國立交通大學

資訊學院

資訊科學與工程研究所

博士論文

應用不變量澤爾尼克矩描述元

進行影像之表示、比對及辨識

Image Representation, Matching, and Recognition Using Invariant Zernike Moment Descriptors

研 究 生：孫 樹 國

指導教授：陳　稔　教授

中華民國九十八年七月

應用不變量區域描述元進行影像之表示、比對及辨識

Image Representation, Matching, and Recognition Using Invariant Zernike Moment Descriptors

研究生：孫樹國　　Student：Shu-Kuo Sun

指導教授：陳　稔　　Advisor：Zen Chen

國立交通大學

資訊學院

資訊科學與工程研究所

博士論文

A Dissertation

Submitted to Institute of Computer Science and Engineering
College of Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Computer Science

July 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年七月

應用不變量澤爾尼克矩描述元
進行影像之表示、比對及辨識

研究生： 孫樹國　　　　　　　　　　指導教授： 陳稔博士

國立交通大學

資訊學院

資訊科學與工程研究所

摘 要

本論文在探討三維電腦視覺中，利用二張或更多張不同視度或不同光照條件下所拍攝的景物影像來進行景物分析、辨識及套合等研究，所需克服影像間存在之幾何轉換(旋轉、尺度變化、平移及幾何變形)、影像亮度轉換 (影像模糊、照度改變、雜訊、影像壓縮等）、部分遮蔽以及影像套合計算效率等問題。

首先，我們提出一個基於澤爾尼克矩相位資訊為主的不變量區域描述元，同時包含精確估算二個特徵區域間旋轉角度的方法來解決旋轉方位對齊的問題，以及一個可以達到高可靠度的比對函式。整體而言，在上述不同的幾何及影像亮度轉換下，這個新的澤爾尼克矩相位描述元較諸目前五個主要方法具有更高的區辨能力，論文中亦包含定性及定量分析來說明這些描述元效能差異的原因。

其次，我們將這個區域描述元延伸到行動裝置服務之商標符號辨識上，它可使用於

企業識別、公司網頁存取、交通安全號誌辨認及安全檢查等相關應用上，在此主要的挑戰是行動裝置拍攝影像時所無法避免的幾何及影像亮度轉換，我們提出二種相似度量測方法分別用於分類及檢索上，實驗顯示我們提出的方法較之既有的三個主要方法具有更好的效能。

最後，我們提出一個不同於傳統之影像套合方法，更有效率的達到不同視點影像套合所需之一對一特徵點對應，此方法是基於事先分析參考影像以獲取重要的資訊來引導影像套合程序之進行。首先，在離線階段先針對參考影像中的特徵點根據下述五個規劃策略來事先建立挑選順序之資料庫: (1)特徵點對影像變形之不變量、(2)對影像雜訊之抵抗力、(3)描述元之區辨能力、(4)模型估算之有效性及(5)影像部份重疊之處理能力。因此，當我們獲得感測影像進行影像套合時，即可更有效率的建立這二張影像間之特徵點一對一對應關係，來估算這二張影像的轉換模型。

Image Representation, Matching, and Recognition Using Invariant Zernike Moments Descriptors

Student：Shu-Kuo Sun Advisor：Dr. Zen Chen

Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University

ABSTRACT

In 3D computer vision a scene in the real world is represented by multiple views imaged under different viewpoints and illumination conditions. The spatial and temporal relationships across these views are important to scene analysis and understanding. To derive these relationships the global and local features of the objects (foreground and background) in the scene are the clues. The local features related to the local object surface patches or regions are more robust to viewpoint change than the global features. In addition, the invariance under the photometric transformations such as blur, illumination, scale, noise, JPEG compression is also receiving great attention.

In this dissertation subjects related to the local image representation, matching, and recognition under the above image variations are addressed. First, a new distinctive image descriptor to represent the normalized regions extracted by an affine region detector is proposed which primarily comprises the Zernike moment (ZM) phase information. An accurate and robust estimation of a possible rotation angle between a pair of normalized regions is then described, which will be used to measure the similarity between two matching regions. The discriminative power of the new ZM phase descriptor is compared with five major existing region descriptors based on the precision-recall criterion. The experimental results involving more than 15 million region pairs indicate the proposed ZM phase descriptor has, overall speaking, the best performance under the common photometric and geometric transformations. Both quantitative and qualitative analyses on the descriptor performances are given to account for the performance discrepancy.

Second, the proposed ZM phase descriptor is further extended to present a new recognition method of logos imaged by mobile phone cameras. The logo recognition can be incorporated with mobile phone services for use in enterprise identification, corporate website access, traffic sign reading, security check, content awareness, and the related applications. The main challenge to applying the logo recognition for mobile phone applications is the inevitable photometric and geometric transformations. The proposed ZM phase recognition method is associated with two similarity measures. The logo classification and retrieval experimental results show that the proposed ZM phase method has the best performance under the typical photometric and geometric transformations, compared with other three major existing methods.

Finally, as for the one-to-one feature matching correspondences in view registration, we propose an efficient registration method different from the traditional methods. We take advantage of preprocessing of the reference image offline to gather the important statistics for guiding image registration. That is, we introduce five planning strategies to sort the feature points in the reference image based on the concepts of (1) feature invariance to image deformation, (2) image noise resistance, (3) distinctive description power, (4) model estimation effectiveness, and (5) partial image overlapping handling capability. Thus, a reference matching database is constructed offline using the above five planning strategies. Then, an online registration process is presented to estimate the transformation model to overlay the reference image over an incoming sensed image. In this way, better registration efficiency can be achieved.

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to my advisor, Dr. Zen Chen, for his kind patience, constant encouragement, helpful guidance, inspiration throughout, and the invaluable training the course of this dissertation. In these years, he has stimulated the research work and teaches me how to learn and how to think interpedently. Especially, he encourages me to challenge what we are used to be. I also express my sincere gratitude to the members of my thesis committee, Professor Chuang and Professor Wang, for their valuable suggestions and comments.

Finally, I am so grateful to my wife, my parents, and my children for their love, support, and tolerance during the dissertation study. This dissertation is dedicated to them.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Problems Statement

In 3D computer vision a scene in the real world is represented by multiple views when imaged under different viewpoints and illumination conditions. The spatial and temporal relationships across these views are important to scene analysis and understanding. To derive these relationships the global and local features of the objects (foreground and background) in the scene are the clues. Global features such as Fourier descriptors describe the scene information as the scene is seen in the 2D image as a whole. The global features are suitable for deriving the relationships when the objects of concern have the same appearances in the different views. Generally, the objects have different surface appearances under different viewpoints, especially when the background and foreground objects are partially overlapped, so the global image features are often not invariant to the viewpoint. On the other hand, the local features related to the local object surface patches or regions are more robust to viewpoint change. In addition, the invariance under the photometric transformations such as blur, illumination, scale, noise, JPEG compression is also receiving great attention. The invariant local features are crucial to most image understanding and computer vision applications including image matching, camera calibration, texture classification, and image retrieval, etc. [1]-[5].

The processing of local features involves three tasks: feature detection, feature description, and feature matching. The local features belong to an interest point (keypoint) or an interest region. Since a single image point carries little information, an interest point must be associated with its surrounding image patch. From this image patch a second moment matrix of image intensities reveals the characteristic structure of the local image region. The keypoint detectors such as Harris corner detector [6] and the SIFT detector [7], which is based on the difference of Gaussians (DOG), utilize a circular window to search for a possible location of a keypoint. However, the image content in the circular window is not robust to affine deformations. Furthermore, the feature points may not be reliable and may not appear simultaneously across the view-point change, as illustrated in Fig. 1.1(b). Recently, a number of local feature detectors using a local elliptical window have been investigated. The affine covariant regions offer a unique solution to viewpoint change, as illustrated in Fig. 1.1(c). Matas et al. [5] presented a maximally stable extremal region (MSER) detector. Tuytelaars and Van Gool [8] developed an edge-based region (EBR) detector as well as an image-based (IBR) region detector. Mikolajczyk and Schmid [9] proposed Harris-Affine and Hessian-Affine detectors. The performances of the existing region detectors were evaluated in [11] in which the MSER detector and the Hessian-Affine detector were the two best.

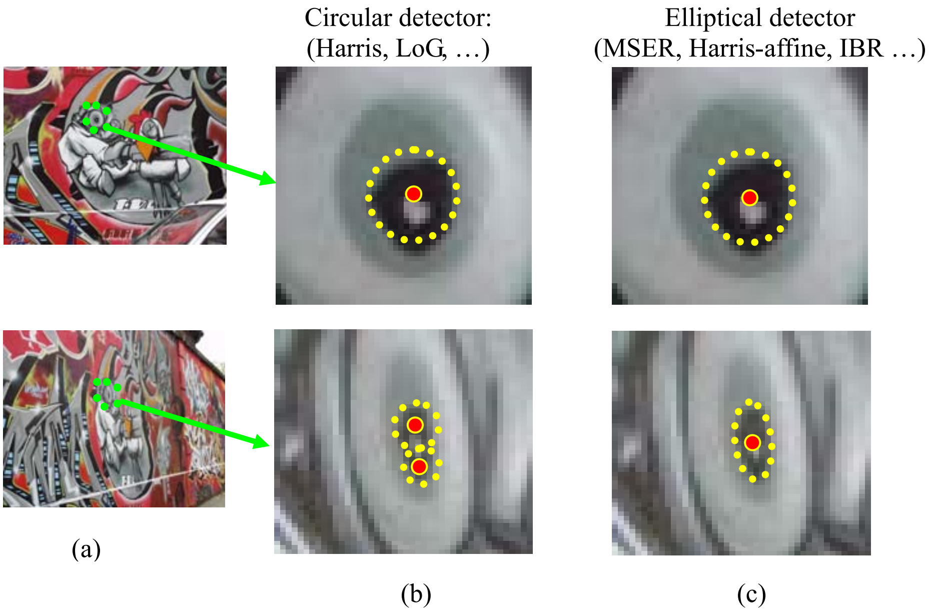


Fig. 1.1: (a) Two images taken from different viewpoints. (b) The detected regions by a circular detector. (c) The detected regions by an elliptical detector.

In the descriptor construction, the detected ellipse-shaped region is first normalized to a circular patch of a fixed size. The normalized circular patch can be shown to be affine invariant up to a rotational ambiguity [10, 33]. A good feature descriptor to describe the normalized circular patch should be invariant (unchanged under the spatial transformation), distinctive (unique in feature description), stable (robust to image deformation) and independent (uncorrelated relation between feature descriptors).

After the region descriptor is determined, a matching function is defined to measure the similarity between regions extracted from different images of the same scene. The merits of various region detectors, coupled with their own region descriptors, are often judged based on the ROC (receiver operating characteristic) curve or the PR (precision-recall) curve.

1.2 Sketch of the Work

In this dissertation, three themes related to the image representation, matching, recognition, and view registration under the aforementioned geometric and photometric transformations are addressed.

In the first theme, the representation and matching power of region descriptors are to be evaluated. A common set of elliptical interest regions is used to evaluate the performance. The elliptical regions are further normalized to a circular one with a fixed size (typically, 41 by 41 pixels). Here a new distinctive image descriptor to represent the normalized region is proposed which primarily comprises the Zernike moment (ZM) phase information. An accurate and robust estimation of the rotation angle between a pair of normalized regions is then described, which will be used to measure the similarity between two matching regions. The discriminative power of the new ZM phase descriptor is compared with five other major region descriptors (SIFT, GLOH, PCA-SIFT, complex moments, and steerable filters) based on the precision-recall criterion. To match the region pairs, a new distance measure based on the ZM phase information is defined. For performance evaluation, important system parameters must be taken into consideration, which include (1) region scene types, (2) region descriptor types, (3) region detector types, (4) region overlap error, and (5) transformation types. From the experimental results involving more than 15 million region pairs the proposed ZM phase has the best overall performance under the aforementioned photometric and geometric transformations. Both quantitative and qualitative analyses on the descriptor performances are given to account for the performance discrepancy.

In the second theme, the proposed ZM phase descriptor is further extended to present a new recognition method of logos imaged by mobile phone cameras. The logo recognition can be incorporated with mobile phone services for use in enterprise identification, corporate website access, traffic sign reading, security check, content awareness, and the related applications. The main challenge to applying the logo recognition for mobile phone applications is the inevitable photometric and geometric transformations encountered when using a handheld mobile phone camera operating at a varying viewpoint during the daytime or the nighttime. The discriminative power of the new logo recognition method is compared with three major existing methods. The experimental results indicate the proposed ZM phase method has the best performance in terms of the precision and recall criterion under the above inevitable imaging variations.

In the third theme, we propose an efficient registration method for the one-to-one feature matching correspondences in view registration. We take advantage of preprocessing of the reference image offline to gather the important statistics for guiding image registration. That is, we introduce five planning strategies to sort the feature points in the reference image based on the concepts of (1) feature invariance to image deformation, (2) image noise resistance, (3) distinctive description power, (4) model estimation effectiveness, and (5) partial image overlapping handling capability. The invariant feature points are extracted from the reference image and a reference matching database is constructed offline using the above five planning strategies. Then, an online registration process is presented to estimate the transformation model to overlay the reference image over an incoming sensed image. In this way better registration efficiency can be achieved.

1.3 Contribution of the Work

The main contributions of this dissertation can be summarized as follows:

(1) To design a new region descriptor and a new matching function based mainly on Zernike moment (ZM) phase information and show the ZM phase information is more distinctive than the ZM magnitude information in terms of image representation and matching.

(2) To propose an accurate estimation of the rotation angle between a region pair to be matched.

(3) To show the proposed ZM phase descriptor has the better overall performance than the five other major descriptors under common geometric and photometric transformations.

(4) To extend the ZM phase descriptor to design a new distinctive logo feature vector and two associated similarity measures for logo recognition, and to show the proposed ZM phase logo feature vector has better recognition and retrieval performance than other three existing methods.

(5) To develop a new view registration method that take advantage of preprocessing of the reference image offline to gather the important statistics for image registration, and achieves better view registration time complexity than other existing methods.

1.4 Dissertation Organization

The rest of this dissertation is organized as follows. Chapter 2 reviews existing literature on local region descriptors, methods for logo recognition, as well as methods for image registration. Chapter 3 presents our Zernike Moment phase based descriptor for local image representation and matching. Chapter 4 extends our Zernike Moment phase based descriptor to logo recognition. Chapter 5 presents five offline planning strategies and an online registration process for high-efficiency perspective view registration. Finally, Chapter 6 closes the dissertation with a summary of our work and a discussion on possible extensions and future research directions.

Chapter 2

Previous Work

2.1 Region detectors and descriptors

A region descriptor is needed to derive the region features for region representation and matching after the regions of interest are detected. Here a brief introduction of five major classes of the existing descriptors is briefly given to explore their strengths and weakness in order to compare them with the proposed ZM phase based descriptor. An excellent review on the existing descriptors can be found in [12]-[13].

(1) Filter-based Descriptors:

This class of descriptors includes steerable filters [14] and Gabor filters [15]. The steerable filter descriptor uses quadrature pairs of derivatives of Gaussian and their Hilbert transforms to synthesize any filter of a given frequency with arbitrary phase. On the other hand, the Gabor transform uses a number of Gabor filters tuned to various frequencies and orientations to represent the image patterns. Both the steerable filter and the Gabor filter descriptors need to seek a dominant orientation for image rotation alignment. If the reference and transformed descriptor feature vectors are not aligned well, their matching score will be poor. Besides, these descriptors are not totally orthogonal and their feature vector dimensions are generally low, so their discriminative powers are limited.

(2) Moment-based descriptors:

The first class of moment-based descriptor is the geometric (or regular) moments. The (p+q) order moment of an intensity or gradient image $f(x,y)$ is defined as follows

$$m_{pq} = \sum_x \sum_y x^p y^q f(x,y), \quad p,q = 0,\ 1,\ 2,\ ...$$

Based on the geometric moments, a set of moment invariants can be derived from the nonlinear combinations of geometric moments to achieve affine invariance [16], [32]. The main problem with the geometric moments is that it is difficult to derive a sufficient number of invariants to describe complex shapes. Moreover, the higher-order moments are more sensitive to image noise than the lower-order moments. Therefore, the geometric moment invariants are usually suitable only for describing simple images [17].

The second moment class is the complex moments of the form $K_{mn}(x,y) = \sum_x \sum_y (x+iy)^m (x-iy)^n f(x,y)$ where $f(x, y)$ is an image intensity function [18], [19]. Any rotation of the image changes the phases of the complex moments, but not the magnitudes. That is, the magnitudes of the filter responses are rotational invariant. There are 16 filters, defined by $m + n \leq 6$ and $n \leq m$, available for image patch description. This low dimensional rotational invariant descriptor generally has a poor discriminative performance [12].

(3) Distribution-based descriptors:

This class of descriptors includes SIFT [7], GLOH [12], PCA-SIFT [22], spin image and RIFT descriptors [3]. They use the distributions of the image content to represent the features of the image region.

The SIFT descriptor is represented by a 3D histogram of gradient locations and orientations. The histogram of the gradient orientations is quantized in 8 bins and the region is partitioned into a 4×4 location grid, resulting in a feature vector of dimension 128. Although the gradient histogram provides stability against deformations of the image pattern, the grid partition of the measurement region has the boundary effect problem. Gaussian smoothing and tri-linear interpolation can be called to alleviate this problem. More importantly, SIFT requires an accurate dominant (gradient) orientation for image rotation alignment.

The PCA-SIFT descriptor is a dimension-reduced version of SIFT (dimension reduced from 3042 to 36 or lower) based on an eigenspace obtained by applying PCA to a collection of 21,000 image patches. On the other hand, the GLOH descriptor is also an extension of the SIFT descriptor. Instead of sampling gradient orientations in a rectangular grid, GLOH is defined in a log-polar location grid with 17 location bins. These location bins, together with 16 gradient orientation bins, form a feature vector of dimension 272. With PCA the feature dimension is reduced to 128 based on a training data set of 47,000 image patches.

The SIFT and its variants depend on a dominant orientation of the normalized patch to achieve the rotation invariance. However, according to the experience of Lazebnik et al. reported in [3], the dominant orientation estimation tends to be unreliable, especially for normalized Laplacian regions in which strong edges at the center are often not available.

(4) Derivative-based descriptors:

This type of descriptors uses local derivatives, called “local jets”, to construct the differential invariants, which are rotationally invariant [23]. Schmid and Mohr [2] derive a set of differential invariants in terms of polynomials of local derivatives up to the third order for image retrieval. The derivative-based descriptors face with some problems: (a) the dimension of the rotationally invariant differential invariants is generally low [12], and (b) the differential invariants are often sensitive to image blur or image noise if smoothing operation is not used beforehand. (The steerable filters can be also classified as a derivative-based descriptor.)

(5) Others:

Besides the above basic descriptor types, there are other extended descriptors including (i) color-based descriptors [21] which utilizes the color information for feature representation, (ii) textons [3], which are based on the responses of a texture image to a filter bank, can categorize the large-scaled texture images. In this paper, only the basic descriptors of the first four classes are concerned.

2.2 Logo Recognition

A logo is a graphic entity containing colors, shapes, textures, and perhaps text as well, organized in some spatial layout format. There are four major classes of the existing feature used for logo recognition:

(1) Color features:

Color feature are often easily obtained from the logo image. The color histogram [54] is probably one of the most popular gross representations of the foreground object in which the precise spatial information is lost, so an exact matching is generally impossible. Since a logo may be designed with a few setting of color combinations, color will be ignored as far as the unique identity of a logo (represented as an intrinsic graphic pattern) is concerned.

(2) Text features:

The text in the logo is often modified to add to its aesthetic appealing, its segmentation for the OCR processing may not be easy and also unnecessary for logo identification. The whole text can be viewed as part of the logo and handled with others by a general shape analyzer.

(3) Texture features:

Similarly, if a logo contains texture patterns, the texture patterns can be treated as a graphic pattern and, again, handled with other parts together. In the end, a logo representation is boiled down to an integrated shape pattern or a set of sub-logo shape patterns. Hence, shape analysis of the logo is the main concern here.

(4) Shape features:

Different methods using different shape features for logo classification have been proposed in the literature. Edge histogram descriptor (EHD) [58] is an MPEG-7 texture descriptor that captures the spatial distribution of edges. EHD is represented by a histogram of the gradient orientations which is quantized in 5 bins and the region is partitioned into a 4×4 location grid, resulting in a feature vector of dimension 80. Although the gradient histogram provides stability against mild deformations of the image pattern, the grid partition of the support region will lead to the non-smooth boundary feature values, i.e., the so-called boundary effect problem.

Recently, some researchers using Gabor transform and wavelet transform for pattern recognition [55]-[56]. However, the set of Gabor filters is not orthogonal, and thus reduce its discriminative power. On the other hand, the wavelet transform has the advantages of multiple resolutions and reconstructability, but it is not rotational invariant (so is the Gabor transform). Therefore, both transforms need to solve the rotation problem first based on some orientation information.

To achieve rotation invariant, an alternative method using a ring projection structure is suggested in which the absolute sums of the sub-band coefficients (LH, HL and HH) of wavelet transform are accumulated within a specific number of rings [59]. However, the ring projection will lose the spatial information in the radial direction. As a consequence, a logo and its mirror version have the same ring projection profiles, and, therefore, become indistinguishable. More impotently, most of the above methods cannot work properly under photometric and geometric image transformations, as shall be seen.

2.3 View Registration

View registration is a process of overlaying images of the same scene taken at different imaging conditions [60-64]. View registration applications include satellite image registration [65, 66, 82, 83], medical view registration [61, 62], object recognition [69-70], motion tracking [71-73], image mosaic [74], automatic cartography [75], fundamental matrix estimation [76], and perspective reconstruction [74, 75]. Good survey on view registration can be found in [60-64].

Due to the variations in viewpoint, illumination and the sensor noise, the feature points may not be reliable and may not appear simultaneously across the multiple views. Therefore, the point correspondence validation is not a trivial task. One may skip the point correspondence matching and estimates the transformation model directly using an appropriate number of feature point pairs. Traditionally, there are three major ways for the direct registration model estimation:

(1) Clustering technique:

The clustering technique [85] takes an appropriate number of point pairs, say *r*, from a total of available point pairs, say *n*, to compute the occurrence histogram of each set of possible model parameters and picks the histogram cell with the maximum cluster size as the best solution model. This is a complete (or exhaustive) search for the best model.

(2) Random search for a correct model:

The method is to randomly select an *r*-point combination of an *n*-point set to instantiate a model [74]. After a pre-specified number of random trials, the model with the largest consensus set found are chosen as the final model; the model correctness depends on the size of the consensus set.

(3) Ordered search for a probable model:

Recently, another way was proposed to search for a correct model. That is, the set of ${}_nC_r$ possible models is sorted according to some goodness measure and an ordered search is conducted until an acceptable model is found [66]. This is an ordered search for a probable model.

Table 2.1 lists the major point-based view registration methods under four different transformation models: rigid transform [79], similarity transform [80, 81], affine transform [66, 82, 83], and 2D perspective projection (or homography) [84], together with their search strategy and time complexity. The transformation model estimation is through solving a system of linear equations in terms of 3, 4, 6, or 8 transformation parameters. We have observed that various countermeasures were taken to reduce the time complexity of the view registration method.

TABLE 2.1 A PARADIGM OF VIEW TRANSFORMATION ESTIMATION METHODS

Model type	Method	Search strategy	Time complexity[#]
Homography	The proposed method	Ordered	$O(m)$
	Suk and Flusser [84, 2000]	Random/Complete	$O(n^5m^5)$
Affine	Bentoutou et al. [83, 2005]	Ordered	$O(nm)$[*]
	Yang and Cohen [66, 1999]	Ordered	$O(nm)$[*]
	Flusser and Suk [82, 1994]	Ordered	$O(nm)$[*]
Similarity	Dufournaud et al. [81, 2004]	Random	$O(n^2m^2)$
	Wang and Chen [80, 1997]	Complete	$O(n^2m^2)$
Rigid	Isgrò and Pilu [79, 2004]	Random	$O(n^3m)$

[#] n and m are the total numbers of feature points in the reference and sensed images, respectively.

[*] These methods estimate the affine model only once based on the best matched point pairs found from the two images.

Chapter 3

A Zernike Moment Phase Based Descriptor for Local Image Representation and Matching

3.1 Introduction

Local features robust to common photometric transformations (blur, illumination, scale, noise, and JPEG compression) and geometric transformations (rotation, scale, translation, and viewpoint) are crucial to most image understanding and computer vision applications including image matching, camera calibration, texture classification, and image retrieval, etc. [1]-[5].

In this chapter, the representation and matching power of region descriptors are to be evaluated. A common set of elliptical interest regions is used to evaluate the performance. The elliptical regions are further normalized to a circular one with a fixed size. The normalized circular regions will become affine invariant up to a rotational ambiguity. Here a new descriptor, called the Zernike moment phase based descriptor (or ZM phase in short), is proposed. The phase information of a signal is more informative than the magnitude information for signal reconstruction was demonstrated by Oppenheim [34]. The robustness of local phase information for measuring image velocity and binocular disparity was studied in [35-36]. Recently, outputs of complex-valued steerable filter quadrature pairs taken as the separate feature elements for the design of a local image descriptor were proposed in [37-38], instead of combining the magnitudes of the quadrature pair into a single feature element, as done in [12]. They empirically showed that their individual local descriptors have better performance than the gradient-based SIFT descriptor or differential invariants under the affine geometric deformation and lighting variation. However, the feature vector containing the separate steerable filter quadrature pair outputs is not an orthogonal vector itself. If the orthogonal descriptor is used instead, the features are uncorrelated and more informative. So we shall seek a genuine orthogonal feature vector to derive a novel local descriptor with a higher descriptive power.

The discriminative power of the new ZM phase descriptor is compared with five other major region descriptors based on the precision-recall criterion using the set of test images given in [12] plus some new images. To match the region pairs, a new matching function based on the ZM phase information is defined. For performance evaluation, important system parameters are taken into consideration, which include (1) region scene types, (2) region descriptor types, (3) region detector types, (4) region overlap error, and (5) transformation types. The experimental results involving more than 15 million region pairs indicate the proposed ZM phase has the best overall performance. Both quantitative and qualitative analyses on the descriptor performances are provided to account for the performance discrepancy.

The chapter is organized as follows. Section 2 introduces the Zernike moment (ZM) transformation and the ZM basis filters. Section 3 proposes the ZM phase descriptor along with a matching function, and discusses the discriminative powers of the ZM magnitude components and the ZM phase components. In Section 4 the discriminative power of the new descriptor is compared with five existing region descriptors based on the precision-recall criterion, while taking important system parameters into consideration. In Section 5 both quantitative and qualitative analyses on the descriptors are provided to account for the descriptor performance discrepancy.

3.2 Fundamentals of Zernike Moments

Zernike moments (ZMs) have been used in object recognition and image analysis regardless of variations in position, size and orientation [20], [24]-[28]. Basically, the Zernike moments are the extension of the geometric moments by replacing the conventional transform kernel $x^m y^n$ with orthogonal Zernike polynomials. The relationships between the Zernike moments and geometric moments can be established [39]. The ZM coefficients are the outputs of the expansion of an image function into a complete orthogonal set of complex basis functions $\{V_{nm}(\rho,\theta)\}$. Teh and Chin [20] show that among many moment based shape descriptors, Zernike moment magnitude components are rotationally invariant and most suitable for shape description.

The Zernike basis function $V_{nm}(\rho, \theta)$ with order n and repetition m is defined over a unit circle in the polar coordinates as follows:

$$V_{nm}(\rho,\theta) = R_{nm}(\rho)e^{jm\theta} \quad \text{for} \quad \rho \le 1, \tag{3.1}$$

where $\{R_{nm}(\rho)\}$ is a radial polynomial in the form of

$$R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \frac{(n-s)!}{s!(\frac{n+|m|}{2}-s)!(\frac{n-|m|}{2}-s)!}\rho^{n-2s} \quad .$$

Here n is a non-negative integer and m is an integer satisfying the conditions: n-$|m|$ is even and $|m|<n$.

The set of basis functions $\{V_{nm}(\rho, \theta)\}$ is orthogonal, i.e.,

$$\int_0^{2\pi}\int_0^1 V^*{}_{nm}(\rho,\theta)V_{pq}(\rho,\theta)\rho d\rho d\theta = \frac{\pi}{n+1}\delta_{np}\delta_{mq} \quad \text{with} \quad \delta_{ab} = \{ \begin{matrix} 1 & a=b \\ 0 & otherwise \end{matrix} \quad . \tag{3.2}$$

The two-dimensional ZMs for a continuous image function $f(\rho, \theta)$ are represented by

$$Z_{nm} = \frac{n+1}{\pi}\int_0^{2\pi}\int_0^1 f(\rho,\theta)V^*{}_{nm}(\rho,\theta)\rho d\rho d\theta = \frac{n+1}{\pi}\int_0^{2\pi} \mathrm{e}^{-jm\theta}\int_0^1 f(\rho,\theta)R_{nm}(\rho)\rho d\rho d\theta. \quad (3.3)$$

For a digital image function the two-dimensional ZMs are given as

$$Z_{nm} = \frac{n+1}{\pi}\sum_{(\rho,\ \theta)\in \text{unit disk}}\sum f(\rho,\theta)V^*{}_{nm}(\rho,\theta) \quad . \quad (3.4)$$

The Zernike moments can be viewed as the responses of the image function $f(\rho, \theta)$ to a set of quadrature-pair filters $\{V_{nm}(\rho, \theta)\}$. To this end, Fig. 3.1 depicts some examples of $V_{nm}(\rho,\theta)$. Notice that the real and imaginary functions of each basis function $V_{nm}(\rho,\theta)$ are out of phase by π/2; namely, they form quadrature pairs of filters. In addition, repetition m indicates m sector cycles of the function values along the azimuth angle θ, while n and m jointly specify a different number of annular patterns of the function.

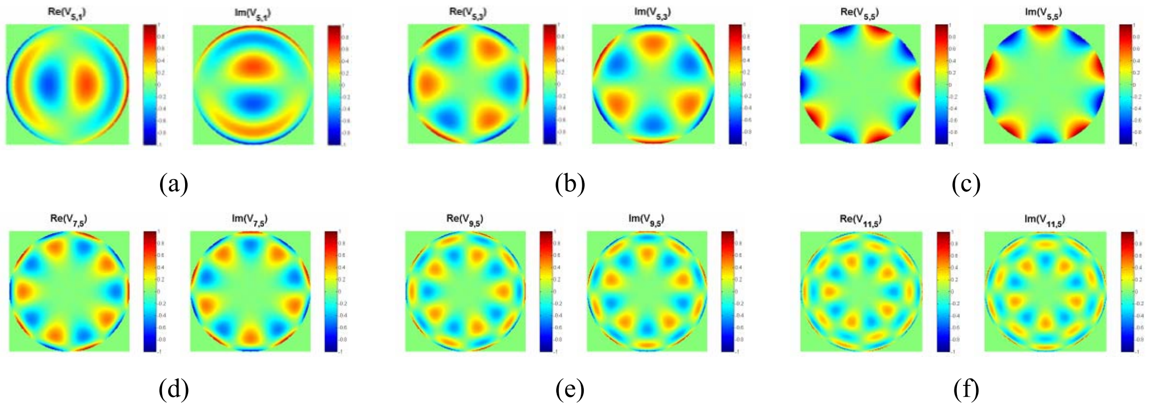


Fig. 3.1: Plots of the real part and imaginary part of $V_{nm}(\rho, \theta)$ for a fixed n: (a) $V_{5,1}$, (b) $V_{5,3}$, (c) $V_{5,5}$, and for a fixed m: (d) $V_{7,5}$, (e) $V_{9,5}$, and (f) $V_{11,5}$.

3.3 Design of A Zernike Moment Phase Based Descriptor

We shall use the ZM phase information to design a novel region descriptor. Let the Zernike moments be sorted by m and n in order. The total number of ZM moments of the same repetition m is equal to $\left\lfloor \frac{N-m}{2} \right\rfloor$+1. Table 3.1 gives the sorted list of the 42 complex ZM moments for the case where the maximum order N and maximum repetition M are both equal to 12.

The sorted Zernike moments form a feature vector $\vec{P}$ as follows:

$$\vec{P} = [|Z_{11}|e^{j\varphi_{11}}, |Z_{31}|e^{j\varphi_{31}}, \cdots\cdots, |Z_{NM}|e^{j\varphi_{NM}}]^T, \tag{3.5}$$

where $|Z_{nm}|$ is the ZM magnitude，and φ_{nm} is the ZM phase. Here the Zernike moments $|Z_{nm}|e^{j\varphi_{nm}}$ with m = 0 are not included, since they provide no information regarding the image matching. Zernike moments with m *<0* are not included, either, since they can be inferred through $Z_{n,-m} = Z^*{}_{nm}$.

TABLE 3.1. LIST OF ZMS SORTED BY n AND m IN SEQUENCE FOR THE CASE WHERE $(n, m) = (12, 12)$

m	Moments	No. of moments		m	Moments	No. of moments
1	$Z_{11}, Z_{31}, Z_{51}, Z_{71}, Z_{91}, Z_{11,1}$	*6*		**7**	$Z_{77}, Z_{97}, Z_{11,7}$	*3*
2	$Z_{22}, Z_{42}, Z_{62}, Z_{82}, Z_{10,2}, Z_{12,2}$	*6*		**8**	$Z_{88}, Z_{10,8}, Z_{12,8}$	*3*
3	$Z_{33}, Z_{53}, Z_{73}, Z_{93}, Z_{11,3}$	*5*		**9**	$Z_{99}, Z_{11,9}$	*2*
4	$Z_{44}, Z_{64}, Z_{84}, Z_{10,4}, Z_{12,4}$	*5*		**10**	$Z_{10,10}, Z_{12,10}$	*2*
5	$Z_{55}, Z_{75}, Z_{95}, Z_{11,5}$	*4*		**11**	$Z_{11,11}$	*1*
6	$Z_{66}, Z_{86}, Z_{10,6}, Z_{12,6}$	*4*		**12**	$Z_{12,12}$	*1*

3.3.1 The Image Description power of the ZM Magnitude Components and the ZM Phase Components

Let the Zernike moments of a reference image and its rotated version be $Z_{nm}^{ref}, Z_{nm}^{rot}$, respectively. Then it is well known that [24], [28]

$$Z_{nm}^{rot} = Z_{nm}^{ref} e^{-jm\alpha}, \tag{3.6}$$

where $\alpha \in [0, 2\pi]$ is the rotation angle.

Therefore, the magnitudes of Zernike moments of the two images are the same, i.e., $\left|Z_{nm}^{ref}\right| = \left|Z_{nm}^{rot}\right|$, but their phase difference (or phase shift) is given by

$$\Omega_{nm} \equiv \arg\left(\frac{Z_{nm}^{rot}}{Z_{nm}^{ref}}\right) = m\alpha, \quad 0 < \Omega_{nm} \leq 2m\pi, \text{ or} \tag{3.7}$$

$$\Phi_{nm} = (\varphi_{nm}^{ref} - \varphi_{nm}^{rot}) \bmod 2\pi = (m\alpha) \bmod 2\pi, \quad 0 < \Phi_{nm} \leq 2\pi. \tag{3.8}$$

In the following, under a mixture of rotation, inversion, and flipping operations, the Zernike moments of a reference image can be shown to be rotationally invariant in terms of the magnitudes, but not the phases.

Let a rotated-and-inverted (the inverted is in terms of gray values) image version of the reference image $f^{ref}(\rho,\theta)$ be denoted by $f^{rot-inv}(\rho,\theta) = 255 - f^{ref}(\rho,\theta+\alpha)$. It can readily shown that their magnitudes are equal: $\left|Z_{nm}^{ref}\right| = \left|Z_{nm}^{rot-inv}\right|$ and their phase difference is given by

$$\Phi_{nm} = [\varphi_{nm}^{ref} - \varphi_{nm}^{rot-inv}]\mathrm{mod}2\pi = \left[\varphi_{nm}^{ref} - (\varphi_{nm}^{ref} - m\alpha + \pi)\right]\mathrm{mod}2\pi = \left(m\alpha - \pi\right)\mathrm{mod}2\pi. \tag{3.9}$$

Next, let a rotated-and-mirrored version of the reference image $f^{ref}(\rho,\theta)$ be denoted by $f^{rot-mirror}(\rho,\theta) = f^{ref}(\rho,\pi-(\theta+\alpha))$. Then it can be shown that their magnitudes also are equal: $\left|Z_{nm}^{rot-mirror}\right| = \left|Z_{nm}^{ref}\right|$ and their phase difference is given by

$$\Phi_{nm} = (\varphi_{nm}^{ref} \text{ - } \varphi_{nm}^{rot-mirror}) \bmod 2\pi = [2\varphi_{nm}^{ref} \text{ - } m(\pi-\alpha)] \bmod 2\pi\,. \tag{3.10}$$

3.3.2 Zernike Moment Phase Descriptor and Its Similarity Measure

From above, it can be seen that the phase information of Zernike moments is more informative than the magnitude information in terms of the discriminative power. Therefore, a new image region descriptor is proposed which is mainly based on the phase components of the feature vector, while the magnitude components are used only as the weighting factors.

Let $I^r(x,y)$ and $I^t(x,y)$ as the reference and transformed image regions with their respective ZM feature vectors $\vec{P}_r = \{\left|Z^r{}_{nm}\right| e^{i\varphi_{nm}^r}\}$ and $\vec{P}_t = \{\left|Z^t{}_{nm}\right| e^{i\varphi_{nm}^t}\}$. Here the transformed image can be either a rotated version of the reference image or a different image. If there exists a rotation angle $\hat{\alpha}$ between $I^r(x,y)$ and $I^t(x,y)$, then $\left|\Phi_{nm} - (m\hat{\alpha}) \bmod (2\pi)\right|$, which denotes the absolute phase difference between the two image regions after the rotation alignment, is equal to 0; otherwise, $\left|\Phi_{nm} - (m\hat{\alpha}) \bmod (2\pi)\right|$ is a nonzero value in the interval (0, 2π) and $\hat{\alpha}$ is simply a putative estimate of a non-existent rotation angle. To derive a reliable estimate using all available phase differences $\{\Phi_{nm}\}$, we define a weighted, normalized phase difference to check the existence of a rotation angle $\hat{\alpha}$ as follows:

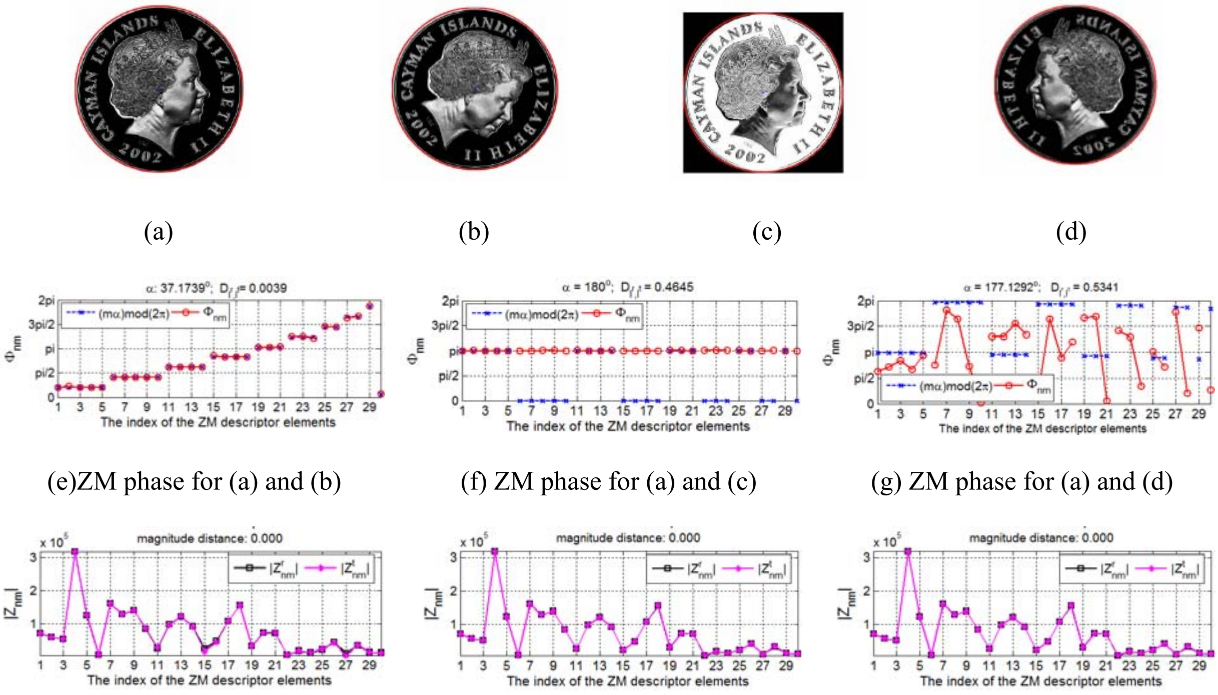
$$D_{I^r,I^t} = \sum_m \sum_n w_{nm} \frac{\min\{|\Phi_{nm} - (m\hat{\alpha})\,\mathrm{mod}(2\pi)|,\ 2\pi - |\Phi_{nm} - (m\hat{\alpha})\,\mathrm{mod}(2\pi)|\}}{\pi}, \tag{3.11}$$

where $\Phi_{nm} = (\varphi^r_{nm} - \varphi^t_{nm})\,\mathrm{mod}(2\pi)$, $\hat{\alpha}$ is the estimated rotation angle to be described later, and w_{nm} is a normalized weighting factor of the form

$$w_{nm} = \frac{|Z^r_{nm}| + |Z^t_{nm}|}{\sum_{n,m}(|Z^r_{nm}| + |Z^t_{nm}|)} \tag{3.12}$$

such that the phase components associated with small magnitudes are weighted less. The weighted, normalized phase difference D_{I^r,I^t} lies in the interval [0, 1] and is dimensionless since it is derived from ratios of angles.

Figs. 3.2(a)-2(d) show a reference coin image and its three variants: a rotated one (with a rotation angle 37.22^{o}), an inverted one, and a mirrored one, as described above. Image matching between the reference and each variant based on either the phase components or the magnitude components of Zernike moments are shown in Figs. 3.2(e) - 3.2(j) where the ZM order (*n, m*) ranges from (1, 1) to (10, 10). The estimated values of $(m\hat{\alpha})\,\mathrm{mod}(2\pi)$ are colored in blue and are connected for components with the same *m* values. The actual phase differences Φ_{nm} are shown in the red color. On the other hand, the ZM magnitude components for each pair of images are colored in purple. Notice that the magnitude component diagrams are the same for all the three pairs, but the phase component diagrams are different. Therefore, the phase components have a better discriminative power than the magnitude components.



(a) (b) (c) (d)

(e)ZM phase for (a) and (b) (f) ZM phase for (a) and (c) (g) ZM phase for (a) and (d)

(h) ZM magnitude for (a) and (b) (i) ZM magnitude for (a) and (c) (j) ZM magnitude for (a) and (d)

Fig. 3.2: (a) The reference coin image. (b) A rotated variant of the reference coin image (with a rotation angle 37.22°). (c) An inverted variant. (d) A mirrored variant. (e)-(g) The diagrams of the ZM phase differences (h)-(j) The diagrams of the ZM magnitude components.

3.3.3 Estimation of the Rotation Angle from a Rotated Image

In [29] Kim and Kim represented the rotation angle between an original image and its rotated image through the use of the Zernike moment phase shift as

$$\Omega_{nm} = (\varphi_{nm} + 2k_1\pi) - (\varphi^r_{nm} + 2k_2\pi) = \Phi_{nm} + 2\pi k_{nm} = m\alpha . \qquad (3.13)$$

They then proposed a probabilistic model $P(\hat{\alpha}) = \sum_m \sum_n \xi_{nm} P(\hat{\alpha} \mid n, m)$ to estimate the rotation angle α where ξ_{nm} is the weighting factor proportional to the ZM magnitude $|Z_{nm}|$. For each possible solution $\hat{\alpha}_{nm} = \frac{\Phi_{nm}}{m} + \frac{2\pi}{m} k_{nm}$, they used a probability density function

$P(\hat{\alpha}\,|\,n,m)=\frac{1}{m}\sum_{k_{nm}=0}^{m-1}\delta\left\{\hat{\alpha}-\left(\frac{\Phi_{nm}}{m}+\frac{2\pi}{m}k_{nm}\right)\right\}*G(\hat{\alpha},\sigma)$, a convolution of an impulse train with a scaled Gaussian kernel, to estimate α. Notice that the estimation is done in discrete angle steps. In order to be accurate, the estimation step size must be as small as possible. Let the estimation step size is 0.01^{o}. For the case where (N, M) = (10, 10), there are 30 generated Zernlike Moments $\{Z_{nm}\}$. From each fixed Zernike moment Z_{nm} an estimator of the rotation angle is given by $\hat{\alpha}_{nm}=\frac{\Phi_{nm}}{m}+\frac{2\pi}{m}k_{nm}$. There are 30 such estimators. To find the common solution to the rotation angle α using these 30 estimators, a common histogram with a bin size of 360×100 (assuming the estimation step size is 0.01^{o}) is used to tabulate the possible rotation angle produced by the 30 estimators. Therefore, the total number of histogram bin values computed is $360\times 100\times 30$ (=1,080,000), which is rather large. In addition, the method may face the ambiguity in multiple peaks in the histogram constructed.

Here a new estimation method of the rotation angle $\hat{\alpha}$ is proposed, which is implemented in the continuous angle space rather than in the discrete space. The basic idea behind the proposed method for estimating the rotation angle $\hat{\alpha}_m$ is to avoid the m ambiguities in the value of k_{nm}. Instead, the rotation angle $\hat{\alpha}$ can be found from the phase difference using any two adjacent Φ_{nm} and $\Phi_{n,m-1}, m\neq 0$, through

$$\alpha=m\alpha-(m-1)\alpha=(\Phi_{nm}+2\pi k_{nm})-(\Phi_{n,m-1}+2\pi k_{n,m-1}) \tag{3.14}$$

$$=(\Phi_{nm}-\Phi_{n,m-1})\bmod 2\pi\,,\quad m\neq 0\,.$$

Since m = 1, 2, .., M, n = 1, 2, ..,N, there are $\sum_{m=1}^{M}(\left\lfloor\frac{N-m}{2}\right\rfloor+1)$ ways to compute the rotation angle $\hat{\alpha}$. A more robust estimation is to weight the estimated angles by the individual magnitude $|Z_{nm}|$.

An iterative computation of the rotation angle $\hat{\alpha}$ using all available Zernike moments sorted by m is given below:

The ZM phase-based rotation angle estimation algorithm

```
Initialization: $\hat{\alpha}_0 = 0$ and $c_0 = 0$
For $m = 1, 2, \ldots, M$
    For $n = m, m+2, .., m+2\left\lfloor \frac{N-m}{2} \right\rfloor$
        $\delta_{nm} = [(\Phi_{nm} - (m-1)\hat{\alpha}_{m-1}] \bmod 2\pi$
        $w_{nm} = \frac{|Z_{nm}^{r}| + |Z_{nm}^{t}|}{2}$
    End
    $s_m = \sum_{k=0}^{\left\lfloor \frac{N-m}{2} \right\rfloor} \frac{w_{m+2k,m}}{m}$
    $\delta_m = \frac{1}{s_m} \sum_{k=0}^{\left\lfloor \frac{N-m}{2} \right\rfloor} \frac{w_{m+2k,m}}{m} \delta_{m+2k,m}$
    $\hat{\alpha}_m = \frac{1}{c_{m-1} + s_m}(c_{m-1}\hat{\alpha}_{m-1} + s_m\delta_m)$
    $c_m = c_{m-1} + s_m$
End
$\hat{\alpha} = \hat{\alpha}_M$
```

3.4 Experimental Results for Performance Evaluation

We will examine the system performance with respect to important system parameters including (1) region scene types, (2) region descriptor types, (3) region detector types, (4) region overlap error, and (5) transformation types. The region scene types under consideration are the structured and textured scenes. The test images available at the website [30], plus some new images, are used in the experiments. The transformation types considered here contain the common photometric transformations (blur, illumination, noise, and JPEG compression) and geometric transformations (rotation, scaling, translation, and viewpoint). Fig. 3.3 shows the representative test image pairs taken for the textured and structured scenes.

In regard to the region descriptor types we include the proposed ZM phase and five popular descriptors: SIFT, GLOH, PCA-SIFT, steerable filters, and complex moments. In the beginning of the experiment, we need to choose a region detector in order to extract the regions of interest from the given image. Here we decide to choose either MSER detector or Hessian-affine detector. Once the region detector type is decided, the program codes available at the website [30] are used to obtain (a) regions of interest, (b) the dominant orientation in a region image and (c) the descriptor feature vectors of SIFT, GLOH, PCA-SIFT, steerable filter and complex moment for each region of interest. Then we run our program codes to generate our ZM phase descriptor and to calculate the similarity measures and generate the precision-recall curves to evaluate the descriptor performances, as done in [12]. Totally, there are 8 types of transformations, 2 types of scenes, and at least 4 image pairs for each transformation. On the average, one image pair generates 250,000 (= 500×500) region pairs for matching. All together the experiments involve more than 15 million region pairs.

Table 3.2 lists the typical feature vector dimensions of the six descriptors used in the experiments. Later, a discussion on the feature dimensionality will be provided.

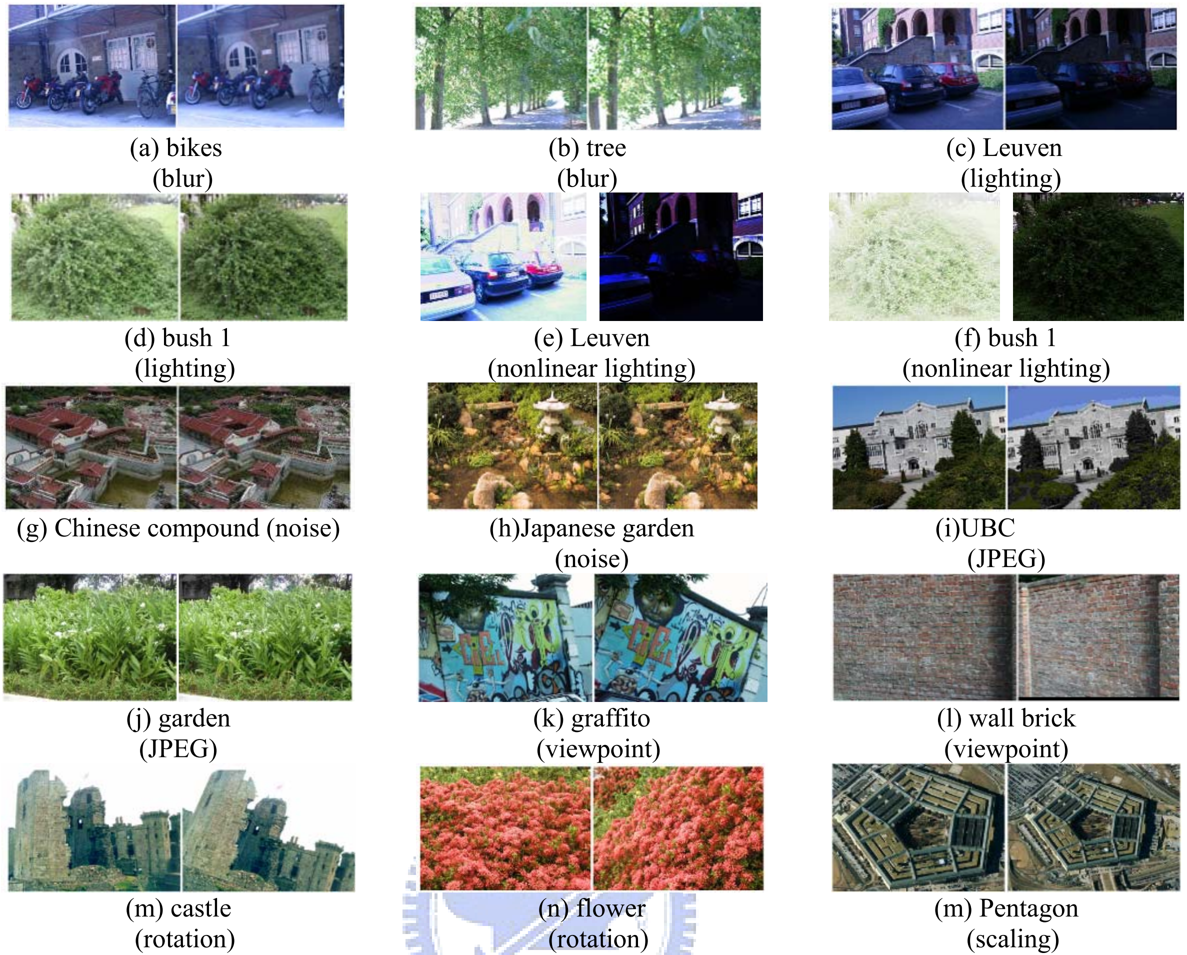

(a) bikes (blur)
(b) tree (blur)
(c) Leuven (lighting)
(d) bush 1 (lighting)
(e) Leuven (nonlinear lighting)
(f) bush 1 (nonlinear lighting)
(g) Chinese compound (noise)
(h)Japanese garden (noise)
(i)UBC (JPEG)
(j) garden (JPEG)
(k) graffito (viewpoint)
(l) wall brick (viewpoint)
(m) castle (rotation)
(n) flower (rotation)
(m) Pentagon (scaling)

Fig. 3.3: Representative test image pairs taken from the textured and structured scenes under a specified photometric or geometric transformation.

TABLE 3.2 THE TYPICAL FEATURE VECTOR DIMENSIONS OF THE SIX DESCRIPTORS

Descriptor	SIFT	GLOH	ZM phase	PCA- SIFT	complex moments	steerable filters
Feature dimensionality	128	128	42	36	15	14

3.4.1 Performance Evaluation Criteria – PR curve

For region matching, the extracted regions of the reference and transformed images are examined for (a) their distance measure and (b) their spatial overlap error under the applied transformation. There are three strategies for region matching proposed in [12]: (a) the threshold-based matching, (b) the nearest-neighbor-based matching, and (c) two-nearest-neighbor-based matching. Although these three matching methods are functionally different, their ranking results of the performances of the various descriptors are virtually the same; the first one is generally recommended [12], [38]. Therefore, we adopt the threshold-based matching strategy in which the distance measure between a region pair is compared to a given distance threshold, D_t.

On the other hand, the region overlap error is represented by the overlap ratio between the region intersection area and the region union area under the known planar homography [12], [31], that is, $O_e = 1 - (A \cap H^T BH)/(A \cup H^T BH)$, where A and B are the two matching regions and H is the given homograph between the two region patches. A region pair is called a match if it passes the region similarity test, namely, the distance measure between the image pair does not exceed the distance threshold D_t; otherwise, no match is found. A match is said to be correct, if the region pair also passes the region overlap test given by $O_e < O_t$ for a given overlap error threshold O_t. A match is said to be false, if the pair fails the region overlap test. Sometimes, with a tight overlap error threshold, say $O_t = 0.1$, even though the two regions pass the region similarity test, but they fail the region overlap test due to $O_t<O_e<1$. It seems not very fair to call such a pair a false match when compared to a typical false match whose region overlap error O_e is equal to 1; namely, the two regions do not intersect and are, therefore, not related at all. Hereafter, a matching pair with a region overlap error in between such that $O_t<O_e<1$ is considered as a “don’t care” pair. In other words, the new definition of a false match is a match that passes the region similarity test and its region overlap error O_e must be equal to 1.

It is important to realize a fixed distance threshold cannot be used to evaluate the descriptor performances. Instead, a precision-recall (PR) curve, created by varying the distance threshold, must be used.

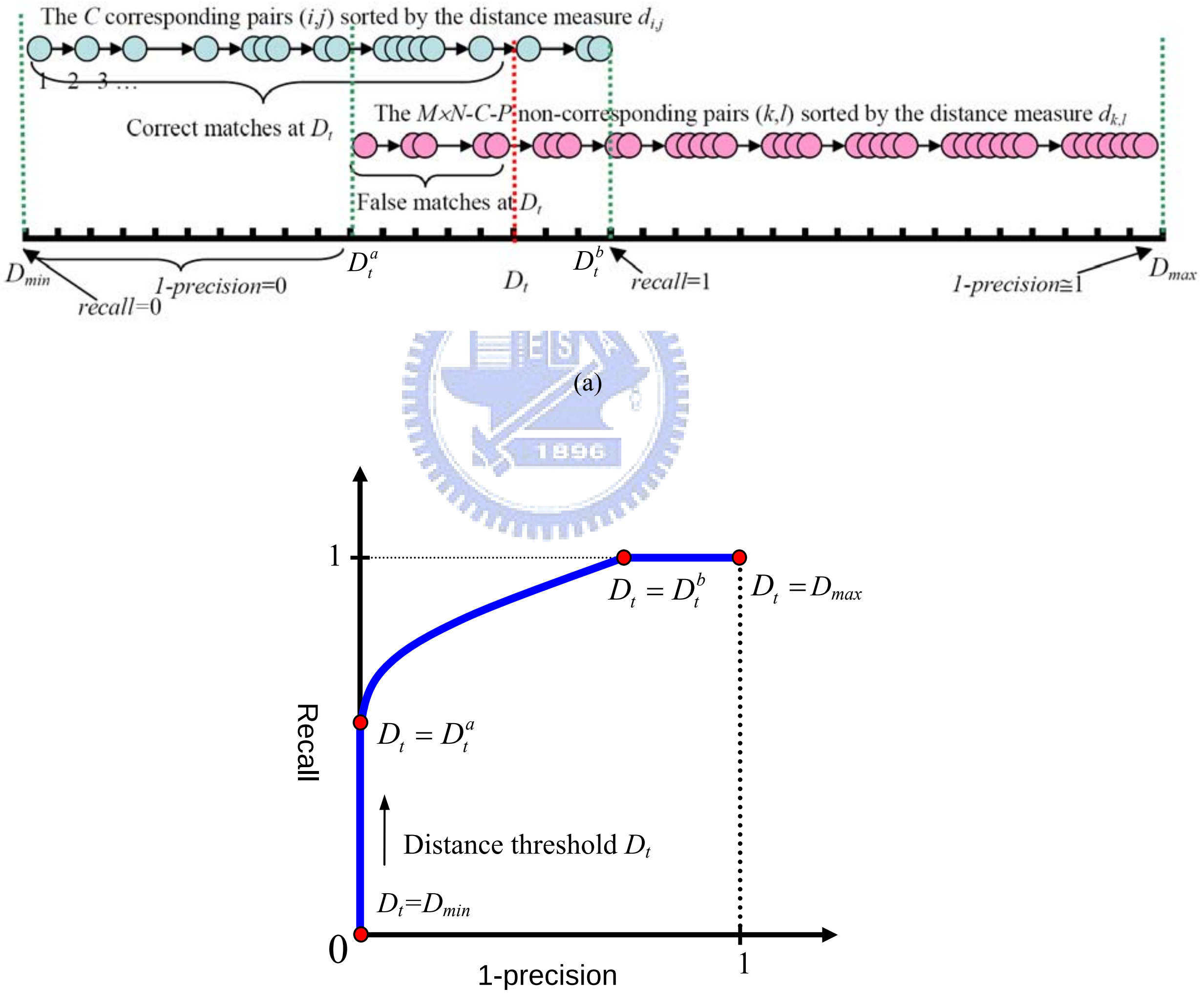
Recall is the ratio of the number of correct matches to the number of corresponding region pairs satisfying the region overlap test: $O_e < O_t$.

$$recall = \frac{\#\,\text{correct matches}}{\#\,\text{correspondences}}. \tag{3.15}$$

Precision is the ratio of the number of correct matches to the total number of correct and false matches:

$$1\text{-}\,precision = \frac{\#\,\text{false matches}}{\#\,\text{correct matches} + \#\,\text{false matches}}. \tag{3.16}$$

Fig. 3.4 depicts a PR curve generation process. Assume there are M, N regions detected in the reference and transformed images, respectively. The regions in the two images form $M \times N$ matching region pairs. Among these $M \times N$ pairs let the number of corresponding region pairs, which are each with a region overlap error O_e smaller than the specified bound O_t, be C. Also, let the number of the "don't care" pairs be P. Now sort the C corresponding pairs and the $M \times N\text{-}C\text{-}P$ non-corresponding pairs, respectively, by their distance measures $d_{i,j}$ in an ascending order. The range of distance measures for the set of C corresponding pairs generally overlaps with that of the set of non-corresponding pairs. Start to increase the distance threshold D_t from the minimum value D_{min} to the maximum value D_{max}. The recall value is initially equal to zero, so is the value of (1-precision). As D_t passes over D_{min}, more and more correct matches occur and the recall value is increasing, while the (1-precision) value remains 0 since there have been no false matches so far. When D_t reaches the minimum distance measure D_t^a of the non-corresponding region pairs, false matching pairs begin to appear and the value of 1-precision is increasing from 0. Notice that the recall is always monotonically increasing and reaches 1 when the distance threshold is equal to the maximum distance measure D_t^b of the C corresponding region pairs. At the end, when the distance threshold is equal to D_{max}, the (1-precision) value approaches 1. Be aware that the (1-precision) value is monotonically increasing when D_t is sufficiently large, but it may decrease at the early stage, if the relative growth rate of false matches is smaller than that of the correct matches.



(a)

(b)

Fig. 3.4: The PR curve generation process. (a) The correct matches and false matches associated with a varying distance threshold D_t. (b) The generated PR curve.

3.4.2 Evaluation on Region Detector Types and Region Overlap Error

As mentioned above, the best two region detectors, MSER and Hessian-affine, are reported in [11]. We shall present the evaluation results for these two detectors side by side.

Fig. 3.5 and Fig. 3.6 show the region detection results for both textured and structured scenes, and the two curves about the relation between recall and region overlap error and that between the number of correct matches and region overlap error using Hessian-affine regions and MSER regions, respectively. There are around 400 regions extracted by either detector. The number of correct matches and the number of correspondences for each overlap error are computed for a single section of overlap errors ranging from the previous one to the current one. For instance, the score for 20 percent is computed for the overlap error interval from 10 percent to 20 percent. Also, the recall values are calculated, by keeping the precision at 0.5, as done in [12].

We observe that the top black line, which shows the number of region correspondences dictated by the given overlap error bound O_e, bounces back at overlap error 40%. This is due to a natural increase in the region correspondences at the given higher region overlap error bound, resulting in “one-to-many” or “many-to-one” overlapped region pairs extracted from the reference and sensed scenes. Usually these new corresponding region pairs are less similar when compared to those at a smaller overlap error bound, causing a drop in the number of new correct matches. On the other hand, for a small overlap error bound the correspondences are mostly the “one-to-one” overlapped region pairs.

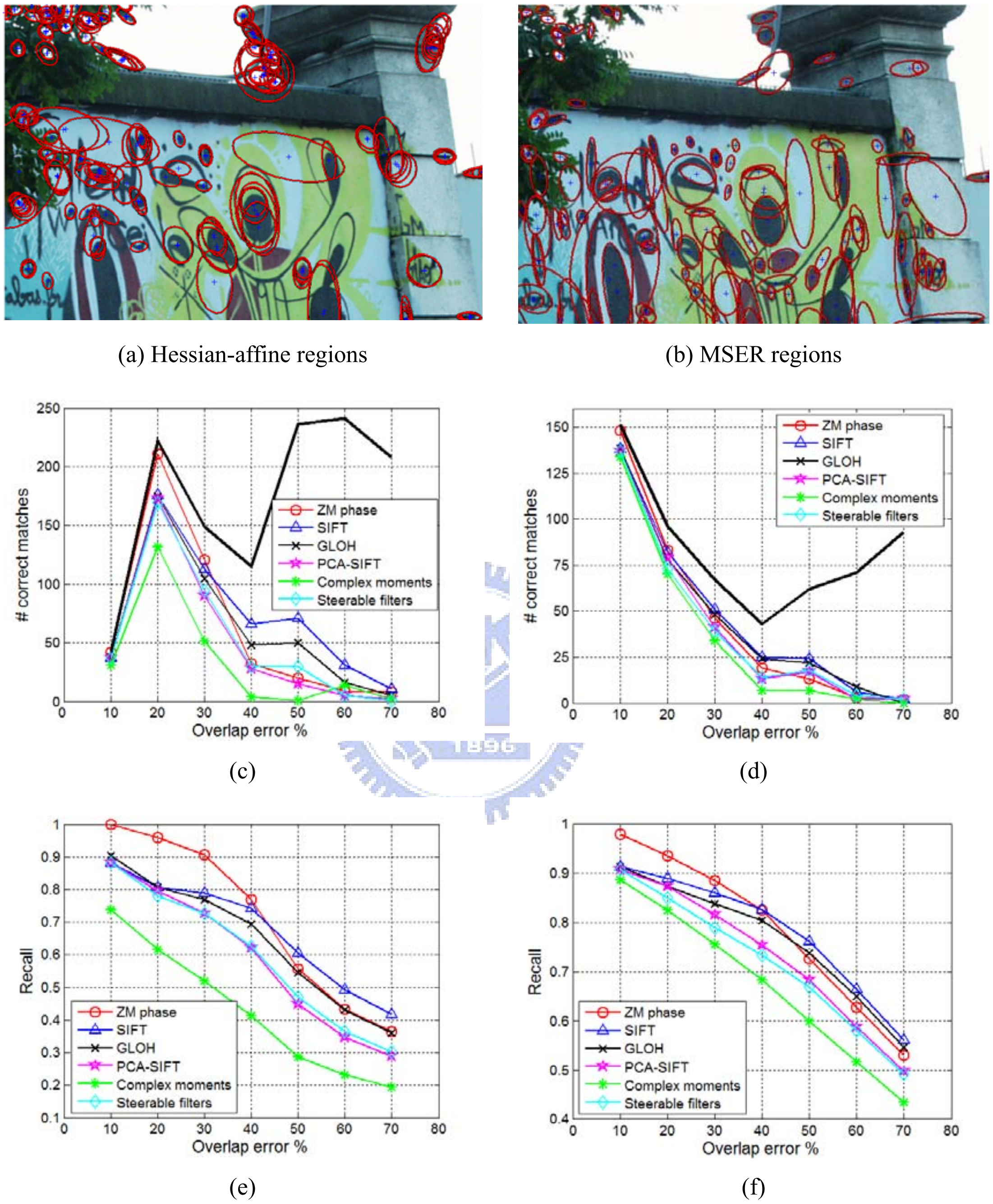


Fig. 3.5: Evaluation for different overlap errors for structured scene. (a)–(b) Detected Hessian-affine regions and MSER regions under viewpoint change for structured graffti scene. (c)-(d) The number of correct matches vs. the overlap error. Also, the top black line shows the number of region correspondences detected. (e)-(f) Recall vs. the overlap error.

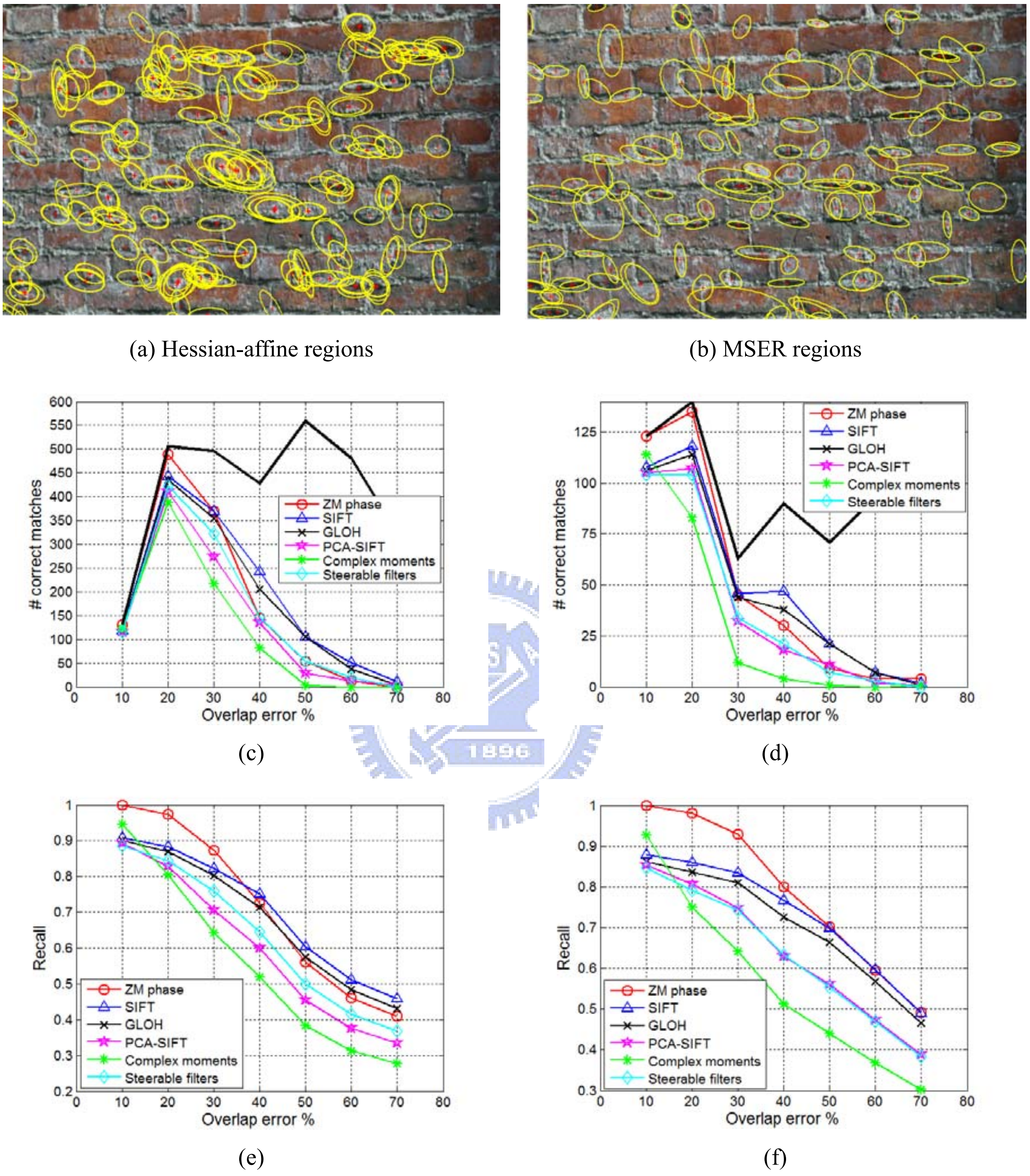
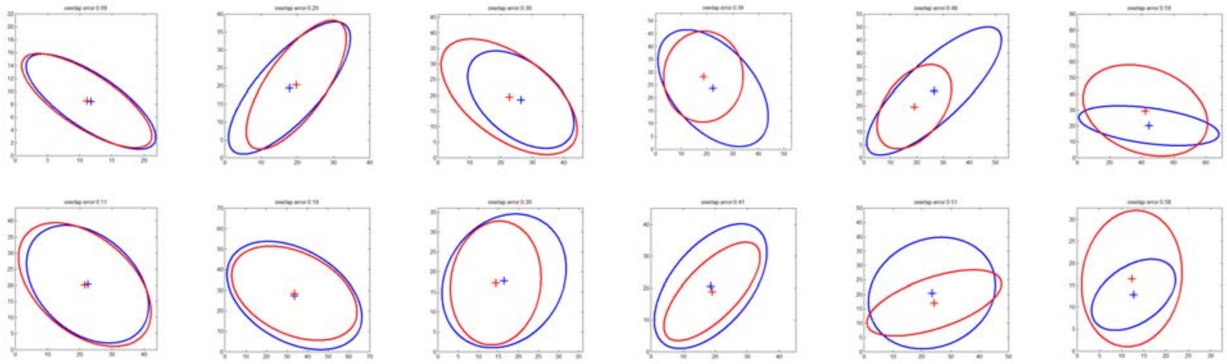


Fig. 3.6: Evaluation for different overlap errors for textured scene. (a)–(b) Detected Hessian-affine regions and MSER regions under viewpoint change for textured brick scene. (c)-(d) The number of correct matches vs. the overlap error. Also, the top black line shows the number of region correspondences detected. (e)-(f) Recall vs. the overlap error.

We observe that the proposed ZM phase descriptor has a higher recall vs. region overlap error curve than other descriptors for the region overlap error in the interval [0.1, 0.4] for both sets of Hessian-affine and MSER regions. The portion of curve is less meaningful when O_t gets larger. This is because when O_t gets larger, the corresponding regions are less similar, as indicated in Figs. 3.7. As mentioned above, when the overlap error bound increases over 0.4, the intersection area of these new corresponding region pairs becomes smaller, resulting in the drop of the number of correct matches and the decrease in the recall value under a fixed precision level (0.5 in this case). At a large overlap error bound the Zernike phase maintains the same tight control on the similarity matching of the new corresponding pairs based on the orthogonal moment features, so the increase in the new correct matches is rather small. On the other hand, SIFT and GLOH have less stringent control on the similarity measure based on the 8-gradient orientation bin tabulation on the 4×4 location grid, so there are more new correct matches when the overlap error bound increases.



$O_\varepsilon = 0.1$ $O_\varepsilon = 0.2$ $O_\varepsilon = 0.3$ $O_\varepsilon = 0.4$ $O_\varepsilon = 0.5$ $O_\varepsilon = 0.6$

Fig. 3.7: The examples of the detected region pairs with different overlap errors O_ε ranging from 0.1 to 0.6. The ellipses indicate the region boundary with blue color and red color for reference region A and the transformed region given by H^TBH, respectively. The cross symbols show the key point positions.

We should not bother considering the corresponding region pairs associated with a large overlap error bound, since many belong to "one-to-many" or "many-to-one" correspondences. The inclusion of these less similar pairs or outliers will result in the erroneous estimations in the later stages such as in the estimations of homography, fundamental matrix and epipolar geometry, etc. Therefore, we set the O_t value to 0.3 rather than 0.5 used in [12].

From now on, only MSER regions will be considered in the later experiments, since the descriptor performance characteristics are similar for MSER and Hessian-affine regions.

3.4.3 Evaluation on Transformation Types

Since the elliptical region is already normalized into a circular image, the normalized region is affine invariant. Nevertheless, the normalized region is not necessarily invariant to rotation. Thus, for most of the descriptors including SIFT, SIFT variants and the steerable filters, the image rotation problem must be solved first by finding a dominant gradient orientation. Similarly, the circular image intensity normalization has made the region descriptor robust to intensity scaling and offset, but not to image blur, image noise, image compression, and the illumination change.

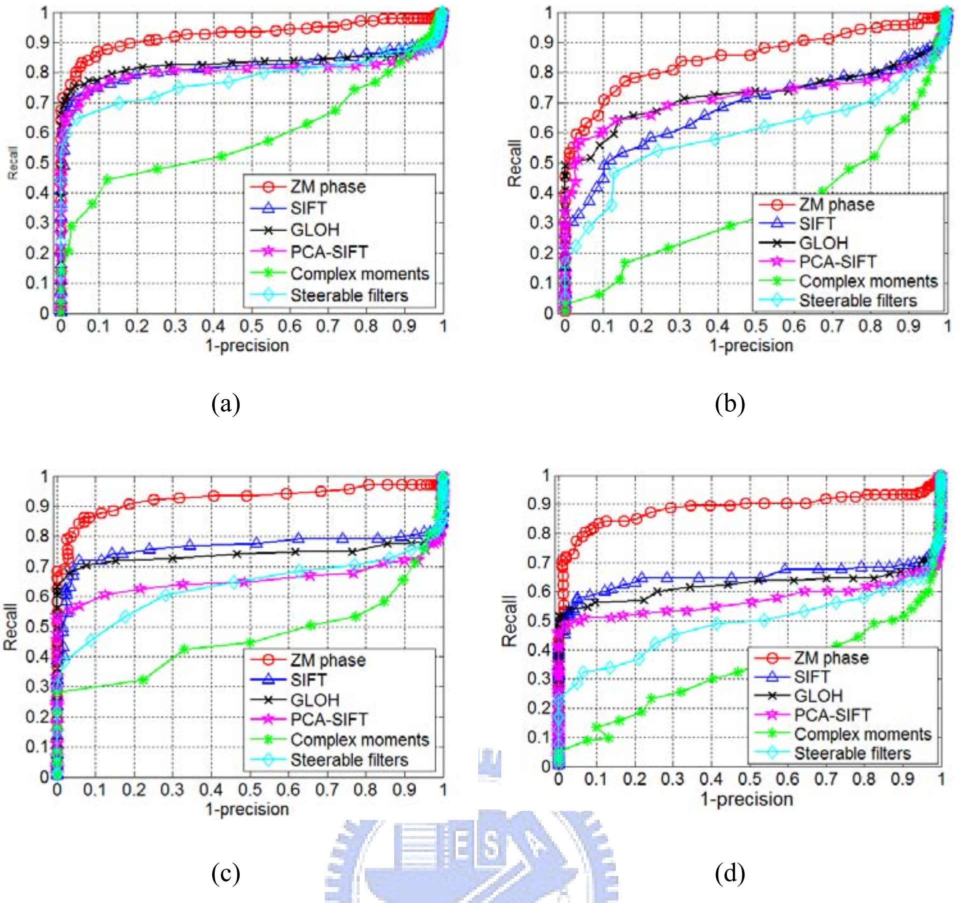
In image registration the two images can be taken by a single camera or different cameras, and the images can be taken during a short period or on different days. These shooting scenarios determine the type of image transformation encountered. For instance, if the two images are shot by different cameras or at different periods, the photometric conditions of the two shootings will be different, not to mention the possible viewpoint change. In general, a geometric transformation is accompanied by some sort of photometric change due to differences in the camera setting and the surface reflection angles.

A) Robustness under Photometric Transformations

To focus on the effects of photometric transformations, we try to avoid the effect of a geometric transformation by setting the region overlap error threshold O_t to a small value (0.2~0.3). Overall speaking, the ZM phase obtains the best performance results for all textured scenes under all type of photometric transformations and for the structured scenes under image blur and nonlinear lighting. The performances of the ZM phase, SIFT, GLOH and PCA-SIFT are comparable for the structured scenes under affine lighting change, image noise and JPEG when the value of 1 – precision is very small. The analysis on these performance results will be given later.

(*i*) *Image Blur*

The performance is measured under image blur introduced by changing the camera focus setting. Figs. 3.8(a)-3.8(b) show the respective PR curves for the bike structured scene (see Fig. 3.3(a)) with minor blur and severe blur, while Figs. 3.8(c)-3.8(d) show the respective PR curves for the tree textured scene (see Fig. 3.3(b)) with minor blur and severe blur. The performance ranking indicates that the best descriptor is ZM phase for both the structured and textured scenes considered. On the other hand, SIFT performs better than its variants, GLOH and PCA-SIFT, for the textured scene, while its variant performs better for the structured scene, as reported in [12]. The last ranking position is the complex moments. This is because its low dimensional feature vector (15 in this case) and its exclusive use of the moment magnitudes without the phase information.



(a)

(b)

(c)

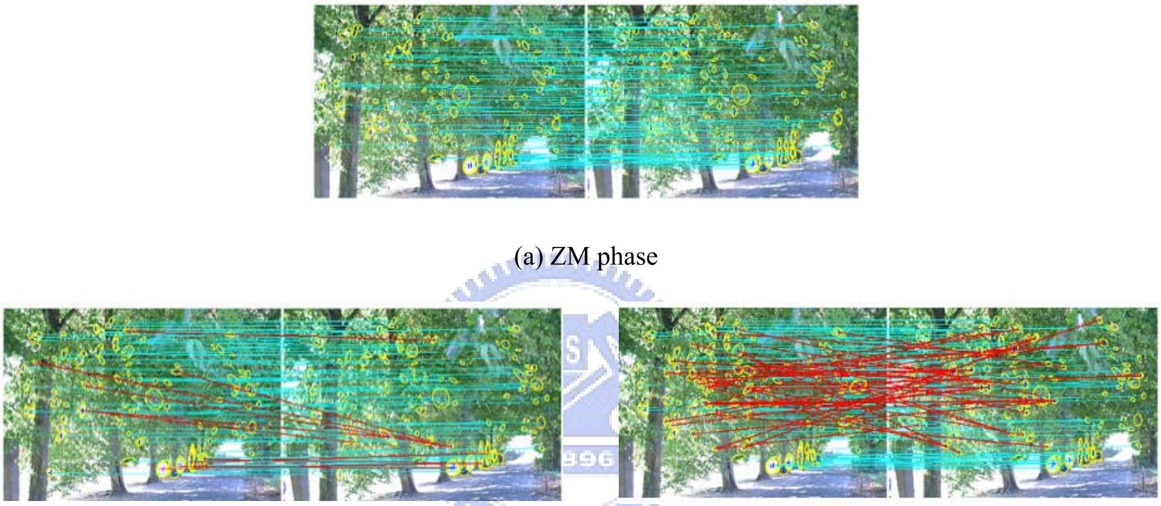
(d)

Fig. 3.8: The PR curves for the structured bike scene with (a) minor blur (b) severe blur. The PR curves for the textured tree scene with (c) minor blur (d) severe blur, all with O_t = 0.3.

To show the performance discrepancies between the top best three descriptors (ZM phase, GLOH and SIFT) under image blur, Table 3.3 shows the matching statistics for the bike structured scene and the tree textured scene with a region overlap error of 0.3 and a recall value of 0.6. Fig. 3.9 depicts the correct and false region matches for the tree textured scene, when using ZM phase, GLOH and SIFT, respectively. There are 0, 11 and 42 false matches (shown by red lines) for ZM phase, SIFT and GLOH, respectively. All these descriptors have 112 correct matches (shown by green lines).

TABLE 3.3 THE MATCHING STATISTICS FOR THE BIKE STRUCTURED SCENE AND TREE TEXTURED SCENE, ALL WITH O_t = 0.3 AND RECALL = 0.6.

Scene	# MSER		# correspondences	ZM phase			SIFT			GLOH		
	Left Image	Right image		Threshold D_t	# correct	# false	Threshold D_t	# correct	# false	Threshold D_t	# correct	# false
Structured (bikes)	449	387	161	0.167	97	4	0.183	96	35	1600	96	14
Textured (tree)	631	531	186	0.179	112	0	0.220	112	11	1543	112	42



(a) ZM phase

(b) SIFT

(c) GLOH

Fig. 3.9: The correct matches (in green) and false matches (in red) obtained by the descriptors, respectively, all with *recall* = 0.6 and O_t = 0.3.

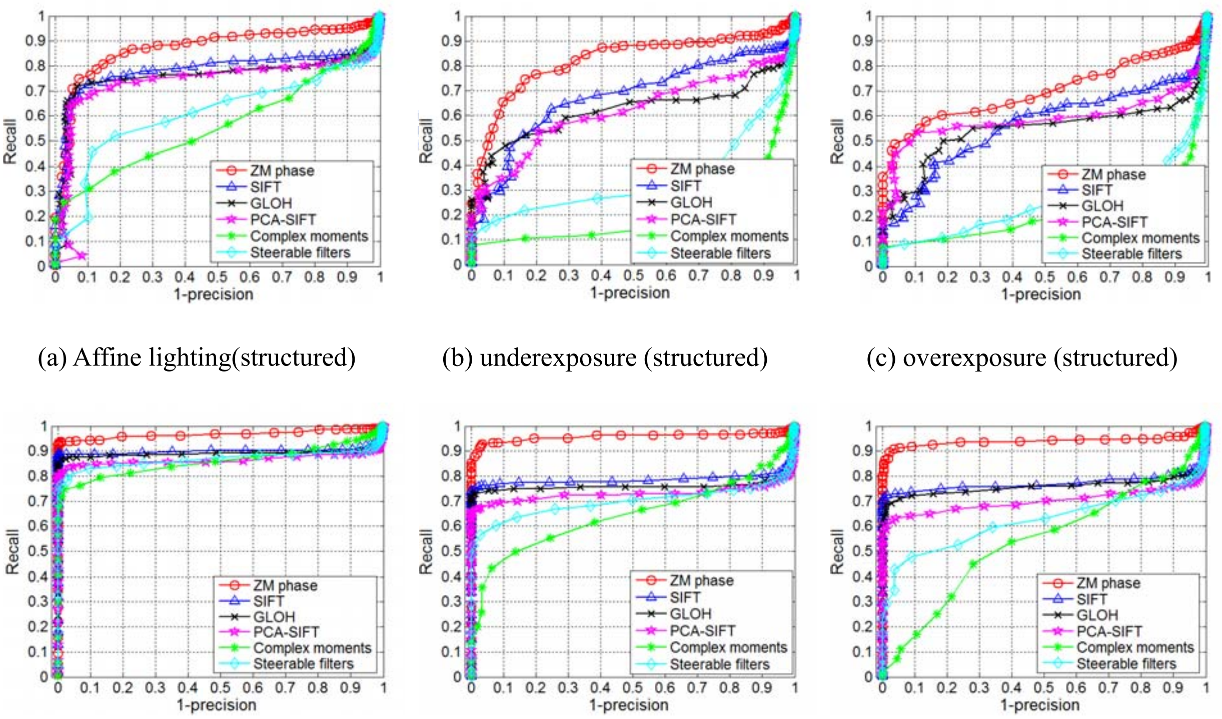
(*ii*) *Illumination Change*

a) *Affine Lighting Change*

To evaluate the descriptor performances under illumination changes, a collection of images has been taken by changing the camera iris settings. Figs. 3.10(a) and 3.10(d) show the PR curves for the Leuven structured scene and the bush 1 textured scene shown in Figs. 3.3(c) and 3.3(d), respectively. The best three descriptors in order are ZM phase, SIFT, and GLOH for the bush 1 textured scene and the situation remains the same for the structured scene except when the value of 1 – precision is less than 0.03.

b) *Nonlinear Lighting Change*

The nonlinear lighting is quite common in practice. Figs. 3.10(b) and 3.10(c) shows the PR curves under the underexposure and overexposure lighting for the Leuven structured scene shown in Figs. 3.3(e). Figs. 3.10(e) and 3.10(f) shows the PR curves under the underexposure and overexposure lighting for the bush 1 textured scene shown in Figs. 3.3(f). In comparison with the PR curves in Figs. 3.10(a) and 3.10(d) under affine lighting change, it can be seen that the performances of the SIFT-based descriptors become significantly worse. To the contrary, the performance results of the ZM phase have only a small change, especially in the case of the textured scene. This will be explained later.



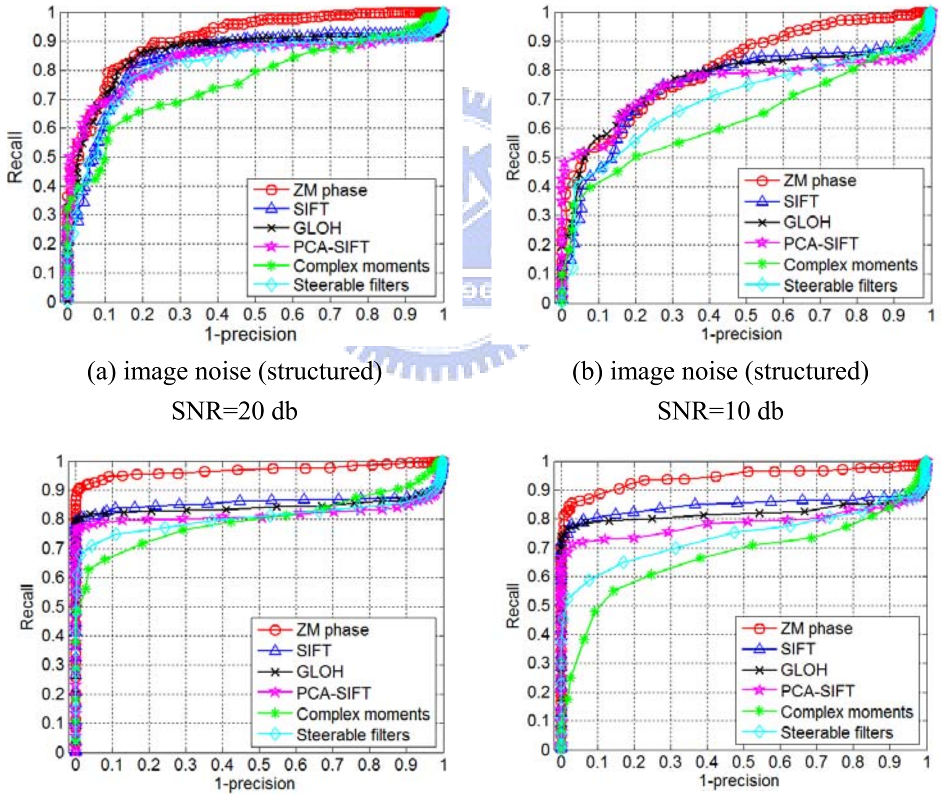
(a) Affine lighting(structured) (b) underexposure (structured) (c) overexposure (structured)

(d) Affine lighting(textured) (e) underexposure (textured) (f) overexposure (textured)

Fig. 3.10: The PR curves for the Leuven structured scene with (a) affine lighting change (b) non-linear lighting change (underexposure), (c) non-linear lighting change (overexposure). The PR curves for the bush 1 textured scene with (d) affine lighting change (e) non-linear lighting change (underexposure), (f) non-linear lighting change (overexposure), all with $O_t = 0.3$.

(*iii*) *Image Noise*

The performances are evaluated by adding a different amount of Gaussian noise to the images. Figs. 3.11(a) and 3.11(b) show the PR curve for the structured Chinese compound scene (see Fig. 3.3 (g)) with two different noise levels (SNR=20 and 10), respectively. Figs. 3.11(c)-(d) show the PR curve for the Japanese garden textured scene (see Fig. 3.3(h)). The ZM phase has the best overall result among all the descriptors for the textured scene and is comparable to the SIFT-based descriptors for the structured scene.



(a) image noise (structured) SNR=20 db

(b) image noise (structured) SNR=10 db

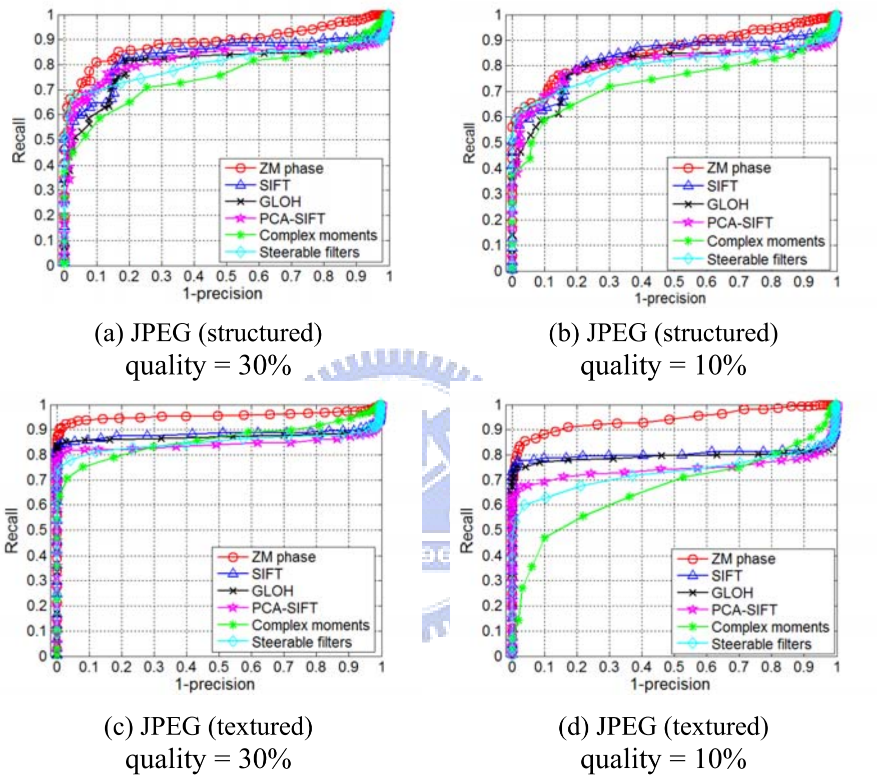
(c) image noise (textured) SNR=20 db

(d) image noise (textured) SNR=10 db

Fig. 3.11: The PR curves for the Chinese compound structured scene under image noise with (a) SNR=20 db, (b) SNR= 10 db. The PR curves for the Japanese garden textured scene under image noise with (c) SNR=20 db, (d) SNR=10 db, all with $O_t = 0.3$.

(*iv*) *JPEG Compression*

Figs. 3.12 depict the PR curves under JPEG compression for the structured UBC scene shown in Fig. 3.3(i) and the textured garden scene shown in Fig. 3.3(j), respectively. The qualities of the compressed images range from 10 to 30 percent of the original one. The performance ranking is similar to that under the noise attack.



(a) JPEG (structured)
quality = 30%

(b) JPEG (structured)
quality = 10%

(c) JPEG (textured)
quality = 30%

(d) JPEG (textured)
quality = 10%

Fig. 3.12: The PR curves for the structured UBC scene under JPEG compression with quality = (a) 30%, (b) 10%. The PR curves for the textured garden scene with quality = (c) 30%, (d) 10%, all with $O_t = 0.3$.

B) Robustness under Geometric Transformations

To focus on the effects of geometric transformations, we try intentionally not to change the photometric conditions. As shall be seen, under all geometric transformations, the ZM phase performs best for all textured scenes, but is comparable to the SIFT-based descriptors for the structured scenes when the value of 1 – precision is less than 0.05.

(*i*) *Viewpoint Change*

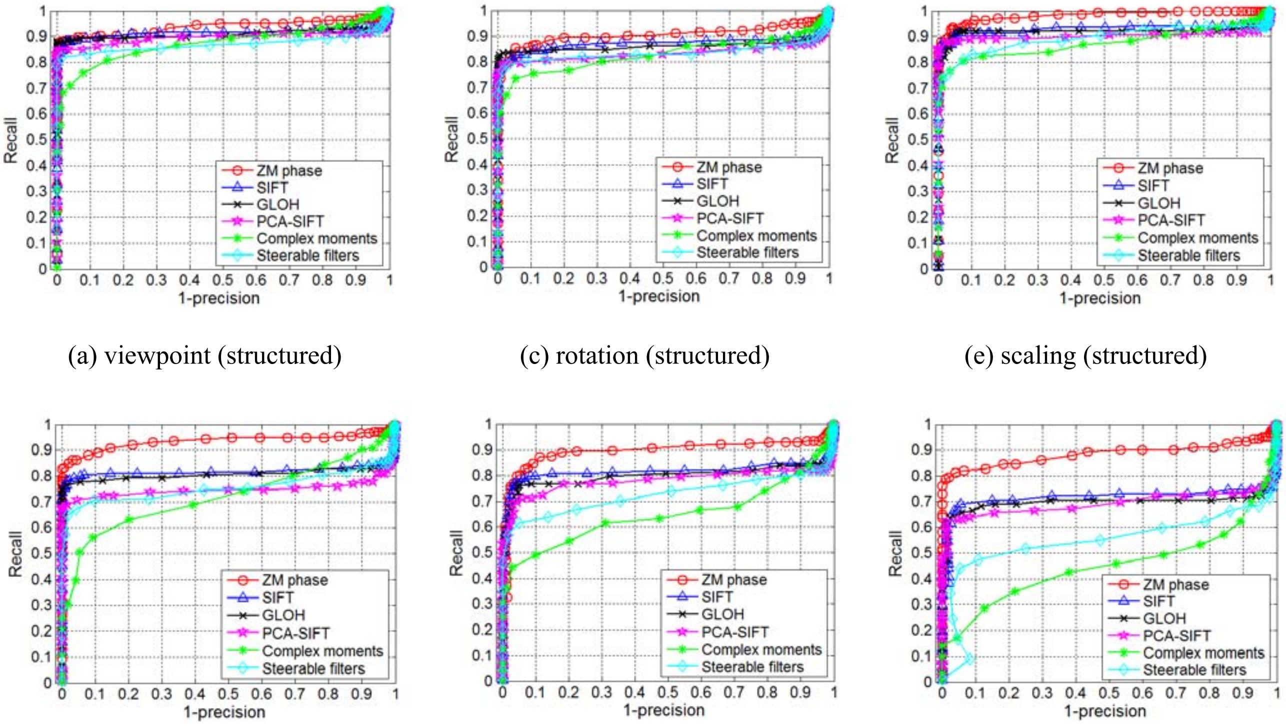
We use six images of the textured and structured scenes taken under a viewing angle ranging from 10 to 50 degrees. Figs. 3.13(a) and 3.13(b) give the PR curves for structured graffiti scenes (see Fig. 3.3(k)) and the textured brick scenes (see Fig. 3.3(l)), respectively. The ranking of the best four descriptors remain unchanged for the specified range [10^{o}, 50^{o}] of the viewing angle. The ZM phase descriptor clearly overpowers the five other descriptors for the textured scene, but not so for the structured scene.

(*ii*) *Rotation Change*

The images considered are taken by rotating the camera axis from 30^{o} to 45 ^{o}. The descriptors for the structured castle scene (Fig. 3.3(m)) and the flower textured scene (Fig. 3.3(n)) under image rotation are evaluated. Figs. 3.13(c)-3.13(d) show the PR curves for the scenes, respectively. The ranking of the top three descriptors remains the same throughout the range of rotation angle and it is similar to the case of viewpoint change.

(*iii*) *Scale Change*

Figs. 3.13(e)-3.13(f) show the performance measures for the descriptors under the scale change using the Pentagon structured scene (Fig. 3.3(m)) and textured bush 2 scene (Fig. 3.3(n)), respectively. The scaling factor is close to 2. The performance rankings are similar to the above two cases of geometric transformations.



(a) viewpoint (structured)

(c) rotation (structured)

(e) scaling (structured)

(b) viewpoint (textured)

(d) rotation (textured)

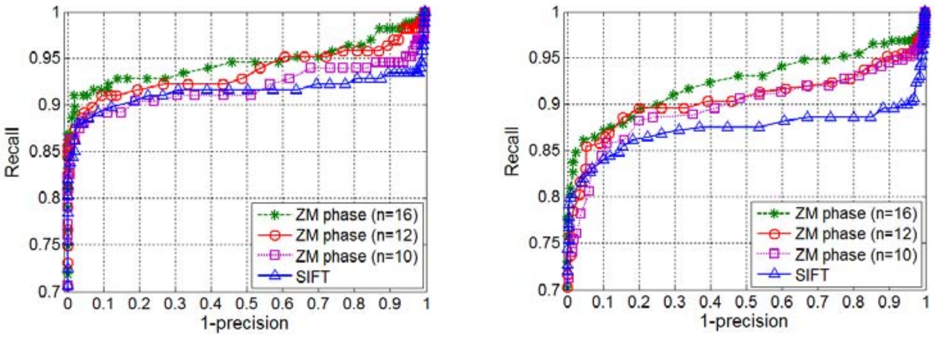
(f) scaling (textured)

Fig. 3.13: The PR curves under geometric transformation, all with $O_t = 0.3$

3.4.4 Evaluation on Feature Dimensionality

To extend the SIFT descriptor both GLOH and PCA-SIFT increase the feature size and then apply PCA to reduce the feature dimensionality. The features of these descriptors are originally correlated and become orthogonal after the application of PCA. However, their optimal dimensions are determined by the training images in the database.

The utilization of Zernike moments up to a higher order generally leads to a more accurate estimate of the region rotation angle and a better image representation power. Fig. 3.14 depicts the PR curves for two structured scenes under two different attacks when the ZM descriptor uses moments of order N up to 10, 12, and 16, respectively. The corresponding feature dimensions are 30, 42, and 72. It can be seen that the descriptor performance becomes better as the feature dimension gets increased. The selection of order N = 12 is a tradeoff between the computational complexity and the descriptor performance.



(a) graffito scene (viewpoint change) (b) castle scenes (rotation change)

Fig. 3.14: The PR curves for ZM phase with the maximum order N = 10, 12 and 16, together with the associated PR curves of SIFT for two structured scenes under two different attacks, all with O_t = 0.3.

3.5 Analysis on Performance Evaluation Discrepancies and Time Complexity Analysis

Since the complex moments and the steerable filters are never ranked in the first position throughout the experiments due to their low feature dimensions chosen, they will be ruled out for further consideration. The SIFT, GLOH, and PCA-SIFT have similar performance results under all the transformations reported. In the following, it is sufficient to compare the performances of SIFT and the ZM phase.

A) The Effect of Image Intensity Fluctuation on the Descriptor Performance

We give a rule of thumb or a simplified explanation why the ZM phase descriptor performs better than other existing descriptors under non-uniform image intensity fluctuation, since an exact analysis varies with the underlying image and, therefore, is rather complicated. First of all, the transformed image is obtained from the reference image according to a given photometric or geometric transform, so their image pattern structures are correlated. After the affine intensity normalization, their image intensity distributions become closer and tangled. Next, the phase difference of the ZM phase descriptor is computed as

$$\Delta\varphi_{nm} = \varphi_{nm}^{tran} - \varphi_{nm}^{ref} = \tan^{-1}\left(\frac{\operatorname{Im}(Z_{nm}^{tran})}{\operatorname{Re}(Z_{nm}^{tran})}\right) - \tan^{-1}\left(\frac{\operatorname{Im}(Z_{nm}^{ref})}{\operatorname{Re}(Z_{nm}^{ref})}\right), \tag{3.19}$$

where $\frac{\operatorname{Im}(Z_{nm}^{tran})}{\operatorname{Re}(Z_{nm}^{tran})} = \frac{\operatorname{Im}(Z_{nm}^{ref}) + \Delta\operatorname{Im}(Z_{nm})}{\operatorname{Re}(Z_{nm}^{ref}) + \Delta\operatorname{Re}(Z_{nm})}$ with $\Delta\operatorname{Re}(Z_{nm})$ and $\Delta\operatorname{Im}(Z_{nm})$ being the real and imaginary ZM components of the difference image between the reference and transformed images. Since the image structures of the transformed and reference images are similar, so it is likely that the phase angles of the reference and transformed images are in phase (*i.e.*, no phase difference after the image rotation alignment), especially when their ZM magnitudes are both large. The weighted sum of the absolute phase differences is, therefore, close to zero. On the other hand, the probability that the reference and transformed images are out of phase (a significant phase difference) is small. Consequently, most of the ZM moment counterparts of the image pair support the single majority of the estimated rotation angle, even though there is some fluctuation in the ZM magnitudes. This leads to the accurate rotation angle estimation when using the ZM phase.

On the other hand, the SIFT based methods utilize the gradient information. The local gradient angles in the transformed image remain considerably unchanged (except under image blur which causes the gradient angles damaged), but their gradient magnitudes change somewhat non-uniformly. Besides, there are generally several different gradient angles found in an image especially for the textured image. (This may not be the case for structured scenes with a distinguished edge orientation.) Therefore, the 36-bin orientation histogram will contain multiple candidates on the histogram ballot. When the gradient magnitudes change non-uniformly, the vote counting of the multiple candidates will change. This leads to a change of the dominant orientation in the transformed image. It, in turn, triggers further non-linear changes in the 128 dimensional SIFT feature vector, regardless of the unit length feature vector renormalization at the end. This is why the performance of the SIFT based methods generally degrades under a given transformation especially for the textured scenes. We shall use an example to justify our above reasoning.

Figs. 3.15 to 3.18 present four experimental results for the performance comparison between ZM phase and SIFT under non-linear lighting change (a power-law (gamma) transform with gamma = 3), JPEG compression (the quality of the transformed image is 5 percent of the reference one), viewpoint change and scaling change, respectively. The four figures are in the same format. Part (a) of the figures shows the region pair before and after affine intensity normalization in the gray color or in the pseudo color for better visualization, along with their difference images and difference intensity histograms. We can observe that the image structure of the transformed and difference images look similar to that of the reference image. This likely leads to the nearly equal real and imaginary parts of the ZM moments for the region pair except for a few components under the non-uniform intensity change, as indicated in part (b) of the figures. Therefore, the majority of the weighted phase differences are nearly zero, as shown in part (c) of the figures. On the other hand, the non-uniform intensity fluctuation causes the dominant orientation histogram and the 128 dimensional SIFT feature vectors to change non-uniformly, resulting in an expected greater dissimilarity between the two images shown in part (d) of the figure, as expected.

Non-linear lighting

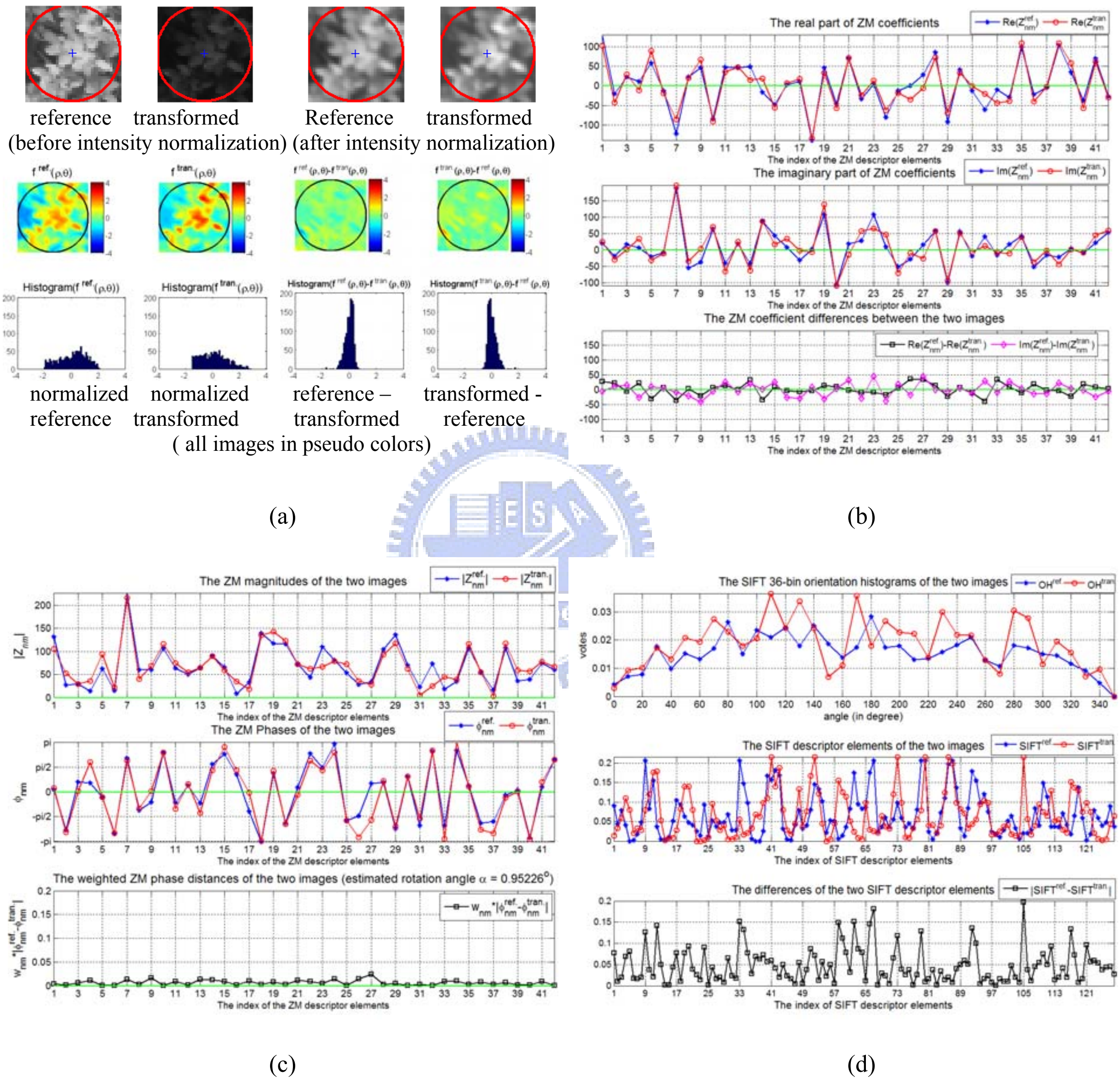
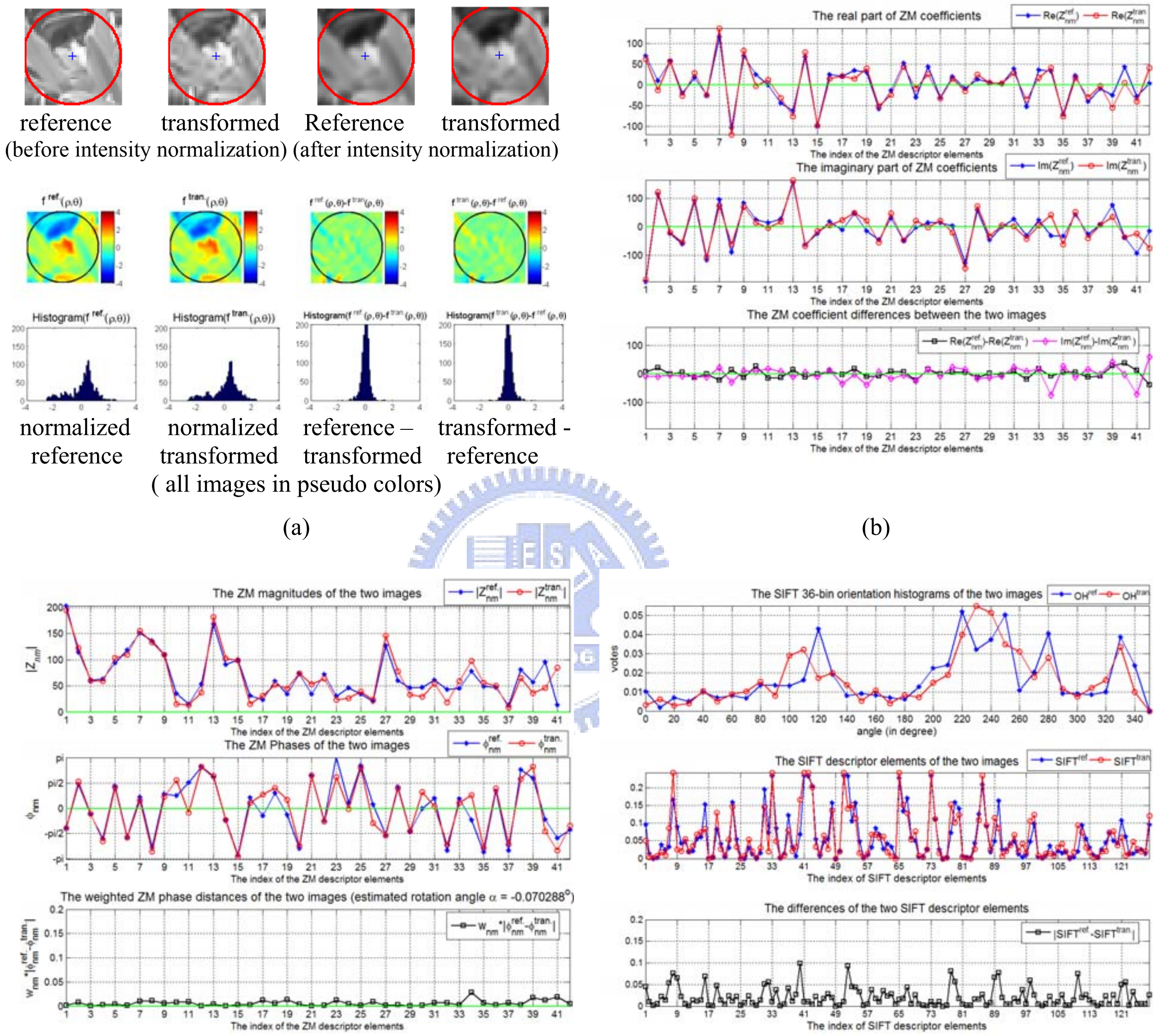


Fig. 3.15: A performance comparison of ZM phase and SIFT under non-linear lighting change. The detected ellipse-shaped regions are normalized to a circular patch through the affine normalization process beforehand.

JPEG compression

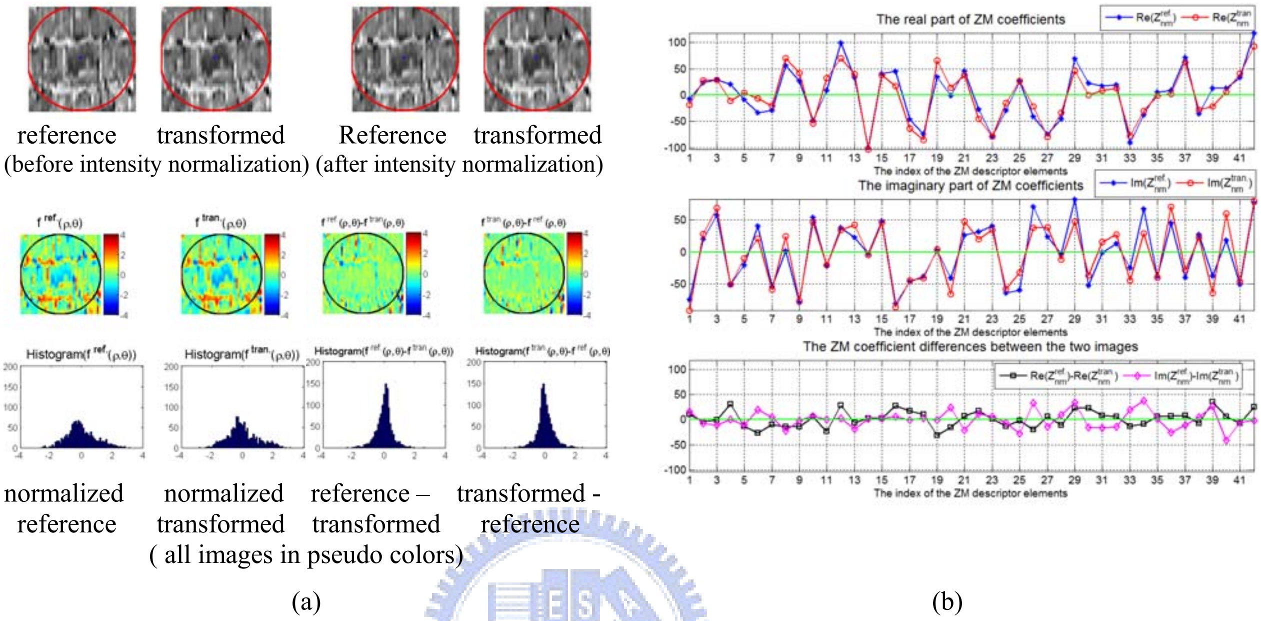


(a) (b)

(c) (d)

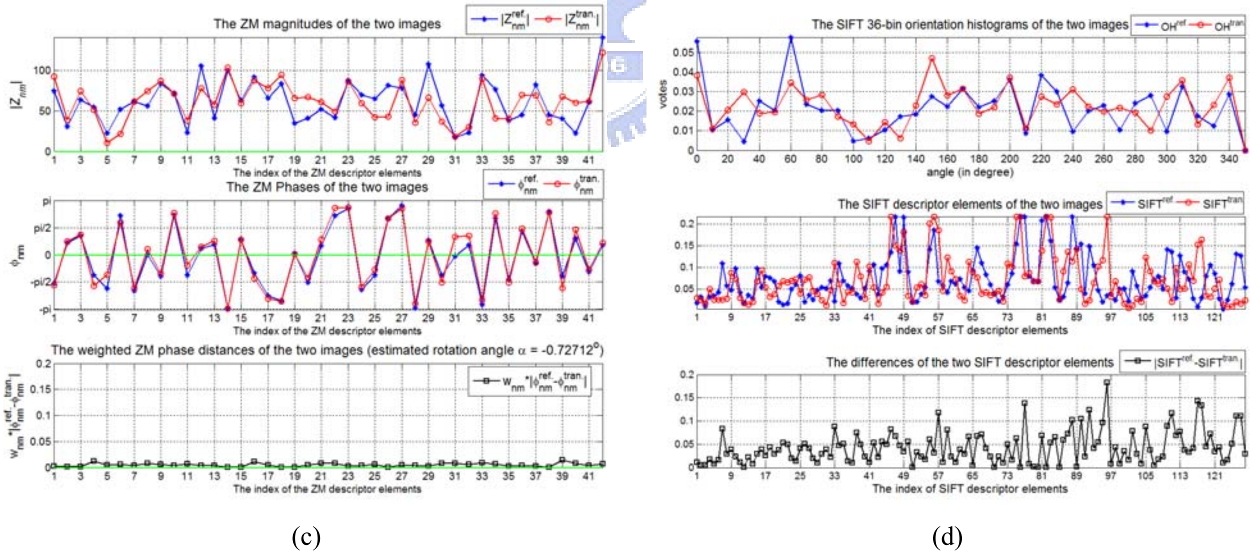
Fig. 3.16: A performance comparison of ZM phase and SIFT under JPEG compression. The detected ellipse-shaped regions are normalized to a circular patch through the affine normalization process beforehand.

Viewpoint change



(a)

(b)

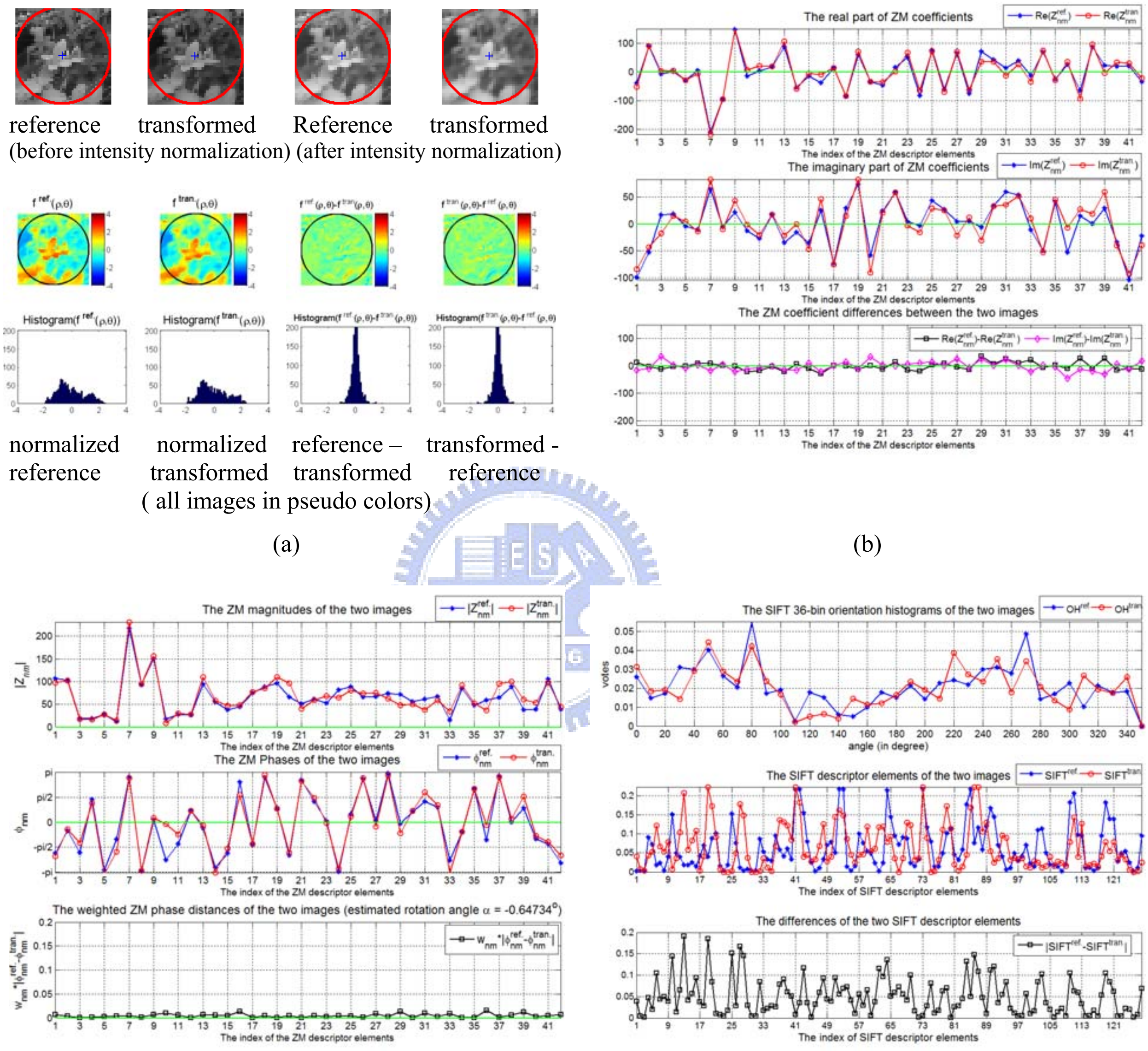


(c)

(d)

Fig. 3.17: A performance comparison of ZM phase and SIFT under Viewpoint change. The detected ellipse-shaped regions are normalized to a circular patch through the affine normalization process beforehand.

Scaling



(a)

(b)

(c)

(d)

Fig. 3.18: A performance comparison of ZM phase and SIFT under scaling change. The detected ellipse-shaped regions are normalized to a circular patch through the affine normalization process beforehand.

In summary, noise, lighting change, compression, and blurring belong to the photometric transformation type which causes the image intensities to vary. On the other hand, viewpoint change, scaling and rotation belong to the geometric transformation type which first relocates the positions of the image points, and then requires some sort of intensity interpolation to compute the image intensities at the new image points; the new image intensities contain some non-uniform fluctuation (except for the rotation transformation which generally causes a very minor intensity fluctuation). We can apply the above-mentioned reasoning to conclude the ZM phase descriptor is generally more robust than the SIFT-based methods under these transformations especially for the textured scenes which generally containing the complex edge orientation information.

B) Rotation Angle Error Statistics and Its Effect on the Descriptor Performance

The descriptor performance discrepancy can be attributed to the different rotation angle estimation errors of the descriptors. The dominant orientation of the SIFT based descriptors relies on the peak detection in the 36-bin histogram of the gradient directions obtained from the region image, while the ZM phase descriptor computes the image rotation angle from the weighted sum of the ZM phase differences. Table 3.4 breaks down the estimated rotation angle errors (ε_{angle}) under the categories of 5, 10, 20, and 30 degrees for both textured scenes and structured scenes under all transformations except the viewpoint change. The rotation angle errors are evaluated by computing the estimated rotation angle for all normalized corresponding region pairs, and then compare them with respect to the actual angle. The actual angle can be obtained by the ground truth homographies given from [30], which are almost a similarity transform. The rotation angle error statistics are not available under the viewpoint change, since the associated rotation angle between two regions under viewpoint change is not fixed.

TABLE 3.4 THE ROTATION ANGLE ESTIMATION ERRORS FOR ALL CORRESPONING REGION PAIRS SPECIFIED BY $O_t = 0.3$.

Transform type	Scene type	method	$\varepsilon_{angle} < 5^o$		$\varepsilon_{angle} < 10^o$		$\varepsilon_{angle} < 20^o$		$\varepsilon_{angle} < 30^o$	
			Avg	%	Avg	%	Avg	%	Avg	%
blur	Textured (tree)	ZM	1.801	86.022	2.333	97.312	2.502	98.925	2.502	98.925
		SIFT	2.484	39.785	4.134	59.140	6.090	72.581	7.787	80.108
	Structured (Bikes)	ZM	1.663	92.547	2.027	98.758	2.162	100	2.162	100
		SIFT	2.147	51.553	3.605	72.671	4.543	80.124	5.080	82.609
affine lighting	Textured (bush 1)	ZM	0.764	96.755	0.947	99.705	0.979	100	0.979	100
		SIFT	1.747	68.437	2.724	84.366	3.366	89.676	3.426	89.971
	Structured (Leuven)	ZM	1.506	93.662	1.641	98.775	2.115	100	2.115	100
		SIFT	1.726	62.676	2.631	78.873	3.313	83.803	4.071	86.620
non-linear lighting (underexposure)	Textured (bush 1)	ZM	1.099	94.561	1.284	97.908	1.363	98.745	1.363	98.745
		SIFT	2.002	53.556	3.115	69.874	4.327	79.498	4.532	80.335
	Structured (Leuven)	ZM	1.220	93.662	1.481	95.592	1.584	99.296	1.715	100
		SIFT	1.906	63.380	2.944	80.282	3.308	83.803	3.463	84.507
noise	Textured (Japan garden)	ZM	1.564	92.857	1.838	98.352	1.893	98.901	1.893	98.901
		SIFT	2.423	42.857	4.071	65.385	5.859	80.121	6.765	83.516
	Structured (Compound)	ZM	1.349	93.293	1.666	99.085	1.763	100	1.763	100
		SIFT	1.781	69.207	2.814	85.671	3.377	90.244	3.377	90.244
JPEG	Textured (garden)	ZM	1.318	93.817	1.654	100	1.654	100	1.654	100
		SIFT	2.107	50.269	3.722	73.387	4.948	83.871	5.272	85.215
	Structured (UBC)	ZM	1.112	93.158	1.326	96.842	1.552	98.947	1.552	98.947
		SIFT	1.852	68.947	2.724	82.632	3.162	86.316	3.419	87.368
Rotation	Textured (flower)	ZM	1.310	97.692	1.370	99.231	1.370	99.231	1.370	99.231
		SIFT	2.346	54.483	3.910	81.379	4.662	89.655	4.973	91.034
	Structured (castle)	ZM	1.061	98.755	1.117	100	1.117	100	1.117	100
		SIFT	1.777	74.274	2.544	87.552	2.963	91.286	2.963	91.286
Scaling	Textured (bush 2)	ZM	1.414	92.623	1.625	97.541	1.840	99.180	1.840	99.180
		SIFT	2.222	53.279	3.519	70.492	4.340	76.230	5.390	80.328
	Structured (Pentagon)	ZM	0.913	98.551	0.999	100	0.999	100	0.999	100
		SIFT	1.356	78.261	2.154	90.580	2.529	93.478	2.529	93.478

From Table 3.4 the average rotation angle errors of the ZM phase is smaller than those of SIFT for the structured scenes and textured scenes when $\varepsilon_{angle} < 30^o$. More importantly, the coverage percentage is more than 86% for ZM phase and around 40% to 78% for SIFT when $\varepsilon_{angle} < 5^o$. The coverage percentage is computed as the ratio between the number of region pairs with rotation angle estimation error (ε_{angle}) less than a specific value (ε_t = 5°, 10°, 20° or 30° in Table IV) and the total number of correspondence:

$$\text{coverage percentage}=\frac{\#\text{ corresponding pairs with }\varepsilon_{angle}<\varepsilon_t}{\#\text{ correspondences}}. \tag{3.18}$$

The large rotation angle errors of SIFT are due to the big error caused by the ambiguity in the multiple dominant orientation peaks. This is the main reason why the SIFT performance becomes poor.

Lowe [7] suggested solving the multiple dominant orientation problem by creating multiple keypoints at the same location but with one of the dominant orientations (In this case there is no clear rule for counting the multiple keypoints as correct or false matches in generating the PR curves). In Fig. 3.19 the PR curves for the flower textured scene under image blur is plotted with the removals of region pairs with a rotation angle error no less than 10 °, 20 °, 30°, and 360°, respectively. The ZM phase performs better than SIFT for rotation angle errors not exceeding 20 °, 30°, and 360°, but not for the case of rotation angle errors <10°, where SIFT does not face the multiple dominant orientation problem, as described previously.

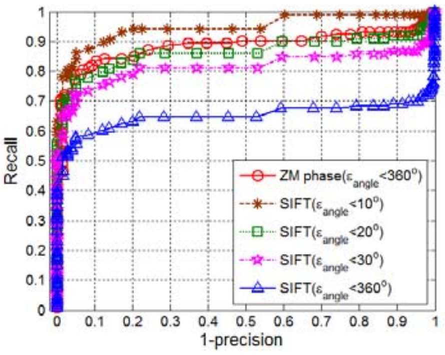


Fig. 3.19: The PR curves for tree textured scene under image blur with the removal of regions with a rotation angle error not exceeding a specified level of 10 °, 20 °, 30°, and 360°, respectively.

C) The Effects of Feature Dimensionality and Feature Orthogonality on the Descriptor Performance

Generally speaking, the high dimensional feature vector contains more descriptive information at the expense of memory space. For example, PCA-SIFT and GLOH start with a feature dimension of 3042 and 272, respectively. However, the components of these feature vectors are correlated and partially redundant. By the application of PCA (principal component analysis) a subset of eigenvectors associated with the larger eigenvalues can be extracted and the projection of the original feature vector to the sub-eigenspace reduces the original dimension down to 128 or even smaller. The dimensionality reduction can be determined based on the percentage of the sum of eigenvalues retained.

We know the ZM phase applies a set of orthogonal ZM moments to design the feature vector such that the feature components are mutually independent and more informative. With the same dimensionality (or the same memory space) the set of orthogonal features generally results in a better descriptive power to distinguish the different image patterns embedded in the textured scenes. However, when the image patterns in the scenes are highly similar, it require a higher feature dimensionality in order to reflect the subtle pattern difference, as indicated previously in Fig. 3.14.

D) Time Complexity Analysis

The computation time for evaluating the descriptor performance consists of the region extraction time, the descriptor feature vector construction time and the region matching time. Because all descriptors use the same set of regions of interest detected, so their region extraction times are the same. As for the feature vector construction time, the numbers of multiplications and additions required to compute Zernike moments up to order N for a $q \times q$ image patch are both of order $O\ (N^{2} q^{2})$ [40] . However, this calculation can be speeded up by using the symmetrical properties of Zernike basis functions [41], or achieve in real time performance by using special hardware accumulation grid architecture [42]. As for the region matching including the rotation angle estimation, the numbers of multiplications and additions required by the ZM phase descriptor are both of order $O\ (N^{2})$. Theoretically speaking, the SIFT based descriptor has a shorter region matching time per region pair, compared to the ZM phase descriptor. However, if desired, we can first use the ZM moment magnitude components, which are known rotationally invariant, to compute the distance between two given feature vectors. Only when the magnitude-based distance passes the condition checking, the ZM phase descriptor needs further to calculate the weighted, normalized phase difference to check if there exists a rotation angle between two matching regions.

Chapter 4

Robust Logo Recognition for Mobile Phone Applications

4.1 Introduction

With the rise of affordable digital cameras mounted on mobile devices, the mobile applications of visual image information have received a great deal of attention. Visual pattern recognition could play a key role in the mobile applications for security check, context recognition, location detection, and museum guidance [43-53]. Fig. 4.1 depicts a scenario of the mobile applications of the logo images. A mobile user directs his or her mobile phone camera to a logo of interest and captures an image in the camera field of view. A software client built in the mobile device initiates submission of the image to the server via 3G or other wireless links. The web-service reads the message and evokes the logo recognition system to identify the logo in the sever logo database. Then the server sends the corresponding corporate identity back to the client, enabling the user to access to the more detailed and specific information.

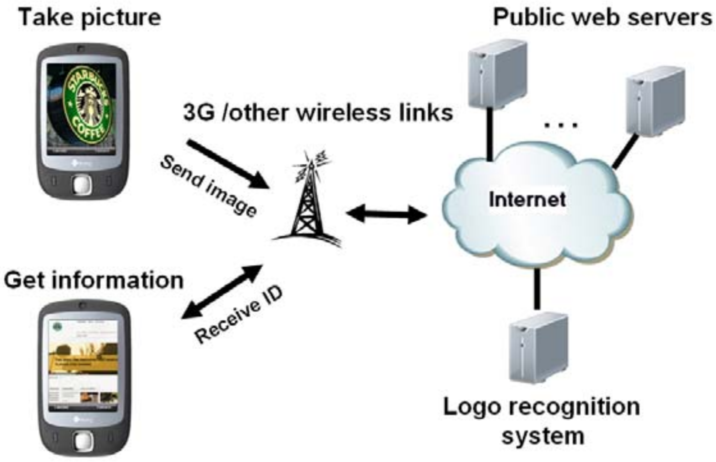


Fig. 4.1: A scenario of the mobile applications for logo recognition.

For a logo recognition system, features related to visual contents are first extracted to describe the logo images. Then, a similarity measure is defined to compare the query image with the target images in a logo database using the extracted features. Next, the target logos most similar to the query image are retrieved. Since the query logo image may be taken by a handheld mobile phone camera operating at a varying viewpoint under different lighting environments (daytime or nighttime), the query image may differ substantially from the database target one due to geometric transformations (viewpoint change, rotation, and scaling change) and photometric transformations (lighting change, noise, and image blur). Therefore, a challenge to the logo recognition system is to extract the features robust to the above inevitable imaging variations.

In this chapter, we propose a logo recognition method based on the ZM phase-based feature vector. To start with, we apply a shape deformation correction process to solve the shape distortion problem caused by a geometric transformation. The normalized logo planar patch can be shown to be affine invariant up to a rotational ambiguity [10]. After the region normalization, a ZM phase-based feature vector will be defined, which is robust to geometric and photometric image transformations. Meanwhile, due to the use of a set of orthogonal filters, the ZM feature vector is more compact and has a greater discriminative power. Experimental results show that the proposed ZM phase based recognition method has better retrieval performance in terms of the precision-recall criterion than the other existing methods.

The chapter is organized as follows. Section 2 introduces the logo shape deformation correction. Section 3 proposes the similarity measure using a ZM phase-based feature vector. In Section 4 the discriminative power of the new ZM phase recognition method is compared with three existing methods based on the precision-recall criterion. Furthermore, an analysis on the performance discrepancy between different logo recognition methods is given.

4.2 Logo Shape Deformation Correction

Since a logo usually lies on a planar surface, the logo image undergoes a homography transformation when the viewpoint is changed. The homography can be shown locally affine, so an affine approximation is commonly made. We shall fit an ellipse to a logo region. The normalized region was shown to be affine invariant up to a rotation change [10].

The ellipse region can be formulated by

$$R = \{(x,y) \mid dx^2 + 2exy + fy^2 \le 1\} \tag{4.1}$$

Where $\begin{pmatrix} d & e \\ e & f \end{pmatrix} = \begin{pmatrix} \mu_{xx} & \mu_{xy} \\ \mu_{xy} & \mu_{yy} \end{pmatrix}$

$$\mu_{xx} = \sum_{x} (x - \overline{x})^2 b(x, y)$$

$$\mu_{yy} = \sum_{y} (y - \overline{y})^2 b(x, y)$$

$$\mu_{xy} = \sum_{x} \sum_{y} (x - \overline{x})(y - \overline{y}) b(x, y)$$

The center $(\overline{x}, \overline{y})$ of the ellipse is obtained by taking mean of the coordinate of all non-zero intensity pixels. The second moment matrix is up to a scale so that the ellipse can cover all the logo pixels. The scale can be determined by finding the maximum distance from the logo center to all the boundary points of the logo:

$$s = \max_{dist} \{ dist = d(x - \overline{x})^2 + 2e(x - \overline{x})(y - \overline{y}) + f(y - \overline{y})^2 \mid (x, y) \in \text{boundary of the logo} \} . \quad (4.2)$$

As a result, the final ellipse is determined with

$$M = s \begin{pmatrix} \mu_{xx} & \mu_{xy} \\ \mu_{xy} & \mu_{yy} \end{pmatrix} . \quad (4.3)$$

Define the affine normalized image of *I'*(x, y) to be

$$I'(x, y) = M^{-\frac{1}{2}} I(x, y) . \quad (4.4)$$

Fig. 4.2 shows examples of the respective original and normalized images of a logo and its two deformed versions. We can see, after the affine normalization process, the normalized images are more similar.

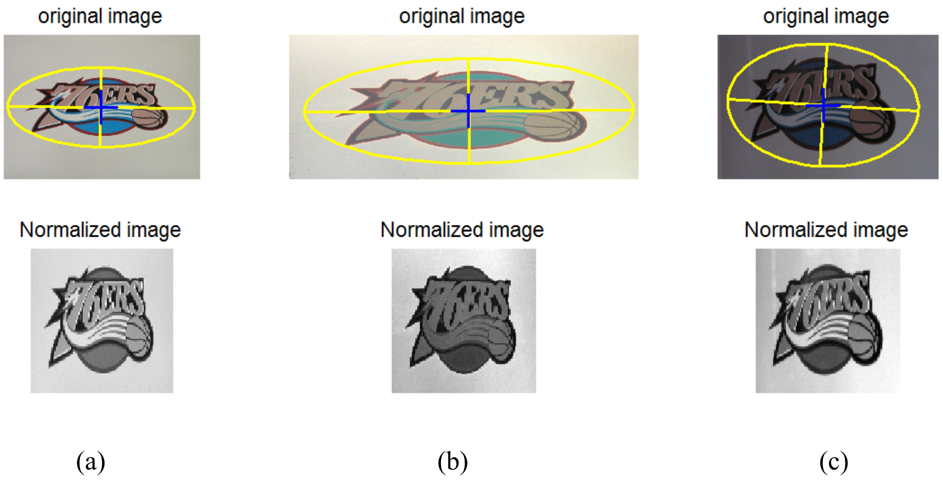


Fig. 4.2: Three logo images taken from different viewpoints and their normalized images. (a) The reference logo image. (b)-(c) The two deformed versions of the reference image. The yellow ellipses show the detected ellipses.

4.3 Logo Similarity Measure Based on the ZM Phase Information

A logo can be viewed as a single integrated graphic entity or a composite of several sub-logos when it contains multiple sub-components. There are two types of logo processing tasks: one is to classify the query logo as one from the database and the other is to retrieve all similar logos in the database. The similarity measures for these two types are defined below.

(a) The similarity measure for the logo classification

Let $L^q(x,y)$ and $L^d(x,y)$ be a query logo and a database logo, respectively, and let their respective ZM feature vectors be $\vec{P}_q = \{|Z^q{}_{nm}|e^{j\varphi^q_{nm}}\}$ and $\vec{P}_d = \{|Z^d{}_{nm}|e^{j\varphi^d_{nm}}\}$. Here both logos are treated as an integrated graphic entity each. Here the query logo can be either a rotated version of the database logo or a totally different one. A similarity measure using the weighted ZM phase differences is expressed by

$$S(\vec{P}_q,\vec{P}_d)=1-\sum_{m}\sum_{n} w_{nm}\frac{\min\{\left|\Phi_{nm}-(m\hat{\alpha})\bmod(2\pi)\right|,2\pi-\left|\Phi_{nm}-(m\hat{\alpha})\bmod(2\pi)\right|\}}{\pi}, \qquad (4.5)$$

where

$$w_{nm}=\frac{\left|Z_{nm}^{q}\right|+\left|Z_{nm}^{d}\right|}{\sum_{n,m}(\left|Z_{nm}^{q}\right|+\left|Z_{nm}^{d}\right|)}, \text{ and}$$

$\Phi_{nm}=(\varphi_{nm}^{q}-\varphi_{nm}^{d})\bmod 2\pi$ is the actual phase difference.

The rotation angle $\hat{\alpha}$ is determined by an iterative computation of $\hat{\alpha}_m=(\Phi_{n,m}-\hat{\alpha}_{m-1})\bmod 2\pi$, with the initial value $\hat{\alpha}_0=0$, using the entire information of Zernike moments sorted by *m*. The value range of $S(\vec{P}_q,\vec{P}_d)$ is the interval [0, 1].

(b) The similarity measure for the similar logo retrieval

For the similar logo retrieval, the connected components of the logo are detected first, and then each component is treated as a sub-logo. We compute the ZM feature vector for each sub-logo. Therefore, a logo is represented by a set of ZM feature vectors.

Given a query logo L_q with *N* sub-logos. We compare the query logo L_q with all the logos in the database. Assume a database logo L_d has *M* sub-logos. The similarity measure for the logo pair (L_d, L_q) is computed as the sum of the similarity scores of all matched sub-logo pairs. That is, for each sub-logo C_i^q of the query logo L_q, $i=1,\ 2,\ ...,N$. We find the sub-logo C_j^d of a database logo L_d with the maximum similarity score $S_i=S(C_i^q,C_{j*}^d)=\max_j\{S(C_i^q,C_j^d)\,|\,j=1,\ 2,\ ...,M\}$. If the similarity score is greater than a pre-defined threshold (*e.g.*, 0.8), then (C_i^q, C_{j*}^d) is considered as a matched sub-logo pair. All of the matched sub-logo pairs are further checked to ensure the 1-1 correspondence relation. Assume there are $i_1,\ i_2,\ \cdots,\ i_T$ matched sub-logo pairs in the query logo L_q, the similarity score is computed as

$$Score(L_q, L_d) = \sum_{p=1}^{P} (w_{i_p} \times S_{i_p}), \tag{4.6}$$

where $w_{i_p} = \dfrac{A_{C_{i_p}^q}}{\min(A_q, A_d)}$ with

A_q, A_d and $A_{C_{i_p}^q}$ being the areas of the query logo, a database logo and the i_t-*th* matched sub-logo $C_{i_p}^q$ of L_q, respectively.

4.4 Experimental Results

To evaluate the performance of the proposed ZM phase based recognition method, three experiments are to be conducted. We compare with our proposed method with three other state-of-the-art methods for logo recognition: IZMD [57], EHD [58], and Ring projection [59]. The first experiment is to evaluate the classification power of the four methods by treating the logo as an integrated entity. The second experiment evaluates the precision and recall rates of the four methods in which the logo is considered as a whole. The final experiment demonstrates our proposed method for retrieving the similar traffic signs by treating the logo as a composite of multiple components.

4.4.1 The performance comparison for the logo classification

Fig. 4.3 shows a set of 8 similar logos at a 400×400 pixel resolution which are downloaded from various web sites. In Fig. 4.4 we also download three different views of the first logo in the data set taken under a viewpoint change, an image blur, and a non-linear lighting change, respectively.

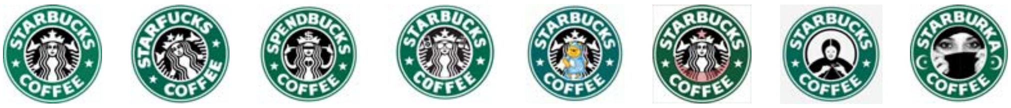


Fig. 4.3: The set of similar logos.

There are some properties of the mobile phone imagery for the logo segmentation. First, the logo is usually placed at the image center. Second, the logo and its background are highly contrasted. Third, the logo generally contains subparts in different colors. Therefore, the segmentation task of the logo image is simpler than a general image segmentation problem. Our segmentation process works in the HSI (Hue-Saturation-Intensity) color space. The major colors of the image are found as the local peaks in the histogram plot of the hue band. Then we apply the k-means clustering to cluster the image pixels of similar color as a group. We select the color clusters located near the center of the logo image. The homogeneous logo regions are then extracted using the selected color clusters. The final segmentation result of the logo positioning at each image center is shown in Fig. 4.4(b).

The segmented logo images are then submitted for a logo query. The color images are transformed to gray-level images before computing the feature vectors by the four methods. After the classification process, the logos in the top three ranks are listed in Figs 4.4(c) – 4.4(f) for the four methods. The correct one is marked with a red box. The results show that ZM phase has the best classification power for the three query logos.

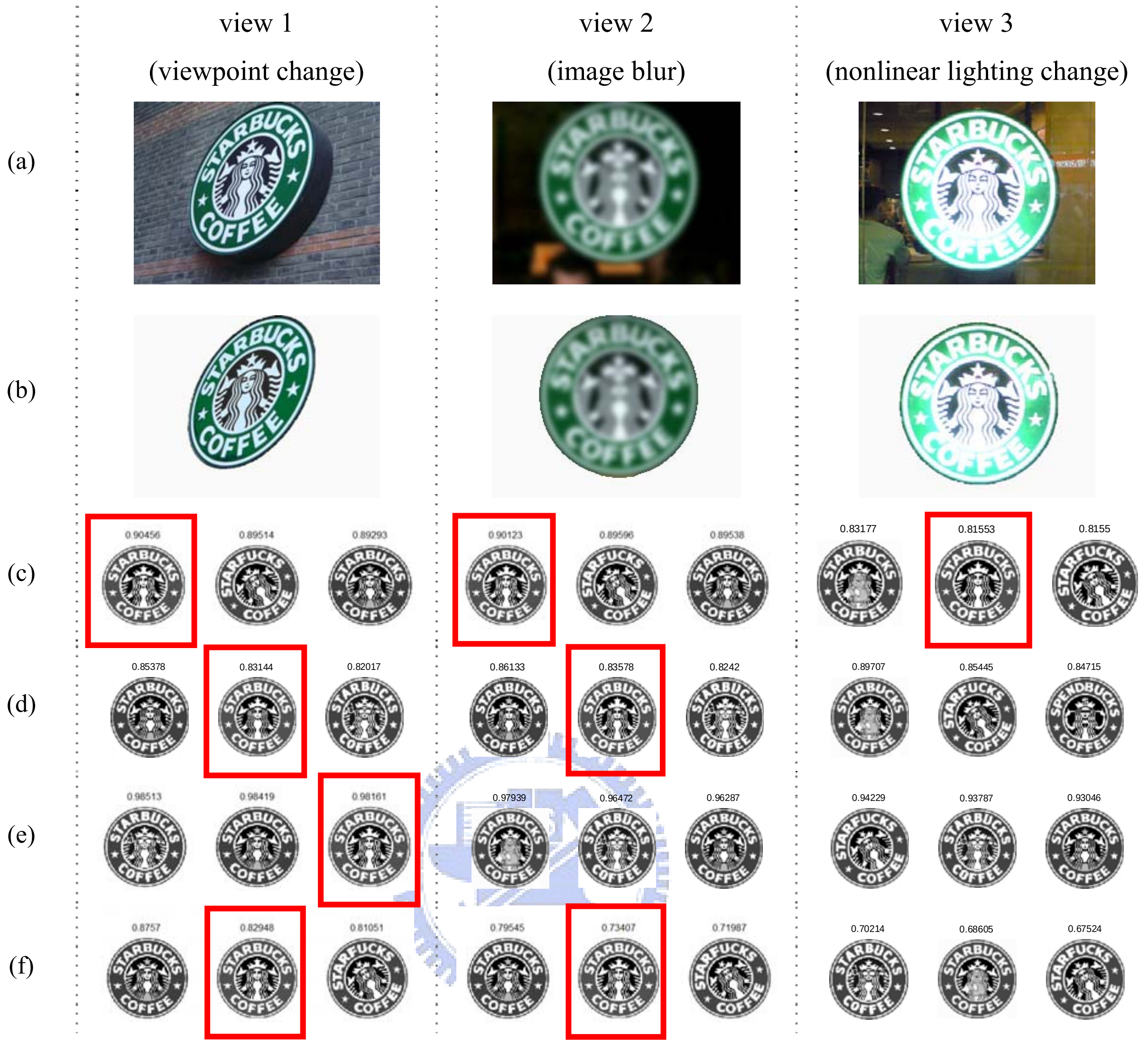


Fig. 4.4: The classification results for three logo queries. (a) Query logos. (b) Segmentation results of the query logos. The logos in the top three ranks determined by (c) ZM phase, (d) IZMD, (e) EHD and (f) Ring projection, respectively.

4.4.2 The performance comparison for logo retrieval

For evaluating the discriminative performances of the four methods, we use a database composed of M logo patterns (M = 300) in the experiment; some representatives of the logo patterns are shown in Fig. 4.5. For each logo in the database, we generate N synthetic images

(N = 10 in this case) under four kinds of imaging variations whose transformation parameters are listed below:

(1) Image blur: via a Gaussian smoothing with mean 0 and standard deviation value σ = 1.2, 1.4, 1.6 ,…., 3, respectively (with an increment of 0.2).

(2) Gamma lighting change: $I_q = I^\gamma$, where I_q is the reference image I raised to a power of γ with γ being 1/3, 2/3, 1,.., 3, respectively (with an increment of 1/3).

(3) Affine deformation: using 10 known planar homographies.

(4) Image noise: via adding Normally distributed Gaussian noise with SNR=5, 7, 9, …., 23. (with an increment of 2)

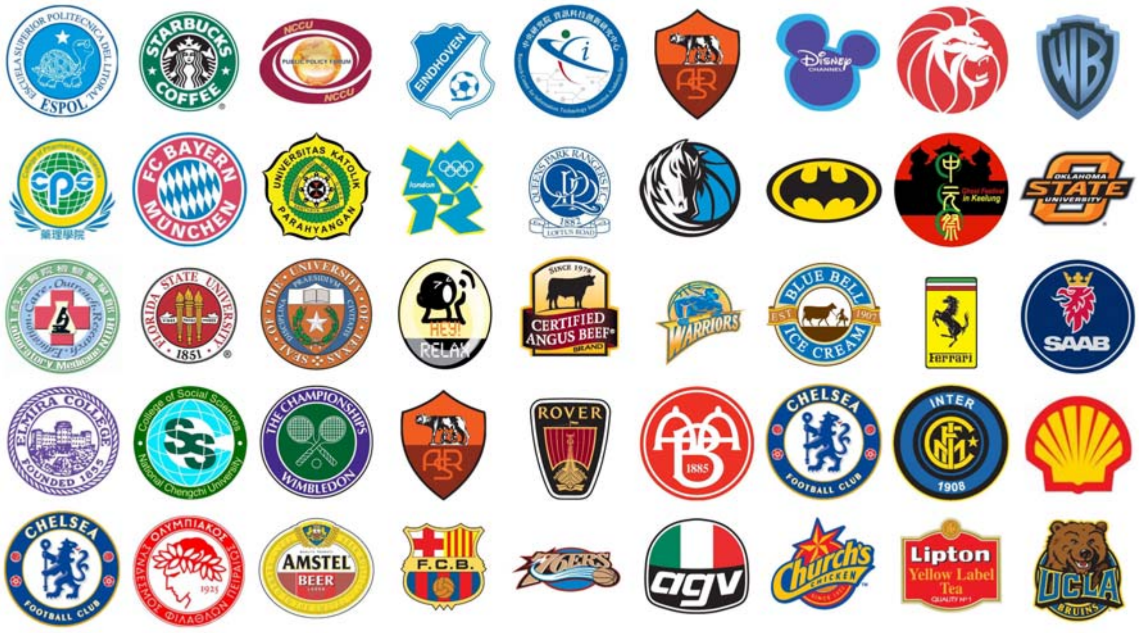


Fig. 4.5: Some of 300 logos used in the experiment.

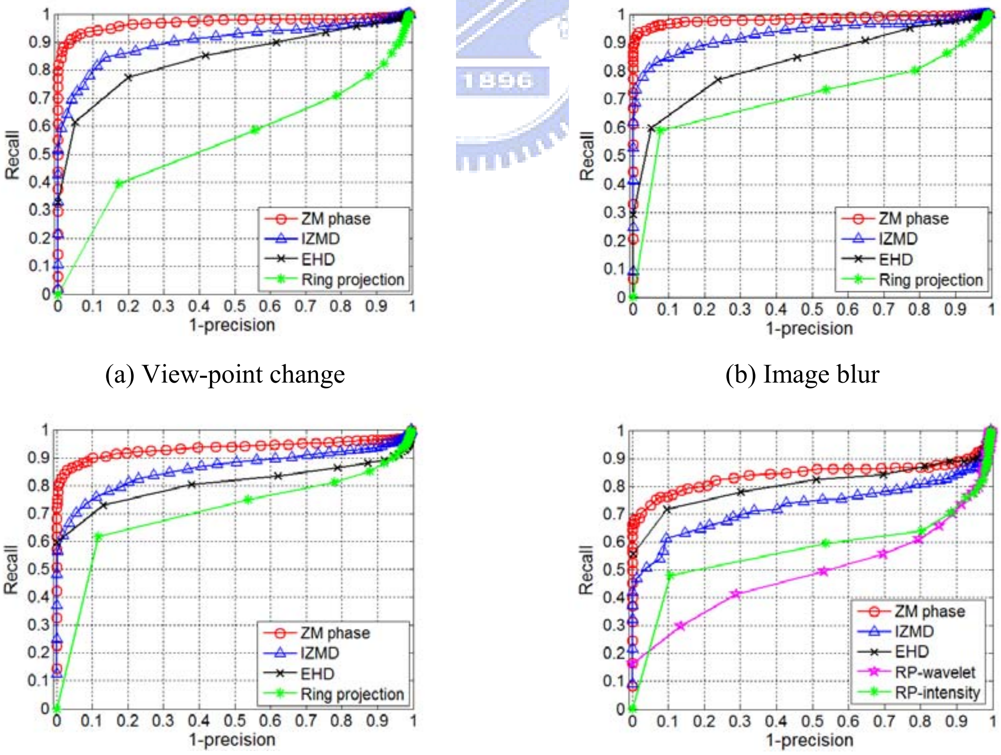
The $M \times N$ (3000=300×10) query logos under each of the above imaging variations are generated. The retrieval performances of the four recognition methods are evaluated based on the precision and recall rates as defined in 3.4.1.

$$\text{Recall} = \frac{\#\text{correct matches}}{\#\text{correspondence}}$$

$$\text{Precision} = \frac{\#\text{correct matches}}{\#\text{correct matches} + \#\text{false matches}}$$

Here, #correspondences = $M \times N$, #correct matches ≦ $M \times N$ and #false matches ≦ $(M\text{-}1) \times (M \times N)$.

Fig. 4.6 shows the results by the PR (Precision vs. Recall) curve. The ZM phase curve is located above other curves in each case, indicating the ZM phase method has the best performance among the four methods under the three given imaging variations which are rather typical.



(a) View-point change

(b) Image blur

(c) Gamma lighting change

(d) noise

Fig. 4.6: The PR curves for retrieval performance evaluations under different kinds of specified transformations.

4.4.3 Traffic sign retrieval by multiple component matching

To show the proposed method for logo retrieval based on the multiple components of the logos, the following experiment is conducted on a downloaded dataset which consists of 100 traffic signs; some representatives of them are shown in Fig. 4.7. Given a query image, the components of the sign are extracted by the hue segmentation and each of the connected components is viewed as a sub-logo. We apply the similarity measure described in Chapter 4.3.2 (b) to compute the similarity scores, and fetch traffic signs in the top 4 ranks from the dataset. Fig. 4.8 shows the top 4 positioned retrieved database logos for two different query images. As expected, the correct target traffic sign is ranked as the top one by the ZM phase method.

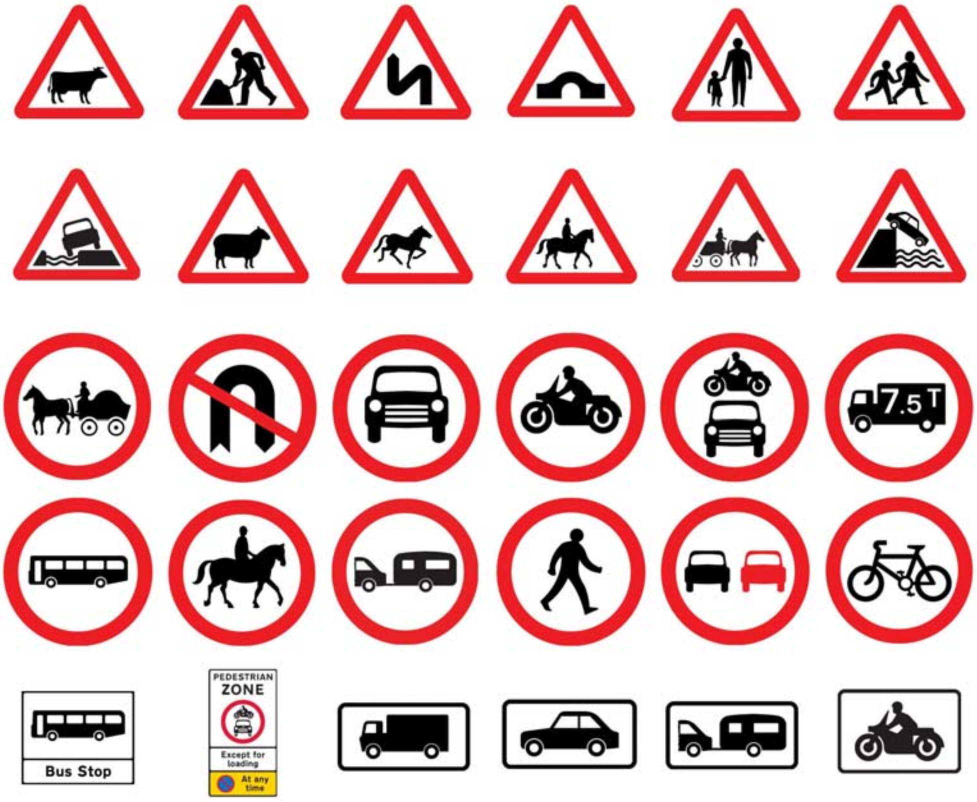


Fig. 4.7: Some of 100 traffic signs used in the experiment.

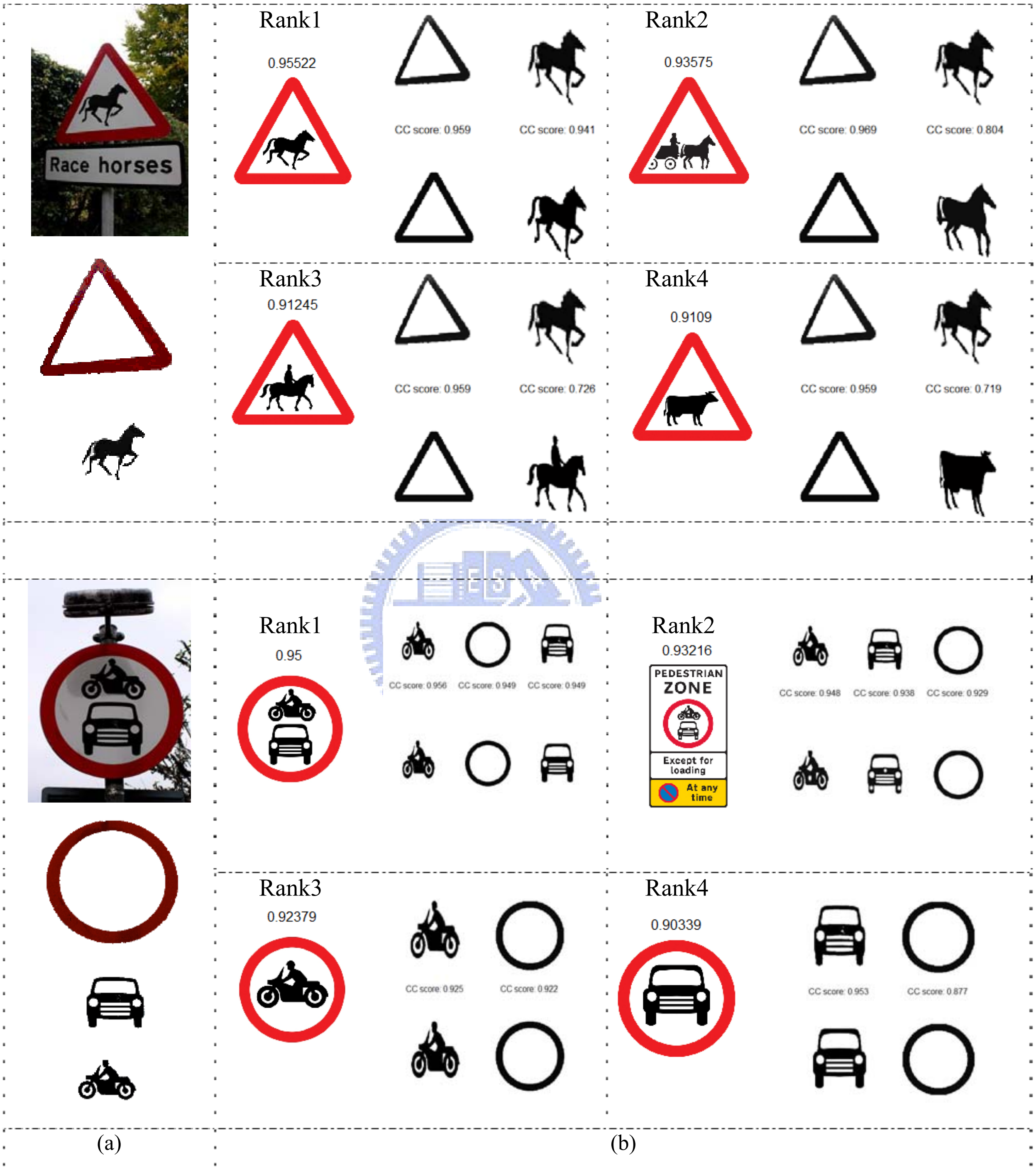


Fig. 4.8: The traffic sign retrieval results. (a) The two query images and the extracted multiple sub-logos. (b) The 4 highest-ranked database logos and their matching scores against the database sub-logos.

4.4.4 An Analysis on Logo Retrieval Results

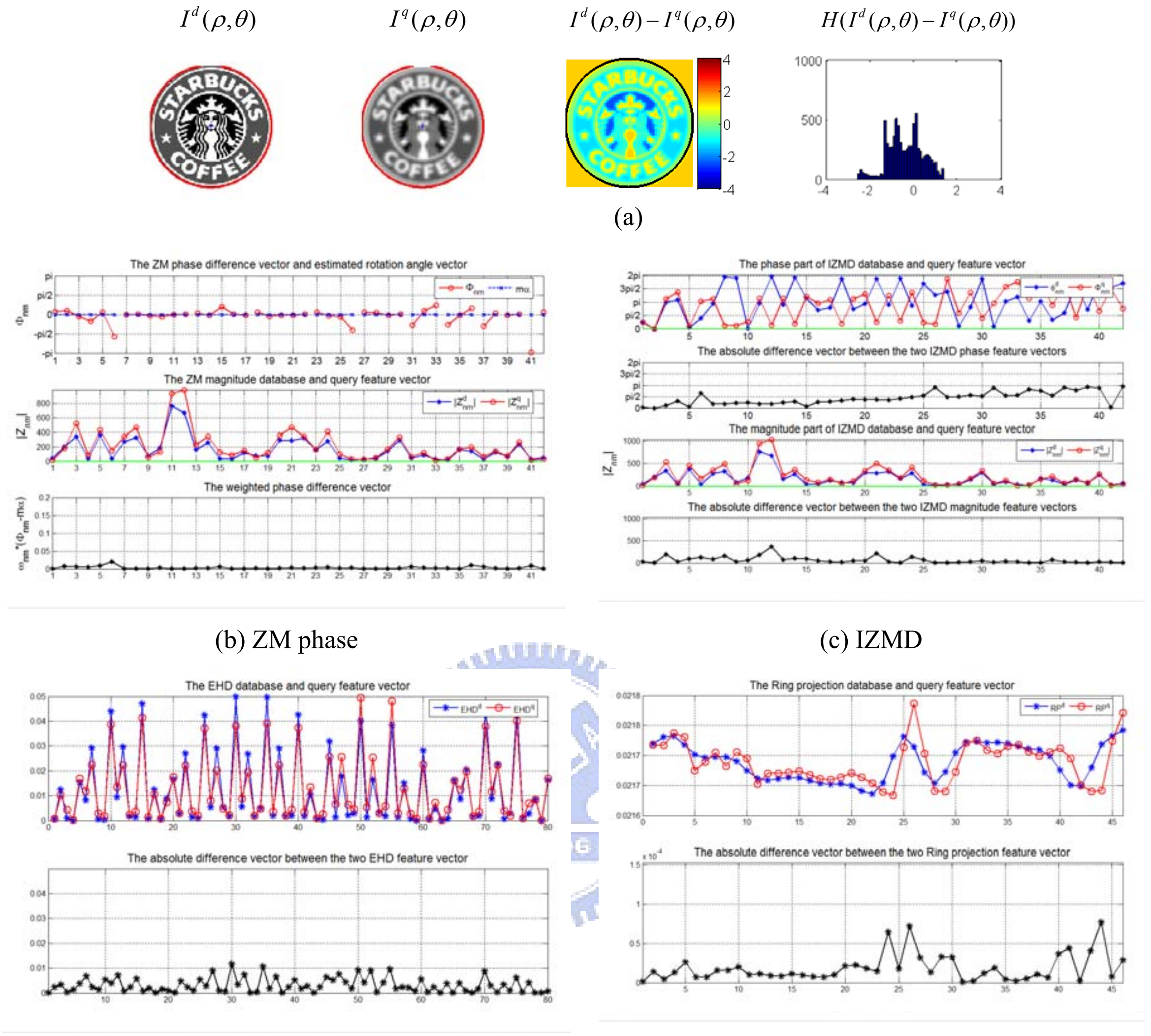
From above, we can observe that the ZM phase method has the best performance among the four recognition methods (ZM phase, IZMD, EHD and Ring projection) under image blur. The photometric and geometric transformations generally lead to an image intensity transformation at the pixel level (image blur is used as an example). To illustrate the performance differences between them, Fig. 4.9 show the intermediate results for the performance comparisons under image blur variation. Fig. 4.9(a) shows a database logo image $I^d(\rho,\theta)$ and its transformed version (i.e., query image $I^q(\rho,\theta)$) under the image blur variation, along with their difference images and histograms of intensity differences. We can observe that the difference image contain some non-uniform intensity fluctuation.

As stated in 3.5.3, the non-uniform intensity fluctuation causes the non-uniformly change in the ZM magnitude. On the other hand, since the image structures of the query and database images are similar, so it is likely that the phase angles of the two images are in phase (*i.e.*, no phase difference after the image rotation alignment). On the other hand, the probability that the two images are out of phase is small. Since our ZM phase similarity score is measured by the phase difference weighted with the ZM magnitude for each order (n, m), the weighted sum of the absolute phase differences is nearly zero, as indicated in Fig. 4.9(b). Consequently, the single majority of phase differences (zero degree) lead to the robustness of the rotation angle estimation and of the ensuing similarity measurement.

On the other hand, the similarity score of IZMD is computed as the weighted sum of the two distances: magnitude distance and phase distance. The non-uniform intensity fluctuation leads to a significant change in the ZM magnitudes, resulting in a change in the similarity score. Furthermore, the IZMD method performs a phase alignment using a fixed order moment (e. g., $\phi_{3,1}$) to achieve the rotation invariance. However, since different logos have different ZM magnitudes, the specific $|Z_{3,1}|$ magnitude may be small. In this case, the $\phi_{3,1}$ phase becomes unstable so that the other phase differences are not close to zero, as shown in the second row of Fig. 4.9(c). As a consequence, the rotation alignment is unstable, so is the similarity measure.

The 4×4 grid partition of the EHD measurement region will face the boundary effect problem, as described previously. Although the local gradient angles in the transformed image remain considerably unchanged (except under a severe image blur which causes the gradient angles destroyed), their gradient magnitudes will change in a non-uniform manner, as indicated in Fig. 4.9(d). It results in a greater dissimilarity between the original and transformed images.

Finally, the ring projection method is based on the sums of the corresponding feature values accumulated in the individual rings, and, thus, are potentially invariant to image rotation. The partition of the ring segments faces the boundary effect, too, and is sensitive to the non-uniform image fluctuation, as indicated in Fig. 4.9(e). Moreover, the ring projection structure loses the spatial information in the individual rings, thus reducing its discriminative power. Consequently, the ring projection has the poor performance, as shown in Fig. 4.6.



(a)

(b) ZM phase

(c) IZMD

(d) EHD

(e) Ring projection

Fig. 4.9: A performance analysis on the ZM phase, IZMD, EHD and Ring projection methods under image blur.

Chapter 5

High-Efficiency Perspective View Registration Using Offline Planning Strategies

5.1 Introduction

The fundamental problem of view registration is to recover a 2D spatial transformation model to overlay two or more images taken under different imaging conditions [60-64]. The image will generally deform under a view transformation. Therefore, the view registration is better based on the local image features instead of the global features. There are two major types of invariant features: points of interest and regions of interest. Both types of features are designated in the image by points. The view transformation model can be estimated with or without the actual establishment of point correspondences between the reference and sensed images first. These methods have different orders of time complexity (refer to Chapter 2). We have observed that various countermeasures were taken to reduce the time complexity of the view registration method.

In this chapter we propose an alternative way to achieve better registration efficiency. We introduce five planning strategies to sort the feature points in the reference image based on the concepts of feature invariance to image deformation, image noise resistance, distinctive description power, model estimation effectiveness, and partial image overlapping handling capability. The feature points are detected using the Gabor filtering technique and a reference matching database is constructed offline using the proposed five planning strategies. Here, we focus on the planning strategies to achieve better registration efficiency. The Gabor feature points can be replaced by any of the invariant feature detectors (e.g. MSER or Hessian-affine), which are with energy value, dominant orientation. Next, an online registration process is presented to estimate the transformation model to overlay the reference image over an incoming sensed image. We take advantage of preprocessing of the reference image offline to gather the important statistics for guiding the sensed image registration. Fig. 5.1 shows the architecture of the proposed method. In this way better registration efficiency can be achieved. Experimental registration results are provided and the computational complexity is analyzed.

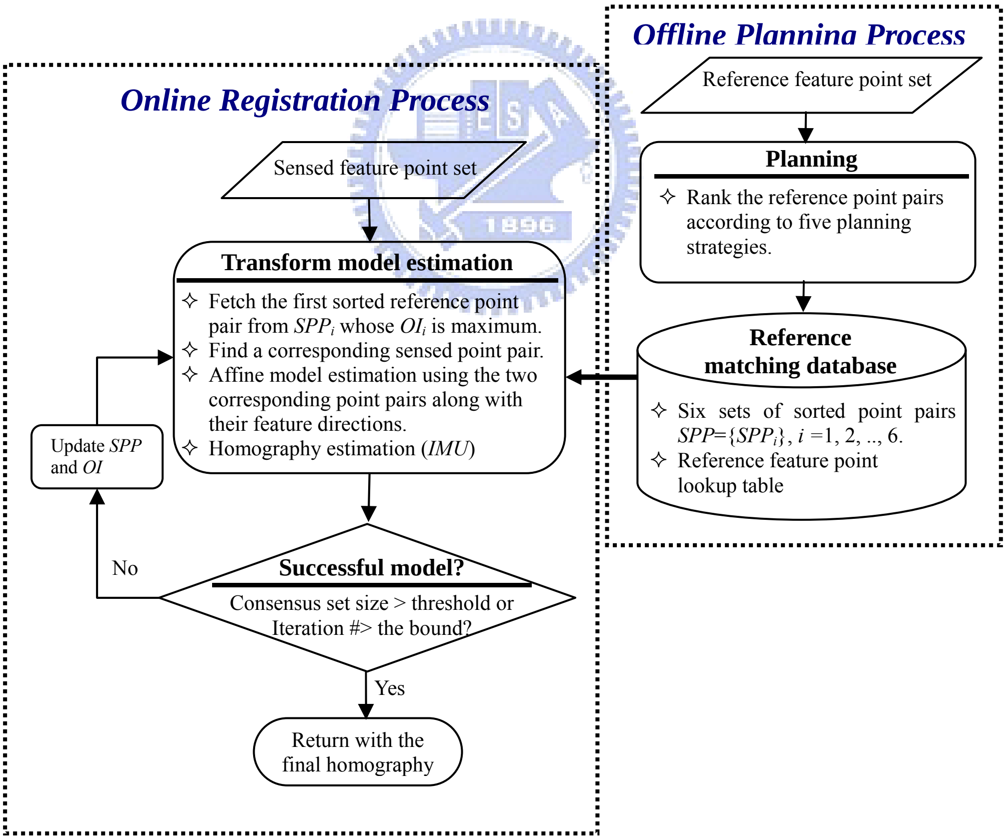

Offline Planning Process
Reference feature point set
Planning
✧ Rank the reference point pairs according to five planning strategies.
Reference matching database
✧ Six sets of sorted point pairs $SPP=\{SPP_i\}$, i =1, 2, .., 6.
✧ Reference feature point lookup table
Online Registration Process
Sensed feature point set
Transform model estimation
✧ Fetch the first sorted reference point pair from SPP_i whose OI_i is maximum.
✧ Find a corresponding sensed point pair.
✧ Affine model estimation using the two corresponding point pairs along with their feature directions.
✧ Homography estimation (*IMU*)
Update *SPP* and *OI*
No
Successful model?
Consensus set size > threshold or Iteration #> the bound?
Yes
Return with the final homography

Fig. 5.1: The architecture of the proposed view registration process

The main concepts of the method include:

(1) To reduce the time complexity we initially approximate the homography model by an affine one, which is then estimated by using only two pairs of matched points, along with their feature point directions, all obtained by the above Gabor filtering technique. The initial transformation model is later iteratively updated to produce the final homography model.

(2) To solve the partial image overlapping problem, we partition the reference image into four sub-regions and construct six region pairs from the four sub-regions. During the online registration we compute the overlap index for the six region pairs so that we can avoid selecting the point pairs from the non-overlapping sub-regions.

(3) We implement five planning strategies to sort the feature points in the reference image based on the concepts of feature invariance to image deformation, image noise resistance, model estimation effectiveness, distinctive description power, and partial image overlapping handling capability to construct a reference matching database offline. This database will be used in the later online sensed image registration.

The rest of the chapter is organized as follows. Section 2 introduces the invariant feature point extraction using the Gabor filtering technique. In Section 3, we discuss how the affine transformation can be determined by using only two feature points along with their feature directions. Next, we refine the transformation model by applying an iterative process. Section 4 describes an off-line reference matching database construction using five planning strategies in order to select two good starting reference point pair to invoke a later online view registration. The concept of an overlap index is introduced to handle the partial image overlapping problem. Section 5 illustrates the on-line registration process. Experimental results and performance analysis are given in Section 6. Finally, we give an analysis of the algorithm computational performance in Section 7.

5.2 Feature Point Extraction by Gabor Filtering

In our previous work [78], we apply a multi-scale and multi-orientation Gabor filtering technique to obtain a set of feature points from an image. Let $I(x,y)$ be the input image. For a set of scales $s \in \{1, 2, 3, ..., S\}$, and a set of orientations $\theta_l = l \times \Delta\theta$, $l = 1, 2, ..., L$ ($\Delta\theta$ is a divisor of π), the image responses to the multi-scale and multi-orientation Gabor filters are described by a convolution operation:

$$R^{s,l}(x,y) = I(x,y) * g^{s,l}(x,y)\,, \tag{5.1}$$

where

$$g^{s,l}(x,y) = \frac{1}{2\pi\sigma_s^{\,2}} \exp\{-\frac{(x')^2 + (y')^2}{\sigma_s}]\} \sin(w_s x'), \text{ or}$$

$$g^{s,l}(x,y) = \frac{1}{2\pi\sigma_s^{\,2}} \exp\{-\frac{(x')^2 + (y')^2}{\sigma_s}]\} \cos(w_s x')\,.$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta_l & \sin\theta_l \\ -\sin\theta_l & \cos\theta_l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

and $w_s = 3\pi / 4\sigma_s$.

The absolute responses to the filters at a common scale and L orientations is summed up at each image point and the maximum sum over the S scales is searched, as shown below

$$E(x,y) = \max_{s \in S}\{E^s(x,y) = \sum_{l=1}^{L} \left| R^{s,l}(x,y) \right| \}\,. \tag{5.2}$$

This maximum energy $E(x, y)$ is taken to reflect the actual energy at the image point $p_i(x, y)$

and the particular scale at which the maximum energy occurs is called the dominant scale, $s_{p_i}^{d}$. The corresponding orientation l associated with the maximum filter response $R^{s,l}$ at the dominant scale is called the dominant orientation or feature point direction, $\theta_{p_i}^{d}$, at the image point. The Gabor-filtered feature points with a local maximum energy can be shown to be robust to the local image deformation.

For measuring the similarity between two matching points, the image patches centered at the Gabor feature points are represented by the ZM phase descriptor, and the weighted, normalized phase differences are computed as in Chapter 3.

5.3 View Transformation Model Estimation

5.3.1 Affine approximation to the homography model

We shall first approximate a homography by an affine model. The affine transformation involves six parameters, so we need at least three matched point pairs to estimate the six parameters. However, we use only two pairs of feature points. The third point pair required for the model estimation is a virtual point pair obtained from the intersection of the two dominant orientations associated with the two feature points in each of the two images. Any other type of invariant feature points can substitute our feature points if they have an accompanied feature point direction, too. The advantage of using two point pairs instead of three pairs is to reduce the computational complexity, as shall been seen later.

Under the affine transformation the relationships between the two matched point pairs (p_k, p_k') and (p_l, p_l') along with their associated feature point directions $(\vec{e}_k, \vec{e}_k')$ and $(\vec{e}_l, \vec{e}_l')$

are given by a matrix A, i.e.,

$$\vec{X}'_{p_i} = \begin{bmatrix} x'_{p_i} \\ y'_{p_i} \\ 1 \end{bmatrix} = \begin{bmatrix} s_1 \cos\theta & -s_1 \sin\theta & t_x \\ s_2 \sin\theta & s_2 \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{p_i} \\ y_{p_i} \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{p_i} \\ y_{p_i} \\ 1 \end{bmatrix} = A\vec{X}_{p_i} \quad \text{and}$$

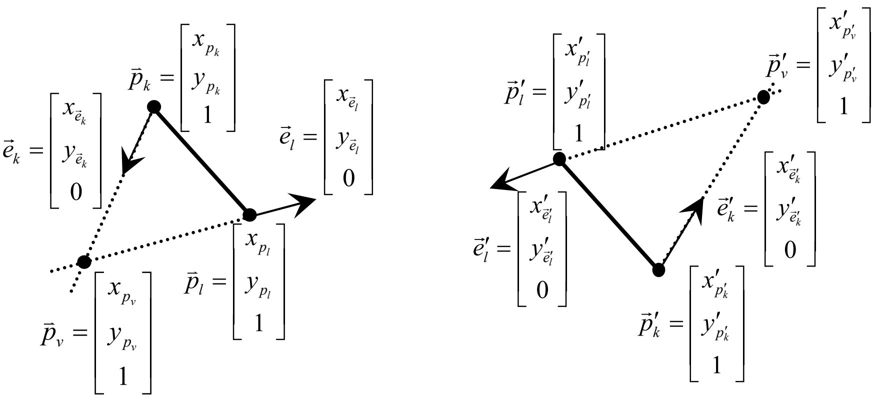
$$\begin{bmatrix} x'_{\vec{e}'_i} \\ y'_{\vec{e}'_i} \\ 0 \end{bmatrix} = A \begin{bmatrix} x_{\vec{e}_i} \\ y_{\vec{e}_i} \\ 0 \end{bmatrix} \quad i = k, l. \tag{5.3}$$

There are eight linear equations in six unknown parameters of the affine matrix. We can estimate the affine matrix using the singular value decomposition (SVD) technique.

Fig. 5.2 illustrates the two respective pairs of Gabor feature points (p_k, p_l) and (p'_k, p'_l) in the reference and sensed images, together with their associated unit feature point directions $(\vec{e}_k, \vec{e}_l)$ and $(\vec{e}'_k, \vec{e}'_l)$. Note that the feature point directions $\vec{e}_k$, $\vec{e}_l$ and $\overrightarrow{p_k p_l}$ must not be mutually parallel or nearly parallel in order that their extended lines can intersect. That is, they must satisfy the following intersection condition:

$$|(\overrightarrow{e_k} \cdot \overrightarrow{p_k p_l})| < 1 - threshold,\ |(\overrightarrow{e_l} \cdot \overrightarrow{p_k p_l})| < 1 - threshold, \text{and } |(\overrightarrow{e_k} \cdot \overrightarrow{e_l})| < 1 - threshold\,. \tag{5.4}$$

Similarly, the feature point directions $\vec{e}'_k$, $\vec{e}'_l$ and $\overrightarrow{p'_k p'_l}$ in the sensed image must satisfy an identical condition. Denote the estimated matrix A by $T^{(0)}$. It will be used as the initial solution to estimate a more general homography transformation presented below.


$\vec{p}_k = \begin{bmatrix} x_{p_k} \\ y_{p_k} \\ 1 \end{bmatrix}$
$\vec{e}_k = \begin{bmatrix} x_{\vec{e}_k} \\ y_{\vec{e}_k} \\ 0 \end{bmatrix}$
$\vec{e}_l = \begin{bmatrix} x_{\vec{e}_l} \\ y_{\vec{e}_l} \\ 0 \end{bmatrix}$
$\vec{p}_l = \begin{bmatrix} x_{p_l} \\ y_{p_l} \\ 1 \end{bmatrix}$
$\vec{p}_v = \begin{bmatrix} x_{p_v} \\ y_{p_v} \\ 1 \end{bmatrix}$
$\vec{p}'_l = \begin{bmatrix} x'_{p'_l} \\ y'_{p'_l} \\ 1 \end{bmatrix}$
$\vec{p}'_v = \begin{bmatrix} x'_{p'_v} \\ y'_{p'_v} \\ 1 \end{bmatrix}$
$\vec{e}'_l = \begin{bmatrix} x'_{\vec{e}'_l} \\ y'_{\vec{e}'_l} \\ 0 \end{bmatrix}$
$\vec{e}'_k = \begin{bmatrix} x'_{\vec{e}'_k} \\ y'_{\vec{e}'_k} \\ 0 \end{bmatrix}$
$\vec{p}'_k = \begin{bmatrix} x'_{p'_k} \\ y'_{p'_k} \\ 1 \end{bmatrix}$

(a) **(b)**

Fig. 5.2: (a) The point set (p_k, p_l) and the dominant orientation set $(\vec{e}_k, \vec{e}_l)$ in the reference image. (b) The corresponding point set (p'_k, p'_l) and dominant orientation set (e'_k, e'_l) in the sensed image.

5.3.2 Iterative view transformation updating

Finally, we extend the estimation from the affine transformation to a homography

$$\vec{X}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = M\vec{X} \,. \tag{5.5}$$

A homography transformation differs from an affine transformation in the nonzero values of m_{31} and m_{32}. We use the affine solution $T^{(0)}$ as the initial solution to invoke an iterative model updating (IMU) algorithm to obtain the final transformation estimation.

Let $P = \{p_1, p_2, \cdots, p_n\}$ be a set of n feature points in the reference image and $Q = \{q_1, q_2, \cdots, q_m\}$ be a set of m feature points in the sensed image. The IMU algorithm iteratively transforms the sensed feature point set Q back to the reference image space to seek for more corresponding points in P in support of the current transformation model. All the matched point pairs constitute the *corresponding point set* (*CPS*) similar to the consensus set in RANSAC. Those unmatched pairs are viewed as the outliers. A new transformation $T^{(c)}$ is computed using the enlarged $CPS^{(c-1)}$, where c denotes the iteration index. The process is repeated unless $CPS^{(c)}$ converges or the iteration number of the process exceeds a specified bound.

To facilitate the above search for any reference point of P in support of a sensed point under consideration, we use a table lookup technique. The lookup table is built offline and has the same size as the reference image. Each entry of the lookup table is a pointer, pointing to the location address of each reference image point. The content of this memory address contains a sorted list of $\{(p_i, d_i), i\in\{1, 2, 3, .., n\}\}$, where p_i is a reference feature point of P with $d_i < d^{(c)}$ where d_i is the distance between p_i and the image point picked by the pointer. The following is the iterative model updating algorithm:

Algorithm **IMU** (**I**terative **M**odel **U**pdating)

```
Input:
  1. Sets of reference and sensed feature points: $P = \{p_1, p_2, \cdots, p_n\}$ and $Q = \{q_1, q_2, \cdots, q_m\}$.
  2. The initial estimate of the affine transformation $T^{(0)}$ corresponding to a reference point
     pair $(p_{s,i_s}, p_{t,i_t})$ and a selected sensed point pair $(q_k, q_l)$.
Output:
   1. The resultant corresponding point set $CPS^{(r)}$
   2. The resultant homography transformation matrix $T^{(r)}$
Initialization:
  $c = 0$; $d^{(0)}$ = the reference image width/20; $CPS^{(0)} = \Phi$ (empty set)
Begin
  For $c = 1, 2 \ldots, c_{max}$ ($c_{max} = 5$ in our case)
      1. Reset $CPS^{(c)} = \Phi$ (empty set);  $d^{(c)} = d^{(0)} / 2^c$
```

```
    2. For $j$ = 1, 2, … , $m$
        (1) Fetch the point $q_j$ from $Q$ and find any matched point $p_i \in P$ such that
            $\left|T^{(c-1)}(q_j) - p_i\right| \le d^{(c)}$ using the table look-up.
        (2) If there is more than one such point $p_i$ in $P$, find the one with the minimum
            descriptor matching distance with $q_j$. If the matching distance is less than a
            pre-specified threshold, then add the matched pair to $CPS^{(c)}$.
        End For
    3. If $\left|CPS^{(c)}\right| \le \left|CPS^{(c-1)}\right|$, then return $T^{(r)} = T^{(c)}$ and $CPS^{(r)} = CPS^{(c)}$ and stop the process.
    4. Compute the new transformation matrix $T^{(c)}$ using all point pairs in $CPS^{(c)}$.
  End For
End
```

A remark is in order here. Regardless of whether the input reference pair (p_{s,i_s}, p_{t,i_t}) and the selected sensed point pair (q_k, q_l) are actually matched or not, the process will be terminated within 5 iterations. If the iteration number is less than c_{max} (c_{max} = 5 in our case) and the resultant $CPS^{(r)}$ is of sufficiently large size, the solution model $T^{(r)}$ is likely to be correct; otherwise, the solution model is probably wrong. This is because the inputted reference point pair (p_{s,i_s}, p_{t,i_t}) fetched from the database is the best reference pair based on the offline planning strategies. If this pair fails, it means the sensed image is probably not overlapped with a reference image part from which the pair (p_{s,i_s}, p_{t,i_t}) is fetched. More concrete examples are given in the section on the experimental results.

5.4 Off-Line Reference Matching Database Construction with Planning Strategies

By now, we know that the success of the final view registration is determined by two good starting reference points and their respective matched points in the sensed image. In the following we shall present five offline planning strategies for providing the two good starting reference points from a reference matching database to be constructed.

First of all, the image feature points must be robust to image noise. The Gabor filters used contains a Gaussian smoothing factor, so they can resist the image noise impact. Therefore, a good reference feature point should have large response energy, as described below:

Define the normalized energy factor at point p_i as

$$E(p_i) = \frac{E_i - E_{\min}}{E_{\max} - E_{\min}}, \tag{5.6}$$

where E_{max} and E_{min} are the maximum and minimum energy values at the points of the reference feature point set, respectively, and E_i is the energy at p_i.

On the other hand, the stability of the dominant orientation at a feature point can be measured by comparing the filter responses at the dominant orientation and its two neighboring orientations:

$$DO(p_i) = \frac{\left\| R_i^{l_d} \right| - \left| R_i^{l_d+1} \right\| + \left\| R_i^{l_d} \right| - \left| R_i^{l_d-1} \right\|)}{\left| R_i^{l_d} \right|}, \tag{5.7}$$

where $R_i^{l_d}$ is the filter response at point p_i associated with the dominant orientation l_d, l_d

$\in L$.

Define the normalized orientation factor at point p_i as:

$$O(p_i) = \frac{DO(p_i) - DO_{\min}}{DO_{\max} - DO_{\min}}, \tag{5.8}$$

where $DO_{\max} = \max_i \{DO(p_i)\}$ and $DO_{\min} = \min_i \{DO(p_i)\}$.

Strategy 1: Sort the reference feature points p_i, i =1, 2, … , n, in the descending order of their products of normalized energy and orientation factors $E(p_i)$ $O(p_i)$.

As described in Section 5.2.1 on the affine model estimation, any two reference feature points must form a triangle with their associated feature point directions. (Refer to Fig. 5.2)

Strategy 2: Select the possible reference point pairs such that the two associated feature point directions satisfy the intersection condition for constituting a triangle.

In Fig. 5.3 the two points p_i and p_j, together with their dominant orientations $\vec{e}_{p_i}$ and $\vec{e}_{p_j}$, form a triangle (shown by solid lines) with area $A(p_i, p_j)$. The image noise in the two points and their dominant orientations will affect the ensuing view transformation estimation accuracy. An equilateral triangle with a large area is good for the transformation estimation. We should choose such a triangle from the data set. We need to measure the similarity between a triangle and a virtual equilateral triangle constructed by the longest side of the triangle (shown by dashed lines in Fig. 5.3). An effective equilateral triangle similarity measure is given by

$$S(p_i, p_j) = \frac{A(p_i, p_j)}{A_e(p_i, p_j)} = \frac{A(p_i, p_j)}{(\sqrt{3}/4)l_{\max}{}^2};\ \ 0 \le S(p_i, p_j) \le 1, \tag{5.9}$$

where $A_e(p_i, p_j)$ is the area of the virtual equilateral triangle constructed.

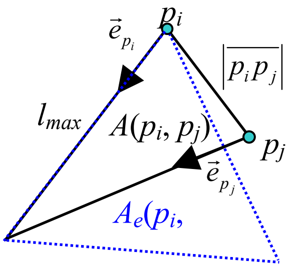


Fig. 5.3: The triangle formed by the two feature points p_i, p_j, and their dominant orientations $\vec{e}_{p_i}$ and $\vec{e}_{p_j}$, together with an equilateral triangle constructed by the longest side $l_{\max}$ of the triangle.

Finally, the effective triangle index is defined by

$$S_e(p_i, p_j) = S(p_i, p_j)\left|\overline{p_i p_j}\right|. \tag{5.10}$$

Strategy 3: Sort the two reference point pairs screened by Strategy 2 in the descending order according to their individual effective triangle indices.

Next, let $N(p_i)$ be the total number of sensed feature points which are found matched to the reference point p_i. Similarly, let $N(p_j)$ be the total number of feature points which are matched to reference point $p_{j.}$ The distinctiveness (or uniqueness) measure of a reference feature point pair (p_i, p_j) is given by

$$U(p_i, p_j) = \frac{1}{N(p_i)N(p_j)}. \tag{5.11}$$

Strategy 4: Sort the reference point pairs in the descending order according to their individual pair-wise distinctiveness measures.

If the reference image is only partially overlapped with the sensed image, the reference feature points in the non-overlapping region will find no matches in the sensed image. We should avoid using these reference feature points for the view registration. To handle this partial image overlapping problem, we partition the whole reference image into four equal sub-regions. We will rank reference point pairs in the six ($C_2^4 = 6$) combinations of sub-region pairs.

Strategy 5: Divide the reference image region into four sub-regions R_1 to R_4 and construct the six region pairs from the four sub-regions to handle the partial image Registration problem.

Now, we are ready to give a process using the above planning strategies to pre-compile a reference matching database to be used in a later online registration to overlay the reference image over an incoming sensed image.

The off-line reference matching database construction process

1. Divide the reference image into four sub-regions R_1 to R_4 (Strategy 5).
2. For each sub-region R_i, i =1, 2, 3, 4, sort the feature points $p_{i,k}$ of R_i in the descending order according to the product of normalized energy and orientation factors $E(p_{i,k})O(p_{i,k})$ (Strategy 1). Retain those reference feature points in each sub-region whose product of normalized energy and orientation factors is greater than a specified threshold. Denote the sorted list of the retained points by $F_i = \{p_{i,1}, p_{i,2}, ..., p_{i,N_i}\}$, i =1, 2, 3, 4.
3. Construct the six possible region pairs from regions R_1 to R_4, denoted by RP = {(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_2, R_3), (R_2, R_4), (R_3, R_4)}. For each region pair (R_i, R_j) of RP find the

Cartesian product (F_i× F_j).

(a) For each Cartesian product (F_i× F_j), (i ,j)∈{(1, 2), (1,3) (1,4), (2,3), (2, 4), (3, 4)}, retain the possible reference point pairs $(p_{i,k}, p_{j,l}) \in F_i \times F_j$ satisfying the intersection condition (i.e.,Strategy 2). Sort all of the feature point pairs $(p_{i,k}, p_{j,l}) \in F_i \times F_j$ in the descending order according to the effective triangle index $S_e(p_{i,k}, p_{j,l})$ (i.e., Strategy 3) and keep those point pairs $(p_{i,k}, p_{j,l})$ with a triangle index $S_e(p_{i,k}, p_{j,l})$ greater than a specified lower bound. Replace the original F_i×F_j by the sorted list of the retained point pairs $(p_{i,k}, p_{j,l})$.

(b) Sort the retained point pairs in F_i×F_j according to the pairwise distinctiveness measure (i.e., Strategy 4): $U(p_{i,k}, p_{j,l}) = \dfrac{1}{N(p_{i,k})N(p_{j,l})}$

Denote the set of six sorted lists of reference point pairs of {F_i×F_j} obtained above by SPP = {SPP_i, i =1, 2, .., 6}. This is called the reference matching database. Table 5.1 lists the construction of SPP_i from four sub-regions. The reference matching database contains pairs of reference points which will be served as the two starting reference points to invoke an affine model estimation and a subsequent iterative model updating process.

TABLE 5.1 THE CONSTRUCTION OF SPP_i FROM THE FOUR SUB-REGIONS.

SPP_i	Corresponding sub-regions	
SPP_1	R_1	R_2
SPP_2	R_1	R_3
SPP_3	R_1	R_4
SPP_4	R_2	R_3
SPP_5	R_2	R_4
SPP_6	R_3	R_4

5.5 Online View Registration

When a sensed image is available, we can start to match the reference feature points in the matching database with the feature points extracted from the sensed image. We use *SPP* = {SPP_i, i =1, 2, .., 6} obtained above to detect the overlapping area between the reference and sensed images. During the view registration process a reference point pair (p_{s,i_s}, p_{t,i_t}) is fetched in order from *SPP* where p_{s,i_s} and p_{t,i_t} are the i_s-th and i_t-th reference feature points from sub-regions R_s and R_t, respectively. If either point fails to find any matched point in the sensed image, then delete all the reference point pairs in *SPP* involving the unmatched reference point, p_{s,i_s} or p_{t,i_t}. We define the size ratio of the updated SPP_i to its initial set as the overlap index. When the overlap index is low for a particular SPP_i, it implies the chance that the two reference sub-regions of SPP_i overlap with the sensed image is also low. An algorithm for the on-line registration process is given below:

Algorithm OLRP (On-Line Registration Process)

Input:

1. The sets of reference and sensed feature points: $P = \{p_1, p_2, \cdots, p_n\}$ and $Q = \{q_1, q_2, \cdots, q_m\}$.
2. The lists of sorted reference point pairs in the six region pairs: *SPP*={$SPP_{i,}$ i =1, 2, .., 6}.
3. The size of initial $SPP_{i,}$: S_i= |SPP_i| for i = 1, 2,…, 6.

Output:

1. The final corresponding point set: $CPS^{(f)}$
2. The final transformation matrix: $T^{(f)}$

Initialization:

Initialize the overlap index OI_i = 1 for i = 1, 2,…, 6.

Begin

```
    For $c$ = 1, 2 … , $c_{max}$ ($c_{max}$ = 5)
        1. Fetch the first element $(p_{s,i_s}, p_{t,i_t})$ from the sorted point pair list $SPP_i$ whose overlap
           index $OI_i$ is maximum (if there is a tie, break the tie arbitrarily).
        2. Find the matched points in the sensed image for each of $p_{s,i_s}$ and $p_{t,i_t}$ based on the
           normalized cross correlation measure. Assume the resulting matched point sets are
           $CM_s = \{q_{k,}\ k = 1, 2, .., n_s\}$ and $CM_t = \{q_{l,}\ l = 1, 2, .., n_t\}$ for $p_{s,i_s}$ and $p_{t,i_t}$, respectively. If
           $CM_s$ (or $CM_t$) is empty, then delete all the reference point pairs involving the
           unmatched reference point, $p_{s,i_s}$ (or $p_{t,i_t}$), from its associated sub-region $R_s$ or $R_t$ and
           $SPP_i$. Update $OI_i = |SPP_i|/S_i$ and go to step 1. If both $CM_s$ and $CM_t$ are not empty,
           continue.
        3. For each $(q_k, q_l)$ in $CM_s \times CM_t$
            (1) Compute the affine transformation matrix $T^{(0)}$ using the two point pairs
                $(p_{s,i_s}, p_{t,i_t})$ and $(q_k, q_l)$ (refer to Section 5.1).
            (2) Invoke the IMU algorithm to determine the homography matrix $T^{(r)}$ using $T^{(0)}$
                as the initial solution and to find $CPS^{(r)}$ (refer to Section 5.2).
            (3) Check the stopping criteria: if the size of the corresponding point set $CPS^{(r)}$ is
                greater than a pre-defined threshold, then return $T^{(f)} = T^{(r)}$ and terminate the
                process with "success"; otherwise, continue.
            End For
        4. Delete the element $(p_{s,i_s}, p_{t,i_t})$ from an involved $SPP_i$ and update the involved overlap
           index $OI_i$.
    End For
**End**
```

The iteration number of the above cycle is bounded by a fixed number (5 in our case), as shall be explained at the end of the next section.

5.6 Experimental Results

A) The Iterative View Registration under the Homography Transformation

We apply our method to register two aerial images. Fig. 5.4(a) shows the reference image of size 500 by 500. A synthetic sensed image with severe perspective deformation is generated and shown in Fig. 5.5(b).

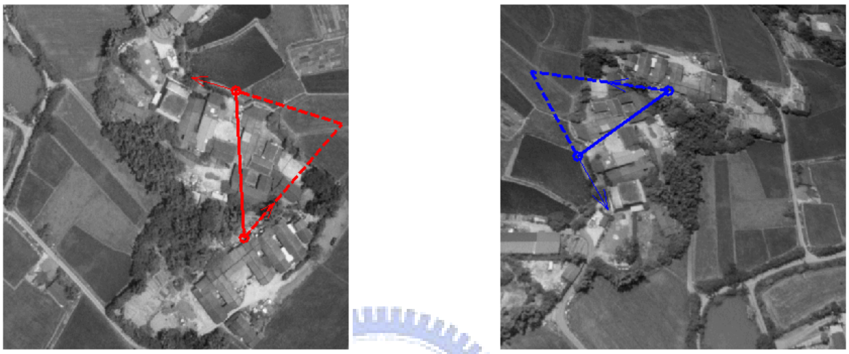
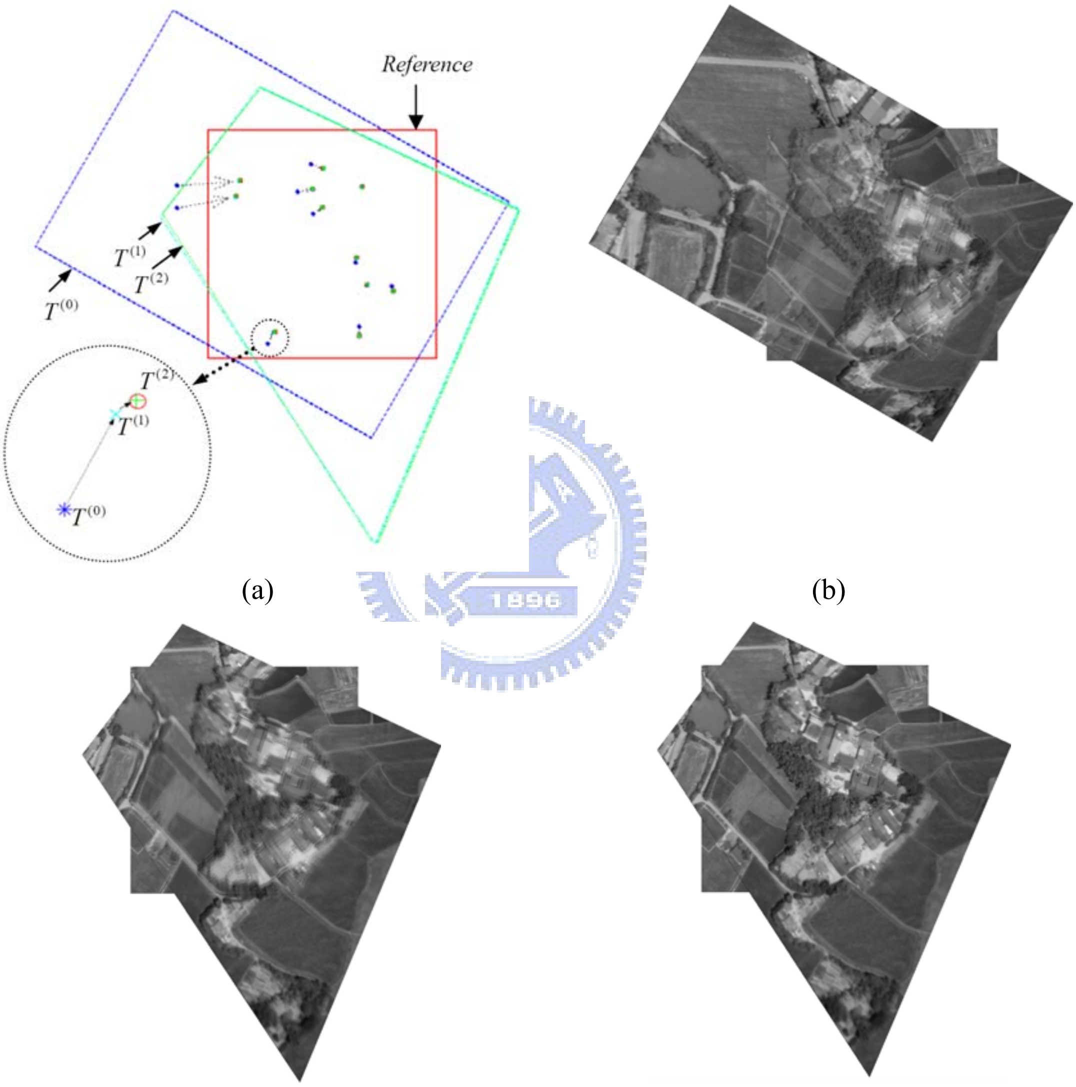

(a) (b)

Fig. 5.4: (a) The reference image. (b) The synthetic sensed image.

The reference features points are extracted using the Gabor filtering technique. A reference matching database is constructed offline using the five planning strategies. Given the sensed image the feature points are extracted first. Then the online registration process is invoked to register the two images. The first starting reference point pair is fetched from the reference matching database and the corresponding sensed point pair is found right away in the case, since this feature point pair is in the overlapping area. Both pairs are shown in the images as the two superimposed triangles. They lead to an affine transformation $T^{(0)}$. Then, the transformation model is updated by the iterative algorithm IMU and converges in two iterations.

Table 5.2 lists three estimated transformation matrices $T(c)$ for c = 0, 1, 2. To demonstrate how the transformation matrix converges, Fig. 5.5(a) shows the feature points and the image boundaries for the reference image and the three transformed sensed images using $T(c)$, $c = 0, 1, 2$. Furthermore, Figs. 5.5(b)–5.5(d) show the registration results under the three transformation models. The RMSE of distances between the sixteen matched point pairs is 0.75 pixels, so it implies the final homography model is rather accurate.



(a)

(b)

(c)

(d)

Fig. 5.5: (a) The partial overlapping between the image boundaries of the reference and three transformed sensed images. (b)-(d) The view registration results under $T^{(0)}$, $T^{(1)}$, and $T^{(2)}$, respectively.

TABLE 5.2 THE TRANSFORMATION PARAMETERS PRODUCED IN THE THREE ITERATIONS

	$M_{3\times3}$								
	m_{11}	m_{12}	m_{13}	m_{21}	m_{22}	m_{23}	m_{31}	m_{32}	m_{33}
$T^{(0)}$	-0.6110	-1.4876	664.0500	1.0264	-0.8557	161.9824	0	0	1
$T^{(1)}$	-0.7977	-0.9341	674.8476	1.0080	-0.7286	173.6283	-0.0005	0.0018	1
$T^{(2)}$	-0.7995	-0.9120	682.9316	1.0400	-0.7318	175.3778	-0.0005	0.0020	1

B) Image Noise Resistance

To demonstrate the usefulness of strategy 1 of the offline planning in combating with image noise, we generate 100 noisy reference image copies by adding Gaussian noise with signal-to-noise ratio 6.2 dB to the original reference image shown in Fig. 5.4(a). Fig. 5.6 shows the effect of image noise on the ranking of the reference feature points according to the descending order of the products of normalized energy and orientation factors $E(p_k)O(p_k)$, $k\in\{1, 2, .., n\}$. The horizontal axis indicates the ranking sequence of the reference feature points before the introduction of image noise. For each of the 100 noisy reference image copies, the feature point ranking process is applied. The vertical axis indicates the new ranking number for each feature point in the horizontal ranking sequence. The mean of the new ranking number is indicated by the marker "*", and the corresponding standard deviation of the new ranking number is indicated by the blue vertical bars centered at the mean rank at each horizontal ranking place. We add a dashed line of slope 45^{o} to serve as the reference line for the ranking change evaluation. Any rank marker located above the reference line indicates a ranking setback under the influence of the image noise, any rank marker located below the line indicates a ranking improvement, and any rank marker located on the line indicates no ranking change. The experimental result shows that the new ranking numbers are fairly close to the old ones. Therefore, the ranking based on the product of $E(p_k)O(p_k)$ is fairly stable in the presence of image noise.

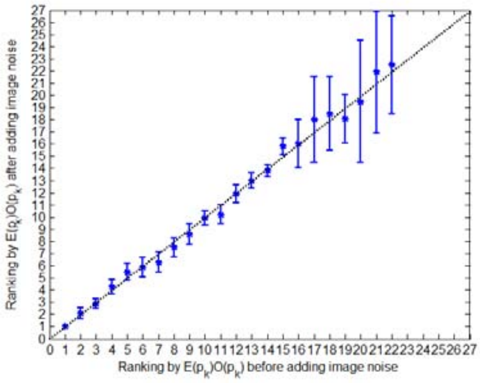
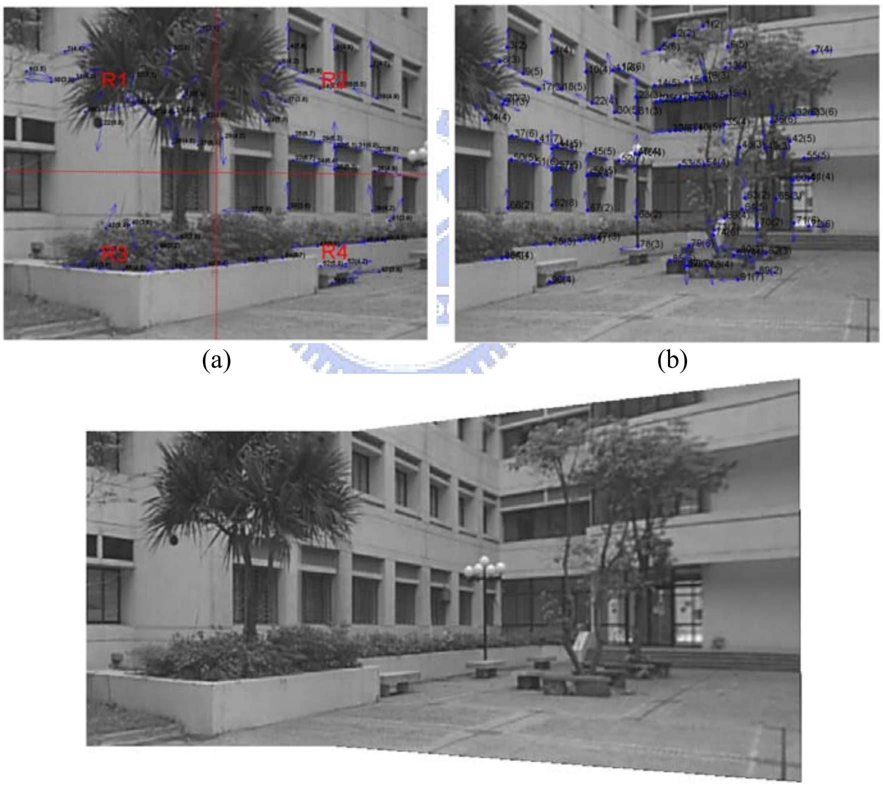


Fig. 5.6: The effect of image noise on the ranking of reference feature points according to the product of normalized energy and orientation factor $E(p_k)O(p_k)$, $k \in \{1, 2, .., n\}$. (See the text).

C) The Efficiency of Online Registration between Two Partially Overlapped Images

In this experiment we use two types of images, building, and landscape painting, to demonstrate the capability of our method in handling the registration of two partially overlapped images. Figs. 5.7(a) and 5.7(b) show the two building images superimposed with the two partitioned lines of the entire image and the labels of extracted feature points. The left image serves as the reference image. It can be seen that the overlapping area contains R_2 and R_4. From Table 5.3, we can explain the importance of using $\{OI_i\}_{i=1,2,\ldots,6}$ to guide the registration process. Initially, we select SPP_6 based on the maximum OI_i value and fetches the first sorted point pair (#40, #57) from it. The processing results show $|CPS| = 0$ and no matched point pairs are found in the sensed image for either reference point of the pair (#40, #57). This indicates that the region pair (R_3, R_4) of SPP_6 is not totally in the overlapping area. Then all entries in the six lists $\{SPP_i\}_{i=1,2,\ldots,6}$ involving one of the reference points #40 and #57 will be removed, and the corresponding overlap indices $\{OI_i\}_{i=1,2,\ldots,6}$ are updated accordingly. Next, we select SPP_1 whose updated OI index is the largest and the leading point pair (#20, #7) of SPP_1 is fetched. The online registration process fails again with a final size

|*CPS*| = 0. It indicates the overlapping area is not found yet. The third attempt chooses SPP_5 whose updated *OI* index is the largest and the reference point pair (#18, #38) is fetched from SPP_5. These two reference points immediately lead to a successful registration with a final size |*CPS*| being 23. Thus, after three attempts (< 5) we find the point pair (#18, #38) that is totally in the overlapping area (R_2, R_4). The execution time for this on-line registration process is 0.312 seconds. Fig. 5.7(c) shows the final registration result of the reference and sensed images.



(a) (b)

(c)

Fig. 5.7: (a)-(b) Two real building images that are partially overlapped. (c) The final registration result.

TABLE 5.3 THE STATISTIC OF THE ONLINE REGISTRATION PROCESS FOR REGISTERING TWO BUILDING IMAGES

Iteration	The source	Point labels of the pair (region involved)	# of Matched points ($N(p_i)$, $N(p_j)$)	Size of *CPS*	Overlap index					
					OI_1	OI_2	OI_3	OI_4	OI_5	OI_6
1	SPP_6	(40(R_3), 57(R_4))	(0, 0)	0	1	1	1	1	1	1
2	SPP_1	(20(R_1), 7(R_2))	(0, 2)	0	1	0.95	0.79	0.76	0.83	0.47
3	SPP_5	(18(R_2), 38(R_4))	(1, 3)	23	0.8	0.75	0.67	0.76	0.83	0.47

We apply the online registration process to another set of three synthetic landscape images shown in Figs. 5.8(a)-5.8(c). The reference image is given in Fig. 5.8(b). The final registration result is given in Fig. 5.8(d).

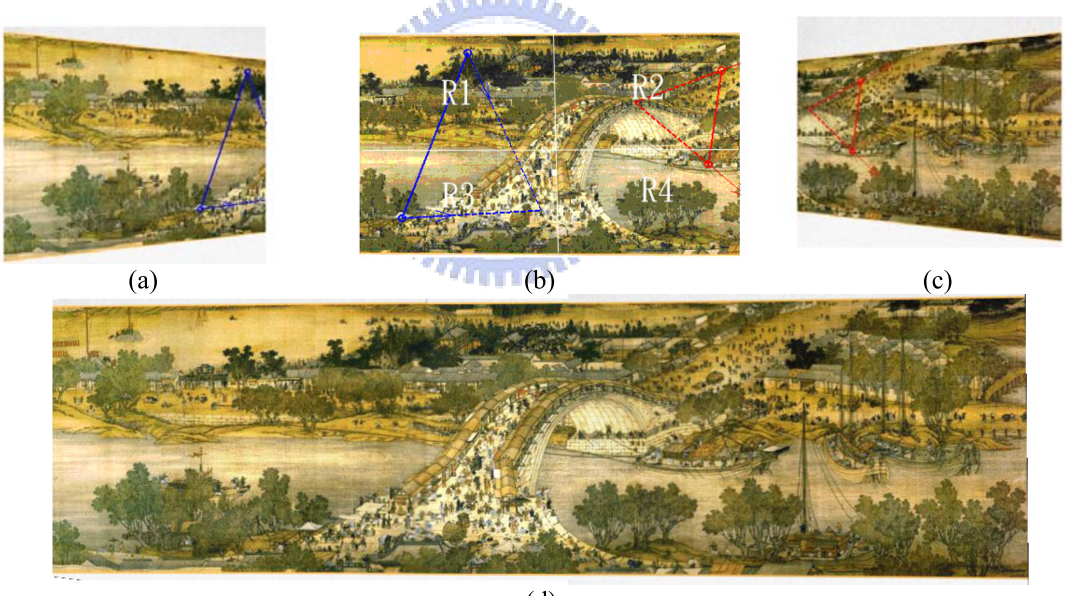


Fig. 5.8: (a) -(c) Three synthetic landscape images used for view registration. (d) The final registration result.

Table 5.4 gives the respective registration efficiencies with and without the five off-line planning strategies. We list the total number of attempts to fetch a reference point pair (*PP*) from the reference matching database *SPP* to complete a successful view registration. We also record the total computational time (T) taken to complete a successful view registration. Without the use of planning strategies, only the IMU algorithm will be employed to find a solution model $T^{(r)}$ using a random drawing of two starting reference points from the set of all possible combinations of the reference point pairs. The model $T^{(r)}$ is correct, if $|CPS^{(r)}|$ is greater than a specified threshold. The IMU process is repeated until a successful view registration is completed (note the reference and sensed images are well overlapped in both cases). The registration statistics are collected for 100 successful runs. Let PP_{avg} denote the average numbers of attempts to randomly draw a reference point pair until a successful view registration is completed and let T_{avg} be the average of the computer execution time taken for completing a successful view registration. The results indicate our method can cut down the computer execution time by using the offline planning strategies. The time reduction benefitted from offline planning strategies is larger, when there are more feature points in the given pair of images.

TABLE 5.4 REGISTRATION EFFICIENCY COMPARISON WITH AND WITHOUT THE OFF-LINE PLANNING STRATEGIES

Image type	Reference image	Sensed image	Registration with offline planning		Registration without offline planning	
			PP	T (sec)	PP_{avg}	T_{avg} (sec)
Building	Fig. 5.7(a)	Fig. 5.7(b)	3	0.312	17.42	6.24
Landscape	Fig. 5.8(b)	Fig. 5.8(c)	5	0.532	24.76	13.04

From our experience the online registration process generally obtain a correct solution within 5 iterations. To put into a more formal statement, under the assumptions that the invariant feature points can be reliably extracted by the feature extractor and that the overlapping area covers at least two sub-regions (a 50% overlapping area ratio), the online registration process will find the overlapping area between the reference and sensed images using the *OI* index within a finite number of attempts (4, most of the time in our case). The ensuing view registration will succeed, since the first sorted point pair fetched from the database *SPP* is totally in the overlapping area and will find a correct matched pair in the sensed image. If the database access number, denoted by N_{OLRP}, exceeds a specified bound (5 in our case), it is likely that the two images are not overlapped at all or only slightly overlapped. So the registration process should be terminated. Of course, we can increase the bound on N_{OLRP} when considering those cases with an overlapping percentage less than 50%.

5.7 Analysis of the Algorithm Computational Performance

The time complexity of the online view registration algorithm does not include the off-line reference matching database construction time. The iterative model updating (IMU) process is a main sub-task of the on-line registration process. The time complexity of the IMU process is first given as follows:

Step 1: Data initialization: $T_{initialization}$.

Step 2: Transform each point of the sensed feature point set to the reference image space and use the table lookup to find any possible matched reference point and, if so, include the matched point in $CPS^{(c)}$: $m(T_{point\text{-}transform} + T_{table\text{-}lookup} + T_{similarity\text{-}measure} + T_{new\text{-}point\text{-}inclusion})$ where m is the size of the sensed feature point set. (The feature point extraction time is not counted in the time complexity of the registration algorithm; refer to Table 5.1.)

Step 3: Check the stopping condition to determine if a final solution is obtained: $T_{stopping\text{-}check}$

Step 4: Compute the new transformation model using the updated $CPS^{(c)}$: $T_{model\text{-}estimation}$

Assume the IMU process stops after a total of N_{IMU} iterations; N_{IMU} is bounded by c_{max} ($c_{max} = 5$ as described in Section 5.2). The total execution time of the IMU process is given by $T_{IMU} = N_{IMU} \times [\ T_{initialization} + m(T_{point\text{-}transform} + T_{table\text{-}lookup} + T_{similarity\text{-}measure} + T_{new\text{-}point\text{-}\ inclusion}) + T_{stopping\text{-}check} + T_{model\text{-}estimation}]$. Only the term in the inner bracket depends on m and the rest are fixed. Therefore, the time complexity of algorithm IMU is of order $O(m)$.

The time complexity of the on-line registration process is given as follows:

Step 1: Fetch the first element (p_{s,i_s}, p_{t,i_t}) from SPP_i with a maximum OI value, $i = 1, 2, 3,..,6$: T_{SPP}.

Step 2: Find the matched point sets CM_s and CM_t with respect to p_{s,i_s} and p_{t,i_t} : $2 \times m \times T_{similarity\text{-}measure}$, where $T_{similarity\text{-}measure}$ is the time for computing the normalized cross correlation between two feature vectors.

Step 3: For a pair $(p_{s,i_s}, p_{t,i_t}) \in SPP$ and its matched pair (q_k, q_l) from the set $\{CM_s \times CM_t\}$

(1) Find the approximate transformation matrix $T^{(0)}$: $T_{model\text{-}estiamtion}$.

(2) Apply the IMU algorithm to obtain the outputs $CPS^{(f)}$ and $T^{(f)}$: T_{IMU}.

(3) Check for the stopping condition: $T_{stopping\text{-}check}$.

Step 4: Update SPP_i and OI_i $i = 1, 2, 3,..,6$: $T_{database\text{-}update}$.

Step 5: Check for the failure condition: $T_{stopping\text{-}check}$.

Assume N_{OLRP} is the total number of attempts to fetch a point pair from the reference matching database SPP for a successful view registration. N_{OLRP} is bounded by 5, as explained above. The computer execution time of the on-line registration process is $T_{OLRP} = N_{OLRP} \times [T_{SPP}$

$+2\times m\times T_{similarity\text{-}measure} + |CM_s|\times|CM_t|\times(T_{model\text{-}estiamtion} + T_{IMU} + T_{stopping\text{-}check}) + T_{database\text{-}update}+ T_{stopping\text{-}check}]$ where the sizes of $|CM_s|$ and $|CM_t|$ are some fixed-sized (e.g., 5-element) subsets of the sensed feature point set. Among the computation times on the right-hand side of the equation, only the second term ($2\times m\times T_{similarity\text{-}measure}$) and the fourth term ($T_{IMU}$) are proportional to m (refer to T_{IMU} described above) and the other terms are deterministic and relatively small. Thus the overall time complexity of the online registration algorithm is of order $O(m)$.

Chapter 6

Conclusions and Future Work

6.1 Summary

In this dissertation, three themes are addressed. In Chapter 3 a new region descriptor, ZM phase, is presented which is robust to common photometric and geometric transformations. A method for an accurate and robust estimation of the rotation angle between two matching regions is described which is implemented in the continuous angle domain without the need of specifying a discrete angle histogram bin resolution. Then a measure for image similarity matching is expressed by a weighted, normalized phase difference. The proposed descriptor is compared with five popular descriptors, SIFT, PCA-SIFT, GLOH, steerable filter, and complex moments, based on the precision-recall criterion with respect to a number of important system parameters. There are more than 15 million region pairs analyzed. The results show that the proposed ZM phase has the leading performance under all photometric and geometric transformations for all textured scenes. As for the structured scenes, the ZM phase has the best performances under image blur and nonlinear lighting, but is comparable to the SIFT-based descriptors under other transformations when the values of 1-precision are small. The analyses on the performance evaluation results are given to account for the performance discrepancy. First, the descriptor performance depends on the estimation accuracy of the rotation angle between two matching regions. Table IV shows the rotation angle estimation error of the ZM phase is better than that of SIFT. Second, the feature dimensionality and feature orthogonality also affect the descriptor performance. Third, the ZM phase is more robust than SIFT-based descriptors under the non-uniform image intensity fluctuation.

In Chapter 4, we extend the proposed ZM phase descriptor to present a new recognition method of logos imaged by mobile phone cameras which can be incorporated with mobile phone services for use in enterprise identification, corporate website access, traffic sign reading, security check, content awareness, and the related applications. The proposed method is compared with three major existing methods, IZMD, EHD, and Ring projection. The logo classification and retrieval experimental results show that the proposed ZM phase method has the best performance under the typical photometric and geometric transformations encountered when using a handheld mobile phone camera operating in the daytime or nighttime. Furthermore, an analysis on the performance evaluation results is given to account for the performance discrepancy between the four different methods.

In Chapter 5, we have developed a new view registration method that consists of two parts: offline planning process and online registration process. Five planning strategies are presented to construct a reference matching database offline. This database is essential to the reduction of time complexity of the online registration, in particular when the reference and sensed images are partially overlapped. This is because we can organize the reference feature points beforehand, aiming at tackling the various problems encountered including image deformation, image noise, point matching ambiguity, model estimation complexity, and partial image overlapping, etc. That is, we take advantage of the preprocessing of the reference image to guide the online registration. Computer simulation results demonstrate the desirable features of our method. A computational complexity analysis is also given, which indicates the time complexity of our online registration algorithm is of order $O(m)$ where m is the size of feature points in the sensed image.

6.2 Future Research

Some research topics for future work are proposed.

(1) Symmetric information extraction

Since symmetry exists widely in the real world, the symmetry detection and localization of symmetry axes is significance for understanding and interpreting the images. The Zernike moments are very suitable for the symmetry detection due to their symmetric and periodical properties. We are currently develop a novel approach which transforms the 2D symmetric image into a 1D periodic curve based on the symmetric properties of the ZMs function. In this way, the symmetric type (rotational or reflection symmetry), fold number and the fold axes can be plainly determined by finding the periodic information from the transformed 1D curve. Furthermore, a unique solution of the rotation angle for a given gray-level or binary image can also be determined.

(2) Content-based image retrieval (CBIR)

We plan to extend the proposed ZM phase-based descriptor for the application of content-based image retrieval. To do this, the interests of regions for a set of labeled training images are first detected and their descriptors are constructed by the proposed ZM phase approach. The collected descriptors are grouped via clustering, and the set of cluster centers are vector quantized. Then a set of vocabularies is established as a set of cluster centers. To organize a codebook, the visual words are constructed from the set of vocabularies. The codebook is stored as an inverted file or a hash table, called the gallery image database. During the image retrieval stage, the corresponding visual vocabularies, visual words are generated for a query image, and then the images in the gallery database are ranked with respect to the query visual word. The most similar gallery images are then output as the query result.

REFERENCES

[1] L. Van Gool, T. Moons, and D. Ungureanu, “Affine/photometric invariants for planar intensity patterns,” in *Proc. Fourth European Conference on Computer Vision,* Vol. II, pp. 642-651, 1996.

[2] C. Schmid and R. Mohr, “Local grayvalue invariants for image retrieval,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 5, pp. 530-535, 1997.

[3] S. Lazebnik, C. Schmid, and J. Ponce, “A sparse texture representation using local affine regions,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp.1265–1278, 2005.

[4] T. Ueshiba and F. Tomita, “Plane-based calibration algorithm for multi-camera systems via factorization of homography matrices,” *Proc. Int’l Conf. Computer Vision,* vol. 2, pp. 966-973, 2003.

[5] J. Matas, O. Chum, M. Urban, T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” *Image and Vision Computing* , vol.22, pp.761–767, 2004.

[6] C. Harris and M. Stephens, “A combined corner and edge detector,” *Alvey Vision Conf.*, pp. 147-151, 1988.

[7] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int’l J. Computer Vision*, vol. 60, no.2, pp. 91–110, 2004.

[8] T. Tuytelaars and L. Van Gool, “Matching widely separated views based on affine invariant regions,” *Int’l J. Computer Vision*, vol. 59, no. 1, pp. 61-85, 2004.

[9] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” *Int’l J. Computer Vision* , vol. 60, no. 1, pp. 63–86, 2004.

[10] T. Lindeberg, and J. Garding, “Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure,” *Image and Vision Computing*, vol. 15, no. 6, pp. 415–434, 1997.

[11] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, and J. Matas, “A comparison of affine region detectors,” *Int’l J. Computer Vision*, vol. 65, no. 1/2, pp. 43–72, 2005.

[12] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

[13] Jing Li, Nigel M. Allinson, "A comprehensive review of current local features for computer vision," *Neurocomputing*, vol. 71, pp.1771– 1787, 2008.

[14] W. Freeman, E. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell*. Vol.13 no.9 pp. 891–906, 1991.

[15] T. S. Lee, "Image representation using Gabor wavelets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.18, no. 10, pp. 959-97, 1996.

[16] Janne Heikkilä, "Pattern matching with affine moment descriptors," *Pattern Recognition*, vol.37, pp.1825 – 1834, 2004.

[17] Dengsheng Zhang, Guojun Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 27, pp. 1–19, 2004.

[18] S. Paschalakis and P. Lee, "Pattern recognition in grey level images using moment based invariant features," *Proc. Seventh International Conference on Image Processing and Its Applications*, vol. 1, pp.245-249, 1999.

[19] F. Schaffalitzky and A. Zisserman, "Multi-view matching for unordered image sets," in *Proc. Seventh European Conf. Computer Vision*, pp. 414-431, 2002.

[20] C.-H. Teh, R.T. Chin, "On image analysis by the methods of moments," *IEEE Trans. Pattern Anal. Mach. Intell*. vol. 10, no. 4, pp. 496–513, 1988.

[21] B. S. Manjunath, J.-R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and Texture Descriptors", *IEEE Trans. Circ. Syst. Video Technol.,* vol.11, no.6, pp.703–715, 2001.

[22] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 506-513, 2004.

[23] L. M. J. Florack, J. Koenderink, B. M, Ter Haar Romeny, J. J. Koenderink and M. A. Viergever, "General intensity transformations and differential invariants," *J. Mathematical Imaging and Vision*, vol. 4, no. 2, pp. 171-187, 1994.

[24] A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp.489–497, 1990.

[25] W.Y. Kim, and Y.S. Kim, "A region-based shape descriptor using Zernike moments," *Signal Processing: Image Communication*, vol. 16, pp. 95-102, 2000.

[26] S. K. Hwang, M. Billinghurst and W. Y. Kim, “Local descriptor by Zernike moments for real-time keypoint matching,” *IEEE Congress on Image and Signal Processing*, pp. 781–785, 2008.

[27] Y. Xin, M. Pawlak, and S. Liao, “Accurate computation of Zernike moments in polar coordinates,” *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 581–587, 2007.

[28] H. Lin, J. Si and G. P. Abousleman, “Orthogonal rotation-invariant moments for digital image processing,” *IEEE Trans. Image Process.*, Vol. 17, No. 3, pp. 272–282, 2007.

[29] W.Y. Kim, and Y.S. Kim, “Robust rotation angle estimator,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 768–773, 1999.

[30] http://www.robots.ox.ac.uk/~vgg/research/affine/.

[31] Z. Chen and H.L. Chou, “A novel 3D planar object reconstruction from multiple uncalibrated images using the plane-induced homographies,” *Pattern Recognition Letters*, vol. 25, no. 12, pp. 1399-1410, 2004.

[32] M. K. Hu, “Visual pattern recognition by moment invariants,” *IRE Trans. Inf. Theory*, pp. 179-187, 1962.

[33] A. Baumberg, “Reliable feature matching across widely separated views,” in *Proc. IEEE Int’l Conf. Computer Vision and Pattern Recognition*, pp. 774-781, 2000.

[34] A. V. Oppenheim and J. S. Lim, “The importance of phase in signals,” in *Proc. Of the IEEE*, Vol. 69, No. 5, pp. 529-550, 1981.

[35] D. J. Fleet and A. D. Jepson, “Stability of phase information,” *IEEE Trans. Pattern Anal. Mach. Intell.,* vol.15, no.12, pp.1253 – 1268, 1993.

[36] D. J. Fleet and A. D. Jepson, “Computation of component image velocity from local phase information,” *International Journal of Computer Vision*, vol.5, no.1, pp. 77- 104, 1990.

[37] G. Carneiro and A. D. Jepson. “Multi-scale phase-based local features,” in *Proc. IEEE Int’l Conf. Computer Vision and Pattern Recognition*, pp. I/736– I/743, 2003.

[38] S. Winder and M. Brown, “Learning local image descriptors,” in *Proc. IEEE Int’l Conf. Computer Vision and Pattern Recognition*, pp.1 – 8, 2007.

[39] M. R Teague, “Image analysis via the general theory of moments,” *J. Opt. Soc. Am.*, vol. 70, no. 8, 1980, pp. 1468-1478.

[40] G. Amayeh, A. Erol, G. Bebis, and M. Nicolescu. “Accurate and efficient computation of high order Zernike moments,” *First Int. Sym. on Vision and Computation, NV, USA*, pages 462–469, 2005.

[41] S. K. Hwang and W. Y. Kim, “A novel approach to the fast computation of Zernike moments,” *Pattern Recognition*, vol. 39, no. 11, pp. 2065-2076, Nov. 2006.

[42] L. Kotoulas and I. Andreadis. ”Real-time computation of Zernike moments”. *IEEE Trans. on Circuits and Sys. for Video Tech.*, 15:801–809, 2005.

[43] A. A. Salah, E. Alpaydin and L. Akarun, “A selective attention-based method for visual pattern recognition with application to handwritten digit recognition and face recognition”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp.420–425, 2002.

[44] J Himberg, K Korpiaho, H Mannila, J Tikanmaki, H, “Time series segmentation for context recognition in mobile devices”, *In Proc. Int’l Conf. Data Mining*, pp. 203–210, 2001.

[45] G Fritz, C Seifert, L Paletta, “A mobile vision system for urban detection with informative local descriptors” *In Proc. Int’l Conf. Computer Vision Systems*, pp. 30–38, 2006.

[46] E Bruns, B Brombach, T Zeidler, O Bimber, “Enabling mobile phones to support large-scale museum guidance”, *IEEE multimedia*, pp. 16–25, 2007.

[47] E. Baratis, E. G.M. Petrakis and E. Milios, “Automatic website summarization by image content: a case study with logo and trademark images,” *IEEE Trans. Knowledge and Data Engineering*, vol. 20, no. 9, pp. 1195-1204, 2008.

[48] G. Zhu and D. Doermann, “Automatic document logo detection,” *In Proc. Int’l Conf. Document Analysis and Recognition*, vol.2, pp. 864 – 868, 2007.

[49] G. Cui, L. Chen, and J. Li, “Billboard advertising detection in sport tv,” *In Proc. Int’l Conf. Signal Processing and Its Applications,* pp.537-540, 2003.

[50] P. Nieto, J.R. Cózar, J.M. González-Linares, N. Guil, *“*A TV-logo classification and learning system,” *In Proc. Int’l Conf. Image Processin*g, pp. 2548 - 2551, 2008.

[51] Wang Jinqiao, Liu Qingshan, Duan Lingyu, Lu Hanqing, and Xu Changsheng, “Automatic TV logo detection, tracking and removal in broadcast video,” *Multimedia*

Modeling (2), pp. 63–72, 2007.

[52] W. Yunqiong, L. Zhifang, and X. Fei, “A fast coarse-to-fine vehicle logo detection and recognition method,” *In Proc. Int’l Conf. Robotics and Biometrics*, pp. 691-696, 2007.

[53] L. Xia, F. Qi, and Q. Zhou, “A learning-based logo recognition algorithm using SIFT and efficient correspondence matching,” *In Proc. Int’l Conf. Information and Automation*, pp. 1767-1772, 2008.

[54] M. J. Swain and D. H. Ballard, “Color indexing,” *Int. J. Comput. Vis.*, vol. 7, pp. 11–32, 1991.

[55] B. S. Manjunath and W. Y. Ma, “Texture features for browsing and retrieval of image data,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol.18, pp. 837–841, Aug. 1996.

[56] A. Hesson and D. Androutsos, “Logo classification using Haar wavelet co-occurrence histograms,” *In Proc. Canadian Conference on Electrical and Computer Engineering,* pp.927-930, 2008.

[57] S. Li, M.-C. Lee, and C.-M. Pun, “Complex Zernike moments features for shape-based image retrieval,” *IEEE Tran. Systems, Man, and Cybernetics—Part A: Systems and Humans*, vol. 39, no. 1, pp.227 - 237, 2009.

[58] C. S. Won, D. K. Park, and S.-J. Park, “Efficient use of MPEG-7 edge histogram descriptor,” *ETRI Journal*, vol.24, no.1, pp. 23-30, 2002.

[59] D.-M. Tsai and C.-H. Chiang, “Rotation-invariant pattern matching using wavelet decomposition”, *Pattern Recognition Letters*, vol.23 pp.191-201, 2002.

[60] L. G. Brown, “A survey of view registration techniques,” *ACM Comput. Surv.* vol.24, no.4, pp.335-376, 1992.

[61] J. B. A. Maintz and M. A. Viergever, “A survey of medical view registration,” *Med. Image Anal.,* vol.2, no.1, pp.1–37, 1998.

[62] T. Mäkelä, P. Clarysse, O. Sipilä, N. Pauna, Q. C. Pham, T. Katila, and I. E. Magnin, “A review of cardiac image registration methods,” *IEEE Trans. Med. Imaging*, vol.21, no.9,

pp.1011-1021, 2002.

[63] B Zitová and J Flusser, "Image registration methods: a survey," *Image Vision Comput.*, vol.21, pp. 977-1000, 2003.

[64] J. Ton and A. K. Jain, "Registering Landsat images by point matching," *IEEE Tran. Geosci. and Remote Sensing*, vol.27, no.5, pp.642-651, 1989.

[65] L.M.G. Fonseca and M.H.M. Costa, "Automatic registration of satellite images", *In Proceedings of Brazilian Symposium on Computer Graphics and Image Processing X*, pp. 219 –226, 1997.

[66] Z. Yang and F. S. Cohen, "Image registration and object recognition using affine invariants and convex hulls," *IEEE Trans. Image Process*, vol.8, no.7, pp. 934-946, 1999.

[67] G. Lei, "Recognition of planar objects in 3-D space from single perspective views using cross ratio," *IEEE Trans. Robot. Automat.*, vol.6, no.4, pp. 432-437, 1990.

[68] H. Lamdan, J. T. Schwartz and H. J. Wolfson, "Affine invariant model-based object recognition," *IEEE Trans. Robot. Automat.*, vol.6 no.5, pp. 578-589, 1990.

[69] D. F. Huber and M. Hebert, "Fully automatic registration of multiple 3D data sets," *Image Vision Comput.*, vol.21, pp.637-650, 2003.

[70] Q. Zheng and R. Chellappa, "Automatic feature point extraction and tracking in image sequences for arbitrary camera motion," *Int. J. Comput. Vision*, vol.15, pp. 31–76, 1995.

[71] C. Shekhar, V. Govindu, and R. Chellappa, "Multi-sensor view registration by feature consensus", *Pattern Recognit.*, vol.32, pp.39–52, 1999.

[72] F. Ola and J. A. Marchant, "Matching feature points in image sequences through a region-based method," *Comput. Vis. Image Und.*, vol.66, no.3, pp.271-285, 1997.

[73] P. Bao and D. Xu, "Complex wavelet-based image mosaics using edge-preserving visual perception modeling," *Computer & Graphics*, vol.23, pp.309-321, 1999.

[74] Martin A. Fischler and Robert C. Bolles, "Random sample consensus: a paradigm for

model fitting with applications to image analysis and automated cartography," *ACM Commun.*, vol.24, no.6, pp.381-395, 1981.

[75] Z. Zhang, R. Deriche, O. Faugeras, Q. T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epiploar geometry," *Artif. Intell.*, vol.78, pp.87-119, 1995.

[76] B.D. Lucas and T. Kanade, "An iterative view registration technique with an application to stereo vision," *In Proc. Image Understanding Workshop*, pp.121-130, 1981.

[77] S. Christy and R. Horaud, "Euclidean shape and motion from multiple perspective views by affine iterations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.18, no.11, pp. 1098-1104, 1996.

[78] S. K. Sun, Z. Chen and T. L. Chia, "Invariant feature extraction and object shape matching using Gabor filtering," *Recent Advances in Visual Information Systems, Lecture notes in computer science*, vol. 2314, Springer Berlin Heidelberg, pp.95-104, 2002.

[79] F. Isgrò, and M. Pilu, "A fast and robust image registration method based on an early consensus paradigm," *Pattern Recognit. Lett.*, Vol.25, pp.943–954, 2004.

[80] W. Wang and Y. C. Chen, "Image registration by control points pairing using the invariant properties of line segments," *Pattern Recognit. Lett.*, vol.18, pp. 269-281, 1997.

[81] Y. Dufournaud, C. Schmid and Radu Horaud, "Image matching with scale adjustment, " *Computer Vision and Image Understanding*, vol.93, no.2, pp.175-194, 2004.

[82] J. Flusser and T. Suk, "A moment-based approach to registration of images with affine geometric distortion," *IEEE Tran. Geosci. Remote Sensing*, vol.32, no.2, pp.382-387, 1994.

[83] Y. Bentoutou, N. Taleb, K. Kpalma, J. Ronsin, "An automatic image registration for applications in remote sensing", *IEEE Tran. Geosci. Remote Sensing*, vol.43, pp. 2127-2137, 2005.

[84] T. Suk and J. Flusser, "Point-based projective invariants," *Pattern Recognit.*, vol.33, pp.

251-261, 2000.

[85] G. Stockman, S. Kopstein, S. Benett, “Matching images to models for registration and object detection via clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.4, pp. 229–241, 1982.

VITA

NAME: Shu-Kuo Sun

BIRTH: Jan. 23th, 1967, Kaohsiung, R.O.C.

EDUCATION:

B.Sc., Department of Survey Engineering, Chung Cheng Institute of Technology, Sep. 1985 – Jun. 1989.

M.Sc., Department of Electronics Engineering, Chung Cheng Institute of Technology, Sep. 1991 – Jun. 1993.

Ph.D., Department of Computer Science, National Chiao Tung University, Sep. 1999 – present.

AWARDS:

Excellent paper of 2008 IPPR Conference on Computer Vision, Graphics and Image processing (with Prof. Zen Chen).

RESEARCH INTERESTS:

Image Processing Pattern Recognition

Computer Vision Remote Sensing