FEATURE-BASED MOTION ESTIMATION AND MOTION SEGMENTATION

XIN CHENG

Feature-based Motion Estimation and Motion Segmentation

Xin Cheng

B.S.E.E., Yan Shan University, China, 1992
M.S.E.E., Shanghai Jiao Tong University, China, 1995

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Engineering

in

Electrical Engineering

Memorial University of Newfoundland

ST. JOHN'S NEWFOUNDLAND CANADA

1999

Abstract

With the advancement of multimedia and hypermedia, video sequence segmentation is becoming more and more important. This thesis presents a new paradigm in the field of video sequence segmentation research. This new approach employs a limited number of landmarks to represent image features and segments video sequences into distinct moving objects according to the relation between different kinds of motion. The strength of this approach lies in the fact that the motion computations are performed only on the landmarks. By adjusting the number and positions of the landmarks, the scheme controls the motion estimation and motion segmentation procedure.

To serve distinct purposes, the scheme uses three different approaches to produce landmarks: the manual mode, the tiling mode, and the automatic mode. For these three modes, the selection of landmarks, motion estimation, clustering, and motion segmentation are applied on standard video sequences. Two new methods for identifying landmarks automatically have been developed and are presented. These methods are compared with typical feature point detectors over different kinds of image pictures, showing that the new methods are more efficient than previous methods. One novel aspect of the scheme is the coupling of automatic landmark identification with motion estimation, producing very accurate motion estimation.

As an application, bubble tracking for multiphase fluid flow is presented and demonstrated. To validate the scheme, a motion analysis platform has been developed to implement the overall scheme. This platform, with comprehensive functions for motion analysis and image processing, makes a practical contribution to video analysis.

Acknowledgements

First, I would like to thank my supervisor, Dr. John Robinson, for all the perspective and wisdom he has put into my graduate studies over these two years. I am especially grateful to Robinson for his expert guidance and the substantial time and effort he has put into my thesis.

I wish to thank Dr. Chan Ching, for first giving me a position to work on image processing two years ago and then for the financial support in the following two years.

My special thanks go to a Ph.D student, Li-Te Cheng, for his wonderful MCL Gallery and much helpful advice.

Many thanks go to my present and former colleagues and friends, Rahim Pira, Charles Robertson, Mohsin Riaz, Moorthy Manoranjan, Andrew Vardy, Qing Song, Xiaomeng Ping, Yan Shu, Jinsong Xu, Dexiang Luo, Zhikai Ding and Meide Qiu. I will sincerely miss the time I have spent in the MCL lab and all my friends there.

Last, but most, I am grateful to my family for being who and for what they are. I thank my elder brother, my dad and my mom for their endless love and support, and for their unwavering confidence in me. They are the sole reason for any progress I have made.

Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Background

Motion analysis involves motion estimation and motion recognition. Motion analysis has many applications including video coding (MPEG), image understanding (depth and structure from motion), obstacle avoidance, object tracking and surveillance, video annotation and indexing and composing. The motion analysis of this thesis includes motion estimation and motion segmentation [1], [16], [7].

This thesis developed from a research program in Multiphase Flow, which is seeking new insights from image processing and computer vision to enrich the understanding of multiphase fluid dynamics. To serve this research, I built a basic modular platform for the processing and computation of multiphase flow images, and I have produced bubble-tracking sequences. The image processing of multiphase flow, therefore, becomes a primary application of my research. From the point of view of computer vision, there is significant interest in other motion analysis and applications. I have thus developed a compact motion analysis platform to serve many purposes.

A practical motivation for the development of motion segmentation algorithms is the eventual introduction to object-based video. Object-based video is one goal that multimedia communications has been pursuing so far [30]. The primary advantages of object-based video arise from several sources. First, the statistics of an object will be more uniform than of blocks in an arbitrary grid dividing the image picture. Second, video objects can be coded by different techniques and updated at different rates. In addition, given *a priori* knowledge, a system can choose to degrade objects differentially in response to limited channel resources. Motion segmentation is still in the research stage and has not become standard. I would like to contribute in this field. **Feature-based motion estimation and motion segmentation** is presented as such a contribution.

1.2 Approach

There are several novel aspects to the approach presented here. Motion segmentation methods typically compute on every pixel within an image picture. To be efficient, the motion segmentation methods need to tolerate all kinds of noise and boundary points, and therefore incorporate complex computations and require a high degree of robustness. **The first novel point** in the approach is that, distinguished from typical motion segmentation methods, the scheme employs a limited number of landmarks to represent image features and the video sequence is then segmented according to the relation between the motions of the landmarks. The advantage of the approach is that since the motion computations are performed only on the landmarks, through the adjustments of the number and positions of landmarks, the motion estimation and motion segmentation procedure can be controlled. This kind of control offers valuable flexibility to motion segmentation. Further, robustness is naturally obtained from this scheme.

One of the primary goals is to implement automatic landmark identification and motion segmentation. **The second novel point** for the strategy is to couple identifying landmarks automatically with motion estimation through the matching strategy, which leads to the accurate motion estimation of the landmarks and therefore provides reliable input to motion segmentation.

The third novel point of the approach is that two efficient methods have been invented for identifying landmarks automatically. The power of the methods is that they are very efficient and flexible enough for many types of pictures.

The fourth novel point in the scheme is the use of a manual approach. This approach introduces subjective judgments and actions, which help to implement motion segmentation for non-rigid objects and further correct errors resulting from motion estimation.

In a video sequence, what we try to segment out could be a real object, part of an object, or the background. Segmented objects are further classified into rigid objects and non-rigid objects. Since rigid objects do not change their shape, their motion is gradual and continuous. Conventional motion estimation techniques are able to capture this kind of rigid motion accurately, which provides reliable input data to the motion segmentation.

Non-rigid moving objects, on the other hand, vary in size and texture, which makes them difficult to recognize and track. The result from the motion estimation therefore cannot be guaranteed to be correct. Accurate motion estimation of highly non-rigid objects remains elusive. My original research, however, has to cope with this problem. Specifically, the original research deals with bubbles, slugs and churns from multi-phase fluid flow. To implement the computations on these fluid non-rigid objects, I choose a manual method to obtain the appropriate solutions. To achieve efficiency, the manual strategy is implemented in an easy and quick method. The motion analysis scheme on multiphase fluid flow has performed computations on fluid objects and provided data for the multiphase fluid research.

How to Implement Motion Segmentation on a Video Sequence

Motion analysis on a multiphase fluid is a special case of the motion analysis of non-rigid objects. Other research is carried out on standard test video sequences. What we normally deal with in a video sequence are rigid objects or non-rigid objects with gradual and continuous changes (if any). Conventional techniques can be used for this case. The results from the motion estimation explain the data of video images with a certain degree of confidence. This kind of motion estimation allows for subsequent motion segmentation.

Motion segmentation separates the image data into distinct objects according to the relation between different kinds of motion. So far, no optimized motion estimation can recover the motion perfectly. Not all the estimated motions are completely correct, which poses a question: **how can I set up a correct and reasonable segmentation?** There are two points arising from this question:

1. Motion segmentation should correctly segment the data into distinct objects based on the estimated motion;

2. Motion segmentation should recognize and correct the errors resulting from the motion estimation.

The first point requires an appropriate method to implement correct segmentation. The second point requires the motion segmentation to be robust. Errors happen at the motion boundaries and noise points. In the scheme developed here, these points are segmented out to prevent them from contaminating the regular data. The scheme uses a limited number of landmarks to represent image features and performs segmentation between different kinds of motion. This kind of correctness is partly judged by human control, which is employed in the manual segmentation.

Manual segmentation serves two purposes: to implement motion segmentation for non-rigid objects and to behave as a benchmark to judge the results from the automatic motion segmentation. My primary work focuses on the automatic motion segmentation, which has broader applications.

The scheme of motion estimation and motion segmentation is outlined in figure 1.1.

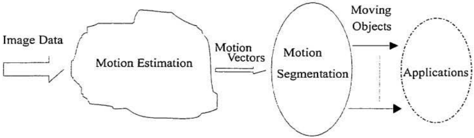


Figure 1.1 The scheme diagram of motion estimation and motion segmentation

1.3 Thesis Overview

The remainder of this thesis is organized as follows:

Chapter 2 reviews conventional techniques for motion estimation and introduces their constraints. This chapter also reviews classification methods in pattern recognition and selects the method for motion segmentation. The strategy for motion estimation and motion segmentation is presented.

Chapter 3 describes the framework of the motion estimation and motion segmentation. I develop a motion analysis platform, which is presented in this chapter. This chapter explains the primary theory in motion analysis and outlines the main stages.

Chapter 4 describes in detail the approaches for identifying landmarks automatically. Four primary methods of identifying landmarks are compared and the most efficient method is selected.

In **Chapter 5,** experimental results on finding landmarks are presented and explained.

Chapter 6 describes in detail the clustering of landmarks. Clustering is performed on the manual method and the automatic method. For each primary method of identifying landmarks, motion estimation is employed to produce the motion vectors; then clustering is performed. The clusters resulting from the four automatic detectors are compared and explained.

In **Chapter 7** video sequences are segmented into distinct moving objects. From the four clustering results from the previous chapter, the most satisfying result is selected. Then, according to this result, the segmentation of the video sequence is implemented. In this chapter, one primary application of the motion estimation and motion segmentation to multiphase fluid flow is also presented.

In **Chapter 8,** the contributions of the thesis are reviewed, and ideas for future work are presented.

CHAPTER 2

MOTION ESTIMATION AND CLASSIFICATION

2.1 Motion Constraints

All methods for motion estimation have to make assumptions about constancy. The assumptions I adopt here are as follows.

Data Conservation

It is assumed that the intensity or gray level of a point in an object remains unchanged between two consecutive video frames. Conventional techniques for motion estimation are based on this primary assumption. This assumption is used more widely than the other two assumptions.

Structure Constraint

This assumption is that the intensity distribution within a small enough patch of the scene changes only gradually.

Temporal Constraint

It is assumed that within a short enough duration, the acceleration of scenes remains constant. This assumption follows fundamental kinetic theory.

These three constraints themselves are assumptions, but they can be violated at the motion boundaries and structural boundaries. Where violation occurs, the results from motion estimation are not reliable. In my research, the violated part is segmented out as outliers from the normal data. Outliers represent noise and all kinds of boundary points.

2.2 Motion Estimation

Conventional techniques [4], [18] for motion estimation fall into four types: one is the gradient-based approach, which has been exploited by many researchers. A second approach is correlation. Feature extraction and matching is another avenue to recover the flow field. The last approach uses spatio-temporal filtering to measure motion vectors [11].

The gradient-based approach is more widely used in research than other three, and the correlation approach is more widely used in applications [4], [18], [11], [25]. I therefore concentrate on these two alternatives.

Gradient-based Approach

Gradient-based optical flow approaches introduced by Horn and Schunck [4], [18] assume the data conservation that the gray level of a pixel remains constant over a video sequence, as described in equation 2.1,

$$\begin{aligned} I(x,y,t) &= I(x+\delta x, y+\delta y, t+\delta t), \\ &= I(x+u\delta t, y+v\delta t, t+\delta t) \end{aligned} \qquad (2.1)$$

where $I(x,y,t)$ represents the intensity at a pixel (x, y) at time t; $[u\ v]^T$ is the horizontal and vertical velocities; δt is small. This equation states that the intensity value at a pixel (x, y) remains unchanged in a later image at a location offset by the optical flow.

Taking the first order Taylor series expansion of the equation 2.1 yields

$$I(x,y,t) = I(x,y,t) + I_x u\delta t + I_y v\delta t + I_t \delta t + \varepsilon \qquad (2.2)$$

where I_x, I_y, and I_t are the first derivatives of the intensity function $I(\ )$ with respect to x, y, and t, and ε embodies the higher-order terms. Dividing by δt gives

$$I_x u + I_y v + I_t = \nabla I \cdot [u \quad v]^T + I_t = 0 \tag{2.3}$$

The error function is thus formed as equation 2.4

$$E_d = (I_x \cdot u + I_y \cdot v + I_t)^2 \tag{2.4}$$

Minimizing the error function yields the motion vector [4],

$$\begin{pmatrix} u \\ v \end{pmatrix} = -\begin{pmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum I_x I_t \\ \sum I_y I_t \end{pmatrix} \tag{2.5}$$

This gradient-based optical flow approach tends to give a good estimate of the true motion of images but there are problems with this approach. First, since a smoothness assumption is implied in optical flow, the approach cannot estimate sudden motion, occlusion or dis-occlusion for example. Second, since the problem is under-constrained, there is no guarantee of a unique solution. Third, this type of optical flow calculation cannot estimate large motions.

Model-based Motion Estimation

Further development of the gradient approach leads to model-based motion estimation. Allowing for the affine model,

$$\begin{aligned} u &= a_0 + a_1(x - x_c) + a_2(y - y_c) \\ v &= a_3 + a_4(x - x_c) + a_5(y - y_c) \end{aligned} \tag{2.6}$$

The motion model is described by six parameters $a_0, a_1, a_2, a_3, a_4, a_5$, where (x_c, y_c) represents the coordinate of some region center. The parameters are derived by using a LSE method to solve equations (2.3) and (2.6).

An affine motion model is more general than a translational model, and therefore is much closer to real motion. This affine model represents rotation, zooming, shears, and all kinds of linear combinations of these movements.

Another more accurate model is the eight-parameter projective model, which reflects perspective motion. These motion models can be further implemented in a multi-resolution scheme for the purpose of optimization.

Block Matching

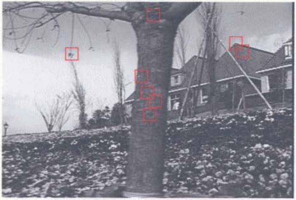
The correlation method is normally implemented as block matching. This method searches for the correlation of a block between two frames. The theory of block matching is in four steps: firstly, two frames are selected, which are close enough in time; secondly, both frames are divided into uniform blocks; thirdly, for each block within a frame (the reference frame), a matching block is searched for in another frame; fourthly, the distance in between the two blocks is denoted as the motion vector. The correlation equation is formulated as,

$$E_d(x,y) = Min\left(\sum_{i=1}^{b}\sum_{j=1}^{b}(I_t(x+i,y+j) - I_{t+1}(x_s+i,y_s+j))^2 \right) \qquad (2.7)$$

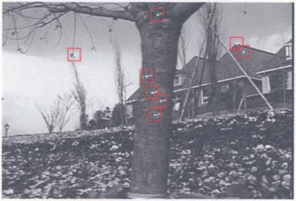
Where $I_t()$ represents intensity function; (x, y) represents the center coordinate of the block to be matched; (x_s, y_s) represents the coordinate of a point within the searching area. The searching area is represented by s; b is the block size. In the scheme, the size of s is chosen as 16 and b is 5.

Block matching gives straightforward computations on the motion at each pixel, and the motion of a pixel will be obtained immediately. But there are two limitations. Only translation is assumed at each pixel in block matching, which is not true for real motion. Another disadvantage is that block matching does not assume any coherence between blocks, which results in some blocks belonging to the same object having inconsistent motion.

An example of motion estimation using block matching is shown in figure 2.1. For each block in figure 2.1(a), block matching searches for the block in figure 2.1(b) that meets equation (2.7). The needles in the second frame represent the motion vectors of the blocks.



(a)



(b)

Figure 2.1 An example of motion estimation using block matching
(a) frame n (b) frame n+1

For these two frames in figure 2.1, the motion can be observed on the tree, which is the moving object.

My Solution for Motion Estimation

In this thesis, block matching is used for motion estimation. The reason comes from the emphasis on landmarks. First, one of my automatic schemes for landmark identification uses a block match search in the current frame. It is natural to use the same strategy in motion estimation. This novel design results in accurate motion estimation of landmarks. Second, since motion computation is performed only on separate landmarks, the limitations on block matching do not impair the scheme. Third, optical flow is not the best solution because of its smoothness constraint, which assumes computation over the whole frame, whereas here motion computation is performed only on a limited number of landmarks.

2.3 Classification

Having estimated the motion of landmarks, my scheme then classifies them on the basis of similarity of motion. I therefore now survey the techniques of supervised and unsupervised pattern classification. This section serves as an overview of principal approaches. For further reading, one is encouraged to refer to relevant textbooks and references [10], [27].

2.3.1 Supervised Methods

During supervised classification, some samples are labeled in advance and used to set up models. These derived models are applied to more samples for the purpose of classification. Supervised methods are classified as parametric or non-parametric. Among typical methods for parameter estimation, statistical parameter estimators are the most efficient. Two primary statistical methods are used to estimate parameters: the maximum likelihood (ML) estimator and the maximum *a posteriori* (MAP) estimator [10], [27]. The maximum likelihood estimator $L(\theta)$ is expressed in equation (2.8).

$$L(\theta) = p_{x_1, x_2, \ldots, x_n; \theta}(x_1, x_2, \cdots, x_n; \theta) = \prod_{i=1}^{n} p_{x_i;\theta}(x_i; \theta) \tag{2.8}$$

θ represents the model parameter to estimate and is derived by maximizing $L(\theta)$.

The maximum *a posteriori* (MAP) estimator is

$$p_{\theta|x_1,x_2,\cdots,x_n}(\theta \mid x_1,x_2,\cdots,x_n)=\frac{p_{x_1,x_2,\cdots,x_n|\theta}(x_1,x_2,\cdots,x_n \mid \theta)\cdot p_\theta(\theta)}{p_{x_1,x_2,\cdots,x_n}(x_1,x_2,\cdots,x_n)} \tag{2.9}$$

$$M(\theta)=p_\theta(\theta)\cdot\prod_{i=1}^{n} p_{x_i|\theta}(x_i \mid \theta) \tag{2.10}$$

The model parameter θ is derived through maximizing equation (2.9), which is the same as maximizing equation (2.10).

Non-parametric methods merge samples according to the distances in between them, and include the nearest-neighbor, Parzen-window, and the linear Fisher discriminant methods [10], [27].

2.3.2 Unsupervised Methods

Unsupervised classification does not use the labeled training samples. Unsupervised methods include parametric and non-parametric methods. Since sample data are normally parametrically identifiable, the maximum likelihood estimator of equation (2.8) and the Bayesian estimator are employed to estimate the parameters of models. The main non-parameter methods are K-means and hierarchical clustering [10], [27].

2.3.3 My Solution for Classification

In my scheme, hierarchical clustering is chosen to merge data. The landmarks are locations with associated motion vectors, but they are not pre-labeled and no other prior information about motion is available. Furthermore the motion cannot be described parametrically. I therefore choose hierarchical clustering for classification, which, as such, is an unsupervised non-parametric method. Another typical non-parametric method, K-means, requires the prior estimation of K, the number of classes, which in my work is usually unknown. The basic algorithm of hierarchical clustering is as follows,

Initially, the number of clusters to be merged is set equal to the number of data to assign; the minimum distance between clusters is found, then those clusters with the minimum distance are merged into one new cluster; the number of the merged clusters is subtracted from the number of clusters to obtain the current number of clusters. The procedure is repeated until some stopping criterion (e.g. number of cluster or minimum distance) is true.

Figure 2.2 illustrates the process.

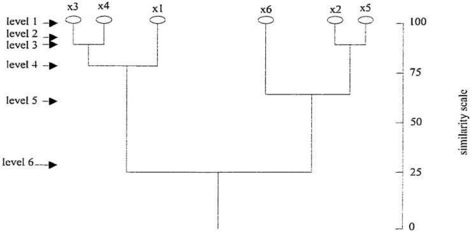


Figure 2.2 Example dendrogram of hierarchical clustering

2.4 Motion Segmentation

Motion segmentation is still in the research stage and has not become standard. There are two primary fashions to motion segmentation. The first fashion is to describe multiple models simultaneously. Wang and Adelson [1] addressed the problem by computing 2D affine motion models. Objects are then represented through different layers. Due to the smoothness assumption in optical flow and the limitation of the specified clustering, the resultant layers cannot describe some types of image data well. Darrell and Pentland [9] used a robust M-estimation and MDL framework to estimate parameters for multiple motions. Black and Yacoob [28] applied robust optical flow and the EM algorithm to their multiple motion estimation.

The second fashion is to segment out multiple motions by solving a dominant model at each stage. Irani and Hsu [16] addressed the problem by computing moving objects over image sequences. Black and Anandan [6] incorporated robust M-estimators in solving the dominant motion estimation.

I present feature-based motion estimation and motion segmentation. I would like to make a contribution in this field. Both styles of motion segmentation have been interlaced in my scheme. The scheme not only segments moving objects simultaneously but also sets up objects one by one.

In the scheme developed here, two approaches are used for segmentation: triangle grouping and nearest neighbor (NN). Since landmarks represent image features and are distributed across the image, they are utilized to make a triangular segmentation of the image. The triangles are then grouped according to the clusters of landmarks to obtain objects. The advantage of this is that the scheme does not need to perform the computation of segmentation on each point. One immediate benefit is the approach tolerates noise and odd points. The reason is that the points within a triangle are plainly rendered and do not distract the segmentation. For the manual process, triangle grouping is used for segmentation because the manual landmarks are easier to control.

Regions from grouped triangles contain moving objects. On the other hand, those segmented regions often include parts of foreign regions. For objects to emerge from the foreign regions, the nearest neighbor (NN) method is employed. I use NN in this way: for each point within an object region, a certain number of nearby landmarks are located; the point is assigned to the cluster in which most of the landmarks are located. NN is mainly used in the automatic process for trimming.

CHAPTER 3

FRAMEWORK

3.1 Methodology: Feature-based Motion Estimation and Motion Segmentation

The entire scheme for motion estimation and motion segmentation is based on landmarks. These landmarks are not usual feature points, but those points which are suitable for matching. A limited number of landmarks are selected to represent image features. It is these landmark motion that are calculated and analyzed. Different methods for finding landmark produce distinct implementations, which serve distinctive purposes. My work studied three methods to produce landmarks: placing landmarks manually; generating landmarks from tiling the picture; and identifying landmarks automatically.

I investigated four methods for automatic landmark identification: the Moravec method [25], the SUSAN method [24], the sub-block matching method, and the variance-based approach (VBA). Moravec is a conventional corner detector; SUSAN is very recent, and is also used for detecting corners. Both these methods assume that corners make good landmarks. One goal of my scheme is to seek for a kind of landmarks that are the best for matching. For this purpose, I investigated two new methods for identifying landmarks: sub-block matching and VBA. All the four methods are efficient for identifying feature points. In particular, sub-block matching and VBA are more efficient for the matching purpose than Moravec and SUSAN. The resultant landmarks will undergo motion estimation and clustering. Motion segmentation utilizes the resulting clusters to produce moving objects. From the segmentations of these four automatic methods, the overall performance can be compared.

Block matching is employed for motion estimation. Hierarchical clustering is used for classifying landmarks. The entire scheme is divided into three processes: manual process, tiling process, and automatic process. During an implementation, at a particular time, only one of these three processes runs.

Image warping and the nearest-neighbor (NN) method are used for motion segmentation. Warping is a common technique in the area of image processing, and for further reading, one is encouraged to refer to [25]. Since the scheme has landmarks at hand, it is wise and efficient to utilize image warping directly to obtain the moving regions. However, in the automatic process, image warping introduces foreign regions for each moving object. NN is, therefore, used to eliminate those invasions and obtain exact moving objects.

3.2 The Platform of Motion Estimation and Motion Segmentation

To support experiments in my research and also to make it convenient for others to implement the entire procedure, a platform has been developed for motion estimation and motion segmentation. This platform implements many functions of motion analysis and image processing, including motion estimation and motion segmentation. Distinctive generators of landmarks and the technique of image warping are included in the platform. The platform of motion estimation and motion segmentation is shown in figure 3.1.

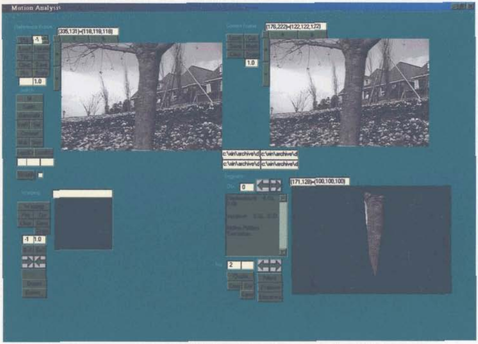


Figure 3.1 The platform of motion estimation and motion segmentation

This platform consists of four primary sections: basic utility functions; landmark identification; image warping; and sequence segmentation. The primary functions of these four sections are described as follows:

Basic Utility Functions (Top of screen)

This section implements basic image processing functions, image loading, image saving, image scaling and image clearing. It also provides functions for tiling images, landmark locating, motion compensation and motion estimation of block matching, and the triangulation of consecutive frames.

Landmark Identification (Panel on left of screen)

This section involves distinctive landmark generators, the Moravec corner detector, the sub-block matching approach and VBA. This section also includes landmark loading, image picture loading, variance calculation, and landmark selecting.

Image Warping (Bottom left of screen)

The primary function of this section is image triangulation from landmarks and warping between consecutive frames. This section also provides for image saving.

Video Sequence Segmentation (Bottom right of screen)

The primary functions of this section embrace the clustering of landmarks, the display of the clustering state, the display of each resultant cluster, the results from motion segmentation, and the display of moving objects. In addition, this section can trigger the display of a video sequence, and the display of the clustering evaluation figure.

3.3 Environments for Programming

The user interface and all the algorithms of image and video processing are done by using the software of Powersoft Power++ developed by Sybase™ company, and MCLGallery, which is built by the PhD student Li-Te Cheng. Powersoft Power++ is a kind of RAD C++ tool for building robust client/server, distributed, and Internet applications. MCLGallery modularizes many basic functions of Power++ into a series of image-oriented components and modules, and makes Powersoft Power++ more convenient for processing image and video. Except for the mclgallery.dll file necessary for the executable file, the release version of the executable file of the entire motion analysis program is 1.4MB in size.

3.4 Manual Process

One of the purposes of the manual process is to obtain the motion segmentation for non-rigid objects. Another purpose is to set up the standard of segmentation for evaluating automatic segmentation. Landmarks are placed manually on the selected frames. Objects are either assumed during the manual process or result from motion segmentation. Once unlabeled landmarks are produced, they will undergo motion estimation and clustering. Further motion segmentation yields moving objects.

3.5 Tiling Process

Tiling process is used to obtain a regular sampling. For two consecutive frames, the scheme tiles one frame, and then performs motion estimation to obtain corresponding landmarks on another frame. These landmarks produced through the tiling process will undergo motion estimation and clustering. Moving objects are obtained from motion segmentation.

3.6 Automatic Process

The more advanced goal of the motion segmentation is to obtain automatic motion segmentation. In the automatic process, landmarks are selected either with the maximum intensity variance within the neighborhood or with the maximum dissimilarity within the neighborhood. Those landmarks produced through the automatic process will undergo motion estimation and clustering. Moving objects are yielded through warping frames according to the landmarks.

The overall framework of motion analysis is outlined in figure 3.2.

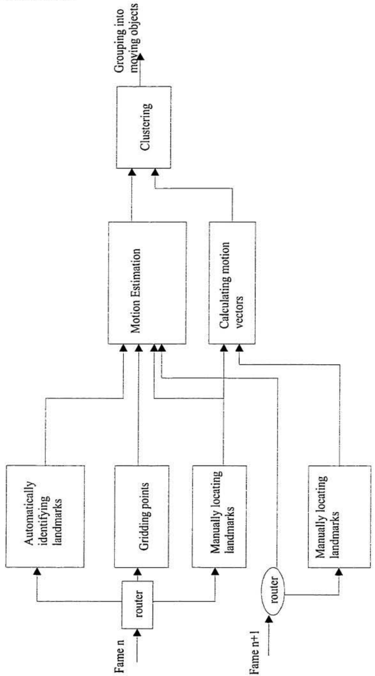


Figure 3.2 The framework of motion estimation and motion segmentation

CHAPTER 4

METHODOLOGY OF FINDING LANDMARKS

The scheme begins with finding landmarks. Different finding landmark methods produce distinct implementations, which serve distinctive purposes. Landmarks are produced in three processes: the manual process, the tiling process, and the automatic process.

4.1 Manually Placing Landmarks

There are two purposes for manual process. One purpose is to deal with non-rigid objects. Since it is very difficult to track non-rigid objects automatically, landmarks are placed manually. Another purpose for the manual process is to serve as a standard for segmentation. An example of manually placing landmarks is shown in figure 4.1. In this example, the pictures are two consecutive frames of the Flower Garden sequence. Different color points represent distinct clusters. There are two clusters on the pictures, which are represented by red and yellow.

The block matching approach can be used to enhance the accuracy of manually placed landmarks by a small shift in position in one or other frame.

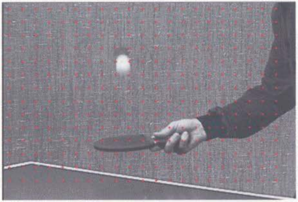
(a)

(b)

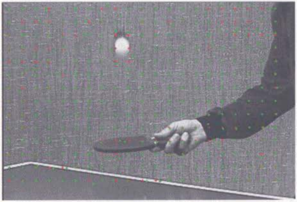
Figure 4.1 An example of placing landmarks manually
(a) frame n (b) frame n+1

4.2 Landmarks from Tiling Picture

A regular pattern is formed through tiling a frame with samples at an even spacing. Landmarks are produced through the tiled frame. An example of tiling pictures is shown in figure 4.2, where the pictures are two consecutive frames from the Table Tennis sequence. Red points are from tiling the picture; green points are the corresponding points resulting from motion estimation.



(a)



(b)

Figure 4.2 An example of tiling frames

(a) frame n (b) frame n+1

4.3 Automatically Identifying Landmarks

The current goal is to identify landmarks automatically. These landmarks capture image features. In the scheme, four primary methods are used to identify landmarks: the Moravec method [25], the SUSAN method [24], the sub-block matching method, and the VBA method. For further reading, one is encouraged to refer to [25], [24].

The Moravec method is a conventional corner detector. The SUSAN method is very recent and also used for detecting corners. Moravec selects corners as those points with the biggest local variance. This approach detects corners efficiently for many pictures. SUSAN generates corners based on the proportion of neighbor points close to the center landmark. For these two methods, corners are used as landmarks. To better serve the matching purpose, I have developed two new methods: sub-block matching and VBA.

Sub-block Matching

The idea of sub-block matching is to find a point that does not match its neighbors very well. At each point, sub-block matching finds the position with the second best matching value within a certain neighborhood of the point. The best match is at the point itself. Then it selects landmarks as those points with the biggest second best matching values. The correlation at pixel (x, y) is formulated in equation (4.1)

$$E_d(x, y) = Min\left(\sum_{i=1}^{b} \sum_{j=1}^{b} (I(x+i, y+j) - I(x_s+i, y_s+j))^2 \right) \qquad (4.1)$$

where $I()$ is the image intensity; (x, y) represents the center coordinate of the block to be matched; (x_s, y_s) represents the coordinate of a point within the searching area; s represents the search area; b represents the block size. In my scheme, the size of s is chosen as 16 and b is 5. The algorithm of sub-block matching is as follows,

For each pixel, a block is fixed around it (5×5 is selected in the experiment); within a certain search area around the pixel, the second best match block to the original is searched for. Since the best matching is the original itself, it is ignored. Then record on the position of the pixel the matching difference with the second best matching pixel. The second best match picture of the *original image is thus generated. Landmarks are generated through taking the largest-valued points from the second best match picture.*

Variance-based Approach (VBA)

VBA combines two methods for identifying landmarks: the round-neighbor approach and the 12-variance approach.

The **four-variance approach** is that for each point, select the least variance among its four orientations (as in figure 4.3) as the variance; then select feature points as those with the variance greater than some threshold.

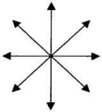


Figure 4.3 A demonstration of four orientations around a point.

The algorithm of the round-neighbor approach includes two principal steps, as follows,

Step 1: *The four-variance approach above is employed to obtain the original landmarks candidates; then the below step is followed to obtain the final landmarks.*

Mini-variance number (MVN) refers to the number of variances (that are greater than a threshold) within a neighborhood.

Step 2: *for each landmark candidate* c_i

if there exist other candidates (a_0,---, a_m) *within its r-pixel radius*

for each pair c_i *and* a_j ($m \geq j \geq 0$)

if **MVN** ***of*** c_i ***is greater than that of*** a_j

remove a_j

else

if c_i*'s MVN less than* a_j*'s*

remove c_i

else

if c_i*'s orientation equals* a_j*'s*

compare the responses p_c *and* p_a *resulting from equation (6)(below)*

if p_c *equals* p_a

remove both c_i *and* a_j

else

if p_c *less than* p_a

remove a_j

else

remove c_i

$$neigh(k) = \begin{cases} 1, & \text{if } \|I(c_i) - I(k)\| \leq threshold \\ 0, & otherwise \end{cases} \quad (5)$$

$$p_c = \sum_k neigh(k) \quad (6)$$

where $I()$ represents the image intensity; k, neighbor point within the d-pixel radius around candidate c_i.

The resultant points are denoted as landmarks. The round-neighbor approach is able to reject those points with the similar pattern to a remarkable degree, and therefore is suitable for identifying landmarks. The 12-variance approach, on the other hand, is able to prevent edge points from being selected as landmarks. Instead of the conventional four variances (figure 4.3), this approach calculates twelve variances around a point. Two from twelve variance patterns are shown in figure 4.3. The 12-variance approach takes the variance of three pixels along an arrow, and then selects the least variance as the current variance. Since these twelve variance patterns emulate edge situations efficiently, this approach eliminates most edge points among the candidate points and captures feature points accurately.

Figure 4.4 Two of twelve variance patterns

VBA combines round-neighbor and 12-variance by applying weights to these two approaches. Weights can be adjusted from two ends: a small value and a large value; then move from two ends to some value in between. If the small value is less than some threshold (1 is assumed in my adjustment of weights), only round-neighbor is utilized. If the big value is greater than 2000(which is also assumed in the adjustment), only 12-variance is utilized; if the value is in between, both methods play a role and their weights explain how important they are. Through adjusting weights, VBA can be tailored to various types of pictures. Moreover, for the same picture, by adjusting weights, multiple outputs with different numbers and positions of landmarks can be obtained. So far, the automatic setting of the weights on a per-picture basis remains an open question.

CHAPTER 5

EXPERIMENTS ON IDENTIFYING LANDMARKS AUTOMATICALLY

This chapter will show the experimental results from four main methods for identifying landmarks automatically: the Moravec method, the SUSAN method, the sub-block matching method, and VBA. The goal is to search for the best matching effect among the four methods.

The tested pictures are selected from distinct but standard video sequences: Flower Garden, Calendar, and Table Tennis.

5.1 Flower Garden

Figure 5.1 shows the experimental results from four methods: Moravec, SUSAN, VBA, and sub-block matching. The number of landmarks is partly determined by the content of picture. For the Flower Garden sequence, 200 landmarks are selected, because this is around the minimum number of landmarks that can cover distinct regions of the picture appropriately. Except for the SUSAN method, the landmarks resulting from the other three methods are distributed fairly evenly and at the same time occupy the feature positions of images.

(a)

(b)

(c)

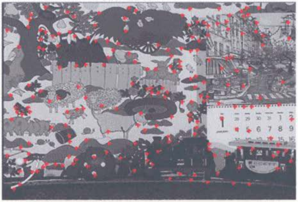
(d)

Figure 5.1 Landmarks of Flower Garden from the four methods

a. landmarks from VBA b. Sub-block matching
c. Moravec method d. SUSAN method

5.2 Calendar

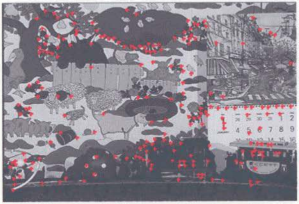
Figure 5.2 shows the experimental results from the four methods: VBA, sub-block matching, Moravec, and SUSAN on the Calendar sequence. For each method, 200 landmarks have been selected, which is again around the minimum number of landmarks to cover the distinct regions appropriately. On the calendar, some points belong to a repeating pattern. These points are not good landmarks because they match other nearby locations. It can be observed that only sub-block matching and VBA reject points within repeating patterns, but Moravec and SUSAN cannot.



(a)

(b)

(c)



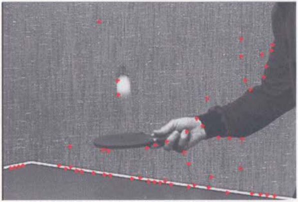
(d)

Figure 5.2 Landmarks of Calendar from the four methods

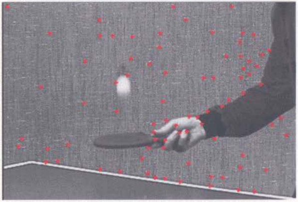
a. landmarks from VBA; b. sub-block matching
c. the Moravec approach; d. The SUSAN approach

5.3 Table Tennis

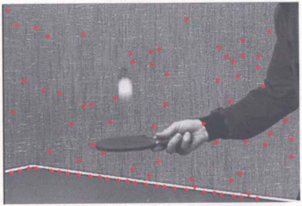
Figure 5.3 shows the experimental results from the four methods, VBA, sub-block matching, Moravec, and SUSAN on the Table Tennis sequence. VBA selects about 60 landmarks, and other three select 80 landmarks. For each method, the number of landmarks is around the optimal number for identifying important regions. It is observed that only sub-block matching rejects all boundary points with a repeating pattern. Sub-block matching searches for those points that do not have self-similarity.



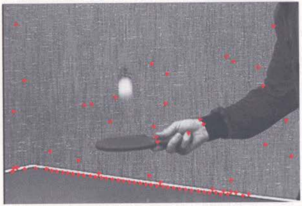
(a)



(b)



(c)



(d)

Figure 5.3 Landmarks of Table Tennis from the four methods

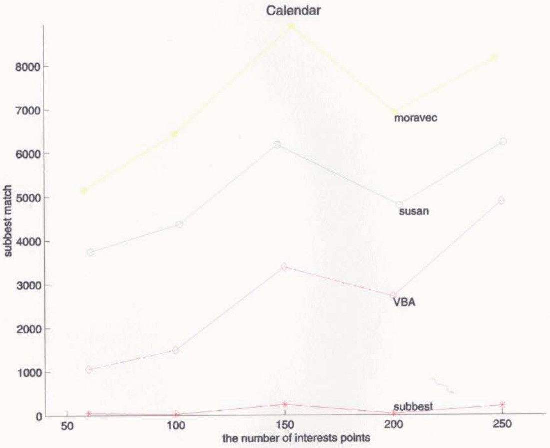
a. landmarks from VBA
b. Sub-block matching
c. the Moravec method
d. SUSAN method

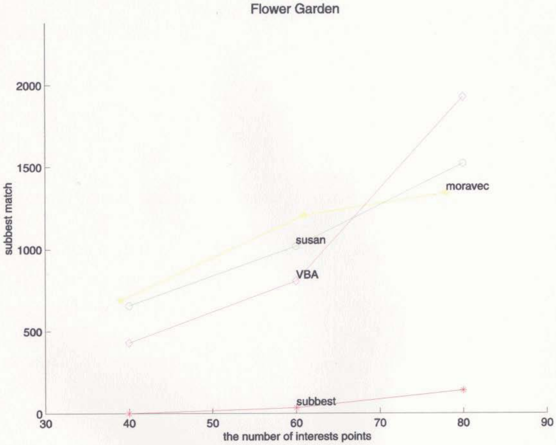
5.4 Comparison to Sub-block Matching

The algorithm of sub-block matching has been recounted in chapter 4. Block matching is a conventional correlation method. The best protection against a block matching a wrong block is to find an area of picture that does not have self-similarity, i.e., that does not repeat itself. Matching a block against its neighborhood is a way of doing this. Sub-block matching searches for a pixel that does not match its neighbors very well and generates the sub-block matching picture which results from recording on the position of the searched pixel the matching difference with the second best matching pixel. The second best match value can be calculated from equation 5.1 at any point. Because of the outstanding performance of the second best matching, it is used as a benchmark to compare with the other methods. At each point where a method places a landmark, the second-best-match value is calculated. These are added together to get a total sub-best match value for the particular set of landmarks. The comparison results are shown in figure 5.4. The lower the curve is, the better the matching effect.

Sub-block matching evolves from the second best matching approach. However, sub-block matching samples at even spacing; thus, it does not find all those points with the biggest second best matching value and therefore works out differently from the second best matching. In figure 5.4, sub-block matching is at the bottom, representing the best matching effect; but it is still over the axis. VBA is next to sub-block matching; Moravec and SUSAN are above them.

For Calendar, the numbers of landmarks for tested sets are 60, 100, 150, 200, and 250. For Flower Garden, the numbers of landmarks for tested sets are 40, 60, and 80. For Table Tennis, the numbers of landmarks for tested sets are 30, 40, 50, and 55.


Calendar
8000
7000
6000
5000
4000
3000
2000
1000
0
subbest match
moravec
susan
VBA
subbest
50
100
150
200
250
the number of interests points


Flower Garden
subbest match
2000
1500
1000
500
0
30
40
50
60
70
80
90
the number of interests points
moravec
susan
VBA
subbest

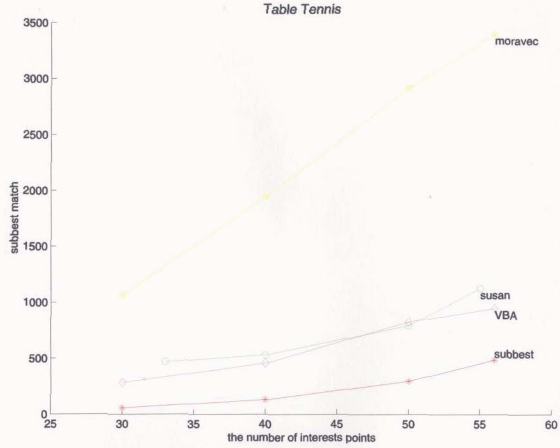


Figure 5.4 *Comparison using sub-block matching as the benchmark*

The labels on the graph represent the sum of corresponding sub-block matching values of all chosen landmarks (that is, interests points in figure) for each method. Therefore, each curve is supposed to rise monotonically; however, to highlight the difference between the four methods and avoid big values, a downward scaling is applied at 200 and 250 of landmarks for Calendar, which explains the fall-down of the curve at 200.

Since sub-block matching has the best matching effect, it falls below the other three curves as expected. VBA is above the sub-block matching curve, but falls below the Moravec curve and the SUSAN curve. This indicates that the matching effect of VBA is better than Moravec and SUSAN, which agrees with the visual judgements of the landmarks.

CHAPTER 6

CLUSTERING OF MOTION VECTORS

Clustering has been performed on three standard video sequences: Flower Garden, Table Tennis, and Salesman. Hierarchical clustering is used in classification. Please refer to Chapter 2 where the algorithm of hierarchical clustering was recounted. For each sequence, clustering is applied to the motion of landmarks obtained from the manual process, the tiling process, and the automatic process. In each case, the actual motion of the landmark is estimated by block matching. Since SUSAN cannot separate landmarks with enough space, it is precluded from the automatic process.

6.1 Robustness in Clustering

Since not all motion vectors are correct due to the errors from the motion estimation, robustness is needed against erroneous vectors. Outliers are defined for clustering as those points with erroneous motion vectors. They should be separated to prevent them from contaminating the clustering of landmarks. Outliers usually occur at motion boundaries and structural boundaries where it is difficult to recover motion vectors correctly. Outliers thus reflect the conditions of motion boundaries and structural boundaries and further help to improve the segmentation. This character of outliers will be utilized in later motion segmentation.

The theory of detecting outliers is developed as follows, Since outliers are few in number, the ratio between them and the total points is used to circumvent outliers. The ratio is controlled through a pre-assumed threshold (as in equation 6.1). In my scheme, outliers are implemented as those points that meet the equation 6.1. The identification of outliers is inserted in the implementation of hierarchical clustering. This implementation tries to capture those erroneous or non-classifiable points. Experiments will show this capture of outliers is very efficient for many types of pictures.

$$\frac{N_{outlier}}{N_{total}} \leq R_{thresh} \qquad (6.1)$$

Where $N_{outlier}$ and N_{total} stand for the numbers of outliers and the frame size; R_{thresh} stands for the ratio threshold.

In the experiments of clustering, different colors represent distinct classes, and the black cross is reserved for outliers.

6.2 Selection of Landmarks

A limited number of landmarks are selected for each sequence. For the automatic process, 200 landmarks are selected for Flower Garden, 180 landmarks for Salesman, and 180 for Table Tennis. The ratio between landmarks and number of pals in the frame is described in table 6.1. The number of landmarks is determined according to the image size and the content of the image picture.

	The number of landmarks	The size of Frame	Ratio of landmark to the frame
Flower Garden	200	84480	0.002367424
Salesman	180	103680	0.001736111
Table Tennis	180	84480	0.002130682

Table 6.1 The ratio between landmarks and frame size

6.3 Clustering of Flower Garden Sequence

Manual Process

The clustering has produced two classes, as shown in figure 6.1. The landmarks on the tree have been grouped into one class, and those on the house and the meadow into another class as the background. One point with erroneous motion vector has been separated as an outlier.

(a)

(b)

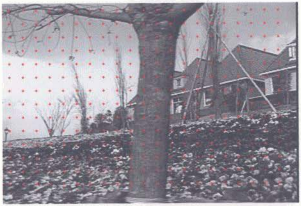
(c)

Figure 6.1 The clustering of manual landmarks for Flower Garden

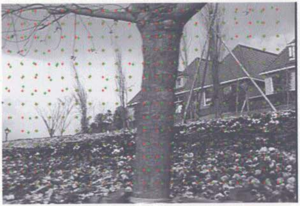
(a) the current frame (b) the reference frame showing pairs of corresponding landmarks
(c) clustering on the landmarks

Tiling Process

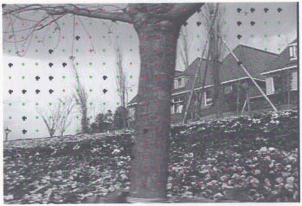
Figure 6.2 shows the two classes obtained from the clustering of tiled landmarks. Those points at motion boundaries or with erroneous motion vectors have been separated as outliers. Although some points have not been clustered correctly, the overall clusters describe the moving objects correctly. In figure 6.2 (b), pairs of corresponding landmarks are represented by two colors, red and green; and the difference in between a pair denotes the landmark's motion.



(a)



(b)



(c)

Figure 6.2 The clustering of tiling landmarks for Flower Garden

(a) the current frame (b) the reference frame with pairs of corresponding landmarks
(c) the clustering

Automatic Process

For each automatic method of identifying landmarks (Moravec, VBA, and sub-block matching), the clustering has produced two classes. These are shown in figures 6.3, 6.4, and 6.5.

Moravec

Two clusters result; one is on the tree, representing the dominant motion; another is on the house and meadow. Some pixels, whose motion cannot be recovered correctly, have been categorized as outliers. Although some pixels on the end of the tree branches are clustered away from the tree, the overall clusters can be observed to describe the moving objects correctly. In figure 6.3 (b), pairs of corresponding landmarks are represented by two colors, red and green; and the difference between a pair denotes the landmark motion.

(a)

(b)

(c)

Figure 6.3 The clustering of Moravec landmarks for Flower Garden
(a) the current frame with the landmarks chosen (b) the reference frame with pairs of corresponding landmarks (c) the clustering

Sub-block Matching

The points on the tree have been clustered into one moving object, representing the dominant motion; the points on the house and the meadow have been clustered into another object. All the points with erroneous motion vectors have been separated out as outliers. Although some points are incorrectly classified, the overall clusters can be observed to describe the moving objects correctly. In figure 6.4 (b), pairs of corresponding landmarks are represented by two colors, red and green; and the difference in between a pair denotes the landmark motion.

(a)

(b)

(c)

Figure 6.4 The clustering of sub-block matching landmarks for Flower Garden

(a) the current frame with the landmarks (b) the reference frame with pairs of corresponding landmarks (c) the clustering

VBA

The overall clusters describe the moving objects correctly. In figure 6.5 (b), pairs of corresponding landmarks are represented by two colors, red and green; and the difference in between a pair denotes the landmark motion.

(a)

(b)

(c)

Figure 6.5 The clustering of VBA landmarks for Flower Garden

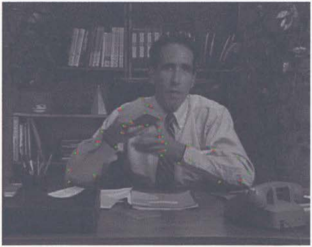
(a) the current frame with the landmarks chosen (b) the reference frame with pairs of corresponding landmarks (c) the clustering

6.4 Clustering of Salesman

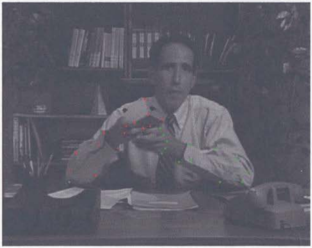
Manual Process

Landmarks on the arms have been clustered into two classes, which are shown in figure 6.6. Two landmarks on the box, which has motions different from both background and the arms, cannot be categorized as any cluster, and are segmented out as outliers.

(a)



(b)



(c)

Figure 6.6 The clustering of manual landmarks for Salesman

(a) the current frame with the landmarks placed manually (b) the reference frame with pairs of corresponding landmarks (c) the clustering

Tiling Process

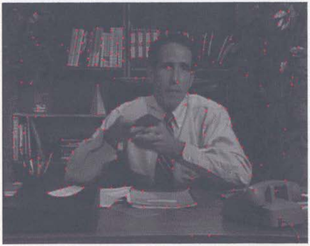
The resulting landmarks from tiling are clustered as outliers that occur at motion boundaries or embody erroneous motion vectors or noise. In tiling, the boundary between noise and moving points is blurred, and it is difficult to tell which is which. Correct clustering does not result from the tiling process.

Automatic Process

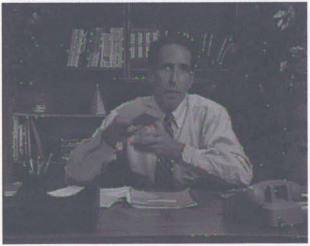
For the three automatic methods of identifying landmarks (Moravec. VBA. and sub-block matching), two classes have been obtained from the clustering. Compared with the other three methods, the landmarks from the SUSAN method cannot be evenly spaced and are not suitable for matching, and therefore the SUSAN performs poorest of automatic processes.

Moravec

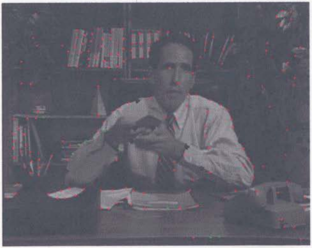
Landmarks have been clustered into two classes. It can be observed that the classification is partly correct. Clustering segments the unchanged scene out as the background, and recognizes the consistent motion of one arm as another cluster. Non-classifiable points are segmented out as outliers, and these represent more complex motions. In figure 6.7 (b), pairs of corresponding landmarks are represented by two colors, red and green; the difference between a pair denotes the landmark motion.



(a)



(b)



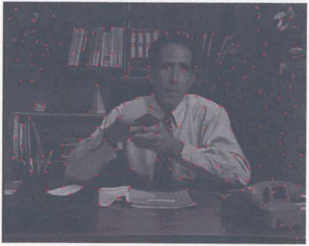
(c)

Figure 6.7 The clustering of Moravec landmarks for Salesman

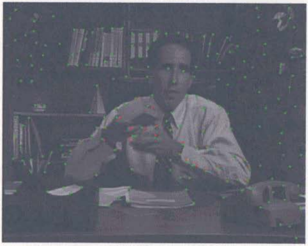
(a) the current frame with landmarks chosen (b) the reference frame with pairs of corresponding landmarks (c) the clustering

Sub-block Matching

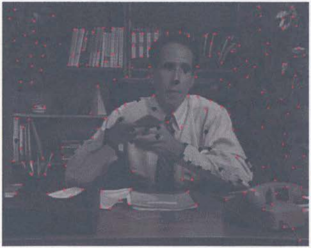
The clustering has generated one class and outliers. Outliers involve all moving points. The class of landmarks comes from the background. In figure 6.8 (b), pairs of corresponding landmarks are represented by two colors, red and green; the difference in between a pair denotes the landmark motion.



(a)



(b)

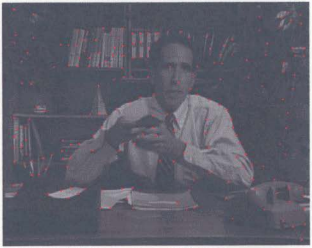


(c)

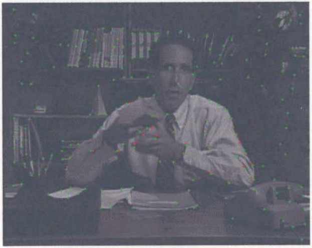
Figure 6.8 The clustering of Sub-block matching landmarks for Salesman
(a) the current frame with landmarks chosen (b) the reference frame with pairs of corresponding landmarks (c) the clustering

VBA

The clustering has generated outliers and two classes. The landmarks on the arms have been uniformly grouped into outliers. In figure 6.9 (b), pairs of corresponding landmarks are represented by two colors, red and green; the difference in between a pair denotes the landmark motion.



(a)



(b)

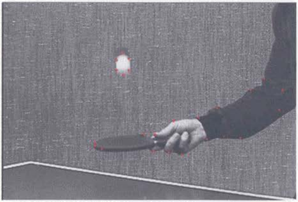
(c)

Figure 6.9 The clustering of VBA landmarks for Salesman
(a) the current frame with landmarks chosen (b) the reference frame with pairs of corresponding landmarks (c) the clustering

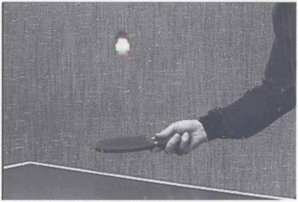
6.5 Clustering of Table Tennis

Manual Process

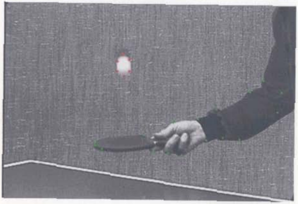
Two classes have been obtained from the clustering of the landmarks, as shown in figure 6.10. The landmarks on the ball have been grouped into one class, and the points on the arm and the bat have been grouped into another class.



(a)



(b)



(c)

Figure 6.10 The clustering of manual landmarks for Table Tennis
(a) the current frame with landmarks placed manually (b) the reference frame with pairs of corresponding landmarks (c) the clustering

Tiling Process

The resulting landmarks from tiling have been grouped into one class and outliers. Outliers embody *noise and scarce points with large motion. As in Salesman, noise and moving points* cannot be distinguished from each other in the tiling case. Correct clustering cannot be obtained from tiling.

Automatic Process

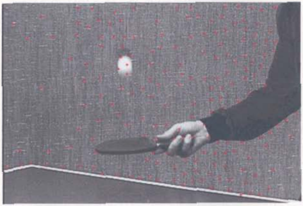
The clustering is performed on the landmarks from three automatic methods of identifying landmarks. The three methods involve Moravec, VBA, and sub-block matching.

Moravec

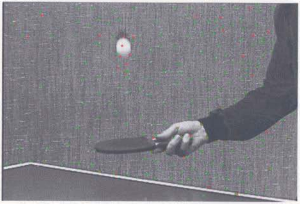
Two classes have been obtained from the clustering. Noise points and the landmarks with motion have been grouped into outliers. Those points on the table edge with same patterns have been erroneously clustered into a cluster because their motion vectors are not correct. As a result, noise and moving points cannot be distinguished from the clustering. The resultant clusters are not correct.

Sub-block Matching

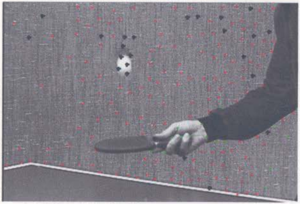
Two classes have been obtained from the clustering. Outliers embody all the points on the ball and some from the background and few from the arm. Actually, outliers form a class and can be further utilized to obtain a moving region. Most of the landmarks on the arm have been clustered into one class. Noise and many landmarks in the background have been grouped into another class. The clustering from sub-block matching can be observed to be correct. In figure 6.11 (b), pairs of corresponding landmarks are represented by two colors, red and green; the difference in between a pair denotes the landmark motion.



(a)



(b)

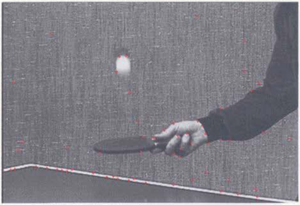


(c)

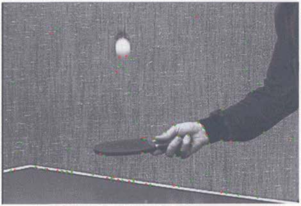
Figure 6.11 The clustering of sub-block matching landmarks for Table Tennis
(a) the current frame with landmarks chosen (b) the reference frame with pairs of corresponding landmarks (c) the clustering

VBA

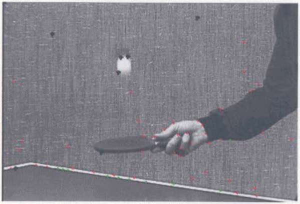
Two classes and outliers have been obtained from the clustering. Noise and the landmarks on the ball have been grouped into outliers. The landmarks on the arm and part of the landmarks on the background have been grouped into one class. Some of the landmarks on the background have been grouped into another class. This classification is not satisfactory; however, it does outline the trend of clustering. Outliers can be further utilized to separate the ball. The result is not as good as sub-block matching, but is better than Moravec. In figure 6.12 (b), pairs of corresponding landmarks are represented by two colors, red and green; the difference in between a pair denotes the landmark motion.



(a)



(b)



(c)

Figure 6.12 The clustering of VBA landmarks for Table Tennis

(a) the current frame with landmarks (b) the reference frame with pairs of corresponding landmarks (c) the clustering

6.6 Discussion

To summarize, among the three processes of manual, tiling, and automatic, it is the manual process that garners the best clustering result because the manually placed landmarks are intended feature points. The tiling process only works for Flower Garden, and fails for Salesman and Table Tennis, which indicates that image-dependent feature points are needed. The automatic process provides for this kind of need. Among the three results from the clustering for the three automatic methods, Moravec works for Flower Garden and Salesman but fails for Table Tennis; whereas VBA and sub-blocking matching work for all three standard sequences. Furthermore, sub-block matching offers better clustering than the other two. The main reason is that the landmarks from sub-block matching are a kind of feature-outstanding points, around which there is no similar point to compete as a match. Moreover, since this approach for identifying landmarks is coupled with motion estimation through the matching strategy, the motion vectors of those points are more reliable than average points. The clustering is then performed on the correct motion vectors and the correct clusters result.

On the other hand, Moravec employs angular information. As this type of identifying landmark method does not have much coherence with the matching strategy, the resultant motion vectors are only partly reliable. The ensuing clustering tries to capture this kind of difference, but it cannot ensure success. To some degree, VBA is in between sub-block matching and Moravec because it is usually successful, but is simple and fast. Whatever method is employed, once the trend of clustering is identified, the clusters must be further refined to set up the accurate segmentation.

Although outliers correspond to erroneous motion vectors, they tend to occur at motion boundaries, and can therefore be used as a class in their own right to identify these boundaries.

CHAPTER 7

EXPERIMENTS ON FINDING MOVING OBJECTS

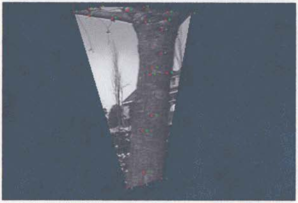
To obtain moving objects from the clustered landmarks, motion segmentation is performed. The **tiling process** is precluded from the segmentation because it cannot provide correct clusters. For the manual process and the automatic process, segmentation experiments have been performed on three standard video sequences: Flower Garden, Salesman and Table Tennis. Please refer to Chapter 2 where the algorithms of motion segmentation were recounted. For the manual process, triangle grouping is used to obtain moving objects. Both triangle grouping and nearest-neighbor are used for the automatic process. Clusters from sub-block matching are utilized for the automatic process.

7.1 Manual Process

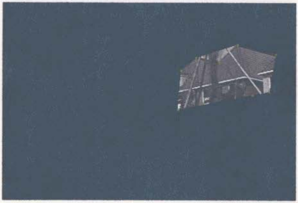
7.1.1 Flower Garden Sequence

The segmentation of Flower Garden is shown in figure 7.1. The landmarks can be observed as clustered correctly because they are chosen from distinct moving regions. According to the clusters, this sequence has been segmented into two moving objects. Figure 7.1 (a) shows the results from the clustering. Since the tree moves relative to the background, it is separated as an independent moving object, shown in figure 7.1 (b), from the background (figure 7.1 (c)). Some background is also combined into the tree object. The reason is that the tree object is obtained by assigning and grouping triangles in the image according to the landmarks. One of the solutions to this problem is to use motion plus other attributes to label each pixel within the segmentation.

(a)



(b)



(c)

Figure 7.1 The manual motion segmentation of Flower Garden

(a) the clustering of the landmarks (b) the first moving object (c) the second moving object

7.1.2 Salesman Sequence

Since landmarks are chosen from distinct moving regions, these landmarks are clustered correctly. According to the clusters, two moving objects have been obtained from the segmentation, which are shown in figure 7.2.

(a)

(b)

(c)

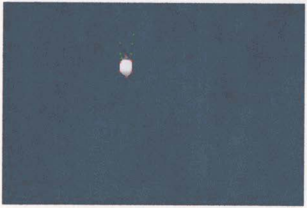
Figure 7.2 The manual motion segmentation of Salesman

(a) the clustering of the landmarks (b) the first moving object (c) the second moving object

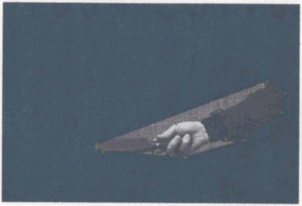
7.1.3 Table Tennis Sequence

Two objects have been obtained from the segmentation of the sequence, which are shown in figure 7.3. Since the landmarks are chosen from distinct moving regions, these landmarks are clustered accurately and moving objects are produced correctly.

(a)



(b)



(c)

Figure 7.3 The manual segmentation of Table Tennis
(a) the clustering of the landmarks (b) the first moving object
(c) the second moving object

7.2 Automatic Process

For the automatic process, sub-block matching is used to produce the landmarks. However, VBA is also efficient for many types of pictures, as demonstrated in the clustering experiments. By adjusting weights, VBA can provide distinctive sets of landmarks with different numbers and positions of landmarks. Therefore, as an alternative, VBA remains its significance for the future. Since sub-block matching offers stable and efficient segmentation, this approach is chosen for the current motion segmentation.

7.2.1 Flower Garden Sequence

One moving object and the background have been segmented out from the sequence, as shown in figure 7.4. The moving object of the tree represents the dominant motion.

(a)

(b)

(c)

Figure 7.4 The segmentation of the Flower Garden sequence

(a) the clustering of landmarks (b) moving object (c) the background

7.2.2 Salesman Sequence

One moving object has been obtained from the segmentation, as shown in figure 7.6. The remainder is segmented as the background.

(a)

(b)

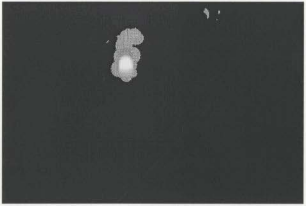
(c)

Figure 7.5 The segmentation of the Salesman sequence

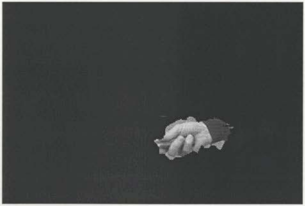
(a) the clustering of landmarks (b) moving object (c) the background

7.2.3 Table Tennis Sequence

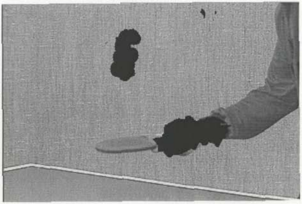
Two moving objects of the ball and the moving hand have been segmented out of the sequence, and the remainder has been segmented as the background, as shown in figure 7.7. The resulting segmentation is correct because it shows the fundamental shapes of different objects, and therefore outlines the trend for further segmentation.



(a)



(b)



(c)

Figure 7.6 The segmentation of the Table Tennis sequence

(a) moving object 1 (b) moving object 2 (c) the background

7.3 Summary

For the manual process, all moving objects have been segmented out accurately. Those intended landmarks are placed across moving objects and have accurate motion vectors.

In the automatic process, clustering performs best for the sub-block matching method. Since this approach for identifying landmarks is coupled with motion estimation through the matching strategy, the resulting motion vectors are more accurate than other landmark detectors. The best clusters are selected as the input to the segmentation. Image pictures are segmented around the clusters. According to this strategy, for a moving region, accurate segmentation results provided that the moving region embraces a large enough number of landmarks.

7.4 Application to Multiphase Fluid Flow

Moving objects in multiphase fluid flow are bubbles, slugs, and churns. These objects are highly non-rigid objects. To obtain data about a bubble, it is virtualized into a globe and landmarks are placed manually on the frames. It is **noted** that the manual process for fluid flow is distinguished from that for standard sequences. For fluid flow, there is no automatic clustering or motion segmentation as for the standard sequences. Moving objects are determined in advance through human control, and landmarks are placed according to distinct objects. Once the manual landmarks are obtained, they are used to calculate the surface areas and the velocities of the objects. Since bubbles are virtualized into globes, through calculating the surface area and the volume of the globe, I obtain approximate data about bubble volumes. To calculate the velocities, a bubble-tracking scheme has been developed. A display of the tracking scheme is shown in figure 7.7.

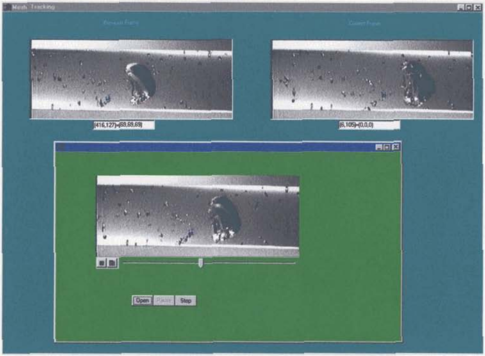


Figure 7.7 The display of bubble tracking

Computation of Flow Images

The segmentation of the flow sequence into three moving bubbles is shown in figure 7.8. Landmarks are placed on those moving bubbles according to the predicted segmentation. The computation of flow involves the calculations of the volume, the surface area, and the velocity of moving bubbles. Although bubbles keep deforming during the movement, they can be considered unchanged within a short enough duration. To approximate the volume, a bubble is virtualized as a globe; corresponding points are placed manually around the maximum circumference of the globe; the volume and surface area of the globe are approximated as those of the bubble. The gravity center of the globe is represented by the mean of the corresponding points

$$(x,y)_{grav\_center} = \frac{1}{N}\sum_{i=1}^{N}(x_i, y_i) \qquad (7.1)$$

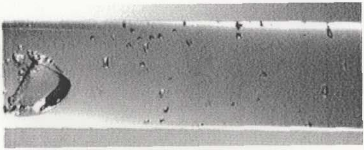
where (x_i, y_i) denotes the coordinate of a boundary point.

Then the velocity

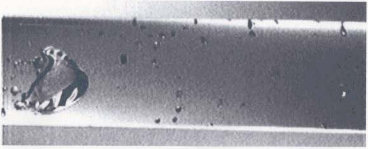
$$(v_x, v_y) = \frac{(s_x, s_y)}{\Delta t} \qquad (7.2)$$

where the numerator (s_x, s_y) denotes the difference in x-axis and y-axis directions between any two positions of the center and the denominator Δt is the corresponding time.

Although this kind of approximate calculation involves error, since the gravity center resulting from the averaging of corresponding points is used to calculate the speed, it is statistically robust against noise errors and therefore the resultant speed is accurate. All the distance is calculated in pixels. To transform image data into the physical distance, it is necessary to record the size ratio between the video frame and the real scene during shooting of the video sequence.



(a)



(b)

(c)

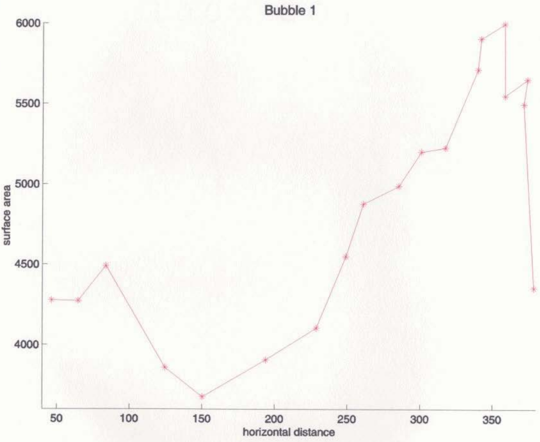
(d)

(e)

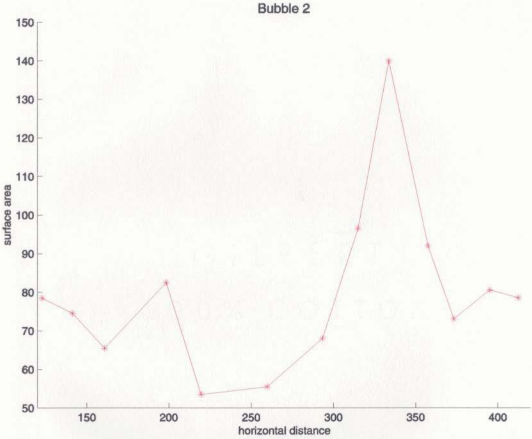
Figure 7.8 Manual segmentation of the flow sequence
(a) frame n (b) frame n+1 (c) object 1 (d) object 2 (e) object 3

The average velocities for the three bubbles have been calculated in pixels/frame. They are 4.3113 pixels/frame, 4.5979 pixels/frame, and 5.1029 pixels/frame. The moving states those velocities present have been verified to agree with results from the multiphase fluid dynamics experiments. If the resulting velocities are converted into physical ones, they should match those from the multiphase fluid experiments. The sizes of the bubbles vary with their horizontal movements. The surface area to the horizontal distance for the bubbles is shown in figure 7.9.

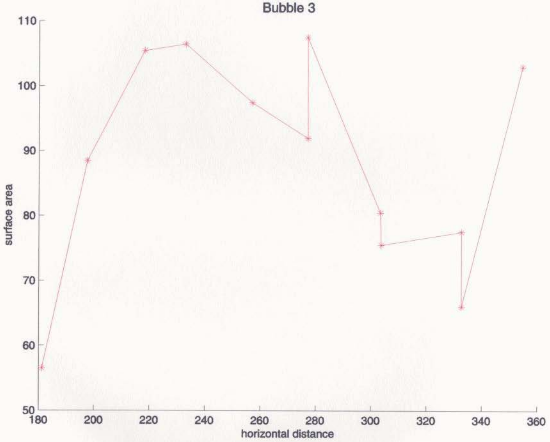
In figure 7.9, for each bubble, the surface area varies with the horizontal movement. Although accurate data on the size cannot be obtained, my approximate calculation provides the comparison among the bubbles in terms of size against velocity. This kind of comparison suggests that at these velocities, the bubbles are highly non-rigid objects, varying greatly in size as they move.


Bubble 1
6000
5500
5000
4500
4000
surface area
50
100
150
200
250
300
350
horizontal distance

(a)


Bubble 2
150
140
130
120
110
100
90
80
70
60
50
surface area
150
200
250
300
350
400
horizontal distance

(b)



(c)

Figure 7.9 Comparison of surface area at different horizontal points

(a) bubble 1 (b) bubble 2 (c) bubble 3

CHAPTER 8

CONCLUSIONS

8.1 Contributions

In this thesis, I have proposed, implemented, and demonstrated a new kind of motion analysis platform for video sequence segmentation. This thesis has made five primary contributions:

1. This thesis has proposed, implemented and demonstrated a new scheme for motion estimation and motion segmentation. The scheme employs a limited number of landmarks to represent image features and performs motion segmentation. Distinguished from typical motion segmentation, the scheme does not need to compute motion on each pixel. Motion computation is performed only on landmarks. The strength of the approach lies in the fact that through the adjustment of the number and positions of landmarks, the scheme controls the motion estimation and motion segmentation procedure. Also, robustness is naturally obtained through the scheme.

2. The scheme couples identifying landmarks automatically with motion estimation through the matching strategy. This approach provides accurate motion estimation of the landmarks, which prepares reliable inputs to motion segmentation.

3. Two efficient methods have been developed for identifying landmarks automatically in this scheme: sub-block matching and VBA, which are more efficient than typical feature point detectors. These methods are powerful enough and very flexible for many types of pictures.

4. The scheme allows human control. Human or subjective control not only plays a key role in the manual approach, but also serves as a standard for motion segmentation.

5. This thesis has made a practical exploration into applications. A platform has been developed to serve comprehensive motion analysis. All types of landmark generators, motion estimation, motion segmentation, and all kinds of controls and result displays are integrated in the platform. The software is fast enough to be utilized in the applications. The power of the software lies in the fact that with many kinds of switches and controls, the platform provides ample flexibility.

8.2 Future Work

Future work can be deployed in two directions: to improve the segmentation of video sequence and to implement more applications of the segmentation. One avenue for future research would be to improve the segmentation; for example by incorporating other attributes, such as texture, position and color, in the motion segmentation.

To further validate this kind of strategy of motion analysis, various applications need to be implemented. For example, object tracking, video editing, video composing, and video coding can have their own strategies of implementation. The development of various applications will allow more examination of the present scheme and help to design better future methods.

CHAPTER 9

REFERENCES

[1] Adelson, Edward, H., and John Y. A. Wang. Layered Representations for Motion Analysis. *Proceedings of Computer Vision and Pattern Recognition Conference*, June 1993.

[2] E. Amir, S. McCanne, and R. Katz. An Active Service Framework and Its Application to Real Time Multimedia Transcoding. *Proceedings of SIGCOMM '98*, Vancouver, British Columbia, September 1998.

[3] H. Balakrishnan, S. Seshan, E. Amir, R. Katz. Improving TCP/IP Performance over Wireless Networks. *Proceeding of the 1st ACM Conf. On Mobile Computing and Networking*, Berkeley, CA, November 1995.

[4] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, 12:1, 43-77, 1994.

[5] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. MACAW: A Media Access Protocol for Wireless LANs. *Proceedings of SIGCOMM '94*. University College London, London, U.K., pp. 212-225, September 1994.

[6] M. J. Black, and P. Anandan. The robust estimation of multiple motions : Parametric and piece-wise smooth flow fields. *Computer Vision and Image Understanding*, CVIU, 63(1), pp. 75-104, Jan. 1996.

[7] V. M. Bove, Jr.. Object-Oriented Television. *SMPTE Journal*, 104, pp.803-807, December, 1995.

[8] Xin Cheng and John A. Robinson. Identifying Rigid Moving Objects from Point-to-Point Correspondence. *Project Report, Faculty of Engineering, Memorial University of Newfoundland*, November, 1998.

[9] T. Darrell and A. P. Pentland. Cooperative Robust Estimation Using Layers of Support. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):474-487, 1995.

[10] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

[11] D. J. Fleet, and A. D. Jepson. Stability of Phase Information. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1993.

[12] Foley, van Dam, Feiner, Hughes, and Phillips. *Introduction to Computer Graphics.* Addison-Wesley, 1994.

[13] D. Heeger. Optical Flow Using Spatiotemporal Filters. *International Journal of Computer Vision*, 1:270-302, 1988.

[14] B. Horn and B. Schunk. Determining Optical Flow. *Artificial Intelligence*, vol. 17, 1981.

[15] P. J. Huber. *Robust Statistics*. Wiley, New York, 1981.

[16] M. Irani, S. Hsu, and P. Anandan. Mosaic based video compression. *Proceedings of SPIE Conference on Electronic Imaging*, volume 2419, February 1995.

[17] M. Irani, S. Hsu, and P. Anandan. Video Compression Using Mosaic Representation. *Signal Processing: Image Communication, special issue on Coding Techiques for Low Bit-rate Video*, vol. 7, No.4-6, pp. 529-552, November 1995.

[18] B. Lucas, and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. *Proceedings of DARPA Image Understanding Workshop*, pp. 121-130, 1981.

[19] R. Mann, A. Jepson, and J.M. Siskind. The Computational Perception of Scene Dynamics. *Proceedings of 4^{th} European Conf. On Computer Vision*, Cambridge, April 1996, pp. 528-539.

[20] D. Murray and B. Buxton. Scene Segmentation from Visual Motion Using Global Optimization. *IEEE Trans. On Pattern. Analysis and Machine Intelligence*, vol. PAMI-9, March 1987.

[21] David A. Patterson, John L. Hennssy. *Computer Organization and Design: The Hardware/Software Interface. Second edition*. Morgan Kaufmann Publisher, San Francisco, 1997.

[22] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Tools for Content-Base Manipultion of Image Databases. *International Journal of Computer Vision*, vol. 18, No.3., pp. 233-254, 1996.

[23] E. P. Simocelli, W. Freeman, E. Adelson, and D. Heeger. Shiftable Multi-scale Transfroms. *IEEE Transactions on Information Theory*, 38:587-607, 1992.

[24] S. M. Smith, and J.M. Brady. SUSAN----A New Approach to Low Level Image Processing. *Technical Report TR95SMSIC, Department of Engineering Science, Oxford University*, 1995.

[25] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision, second edition*. ITP, 1998.

[26] R. Szeliski and H. Shum. Creating Full View Panoramic Image Mosaics and Environment Maps. *Proceedings Of SIGGRAPH*, pp. 251-258, 1997.

[27] Charles W. Therrien. *Decision Estimation And Classification.* John Wiley and Sons, Inc., 1989.

[28] Y. Yacoob and M. J. Black. Parameterized Modeling and Recognition of Activities. *ICCV-98*, Mumbai, India, pp. 120-127, 1998.

[29] J. Zhang, J. Modestino, and D. Langan. Maximum-Likelihood Parameter Estimation for Unsupervised Stochastic Model-Based Image Segmentation. *IEEE Trans. on Image Processing*, Vol. 3, July 1994.

[30] *http://drogo.cselt.it/mpeg/*