this technique which can be combated easily, arise in case of uniform regions. Flow estimation using region-based matching can perform poorly in such regions where there is no intensity or texture change [17]. This well known problem often called the *aperture problem*, is usually overcome by enlarging the reference window.

An issue that is often raised in region based matching is the choice of the reference window with multiple images. In some applications, using the reference window from the first image through out the matching yields good results while in some cases, changing the reference window to the last image in the sequence gives better results. Therefore the choice of the reference window is dependant on the application while more success has been shown when using a constant reference window [20].

2.6 Stereoscopic Camera

A stereoscope is a device in which two identical images with small disparity are viewed through carefully arranged lenses. When the images are viewed through this device, a realistic 3D perception is experienced.

A conventional camera records the 3D world it sees on a 2D image. An image sequence captured using a video camera adds only an extra dimension - time, but not depth. Therefore, we do not perceive depth observing a 2D image or an image sequence. For a long time, enthusiasts worked on techniques that enable the viewer to perceive 3D from 2D images.

We tend to perceive depth using special devices like the stereoscope. The disparity in the images may not be highly apparent when seen directly, but give a 3D perception when viewed together. Usually, these two images are taken using two cameras separated by a small

distance or from a single camera at one position and then studiously displaced by a short distance. This arrangement of camera comprises the *stereoscopic* setup. The use of a single camera adds some complexity although it saves the cost. The complexity arises due to the accuracy needed in taking the two images. In a two-camera system, this problem does not arise because once calibrated, the two camera arrangement can be ported anywhere. The sheer inconvenience and problems in handling such extra devices forced researchers to look for other viable alternatives. This led to the development of fancy wearable goggles and head-mount displays for 3D perception.

Since there are two images in which the disparity corresponds to displaced objects, image segmentation can be performed using stereoscopic devices.

2.7 Moving Aperture Lens

Innovative ideas took shape as the search for a better stereoscopic system was in place. Auto-stereoscope is an improvement over the stereoscope which employ only one camera or image for 3D perception in contrast to stereoscopes.

A novel auto-stereoscopic camera in which the aperture of the camera moves in a circular fashion around the camera center is proposed in [8]. Although this design allows the aperture to move, the camera itself remains stationary. Therefore we call this system the *moving aperture lens*. A video sequence made using this unique system injects parallax into the images, giving the impression of 3D distances. Hence it obviates the need for special devices commonly used for realistic 3D perception.

The working of a moving aperture lens can be understood from Fig. 2.3. (Inset image was taken from [18].) In a conventional camera, the aperture is located at the stationary point

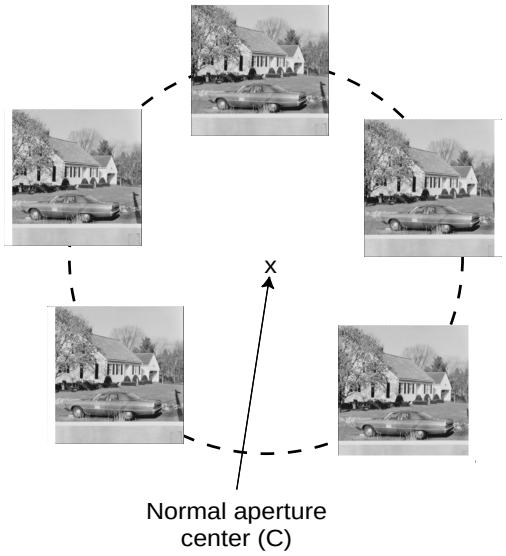


Figure 2.3: Illustration of images as seen at various positions of a moving aperture.

C. But in an auto-stereoscopic camera, the aperture moves in a circle about the center C. The radius of motion is determined by the control settings chosen while using the camera. Although the scene observed through the aperture looks similar, the inherent parallax effect becomes apparent when we view the image sequence played back in real time.

The auto-stereoscopic camera is made in such a way that the camera parameters like the focus, amplitude (or radius) and frequency (revolutions of the aperture per second) of rotation can be varied using individual control knobs. Changing the focus produces an image where the object in focus appears to be defined sharply while objects at some distance from the focus (i.e., on other depth planes) do not appear sharply. In popular parlance, this is the equivalent of an *out of focus* setting. A video sequence made with this setting produces a sequence of images in which the object in focus remains stationary but objects on other depth planes appear to translate in a circular path.

A change in the amplitude of rotation alters the radius of the circular path traced about the point C. Small amplitude implies small displacements (or parallax) and hence can give

a pleasing perception or can be impossible to detect. However large amplitude means large radius of motion which the observer can perceive as *jitter* in the video. High jitter often causes inconvenience to the observer and hence is not recommended for videos made for viewing. Frequency determines the number of rotations about the center and is also a key factor for a comfortable viewing. Empirical results have shown that a frequency of 4.1 Hz gives a pleasing 3D perception to a human observer and hence is ideal for video sequences. An extensive description on the construction and working of this camera is given in [8].

2.8 Range Estimation

In given scene, the distance of an object from the observer can be found using many techniques. The mostly used method is the conventional radar, where a radio-frequency pulse is emitted and the distance is calculated from the time of flight. The equivalent technique in vision applications is the imaging radar [17], where a laser beam is emitted and the distance of objects is found using either time-difference or phase-difference of the reflection.

Another method uses structured lighting where the projection angle of light in the scene is known a priori. Light reflected from the object forms a triangle with the camera and then the range can be estimated [17]. These techniques are called *active methods* of range estimation because some energy has to be emitted in the scene to measure distances.

Unlike the above methods, there also exist *passive methods* where no energy is emitted in the observed scene. One such technique is to use off-centered aperture. Early studies on range estimation using this technique can be found in [14, 13]. Auto-stereoscopic systems can also be used in range estimation as will be shown later.

2.9 Summary

In this chapter, we presented a short introduction to some important concepts, upon which our work is entirely built. For anyone who is already acquainted with these concepts, this chapter will serve as a short review. Despite being short, we hope that this review helps to grasp the underlying ideas and hence prepare ourselves in our journey henceforth.

Chapter 3

Range Estimation from a Moving Aperture Lens

3.1 Preface

The short review of some fundamental concepts in the earlier chapter will be useful as we introduce our methodology and explain our implementation in this chapter. We hope that the reader is now acquainted with the basic ideas, either from his previous knowledge or from our discussion in Chapter 2.

This chapter is arranged as follows. The principle behind our method is given in Section 3.2 followed by a discussion of our setup in Section 3.3. This discussion then lends itself to our methodology in Section 3.4.3.

3.2 Principle

Consider a world coordinate system with its origin at C, the aperture of a camera, as shown in Fig. 3.1. Let F be the point of focus at a distance z from the aperture. Thus the coordinates

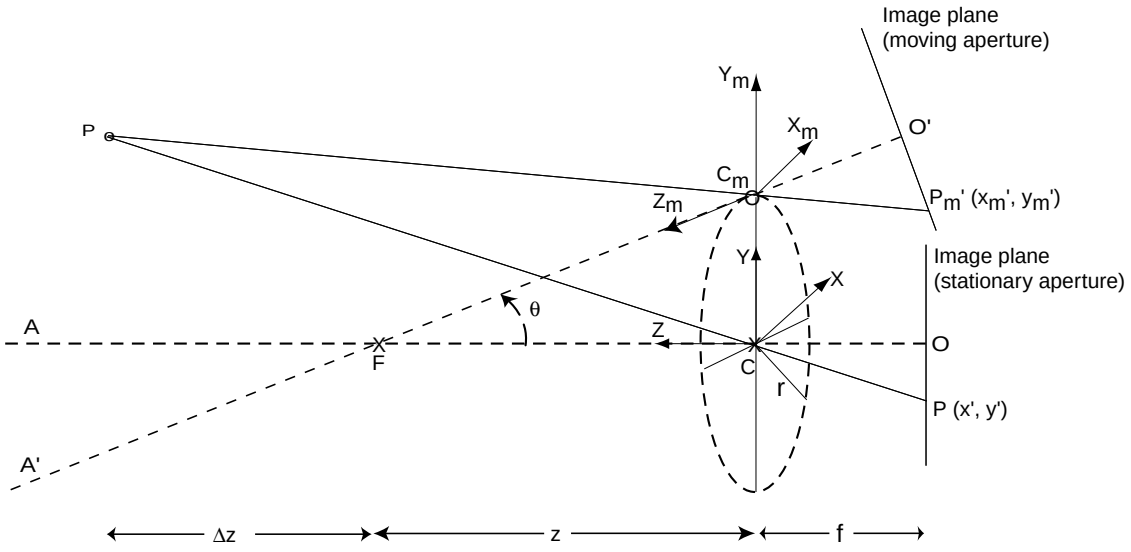


Figure 3.1: Coordinate system and image plane corresponding to a moving aperture.

of F in three-dimensions are $(0, 0, z)$. Consider another point P located at $(x,\ y,\ z+\Delta z)$. For a stationary aperture C, following the discussion in Section 2.2, the point P corresponds to the image plane locations given by

$$x' = f\,\frac{x}{(z+\Delta z)} \quad \text{and} \quad y' = f\,\frac{y}{(z+\Delta z)} \tag{3.1}$$

where f is the focal length of the lens.

Now imagine that the aperture moves in a circle of radius r, around the point C, as illustrated in Fig. 3.1. We represent the moving aperture by the point C_m on the circle. Again, let the point of focus be F at a distance z from the point C_m. The moving aperture with its constant focus forms a cone with F at its apex. Let θ be the angle of the cone, which we shall call as the *scan angle or parallax angle*. (In practice, θ usually is not very large due to implementation reasons.) With respect to C_m, the point P is now located at $(x_m,\ y_m,\ z+\Delta z_m)$ where x_m, y_m, z_m and Δz_m are expressed with respect to the origin now at C_m instead of C. As shown in eqn. (3.1), we can arrive at a similar point on the image

plane which we shall represent as x'_m and y'_m. This new x'_m and y'_m will not be the same as x' and y' due to the shift in aperture and this is equivalent to introducing parallax in the image.

The displacement on the image plane (or parallax) can be written as,

$$\begin{aligned} \Delta x' &= x'_m - x' = f\,\frac{x_m}{(z_m + \Delta z_m)} - f\,\frac{x}{(z + \Delta z)} \\ &= f\,\frac{\Delta z \sin\,\theta + x\cos\,\theta}{(z_m + \Delta z\cos\,\theta - x\sin\,\theta)} - f\,\frac{x}{(z + \Delta z)} \end{aligned} \tag{3.2}$$

Dividing the first numerator and denominator by $\cos\theta$, and using the approximations $z \approx \frac{z_m}{\cos\,\theta}$ and $x\tan\theta \approx 0$, we can combine terms to obtain the displacement,

$$\Delta x' \approx \frac{\Delta z}{(z + \Delta z)}\, f\tan\,\theta \tag{3.3}$$

Thus for a given focal length and scan angle, the resulting image displacement $\Delta x'$, is a function of the focal length f, distance to the point of focus z and the relative object distance Δz. Objects at the plane of focus correspond to the case $\Delta z = 0$, and so no image displacement results. i.e., there is no parallax corresponding to focus. This equation also illustrates the fact that the direction of displacement in the image depends on the sign of Δz.

We can rearrange eqn. (3.3) as follows,

$$\Delta z = \frac{z\,\Delta x'}{f\tan\,\theta - \Delta x'} \tag{3.4}$$

to find the relative distance Δz in the scene. If Δx can be estimated from image-matching techniques, and if the remaining variables are also known, then Δz can be estimated accurately.

3.3 Setup

For our analysis, we acquired a video sequence where an auto-stereoscopic camera was looking at a line of trees as shown in Fig. 3.2. For the sake of convenience in referring the trees, we labelled them $T1, T2, T3$ and $T4$ on the image. Let us also call the region beyond $T4$, the *background*.

The original S-VHS video from the auto-stereoscopic camera was converted to a digital image sequence at 640×480 resolution. Since grayscale images are relatively easier to analyze compared to color images, we converted the images to grayscale for our study.

The dimensions of the image shown in Fig. 3.2 are 350×300 (350 pixels length by 300 pixels height) since we extracted this small region from the 640×480 resolution original image. We decided to clip the image as our region of interest (shown in the image) was adequately covered with this size and also because larger size directly adds to the computational cost.

The single image shown above, is sufficient for a human observer to conclude that there are four trees with bushes grown in between these trees. It can also be inferred that $T1$

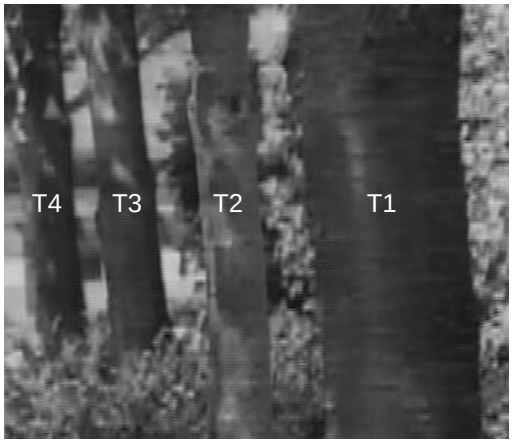


Figure 3.2: Field of view of the moving aperture camera. The trees are labelled $T1 - T4$ on the image, for reference. $T1$ is nearer the observer and trees $T2 - T4$ are increasingly distant.

is closer to the observer while T4 is far away, from a cursory glance. The human visual system possibly utilizes clues like relative size, intensity and texture augmented by its a priori knowledge. Unlike the human visual system, computer vision systems have to rely on various complex algorithms to arrive at a similar conclusion.

When the camera focus changes, we tend to observe some dramatic differences in the resulting images. Fig. 3.3 shows the acquired images for different focus settings when $T1, T2$ and $T4$ are in focus respectively. It can be observed that in Fig. 3.3(a), $T1$ is clearly defined while the background is not well defined. In Fig. 3.3(b) where $T2$ is in focus, we see that the entire scene appears to be better defined in general and in Fig. 3.3(c) we observe that $T1$ appears to be out of focus.

3.4 Methodology

From our discussion in Section 2.7, we know that a moving aperture introduces parallax effect in the captured images. This parallax corresponds to the displacement $\Delta x'$ on the image plane, which is dependant on the distance from the focus Δz (refer (3.3)). Since the aperture moves in a circle, the image flow vectors (or displacement of image points between two images) constitute a circular movement of image points. The radius of this circular motion is none other than the optical flow, which was discussed earlier. For objects in different depth planes, the magnitude of optical flow varies with respect to their distance from focus.

(a)

(b)

(c)

Figure 3.3: Effect of changing focus: (a) $T1$ in focus, (b) $T2$ in focus and (c) $T4$ in focus.

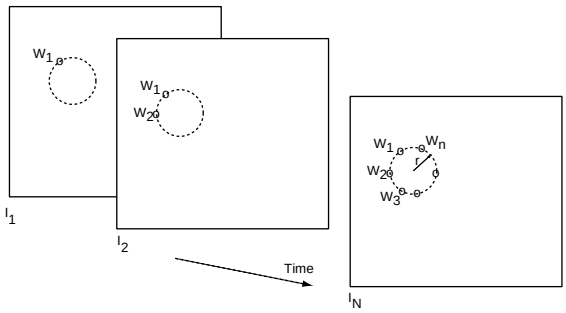


Figure 3.4: Optical flow calculation in images from a moving aperture sequence.

3.4.1 Flow Calculation

Optical flow can be calculated as follows. Consider a sequence of N images $(I_1, I_2, ..., I_N)$ from the moving aperture lens. Let us select an arbitrary image window, centered at location W_1 on the first image I_1 . Using pattern matching techniques like normalized cross-correlation or cross-covariance, we can find a best match to W_1 in the image I_2 (call this point W_2). Similarly we can find the best matching windows in other images $(I_3, I_4, ..., I_N)$ and hence $W_3, W_4, ..., W_N$. This is illustrated in Fig. 3.4.

Due to the circular motion of the aperture, these points $W_1 - W_N$ tend to lie on a circle whose radius r is determined by the distance from the point of focus. We can apply circle fitting procedures (Appendix A) to determine the center of this circular motion and also the corresponding radius (or flow).

We like to emphasize that the number of images $(I_1, I_2, ..., I_N)$ chosen for this analysis should be sufficient enough to fit a circle as otherwise the circle fitting procedure may produce large errors. It is also redundant to select a large N because using overlapping data points for circle fitting may not improve the result significantly.

3.4.2 Segmentation

Once we calculate the optical flow (i.e., radius of motion) in the sequence, we can identify boundaries of objects in various depth planes based on the magnitude of the flow vectors. This yields a result in which object boundaries represent regions in the image and hence a segmented view of the scene.

For example, consider the case when the camera is focussed on $T2$, shown in Fig. 3.5. Since Δz is zero for all points on $T2$, the optical flow $\Delta x'$ is also zero for all points on $T2$. As we already know, the radius of motion (or optical flow) increases when the distance from the point of focus Δz increases. Therefore the flow magnitude for $T4$ is more than that for $T3$. When Δz becomes negative as with $T1$, Δx also becomes negative (refer eqn. (3.3)). But a negative $\Delta x'$ makes no difference to our understanding because it corresponds to the radius and therefore the sign has no effect.

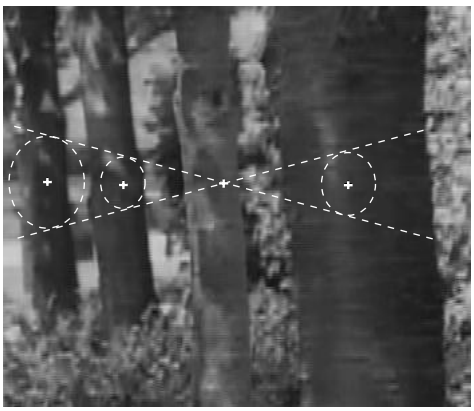


Figure 3.5: Illustration of increasing radius of motion (optical flow) in moving aperture images as distance from the plane of focus increases.

The algorithm for image segmentation can be summarized as follows.

1. Estimate the optical flow field from the series of images using region matching (2.8) and circle fitting (A.6), (A.7), (A.8).

2. Smoothen the calculated radius values using three techniques applied in sequence: measure of fit, maximum radius, and a 3×3 averaging filter. Large values of Q (A.9) indicate a poor measure of fit; thus, a threshold can be used to set these outliers to zero. Next step can be to force all radii larger than the anticipated maximum radius to zero. The anticipated maximum radius can be determined from the scan angle or visually from the video sequence.

3. Segment the scene into distinct regions (with 8-neighbor connectivity) using morphological erosion techniques.

4. Assign unique labels to each identified region [15, 17].

3.4.3 Range Estimation

In addition to image segmentation, the auto-stereoscopic image sequence can be used to estimate the location of the objects (or depth plane) in the scene. Once we know the camera parameters like distance of focus and focal length, we can calculate the distance of objects or depth planes in the scene.

The circle fitting algorithm estimates the center of the circular motion and the radius. If we consider the difference between our initial point in the sequence (W_1) and the calculated center, the direction of shift along the X axis offers a highly valuable information for range sensing. Consider a point in focus F, as shown in Fig. 3.6. As the aperture moves, the point F still continues to be in focus. Therefore the radius of motion is zero and so the center of the circle will not move away from the point F.

For an object (or depth plane) in front of the focus, there will exist some motion. In this case, the difference between the initial point Q from I_1 and the calculated center Q_c

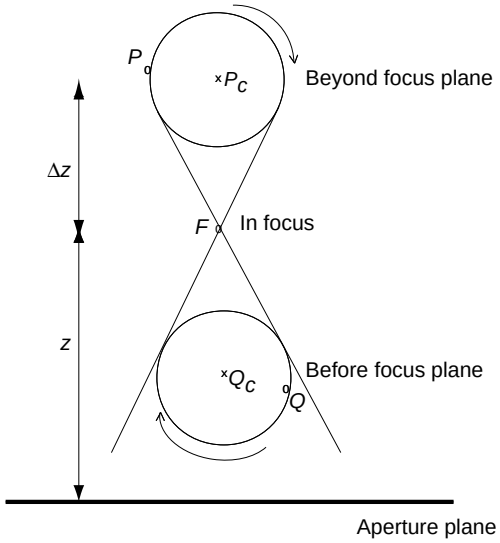


Figure 3.6: Illustration of location estimation.

indicates a shift. In the figure above, the shift from Q to Q_c is to the left and hence can be considered toward the negative X axis (or simply negative). Considering the same image sequence, the initial point P, on a plane beyond the focus will exhibit a positive shift to the calculated center P_c.

This shift information can be utilized to identify the location of objects. In some cases we might see different radius of motion for adjacent regions in which cases it is essential to know the location of objects in the scene. In such cases, this above technique can be employed for location estimation.

The location estimated results when combined with segmentation result gives the range values in the scene. The algorithm for range estimation can be summarized as follows.

1. Apply the segmentation algorithm to segment objects in the scene.

2. For each point in the image, calculate the direction of shift between initial point and calculated center of flow. Plot this data on a 2D plane.

3. Since the principal axes of this data may not be aligned with the regula X and Y axes,

use principal component analysis (PCA) [15, 21] to find the principal axes.

4. Discard outliers that correspond to very large radius values.

5. Transform the points with respect to the calculated principal axes.

6. Data points at the origin (or in focus) correspond to no flow; data points on the left half-plane correspond to image points on one side of the plane of focus and the data on the right half-plane correspond to image points on the other side of the plane of focus.

7. From the flow magnitudes calculated for image segmentations, determine the distance of regions. Thus the regions can be identified as near and far regions.

3.5 Summary

In this chapter we discussed the key principle behind our method and investigated the setup for our work. We also presented a detailed description of our image segmentation approach using the moving aperture lens and also range estimation using the auto-stereoscopic sequence.

Chapter 4

Results

4.1 Preface

In the previous chapter we outlined our methodology for image segmentation and range estimation. In this chapter we present the results obtained using the real video sequence whose scene setup was discussed in Section 3.3.

The video we obtained for our analysis was shot with a frequency of rotation 4.1 Hz and was recorded in NTSC format. Thus digitizing it resulted in approximately 30 images (frames) per second of video or seven images per rotation of the aperture. Therefore seven images are sufficient to determine the circle parameters. For our study, we chose various scenarios with different focus settings.

The accuracy of radius calculations and hence the segmentation results depend on the resolution of our images we use for analysis. Therefore it is extremely important that the resolution of the images be sufficiently high. In our study, the 350×300 resolution was not sufficient enough to calculate the radius precisely. Due to this low resolution, we interpolated our original 350×300 images to 700×600 resolution.

An expert reader, after reading our above statement will jump to the conclusion that interpolating even further will produce more accurate results. But it should be remembered that an increase in image resolution indirectly implies a many fold increase in computational cost and so deciding the balance between resolution and computation is left to the reader depending on the application. In our experiments, we found that interpolations higher than two did not give a significant improvement in results and hence we settled on using the 700×600 resolution images.

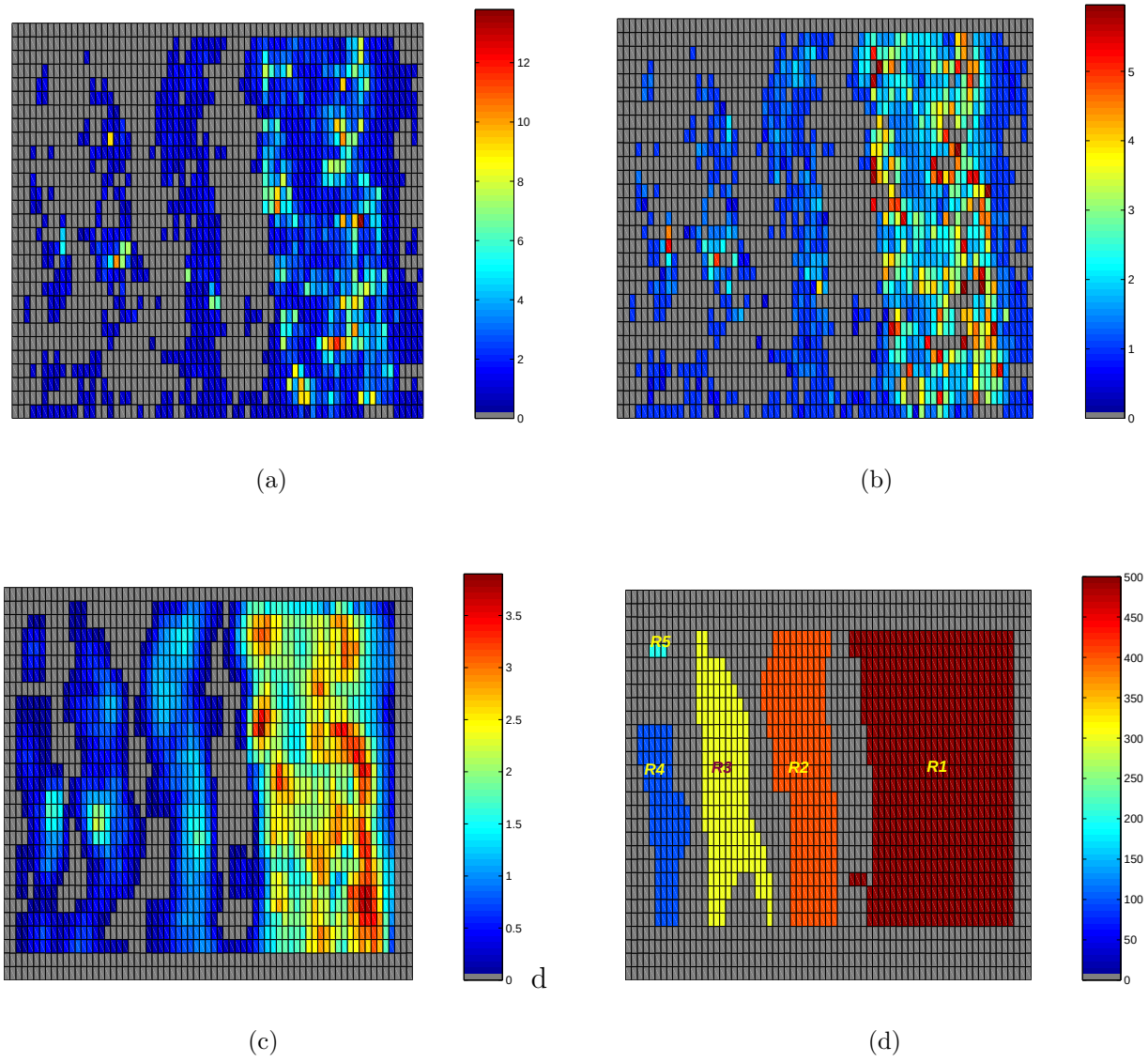
4.2 Far Focus

First we chose to analyze the case when the camera was focussed beyond $T4$ (far focus). In this case, we expect to find zero flow for the image points beyond $T4$ and for regions between the trees. We also intend to find increasing radius on the image, as we move right towards tree $T1$ We like to note here that the low lying bushes in the scene are not located at distinct depth planes but instead extend across the whole scene. So the results obtained in this region of the image can lead to unsatisfactory results and hence deserve a closer observation.

4.2.1 Optical Flow

Figure 4.1(a) shows a plot of the optical flow magnitudes calculated over the entire image. The color scale adjacent to the figure shows the radius values at various points on the image. Blue indicates a small radius of motion (whose value in pixels can be read off the color scale) while red implies a large radius of motion. Although the message is subtle here, it is clear that radius increases as move toward $T1$ in the image.

We notice a few outliers in Fig. 4.1(b), which need to be trimmed down for segmentation.



(a) (b)

(c) (d)

Figure 4.1: Results when the camera focus is beyond $T4$: (a) calculated optical flow magnitudes; (b) flow magnitudes after thresholding based on maximum radius and measure of fit; (c) 3×3 low-pass filter applied to thresholded result; (d) result after morphological erosion and region labelling showing segmented regions corresponding to trees.

Our first step in trimming was to threshold the result based on the maximum anticipated radius which can be obtained from the video or numerically from (3.3). In addition to this threshold, we also used measure of fit (Appendix A) as a criterion to identify and remove the outliers. The result after thresholding is shown in Fig. 4.1(b).

4.2.2 Segmentation

After identifying the outliers we suppressed these points by forcing the radius values to be zeros. We then applied a 3×3 low-pass filter to smooth the data and the result is shown in Fig. 4.1(c). This plot shows a promising sign towards our main goal of image segmentation. We then applied morphological erosion operations to isolate the regions and then a region-labelling procedure to uniquely label each region. The outcome of this operation (Fig. 4.1(d)) distinctly shows the trees $T1-T4$. (We manually added the region labels in our result shown here for convenience.)

Since the background is in focus, the region between the trees which can also be considered as background also are in focus. The trees are the only regions in the image which are not in focus and hence exhibited some optical flow. The results we obtained represents this and hence the success in image segmentation.

Table 4.1 shows the average radius values corresponding to each region. It can be clearly seen that the radius of motion increases as the distance increases from the point of focus.

The computation of numerical distances in the scene using eqn.(3.3), requires that we know the camera parameters. Unfortunately, since we did not have access to these parameters, we were unable to calculate the precise numerical values. Therefore this table which gives radius in pixels demonstrates the feasibility of such numerical calculations.

4.2.3 Range Estimation

Having segmented the trees at different depth planes, we proceeded to estimate the range of trees in the scene. Figure 4.2(a) shows a plot of the differences between the initial point in the sequence and the center calculated from the circle fitting procedure. Applying principal component analysis enabled us to divide this data into three groups - positive shift, negative shift and zero shift. Zero shift in the data indicate that the points is in focus. A positive shift implies some optical flow at the calculated center and corresponds to image points on one side of the focus - either in front of or beyond the current focus. Similarly a negative shift also indicates the presence of flow at the calculated center but also indicates that the image points on the other side of focus. We used a threshold to remove the outliers as followed for optical flow, based on the anticipated maximum radius. The circle in Fig. 4.2(a) indicates the radius threshold of 5 (pixels) as the threshold.

Figure 4.2(b) shows the result after principal component analysis. The multiplicity of each point is not highly apparent in this figure. This result represents the range over the entire image as shown in Fig. 4.2(c). A meticulous observation of this figure shows points in three colors - red, corresponding to points with zero shift, blue and light green corresponding to points on either sides of the focus (in front of or beyond the focus). Now we know which regions lie together and which lie on opposite sides.

Table 4.1: Average radius of identified regions (far focus).

Region	Average Radius (pixels)
R1	1.969
R2	0.829
R3	0.605
R4	0.554
R5	0.522

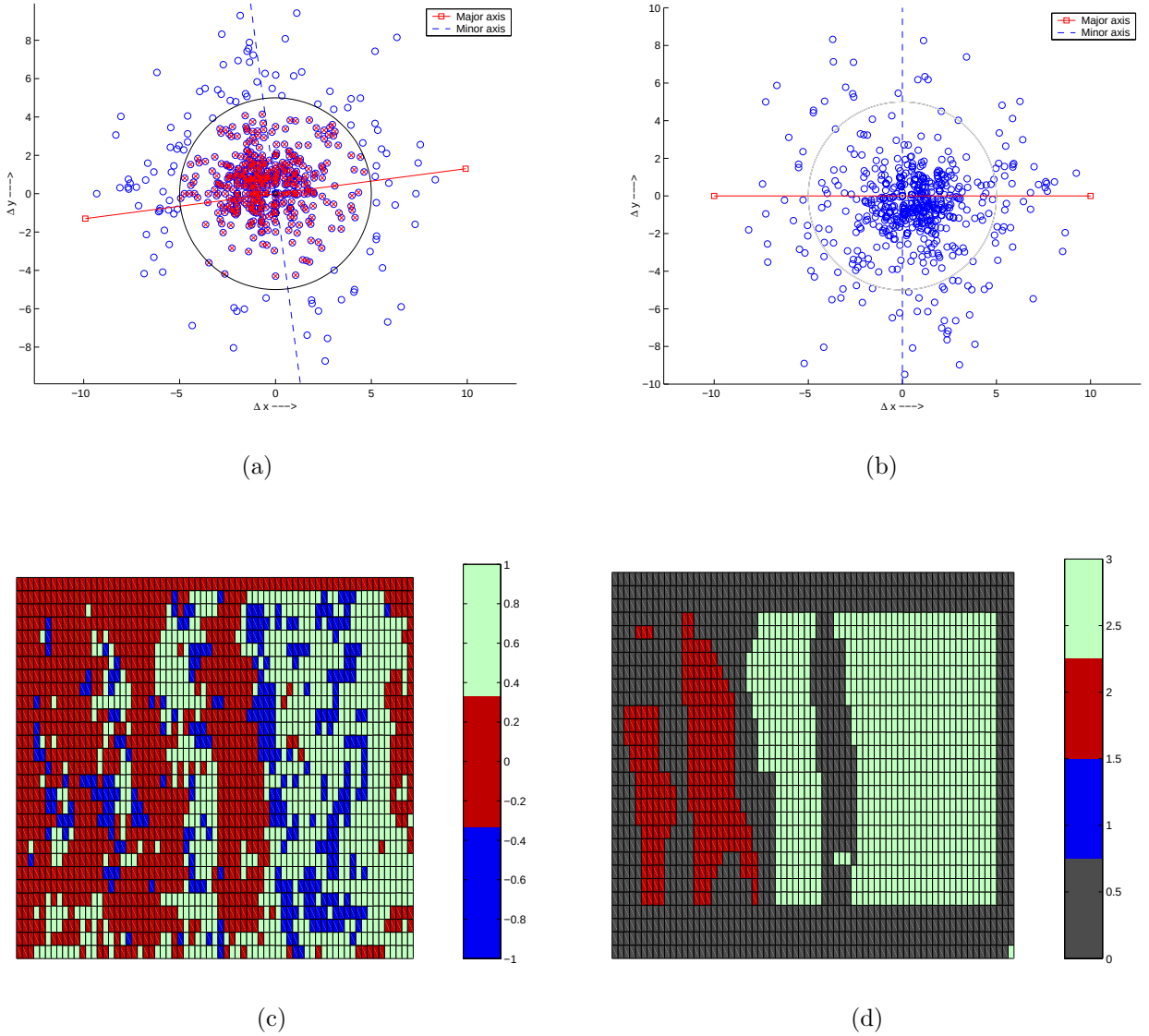


Figure 4.2: Range estimation results for far focus: (a) plot of differences between the initial point and the calculated center; (b) principal component analysis on differences; (c) near-far estimates after principal component analysis; (d) near-far estimates merged with segmentation result.

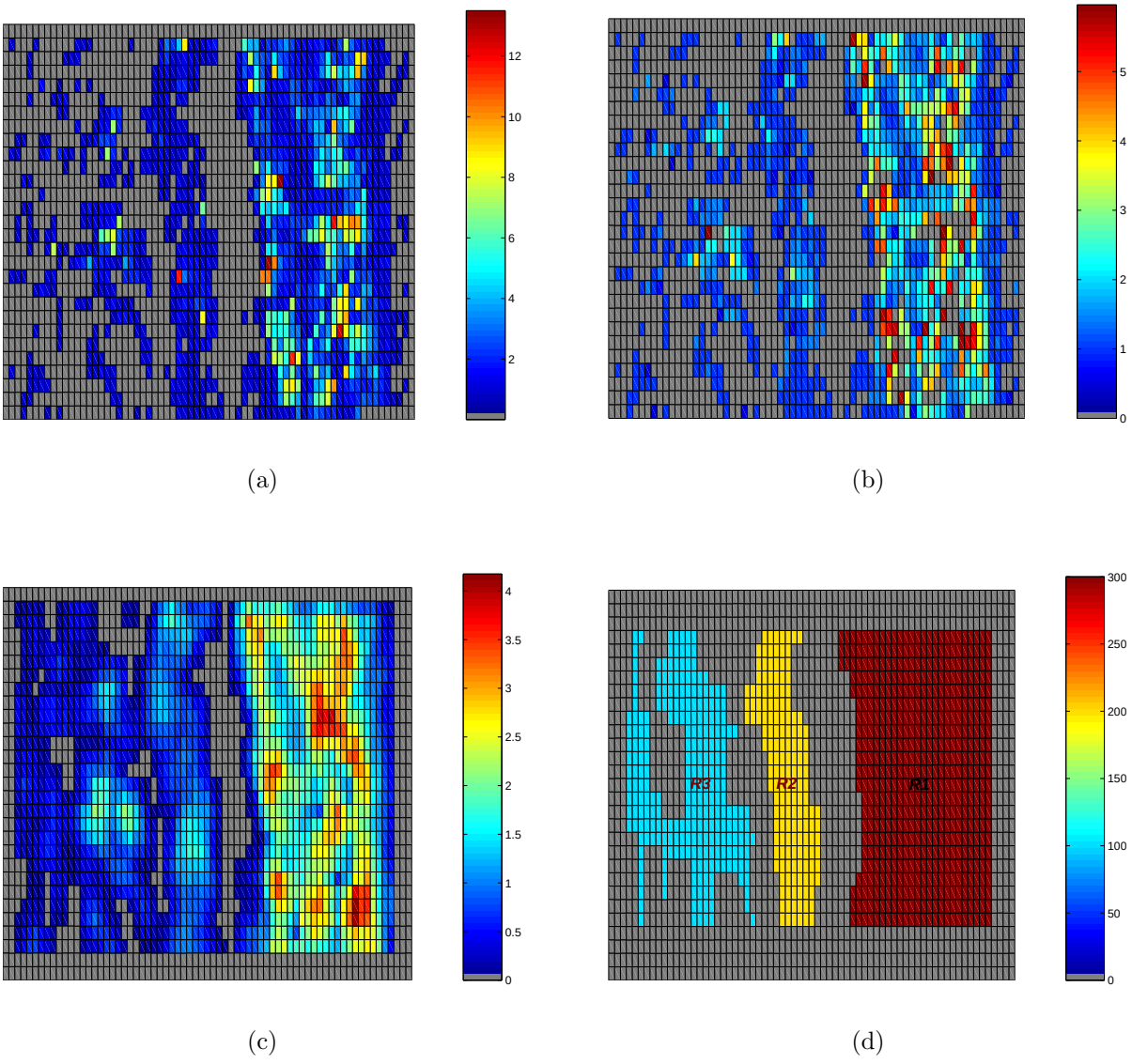
We combined the results from segmentation with labelled regions Fig. 4.1(d) with the location estimates in Fig. 4.2(c) and the result is shown in Fig. 4.2(d). The region shown in red implies the focus while blue and light green regions indicate the other regions not in focus. This figure shows all trees ($T1 - T4$) in one color implying that these trees lie on one side of the focus, which we know is in front of the focus. The reason range estimation gives $T3$ and $T4$ to be in focus although the focus was beyond $T4$ is due to the various thresholding and filtering operations performed. Nevertheless these approximate estimates can be accurately calculated if we have high resolution images and known camera parameters.

The radius information from Table 4.1 provides the necessary information for numerical calculation and the location estimation result suggests the possible location of the region. Thus, in short we have demonstrated the possibility of range estimation using an auto-stereoscopic camera.

4.3 Far Focus - Another Example

We also studied another sequence where the camera was focussed beyond $T4$. In this case also, we obtained similar results as presented earlier. Fig. 4.3 shows the segmentation results corresponding to this sequence.

It can be seen that erosion resulted in only three regions with a connection between for $T3$ and $T4$. If we notice Table 4.2, it is clear that the average radius value increases with increasing distance from the focus. The range estimation results are shown in Fig. 4.4.



(a) (b)

(c) (d)

Figure 4.3: Results when the camera focus is beyond $T4$ (far focus - example 2): (a) calculated optical flow magnitudes; (b) flow magnitudes after thresholding based on maximum radius and measure of fit; (c) 3×3 low-pass filter applied to thresholded result; (d) result after morphological erosion and region labelling showing segmented regions corresponding to trees.

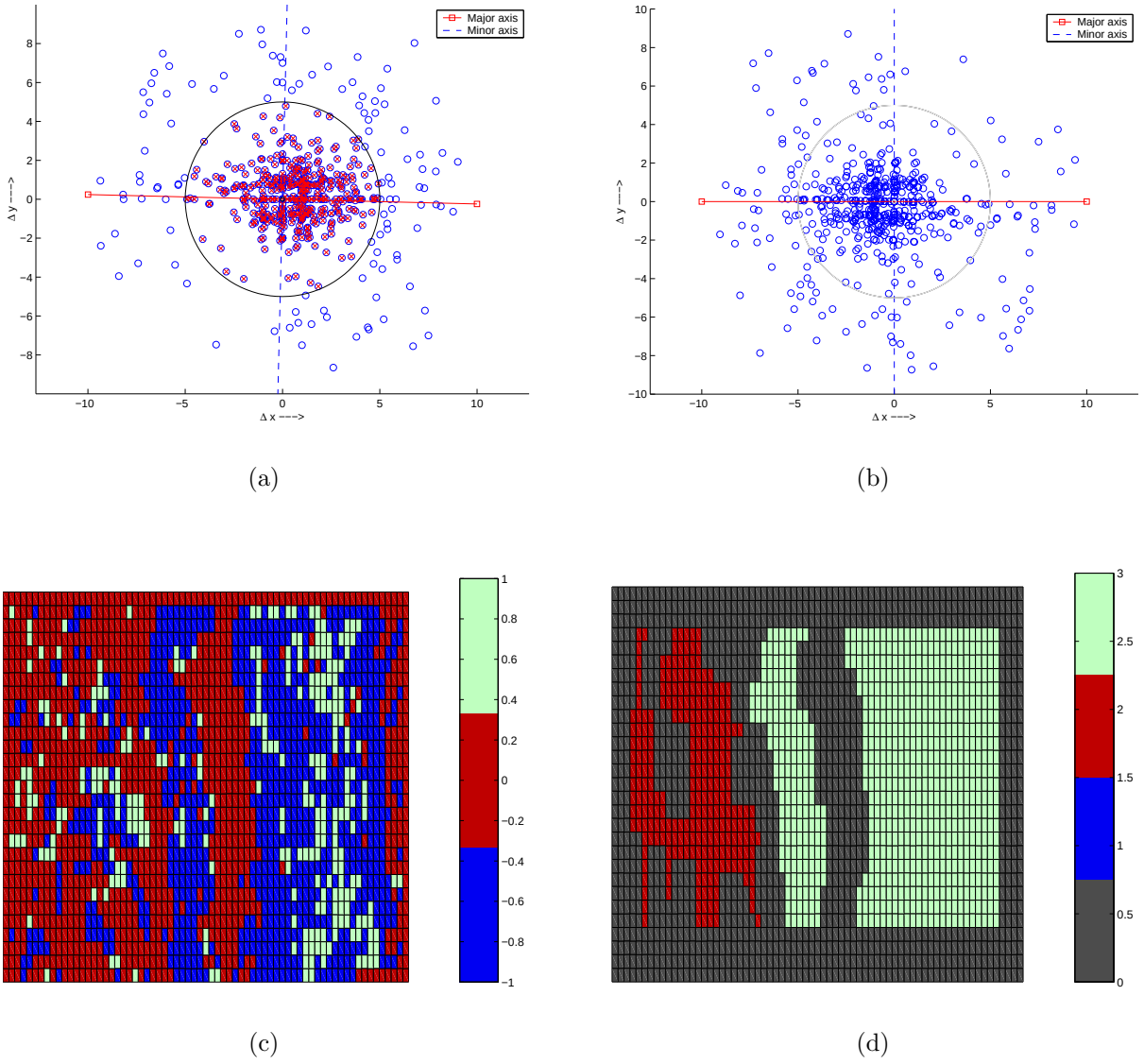


Figure 4.4: Range estimation results for far focus - example 2: (a) plot of differences between the initial point and the calculated center; (b) principal component analysis on differences; (c) near-far estimates after principal component analysis; (d) near-far estimates merged with segmentation result.

Table 4.2: Average radius of identified regions (far focus- 2).

Region	Average Radius (pixels)
R1	1.982
R2	0.927
R3	0.622

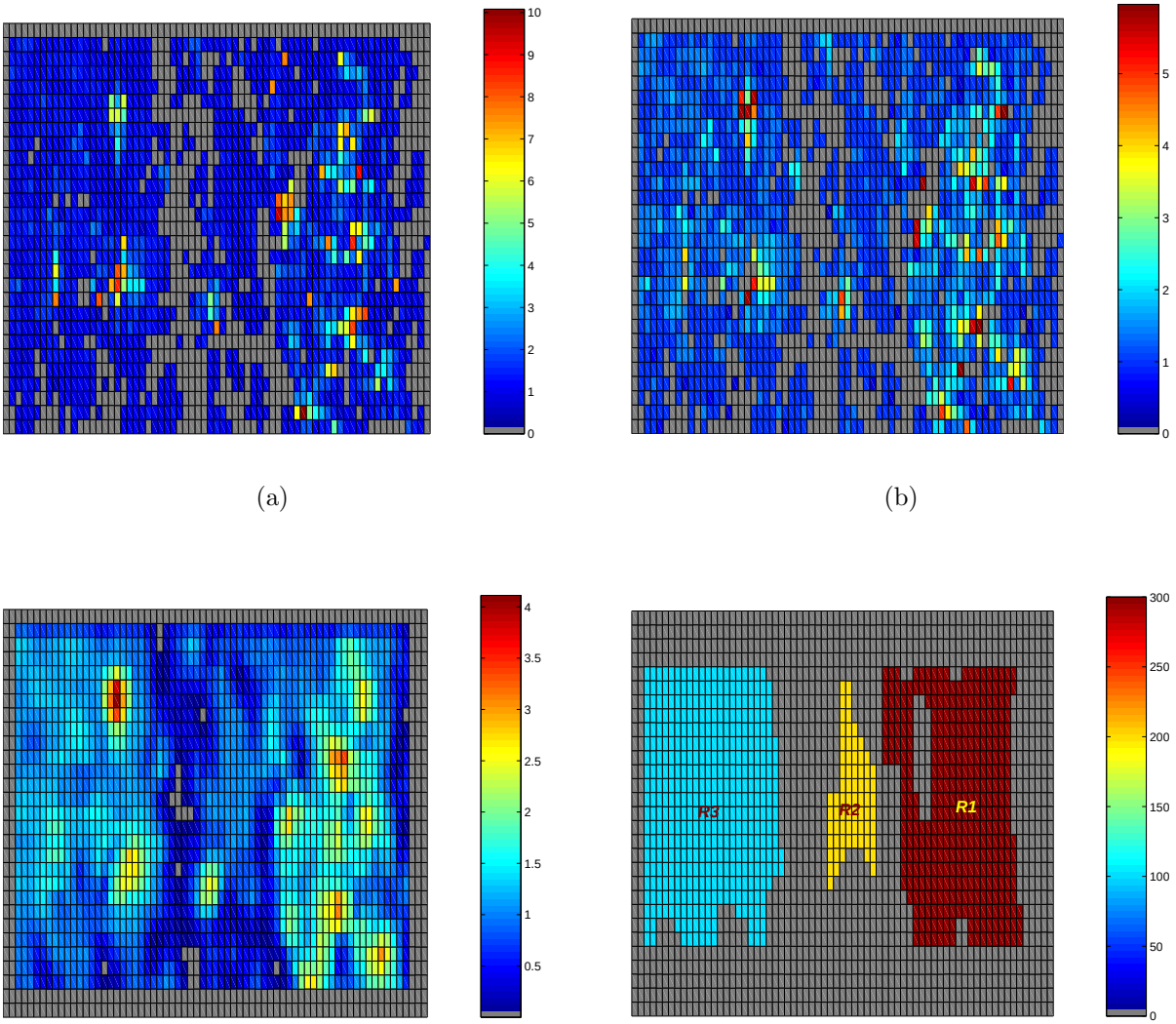
4.4 Mid-range Focus

Next in our analysis, we decided to study the case when the camera is focussed on $T2$ (hence mid-range focus). In this case, the tree $T2$ will exhibit zero optical flow. As we move outwards toward the edges of the image, the distance from $T2$ increases and hence we expect an increasing radius on both directions.

4.4.1 Segmentation

Figure 4.5 shows the similar plots as discussed earlier for the far-focus case. It can be seen here that overall there is more flow in the image and the zero flow region corresponding to $T2$ is apparent from Fig. 4.5(a). The result after thresholding, averaging, morphological eroding and region labelling is shown in Fig. 4.5(d). Again the regions are labelled on the image, mainly for the sake of convenience. This labelled result shows the regions - R1 corresponding to $T1$, R2 corresponding to the region between $T1$ and $T2$ and R3 corresponding to all the region located on the right of $T2$. Thus Fig. 4.5(d) matches our expectation for segmentation.

Table 4.3 lists the radius values for each labelled region. It can be seen that the radius increases as we move away from $T2$ and hence is representative of our desired result.



(a) (b)

(c) (d)

Figure 4.5: Results when the camera focus is on $T2$: (a) calculated optical flow magnitudes; (b) flow magnitudes after thresholding based on maximum radius and measure of fit; (c) 3×3 low-pass filter applied to thresholded result; (d) result after morphological erosion showing segmented regions corresponding to trees.

Table 4.3: Average radius of identified regions (mid-range focus).

Region	Average Radius (pixels)
R1	1.527
R2	0.997
R3	1.306

4.4.2 Range Estimation

In this case, we see that the differences and their principal component analysis plots are as shown in Fig. 4.6. As per our understanding, the estimates in this case should be such that $T1$ lies on one side of the focus while the region between $T1$ and $T2$ together with the region beyond $T2$ should lie on the other side of the focus. Clearly the results shown in Fig. 4.6(c) testify the validity of our understanding.

4.5 Computational Details

To address the computational aspects of our implementations, we used the C++ programming language and Matlab in our analysis. Our programs were run both on Windows NT and SunOS 7 systems. Our implementation to find the best matching window ran for more than $6 - 8$ hours on a Sparc 10 system. This is due to the original size of the images (600×700), size of window we chose (41×21) and the search area we decided on (81×41). But our other implementations (mostly Matlab programs) required less than a minute to analyze the data, once the matching window locations were found.

Although our algorithms were relatively simple, the search for a best matching in the image sequence considerably needed a long time to find the match. With special hardware implementations we believe that this problem can be circumvented and hence should not be

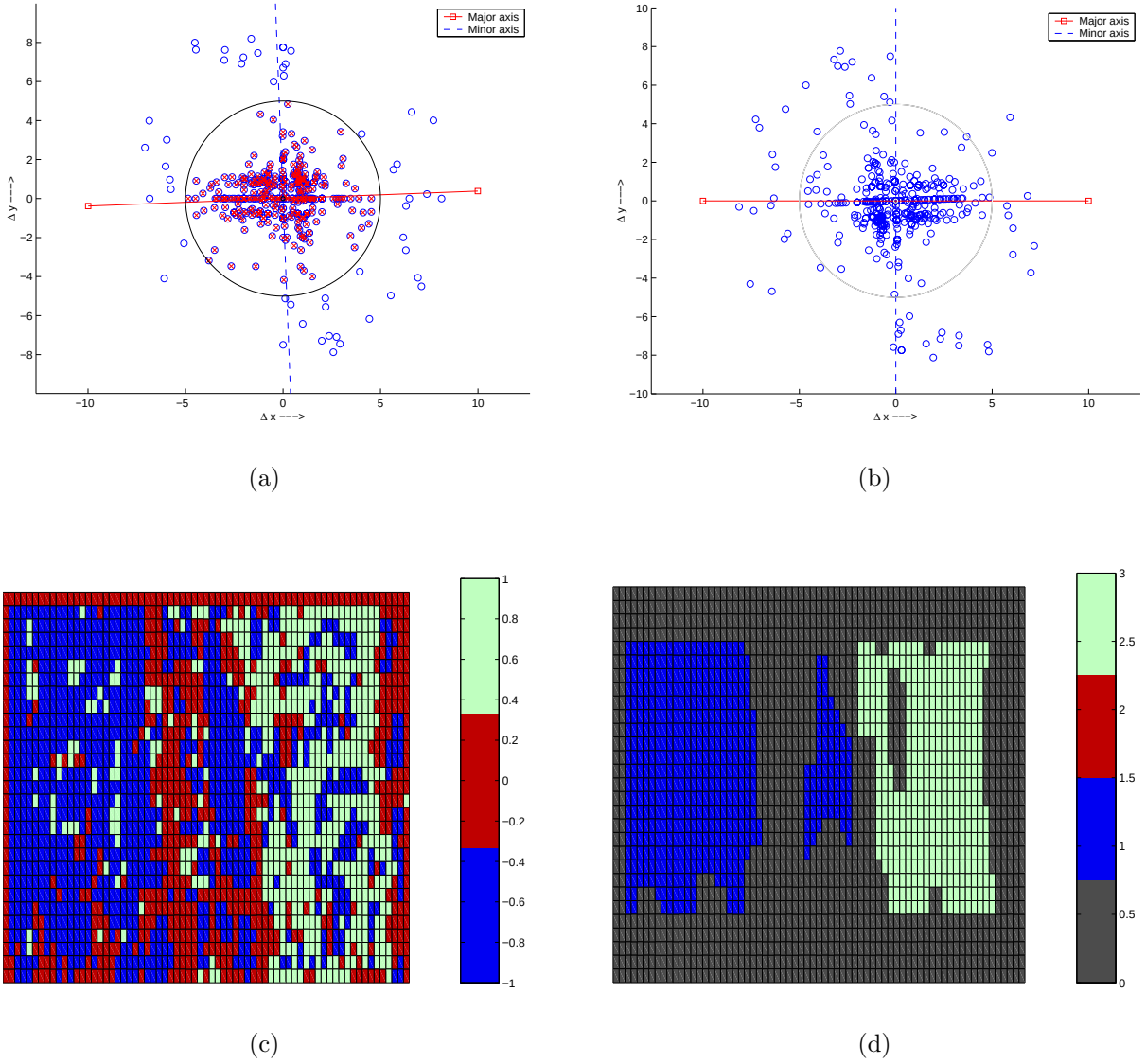


Figure 4.6: Range estimation results for mid-range focus: (a) plot of differences between the initial point and the calculated center; (b) principal component analysis on differences; (c) near-far estimates after principal component analysis; (d) near-far estimates merged with segmentation result.

regarded as a bottleneck.

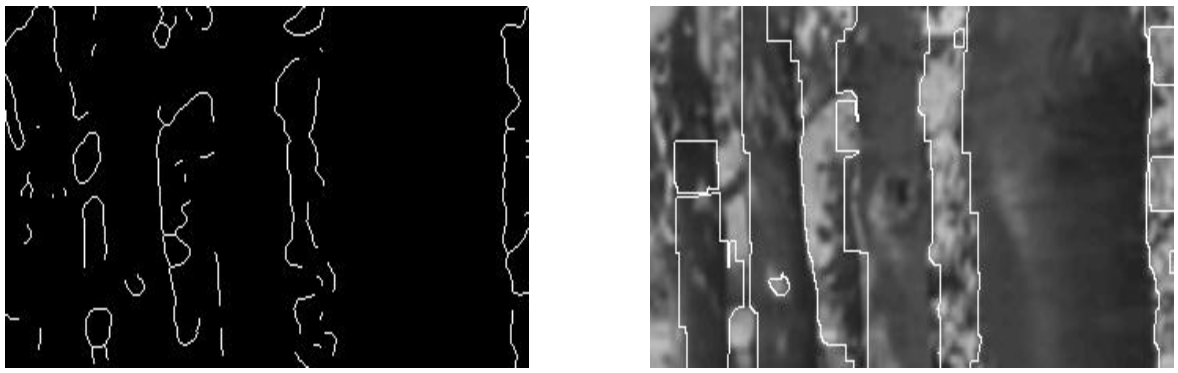
4.6 Disparity Integration

The results of our procedure do not represent the actual edges of the trees. The larger size of identified regions is due to the various operations performed in due course of our analysis like filtering, thresholding and morphological operations. However when used in conjuction with the Disparity Integration(DI) procedure [22], the region edges can be accurately identified which shows the strength in our methodology. The results obtained for the far focus case, when the camera is focussed beyond $T4$ are shown in Fig. 4.7.

4.7 Limitations

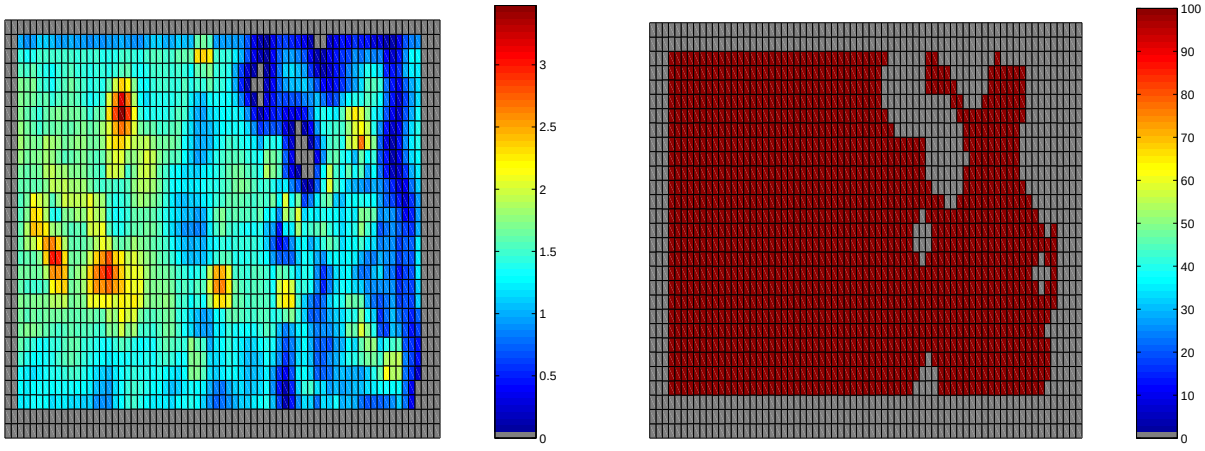
Although the results from our method presented so far are encouraging, it has some limitations. For example, let us consider the case when the point of focus is on the nearest tree T1. Now every object in the scene is behind focus and hence will have some flow. Even though we can estimate their distances theoretically we found it challenging because of the absence of region boundaries.

Figure 4.8 shows the flow magnitudes after filtering and the corresponding result after segmentation, when the camera was focussed on T1. It can be seen that the entire region beyond T1 appears to be one object with motion and hence an inherent difficulty in segmentation in this case.



(a) (b)

Figure 4.7: Disparity Integration algorithm: (a) depth-edge map resulting from the DI algorithm; (b) depth-edge map merged with Fig. 4.1(d) which shows edges of the identified regions. (Images courtesy of Lakshmi R. Iyer.)



(a) (b)

Figure 4.8: Results obtained when T1 is in focus: (a) flow magnitudes after filtering; (b) segmentation after erosion.

4.8 Summary

Through our results we demonstrated that both image segmentation and range estimation are possible using the images from an auto-stereoscopic camera. Since we did not know the focal length and the distance of the focus with respect to the camera in both cases, we could not present a numerical result. Nevertheless our results illustrate the feasibility of this once we know these parameters.

Chapter 5

Conclusion

This thesis has presented our novel approach for image segmentation and range estimation along with the results obtained. By extracting the optical flow information in a video sequence with near and far focus settings, we showed that the boundaries of various objects in the scene can be clearly determined and hence the objects can be easily segmented.

In addition to image segmentation, we can also estimate the range of objects (or depth planes). Consequently we can combine the results from segmentation and the location estimates to compute the location and distance of regions located before and behind the focus.

It is highly important to remember that our method empowers us this capability with only a single moving-aperture camera.

Future Work

In future work, we consider various possible scenarios to perform segmentation and range estimation. For example, we envision scenes with large horizontal surfaces that span the entire scene, non-stationary objects, moving camera and deforming objects. Our analysis will serve as a preliminary investigation into the possibilities of range estimation with moving-

aperture lens. From our results, we strongly hope that further work can address these complex cases.

Our circle fitting methodology deserves more attention and improvement. In our study, we found that our iterative circle fitting procedure mostly converged despite a formal investigation of its convergence properties. Circle fitting itself being a non-linear problem still remains a whole area for research.

As we mentioned earlier we used seven images from the video sequence in our work. For anyone interested in information theory a thought that would have sprout is the redundancy of information used for circle fitting. It is true that we can reduce the number of images chosen from one rotation of the aperture and still determine the flow magnitudes accurately. By choosing less number of images, we can also predict the magnitudes using Kalman filters and yet find the radius values.

A key application which we envisage to benefit from this technique, is passive range estimation. Most of the present day range calculation techniques usually emit some kind of electromagnetic energy in form of radio frequency pulses or laser, in the scene and hence compute the location of objects. In contrast, optical flow techniques do not require any energy to be emitted and still can be used for range estimation.

As discussed in Chapter 2, the technology for 3D transmission and viewer perception is yet to be commercially implemented. By utilizing an auto-stereoscopic camera and our approach, it should be possible to build an efficient system for 3D video transmission. Instead of transmitting every frame in a video sequence, significantly less information can be transmitted using video compression with optical flow. At the receiver end, appropriate hardware implementation can decompress and add parallax effect to the scene, based on the flow magnitudes transmitted. This system thus enables the viewer to experience a 3D

perception of the video transmitted. Other potential applications are video-conferencing, image compositing and autonomous vehicle navigation.

Appendix A

Circle Fitting

Linear regression methods of fitting a line or polynomial to data samples are often encountered in many applications. In general this is a linear problem and hence is easier. Every so often in applications like computer graphics, particle trajectories in high energy physics and lunar craters in astronomy [23], it is desired to fit a circle to the given data.

The problem of fitting a circle to data is comparatively difficult due to the non-linearities involved [24]. Here the goal is, for the given data, find the best values for the center and radius of the circle. Various iterative and non-iterative techniques have been proposed to solve this problem [24, 25, 26, 27, 28]. The choice of a method to solve the problem is influenced by the merits and demerits of each method.

Most of the proposed methods are variations of least-squares approximation technique. A common approach, given a set of $N \geq 3$ points $\{(x_i, y_i)|i = 1...N\}$, is to determine the radius r and location of the center (a, b) that minimizes the error criterion

$$E = \sum_{i=1}^{N} [r_i - r]^2 \tag{A.1}$$

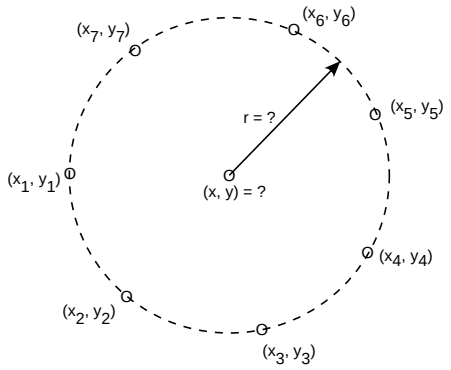


Figure A.1: Circle fitting problem - given seven two-dimensional points, find their center and radius.

where

$$r_i = \sqrt{(x_i - a)^2 + (y_i - b)^2}. \tag{A.2}$$

It is assumed here that the points are non-collinear.

Our work employed an iterative method described in [29]. Taking partial derivatives of (A.1) and setting the resulting equations equal to zero, it is possible to derive the following relations.

$$r = \frac{1}{N}\sum_{i=1}^{N} r_i \tag{A.3}$$

$$a = \frac{1}{N}\sum_{i=1}^{N} x_i + r\frac{1}{N}\sum_{i=1}^{N}\frac{\partial r_i}{\partial a} \tag{A.4}$$

$$b = \frac{1}{N}\sum_{i=1}^{N} y_i + r\frac{1}{N}\sum_{i=1}^{N}\frac{\partial r_i}{\partial b} \tag{A.5}$$

This leads to the following update rules, where the superscript represents the iteration index:

$$r^{(k+1)} = \frac{1}{N}\sum_{i=1}^{N} r_i^{(k)} \tag{A.6}$$

$$a^{(k+1)} = \frac{1}{N}\sum_{i=1}^{N} x_i + \{\frac{r^{(k+1)}}{N}\sum_{i=1}^{N}\frac{a^{(k)} - x_i}{r_i}\} \tag{A.7}$$

$$b^{(k+1)} = \frac{1}{N}\sum_{i=1}^{N} y_i + \{\frac{r^{(k+1)}}{N}\sum_{i=1}^{N}\frac{b^{(k)} - y_i}{r_i}\} \tag{A.8}$$

We keep updating a and b until the change is very small. Experimentally, this procedure has resulted in rapid convergence. To assess the quality of fit, we use the measure

$$Q = \sum_{i=1}^{N} |r_i^2 - \hat{r}^2| \tag{A.9}$$

where, $\hat{r}$ is the final estimate of the radius (i.e. the value of $r^{(k)}$ in the final iteration). A low value for Q implies a good measure of fit (MOF), while a high value for Q indicates a poor fit.

For a detailed discussion on alternative methods of circle fitting [24, 30] are suggested as good references.

Bibliography

[1] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1, pp. 185–203, 1981.

[2] F. Bergholm and S. Carlsson, "A theory of optical flow," *Computer Graphic Models and Image Processing*, vol. 53, pp. 171–188, Mar. 1991.

[3] J. Y. A. Wang and E. H. Adelson, "Layered representation for motion analysis," in *Computer Vision and Pattern Recognition Proceedings*, pp. 361 –366, IEEE Computer Society, June 1993.

[4] J. Y. A. Wang and E. H. Adelson, "Representing moving images with layers," *IEEE Transactions on Image Processing*, vol. 3, pp. 625–638, Sept. 1994.

[5] A. Bab-Hadiashar and D. Suter, "Robust optic flow computation," *International Journal of Computer Vision*, vol. 29, pp. 59–77, Aug. 1998.

[6] K. Satoh, I. Kitahara, and Y. Ohta, "3d image display with motion parallax by camera matrix stereo," in *Multimedia Computing and Systems, Proceedings of the Third IEEE International*, pp. 349 –357, June 1996.

[7] C. A. Mayhew, "Vision III single-camera auto-stereoscopic methods," *Society of Motion Picture & Television Engineers (SMPTE) Journal*, vol. 100, pp. 416–422, June 1991.

[8] C. A. Mayhew and A. Bacs Jr, "Parallax scanning using a single lens," in *Proceedings of Society of Photo-Optical Instrumentation Engineers The International Society for Optical Engineering*, vol. 2653, pp. 154–160, 1996.

[9] J. M. Loomis and D. W. Eby, "Relative motion parallax and the perception of structure from motion," in *Workshop on Visual Motion*, pp. 204–211, 1989.

[10] D. Runde, "How to realize a natural image reproduction using stereoscopic displays with motion parallax," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, pp. 376–386, Apr. 2000.

[11] D. R. Heisterkamp and P. Bhattacharya, "Object and motion recognition using the plane plus parallax displacement of conics," in *Proceedings on Pattern Recognition, Fourteenth International Conference*, vol. 1, pp. 751 –753, June 1998.

[12] M. Bass, ed., *The Handbook of Optics*. McGraw-Hill, second ed., 1995.

[13] F. Blais, M. Rioux, and J. Domey, "Compact three-dimensional camera for robot and vehicle guidance," *Optics & Lasers in Engineering*, vol. 10, pp. 227– 239, 1989.

[14] F. Blais and M. Lecavalier, "Application of the biris range sensor for volume evaluation," in *Proceedings of the Optical 3-D Measurement Techniques III*, pp. 404–413, Oct. 1995.

[15] R. Gonzalez and R. Woods, *Digital Image Processing*. Addison-Wesley, 1992.

[16] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Addison-Wesley, 1992.

[17] R. Jain, R. Kasturi, and B. G. Schunck, *Machine Vision*. McGrawHill, 1995.

[18] Online Image Database at University of Southern California. `http://sipi.usc.edu/services/database/Database.html`.

[19] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, pp. 43–77, Feb. 1994.

[20] J. Shi and C. Tomasi, "Good features to track," *Computer Vision and Pattern Recognition Proceedings*, pp. 593–600, 1994.

[21] G. Strang, *Linear Algebra and Its Applications*. Academic Press, second ed., 1980.

[22] A. Subramanian, L. R. Iyer, A. L. Abbott, and A. E. Bell, "Image segmentation and range sensing using a moving-aperture lens," in *Proceedings of the International Conference on Computer Vision*, July 2000. (to appear).

[23] V. Karimäki, "Fast code to fit circular arcs," *Computer Physics Communications*, vol. 69, pp. 133–141, June 1991.

[24] L. Moura and R. Kitney, "A direct method for least-squares circle fitting," *Computer Physics Communications*, vol. 64, pp. 57–63, Apr. 1991.

[25] I. D. Coope, "Circle fitting by linear and nonlinear least squares," *Journal of Optimization Theory & Applications*, vol. 76, pp. 381–388, Feb. 1993.

[26] C. A. Corral and C. S. Lindquist, "On implementing kasa's circle fit procedure," *IEEE Transactions on Instrumentation and Measurement*, vol. 47, pp. 789 –795, June 1998.

[27] M. Werman and D. Keren, "A novel bayesian method for fitting parametric and non-parametric models to noisy data," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, p. 558, IEEE Computer Society, June 1999.

[28] W. Yi, "A fast finding and fitting algorithm to detect circles," in *International Geoscience and Remote Sensing Symposium Proceedings - IGARSS*, vol. 2, pp. 1187 –1189, IEEE International, Aug. 1998.

[29] D. Eberly, "Least squares fitting of data," tech. rep., Magic Software Inc. `http://www.magic-software.com`.

[30] N. I. Chernov and G. A. Ososkov, "Effective algorithms for circle fitting," *Computer Physics Communications*, vol. 33, pp. 329–333, Oct. 1984.

Vita

Anbumani Subramanian was born in the town of Tiruvannamalai, Tamilnadu (India) in 1975. After his school education at Tiruvannamalai, he went to Coimbatore Institute of Technology, in Coimbatore (India) for a Bachelor of Engineering degree in Electrical and Electronics Engineering. After graduating with Distinction in 1996, he joined IBM Global Services to work as a Software Engineer. He was involved in software development at IBM for about three years from 1996 to 1999. Due to a rejuvenated interest in graduate studies, he came to Virginia Tech to obtain a Master of Science degree in Electrical Engineering. He graduated with an M.S. in May 2001. His interests in Computer Vision and Signal Processing strongly suggested him to pursue his Ph.D..