Wide-baseline stereo for three-dimensional urban scenes

Shu Fei FAN

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

September 2010

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

© 2010 Shufei Fan

THIS PAGE IS INTENTIONALLY LEFT BLANK

Abstract

Like humans, computer vision systems can better infer a scene's 3-D structure by processing its 2-D images taken from multiple viewpoints. While this seems effortless for humans, it is still a challenge for computer vision. Underlying the act of associating the different perspectives is a problem called wide-baseline stereo, which computes the geometric relationship between two overlapping views. Wide-baseline stereo can be problematic when working on images taken of real-life urban environments, due to practical issues such as poor image quality or ambiguity raised by repetitive patterns. We analyze why these factors pose difficulties for current methods and propose principles that can make wide-baseline stereo more effective, in terms of both robustness and accuracy.

We treat wide-baseline stereo as a sequence of three sub-problems: feature detection, feature matching, and fundamental matrix estimation. We propose improvements for each of these and test them on real images of 3-D urban scenes. For feature detection, we demonstrate that when we use both image intensity contrast and entropy-based visual saliency, we are better at repeatably extracting features of a 3-D scene. We use intensity contrast as a cue for obtaining initial feature seeds, which are then evaluated and locally adapted according to an entropy-based saliency measure. We select features with high saliency scores. Experimental comparisons against peer feature detectors show that our method detects more regular structures and fewer noisy patterns. As a result, our method detects features with high repeatability, which is conducive to the subsequent feature matching.

In the case of feature matching, we show that we can match features more robustly when using both local feature appearance and regional image information. We model global image information with a graph, whose nodes contain local feature appearances and edges encode semi-local proximity structure. Working on this graph, we convert traditional feature matching into a graph-matching problem — essentially, we are shifting from a purely local to a context-driven feature matching paradigm. In comparison against local methods, our algorithm performs robustly and is consistently better under difficult wide-baseline conditions, such as repetitive local patterns, under excessive image noise or low resolution inputs.

For the fundamental matrix estimation, we propose to implement a preprocessing step on the feature correspondences before commencing the estimation procedure. This is essentially a registration-based re-alignment on correspondences, where we locally adjust the

position and shape of the feature in one image according to the appearance of its match in the other. Our experiments show that the preprocessing consistently increases efficiency and accuracy of the fundamental matrix estimation.

In summary, we propose a series of algorithms for wide-baseline stereo. Essentially, our methods achieve better robustness and accuracy than current approaches by making use of more image information. By combining entropy-based saliency with intensity contrast, our feature detector is better than its peers at detecting regular man-made structures in the presence of unwanted high frequency patterns regarded as noise. By using neighborhood information, our feature matching method is less sensitive to appearance ambiguity than traditional matching methods. The preprocessing step exploits information contained in both images to refine localization of matched features. These techniques can be especially useful for practical 3-D vision applications, for example, to robustly model or render a 3-D scene based on less-than-ideal input images taken of real-life environments.

Abstract

L'utilisation de plusieurs points de vue d'une scène pour en déterminer sa structure tridimensionnelle est un exercice effectué autant par l'humain que par certains systèmes de vision artificielle. Mais alors qu'il ne requiert aucun effort pour l'humain, il représente un défi pour le domaine de la vision par ordinateur. Un problème sous-jacent à celui d'associer plusieurs perspectives d'une scène est celui de la stéréoscopie pour une longue ligne de base, qui consiste à déterminer les relations géométriques entre deux vues qui se chevauchent. La stéréo pour une longue ligne de base (lorsque les deux points de vue sont éloignés) peut être problématique dans un environnement urbain, en raison d'une qualité parfois pauvre des images, et aussi de l'ambiguïté que peut soulever des formes répétitives. Cette thèse analyse les raisons pour lesquelles ces facteurs peuvent être problématiques pour les méthodes actuelles et propose des principes qui permettent une stéréo plus efficace, autant au point de vue de la robustesse que de la précision.

La stéréo d'images provenant de points de vues éloignés est divisée en trois sous-problémes: la détection de caractéristiques visuelles, leur appariement, ainsi que l'estimation de la matrice fondamentale. Des améliorations sont proposées pour chaque élément, et des expérimentations sur des données réelles de scènes urbaines sont présentées. Pour la détection de caractéristiques, il est démontré que lorsque le contraste en intensité des images ainsi que la saillance visuelle basée sur l'entropie sont utilisés, nous obtenons de meilleurs résultats de détection de caractéristiques de scènes tridimensionnelles. Le contraste en intensité est utilisé pour obtenir des points de départ pour les caractéristiques, qui sont ensuite évalués et adapté localement selon une mesure de saillance basée sur l'entropie. Les caractéristiques ayant obtenues une mesure élevée de saillance sont choisies. Des comparaisons expérimentales avec d'autres méthodes de détection de caractéristiques montrent que la méthode proposée détecte plus de structures régulières et moins de formes bruitées. Par conséquent, la méthode présentée détecte des caractéristiques avec un haut taux de répétabilité, ce qui favorise la procédure d'appariement.

Pour le problème de l'appariement des caractéristiques, il est démontré que l'utilisation de l'apparence des caractéristiques ainsi que d'information autour de ces caractéristiques permet une correspondence plus robuste. L'information globale d'une image est modélisée à l'aide d'un graphe, dont les noeuds contiennent l'apparence locale d'une caractéristique et les arêtes encodent la structure de proximité semi-locale. Le problème d'appariement

des caractéristiques est donc traduit en un problème de couplage de graphes. Le problème n'est donc pas ici traité d'une manière purement locale, puisque le contexte est pris en compte. En comparaison avec les méthodes locales, l'algorithme proposé est robuste et performe mieux dans des conditions difficiles de vues éloignées, avec entre autres des forms répétitives, du bruit excessif dans les images, ou des images d'entrée à basse résolution.

Pour ce qui est de l'estimation de la matrice fondamentale, il est proposé d'implémenter une pré-condition sur la correspondance des caractéristiques avant de commencer la procédure d'estimation. Il s'agit en fait d'un réalignement des correspondances basé sur l'appariement d'images. La position et la forme des caractéristiques sont ajustées localement dans une image selon l'apparence de la caractéristique correspondante dans l'autre image. Les résultats d'expérimentation montrent que la pré-condition augmente l'efficacité et la précision de la procédure d'estimation de la matrice fondamentale.

En résumé, cette thèse propose une série de techniques pour la stéréo lorsque la ligne de base est grande. Les méthodes présentées utilisent plus d'information provenant des images et permettent ainsi une plus grande robustesse et une meilleure précision. La saillance basée sur l'entropie permet une meilleure détection de structures régulières construites par l'humain lorsqu'il y a présence de formes à haute fréquence considérées comme du bruit. En utilisant l'information du voisinage, l'appariement de caractéristiques est moins sensible à des ambiguïtés d'apparence. L'étape de pré-condition exploite l'information contenue dans les deux images afin de raffiner la localisation de caractéristiques correspondantes. Ces techniques peuvent être particulièrement utiles pour des applications pratiques de vision tridimensionnelle, par exemple pour modéliser ou rendre de façon robuste une scène tridimensionnelle dans un contexte réaliste et non idéal.

Dedication

To Fang: your love and support have made this thesis possible.

To Boyuan and Siyuan: you are my source of inspiration.

Acknowledgements

This thesis owes much to my advisor, my mentors, colleagues and friends during my work at the Center for Intelligent Machines of McGill.

I thank my advisor, Frank Ferrie, for guiding me through this expedition. Frank's broad interest and strong curiosity in science and technology is mirrored by the diverse avenues pursued by members of the Artificial Perception Lab. I have benefited a lot from this dynamic group. Amongst others, I have learned from Frank the questions to ask in identifying a research problem, the approaches to undertake in solving a scientific problem, and the ways to communicate in spreading my discoveries. I appreciate the encouragement Frank gave to me in exploring problems that deeply interests me. I appreciate Frank's concern for life of his students. His "family first" attitude gave me ample freedom, and happiness, in balancing work with personal life.

I thank my mentors, Kaleem Siddiqi and James Clark. Their courses have led me into the field of computer vision with great enthusiasm. Their advice and feedback have significantly contributed towards my academic progress. In addition, they went into great length to help me apply funds and to provide advice for my career advancement.

I am very fortunate to have worked with wonderful colleagues and friends at CIM, especially Isabelle, Prasun, Karim, John, Andrew, Prakash, Ruisheng, Rupert, Cathy, Matt, Frank Riggi, Rola, Scott, Carmen, Peter, Sandra, Svetlana, Wei Sun, Li Jie, Stephane, Jianfeng, Adriana, and Jerome. My time at McGill is an enjoyable experience because of the friendship you have offered. In many circumstances, many of you provided me generous help, both intellectually and personally. Extra thanks to Prasun, you have given me many wise advices and countless help, on both my academic and personal life. Cathy provided numerous help to me, sacrificing big chunks of her precious time. Amongst others, Cathy proof-read early draft of almost the entire thesis. Isabelle gave me many valuable suggestions and she translated my abstract into French. Discussions with Rupert has always been pleasant and thought-provoking. I enjoy our collaboration that led to an interesting finding (Chapter 5 of this thesis) — thank you Rupert. Wei Sun mentored me on my first computer vision project, Karim challenged me (in a constructive way) to set a high standard for this thesis, Matt is always energetic in sharing his insights on the state-of-the-art, Rola helped me when I need help the most, ..., and the list goes on.

I sincerely thank Cynthia, Jan, Patrick, and Marlene of CIM. You have provided great

support on administrative and computer issues.

Contents

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Wide-baseline stereo . . . 3
1.2.1 Image formation . . . 3
1.2.2 What is wide-baseline stereo . . . 7
1.2.3 The geometry of stereo . . . 8
1.2.4 Three sub-problems of wide-baseline stereo . . . 10
1.2.5 Closely related areas . . . 14
1.3 Problem statement and contributions . . . 18
1.3.1 Objective . . . 18
1.3.2 Contributions . . . 19
1.3.3 Published work . . . 20
1.4 Structure of the thesis . . . 21

2 Background **23**
2.1 Feature detection . . . 23
2.1.1 Early features . . . 24
2.1.2 Considerations with respect to scale and affine-invariance . . . 26
2.1.3 Region boundary based methods . . . 28
2.1.4 Summarizing remarks . . . 28
2.2 Feature matching . . . 29
2.2.1 Higher-level feature based matching . . . 29
2.2.2 Global-optimization based matching . . . 30
2.2.3 Local feature matching . . . 32

2.2.4 Summarizing remarks . . . 33
2.3 Fundamental matrix estimation . . . 34
2.4 Graph matching in computer vision . . . 35

3 Structure guided salient region detector 38
3.1 Background on the entropy based saliency . . . 40
3.2 The Structure Guided Salient Region . . . 41
3.2.1 Representation of the scale and affine invariant features . . . 42
3.2.2 Seeding using local structure . . . 42
3.2.3 Local salient region adaptation . . . 43
3.2.4 Robust histogram estimation and extension to color images . . . 44
3.3 Performance evaluation on planar scenes . . . 45
3.4 Performance evaluation on 3-D scenes . . . 48
3.5 Conclusions . . . 53

4 Context-consistent feature matching 54
4.1 Background . . . 55
4.1.1 Proposed approach . . . 56
4.2 Salient Feature Graph . . . 58
4.2.1 Nodes . . . 59
4.2.2 Edges . . . 59
4.3 Measuring neighborhood similarity using SFG . . . 61
4.3.1 The Neighborhood Transform . . . 62
4.3.2 Computing Context-Consistency . . . 63
4.4 Matching of *SFG*s . . . 66
4.4.1 Problem formulation . . . 66
4.4.2 Overall algorithm . . . 68
4.5 Performance evaluation . . . 70
4.5.1 Data sets . . . 70
4.5.2 Evaluation criteria . . . 71
4.6 Experimental results . . . 73
4.6.1 Comparison with the local method . . . 74
4.6.2 Evaluating key components . . . 83

4.7 Conclusions 87

5 Better correspondence by registration 90
5.1 Related work 91
5.1.1 Robust **F**-estimation 92
5.1.2 Precise **F**-estimation 92
5.2 Our approach 93
5.2.1 Localization refinement by registration 93
5.2.2 Improved **F**-estimation 95
5.3 Experiments 96
5.3.1 Test on feature localization accuracy 97
5.3.2 Improvement on robust inlier detection 98
5.3.3 Improvement in **F**-estimation accuracy 100
5.3.4 Computing time 100
5.4 Conclusions 101

6 Conclusions and future research 103
6.1 Discussion 104
6.2 Future research 106

References 108

List of Figures

1.1 Pinhole camera model 4
1.2 Camera intrinsic factors 5
1.3 Rotation and translation between the world and camera coordinate frames 7
1.4 Epipolar constraint on point correspondence 9
1.5 Ambiguity in the correspondence between two retinal projections 13
1.6 Depth estimation from stereo disparity 16
1.7 An example stereo images of an urban scene 19

2.1 Pyramidal scale-space representation of an image 27

3.1 Stereo images of the *J-scene* 39
3.2 Features detected on the *J-Scene* 39
3.3 *Graffiti* image set 46
3.4 Performance evaluation on planar scene 47
3.5 Hessian-Affine features detected on the *J-Scene* 49
3.6 MSER Features detected on the *J-Scene* 50
3.7 SGSR features detected on the *J-Scene* 51
3.8 Stereo images of a *ZuBuD* scene 53

4.1 The *Invariant Edge* principle 60
4.2 *Neighborhood Transform* 63
4.3 Node similarity measure by *hypothesis-match* 65
4.4 An example *SFG* image modeling 72
4.5 Image set *Wall* and hand-picked features 76
4.6 Feature matching results on *Wall* 77

4.7 Image set of standard scenes 77
4.8 Image set of challenging scenes 79
4.9 Feature matching results for challenging scenes (1) 80
4.10 Feature matching results for challenging scenes (2) 81
4.11 Feature matching results for challenging scenes (3) 82
4.12 Feature matching results for low-resolution images 84
4.13 Feature matching results for noisy images 85
4.14 Evaluating contribution of the *Invariant-Edge* 86
4.15 Evaluating contribution of the *Neighborhood Transform* 88

5.1 Image sets with estimated epipolar lines. 97
5.2 Accuracy comparison result using ground truth fundamental matrix ($\mathbf{F}_{truth}$) 99
5.3 Accuracy comparison result of **F**-estimation. 101

List of Algorithms

1 Context-Consistent Assignment 69
2 Improved algorithm for computing the fundamental matrix. 96

List of Tables

3.1 Feature matching comparison for the *J-Scene* 52
3.2 Feature matching comparison for the *ZuBuD* scene 53

4.1 Feature matching comparison on standard scenes 78

5.1 The comparison result of improvement on robust estimation 99

Chapter 1

Introduction

1.1 Motivation

Modeling and rendering of visual scenes are widely used in areas such as computer graphics, augmented reality, localization and navigation, etc. Traditionally, three-dimensional (3-D) models of scenes are acquired with specialized devices such as laser scanners, which can be costly and intrusive. With increased demand for visually-appealing modeling and rendering of the environment (e.g., building 3-D models for Google Earth, virtually touring landmarks through Microsoft's Photosynth [113], etc.), low-cost model-acquisition using consumer photo or video-cameras has attracted wide attention in the computer vision community [136][1]. This family of techniques typically assembles images taken from different perspectives of a scene and then fuses them into an integral 3-D model, using some geometric and photometric cues embedded in these images.

Wide-baseline stereo plays a critical role in piecing together these perspectives and establishes the geometric foundation for the entire 3-D reconstruction process. The significance of wide-baseline stereo can be seen in the following description. Suppose we are given two images that both "see" a common scene from different viewpoints. Without any knowledge of how the images were acquired — what were the camera parameters and what was the relative pose between the two cameras — we can use wide-baseline stereo to precisely place the two cameras within a common projective 3-D space [50][67]. If more images and certain priors about the cameras' intrinsic parameters are available, we can even upgrade their relationship into the Euclidean 3-D space [134]. Once we have the images' geometric relationship in the Euclidean space, we can use it to compute the 3-D depth of the scene

using multiple-view stereo algorithms [155].

The key to wide-baseline stereo is to establish and analyze a set of correspondences so as to numerically compute the geometry of a pair of cameras [107][200]. 3-D applications have taken two major approaches to establishing feature correspondences. One family of matching methods is through tracking features across video frames [146][136][135]. Since videos are taken at high frame-rate (e.g., 30 frames per second (FPS)), images differ little between neighboring frames. In this case, feature tracking [103][163] is sufficient to deal with the small image variations. One drawback to this method is that a large number of images have to be processed for even a short sequence of video. The other family is to directly match features across widely separated views. Thanks to the maturity of recent feature detection and description techniques (e.g., SIFT [102]), the second method has been increasingly used by applications that reconstruct 3-D scenes using a sparse collection of images [165][1]. Compared with near-frame images for the video tracking scenario, a pair of images of the latter case can differ a great deal in terms of viewpoint angle, scale, and sometimes illumination. As a consequence, local image structures often undergo drastic shape changes and are frequently occluded in the images. To overcome these difficulties, researchers have devised many detectors for repeatedly extracting image points [102][111][114][180], and have proposed many ways for robustly matching these points [143][109][94][138][137][152][102][170].

I am motivated by the current need for effective methods of matching these widely separated views: how to make sure the algorithm can consistently establish a sufficient number of matches over a variety of real-life environments. The first metric here is robustness, i.e., the ability to consistently produce a large number of correspondences while keeping false matches rare. The second metric is accuracy — to what precision can the algorithm put two images into a common 3-D space [111]. By observing how we humans would approach similar tasks, I form principles and formalize them in my methodologies. The goal of this research is to improve overall robustness and accuracy of wide-baseline stereo matching for real-life images. Eventually I hope that these improvements will facilitate practical 3-D vision applications such as 3-D modeling and rendering [165], augmented reality [20], localization and navigation[60][151], and many others.

In the next section, I will formally introduce wide-baseline stereo — what it is, how it is approached, and how it relates to some other fields. Then I will present my objective, as well as my contributions in the Section 1.3, where I will also list the publications related to this research. In Section 1.4, I will layout the overall structure of this thesis.

1.2 Wide-baseline stereo

Before delving into relating images taken by two cameras, we will first have a look at how an image is taken by one camera — the image formation process (Section 1.2.1). In Section 1.2.2, I will give a formal definition of what I mean by wide-baseline stereo. Then, in Section 1.2.3, I will illustrate the geometry of a stereo system and show how the fundamental matrix encodes the epipolar geometry. After that, I outline the steps involved to solve a wide-baseline stereo problem (Section 1.2.4). And finally, I will make connections with some closely related fields in Section 1.2.5.

1.2.1 Image formation

If we neglect camera radial distortion, the model for image formation can be expressed linearly by a camera's intrinsic and extrinsic parameters, using the *camera projection matrix*.

Intrinsic parameters The simplest and most intuitive model for understanding the formation of images is the basic pinhole camera.

According to the model, all lights incident on the image plane pass through a common pinhole. This pinhole is the camera center. Figure 1.1 illustrates the projection of a world point $\mathbf{X}$, through the camera center $\mathbf{C}$, to the point $\mathbf{x} = (u, v, f)$ on image plane π. The image plane is placed in front of the camera center to facilitate illustration.

By similar triangles, we quickly determine that $u = fX/Z$ and $v = fY/Z$. In projective space, this central projection is a linear mapping from three dimensions to two dimensions. Representing the points with homogeneous coordinates, it can be expressed in matrix form

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}. \tag{1.1}$$

In the above model (Equation 1.1), several factors are neglected to simplify description. Refer to the illustration in Figure 1.2. Usually, image coordinates start at one corner of the rectangular image plane. Sometimes, the effective width and height of a pixel might be different due to camera CCD sensors. And, in some rare cases, the x- and y-axes might be

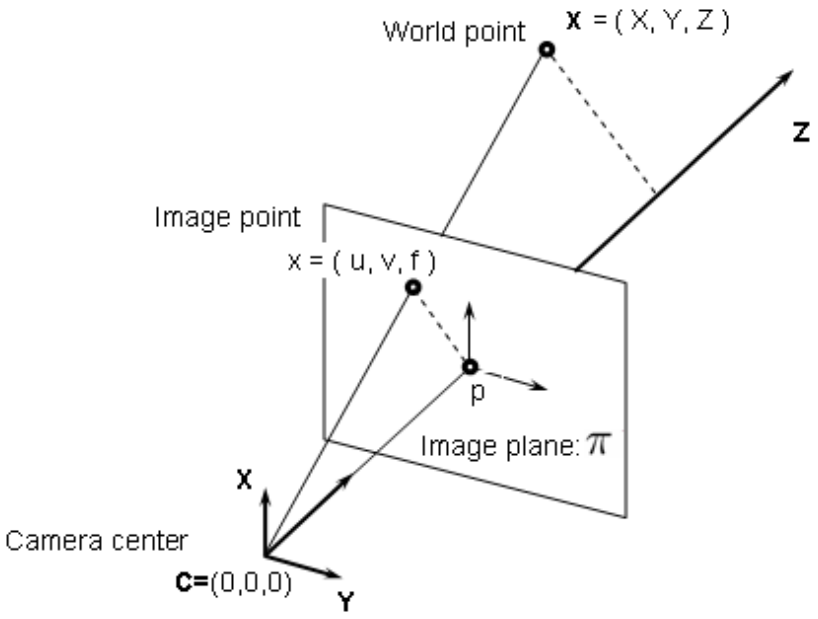


Fig. 1.1 Pinhole camera model. C is the camera center, f is the camera focal length, and p is the principal point (or image center). The camera sits at the origin of the coordinate system and faces z-direction.

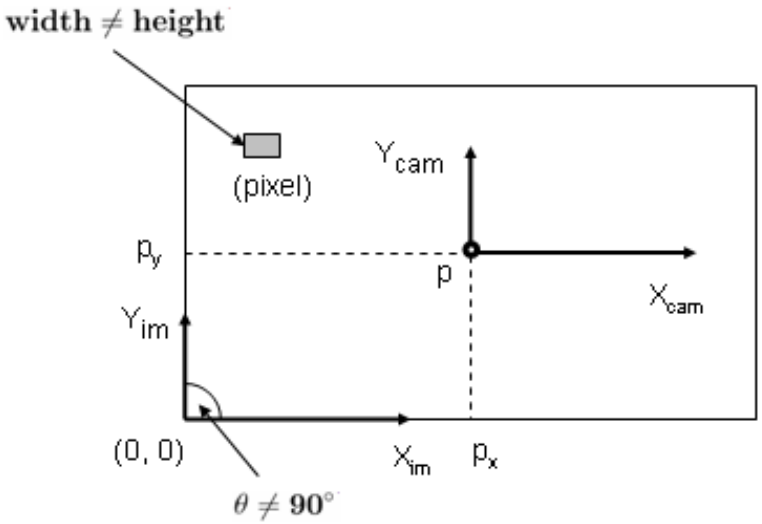


Fig. 1.2 Camera intrinsic factors. This figure shows three factors influencing camera intrinsic parameters: the change from the (X_{cam}, Y_{cam}) coordinate system to the (X_{im}, Y_{im}) coordinates, pixel aspect ratio, and skew of X-Y axis angle.

skewed at an angle that is different from 90°. Taking the above factors into account, the central projection can be generalized to the form (1.2),

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{bmatrix} fm_x & s & o_x & 0 \\ 0 & fm_y & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \tag{1.2}$$

where, m_x and m_y are the numbers of pixels per unit distance along the x and y directions respectively, (o_x, o_y) are the pixel coordinates of the principal point p, and s is the skew parameter. The 3×4 matrix in Equation (1.2) is the *camera projection matrix* $\mathbf{P}$ and it can be decomposed into two parts

$$\mathbf{P} = \mathbf{K}[\ \mathbf{I}_3 \ \ | \ \ \mathbf{0}_3\ ], \tag{1.3}$$

with

$$\mathbf{K} = \begin{bmatrix} fm_x & s & o_x \\ 0 & fm_y & o_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.4}$$

The matrix $\mathbf{K}$ is called the *camera calibration matrix*, which contains all the intrinsic parameters of a camera.

Extrinsic parameters The above discussion considers the lens center of camera as being at the origin and well aligned with the world coordinate system. If the camera is placed at an arbitrary displacement and an arbitrary pose away from the world origin (Figure 1.3), the camera projection model should account for this discrepancy. This information can be conveniently included in the camera projection matrix as follows

$$\mathbf{P} = \mathbf{K}[\ \mathbf{R} \ \ | \ \ \mathbf{t}\ ]. \tag{1.5}$$

In (1.5), the matrix $[\ \mathbf{R} \ \ | \ \ \mathbf{t}\ ]$ contains the camera's extrinsic parameters. $\mathbf{R}$ is its pose represented by a 3×3 rotation matrix, and $\mathbf{t}$ is the 3-D displacement of the camera center relative to the world coordinate frame.

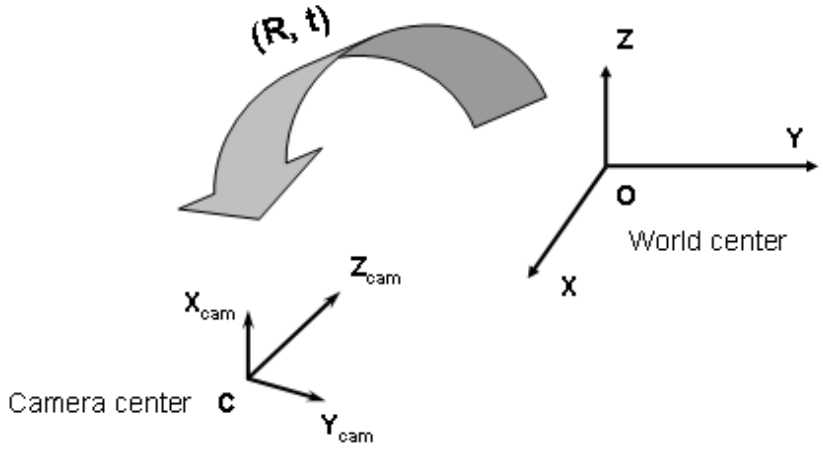


Fig. 1.3 Rotation and translation between the world and camera coordinates.

1.2.2 What is wide-baseline stereo

Wide-baseline stereo, also known as *wide-baseline matching*, uncovers the epipolar geometry between stereo images of a scene by establishing a sparse set of point correspondences. The *baseline* refers to the line joining the centers of the two cameras that took the images ($\mathbf{CC}'$ in the Figure 1.4(a)); and the *wide* indicates that the two cameras are typically widely separated such that the images might undergo some changes in imaging conditions — rotation/translation of cameras, changes of intrinsic camera parameters, and possibly change of illumination. The *epipolar geometry* (cf. definition in Section 1.2.3) is the intrinsic projective relationship between two views of a scene, and depends only on the cameras that took them. Often, the word *wide* conveys another meaning — we know neither the change in camera poses nor their intrinsic parameters. Thus, the principal goal of wide-baseline stereo is to infer the epipolar geometry based solely on the images. Because wide-baseline stereo integrates different perspective views without knowledge of the camera parameters, it enables us to solve a variety of 3-D computer vision problems in real life.

The problem of wide-baseline stereo began to be investigated a little more than ten years ago (e.g., Pritchett and Zisserman [137], Tuytelaars and Van Gool [180]). We should make a clear distinction from the traditional notion of computational stereo (described in Section 1.2.5), which has been researched for more than thirty years (early work includes

[92][109][7]). Although both work on a pair of stereo images, they differ from each other both in their assumptions and in their goals. The computational stereo assumes images are taken by a calibrated stereo system and its goal is to infer depth of the scene from the images, while wide-baseline stereo works on un-calibrated stereo images and its purpose is essentially to calibrate the stereo images, i.e., estimating their epipolar geometry [1]. Thus, wide-baseline stereo matching relieves us from manual calibration of a stereo system, so that image-based 3-D modeling and rendering can be fully automated.

Wide-baseline stereo is the first step of the entire pipeline of structure from motion (SfM) [74][136][66] (note that early SfM algorithms additionally assumed that camera intrinsic parameters are known [100][72][126]). The SfM uses pair-wise epipolar geometry to integrate the entire set of perspective views, and eventually to calibrate all the cameras and to compute the Euclidean 3-D structure of the scene.

Wide-baseline stereo was founded on the theory of the fundamental matrix proposed by Luong [105] (as well as by Faugeras [50][52] and by Hartley et al. [67][68]). The theory states that the epipolar geometry between a pair of stereo images is succinctly encoded in the fundamental matrix, which determines a mapping of corresponding points between two images. The following section will introduce this geometric mapping and its algebraic representation.

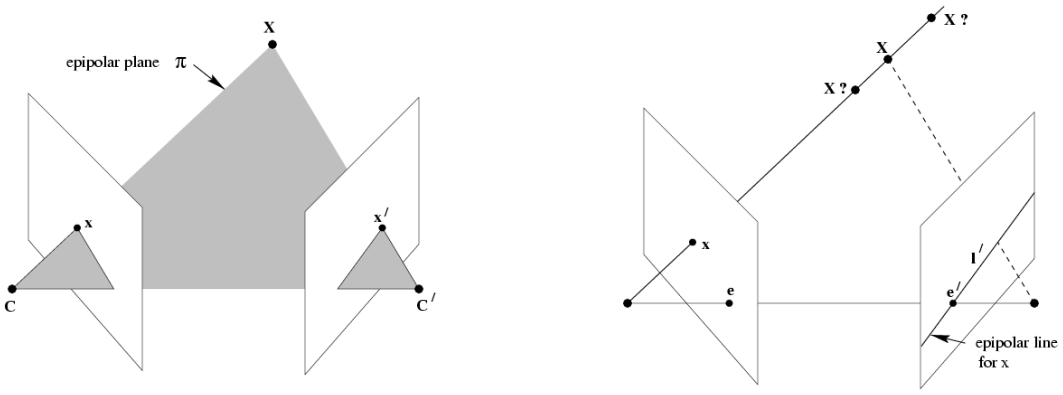
1.2.3 The geometry of stereo

Any two views of a static scene are constrained by a projective relationship — the *epipolar geometry*. This geometry depends on the cameras' intrinsic parameters and their relative pose, and is independent of the scene. However, it can be computed by analyzing images taken of the scene, in the form of estimating an algebraic entity — the *fundamental matrix* $\mathbf{F}$, a 3×3 matrix of rank 2.

Epipolar Geometry Instead of giving a mathematic derivation of how two cameras' parameters exclusively decide the epipolar geometry (cf. Chapter 7 of [175]), I show the *epipolar constraint* with an illustration in Figure 1.4(a). One point $\mathbf{X}$ in 3-D space is imaged by two pinhole cameras centered at $\mathbf{C}$ and $\mathbf{C}'$. Camera $\mathbf{C}$ images $\mathbf{X}$ at $\mathbf{x}$ in the left

[1] Sometimes, the term "wide-baseline stereo" is also used to describe the problem of computational stereo, emphasizing that the calibrated cameras are widely separated, e.g., Strecha et al. [168].

image, while $\mathbf{C}'$ images $\mathbf{X}$ at $\mathbf{x}'$ in the right image. Thus, the baseline $\mathbf{C}\mathbf{C}'$, image points $\mathbf{x}$ and $\mathbf{x}'$, and 3-D point $\mathbf{X}$ are coplanar, on the so-called *epipolar plane* π.



(a) The geometry of two pinhole cameras ($\mathbf{C}$ and $\mathbf{C}'$) shooting a 3-D point $\mathbf{X}$

(b) The epipolar constraint maps a point $\mathbf{x}$ in left image onto a line $\mathbf{l}'$ in the right

Fig. 1.4 Epipolar constraint on point correspondence. Please refer to the text for explanation. (Original Fig-9.1 in Hartley and Zisserman [66], reproduced with permission from the authors.)

Thinking of Figure 1.4(a) from a different perspective, suppose we want to know where the 3-D point $\mathbf{X}$ is through a known image correspondence ($\mathbf{x}$ with $\mathbf{x}'$), and known camera setups ($\mathbf{C}'$ and $\mathbf{C}'$). We can back-project rays $\mathbf{C}\mathbf{x}$ and $\mathbf{C}'\mathbf{x}'$ and intersect them in 3-D space to find the point $\mathbf{X}$. This is called 3-D reconstruction by *triangulation* — one correspondence between two images taken by calibrated cameras uniquely defines a 3-D point.

Now suppose we know neither 3-D point $\mathbf{X}$ nor $\mathbf{x}'$ in the right image (Figure 1.4(b)). 3-D reconstruction would then involve first finding the corresponding point $\mathbf{x}'$ for $\mathbf{x}$ (the *correspondence* problem), and then performing the triangulation. Two planes constrain

the unknown $\mathbf{x}'$ to a line: since $\mathbf{x}'$ lies on both the epipolar plane (as in Figure 1.4(a)) and the image plane ($\mathbf{x}'$ being an image point), $\mathbf{x}'$ can only be somewhere on the intersection of both planes — the line $\mathbf{l}'$. $\mathbf{l}'$ is called the *epipolar line* for $\mathbf{x}$; and the point $\mathbf{e}'$ is the *epipole* of the right image — it is the intersection of the baseline with the image plane. This constraint is called the *epipolar constraint* — it reduces the correspondence problem from searching for $\mathbf{x}'$ over the entire image to searching over the line $\mathbf{l}'$. Every possible location on $\mathbf{l}'$ implies one possible 3-D point $\mathbf{X}$. This is a symmetric relationship — the same illustration can be made to show, on the left image, the epipole $\mathbf{e}$ and the epipolar line $\mathbf{l}$ for $\mathbf{x}'$.

The fundamental matrix F Figure 1.4 shows us that between any pair of images, each point $\mathbf{x}$ in one image can be mapped to a line $\mathbf{l}'$ in the other image. In fact, the line $\mathbf{l}'$ is the projection onto the second image of the 3-D ray generated from $\mathbf{C}$ and $\mathbf{x}$ of the first images .

Algebraically, this mapping is elegantly summarized by the fundamental matrix $\mathbf{F}$. For any pair of matching points $\mathbf{x} \longleftrightarrow \mathbf{x}'$ in two images, where $\mathbf{x} = (a, b, 1)$, $\mathbf{x}' = (a', b', 1)$ are homogeneous representations of point image-coordinates, the fundamental matrix $\mathbf{F}$ — a 3×3 matrix of rank 2 — relates them with a linear equation

$$\mathbf{x}'^{\top}\mathbf{F}\mathbf{x} = 0. \tag{1.6}$$

Geometrically, $\mathbf{F}\mathbf{x}$ is the epipolar line $\mathbf{l}'$, and Equation (1.6) states that $\mathbf{x}'$ lies on $\mathbf{l}'$.

1.2.4 Three sub-problems of wide-baseline stereo

The mapping relationship defined by the fundamental matrix (Equation 1.6) is of significant practical importance: we can compute the epipolar geometry using only image correspondences.

If we are given homogeneous coordinates of one pair of matches $\mathbf{x} = (a, b, 1)$ and $\mathbf{x}' = (a', b', 1)$, they can be related by Equation (1.6) as

$$a'af_{11} + a'bf_{12} + a'f_{13} + b'af_{21} + b'bf_{22} + b'f_{23} + af_{31} + bf_{32} + f_{33} = 0. \tag{1.7}$$

Equation (1.7) is linear in the unknowns entries of $\mathbf{F}$. If we denote these entries with a

9-vector, $\mathbf{f} = (f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33})^\top$, a set of n correspondences will give a set of linear equations of the form

$$\mathbf{Af} = \begin{bmatrix} a'_1a_1 & a'_1b_1 & a'_1 & b'_1a_1 & b'_1b_1 & b'_1 & a_1 & b_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a'_na_n & a'_nb_n & a'_n & b'_na_n & b'_nb_n & b'_n & a_n & b_n & 1 \end{bmatrix} \mathbf{f} = \mathbf{0}. \tag{1.8}$$

Equation (1.8) forms the basis of fundamental matrix estimation: between two images, it can be used to compute the $\mathbf{F}$ based on an appropriate collection of correspondences (not lying on one of the *critical configurations* [66]), assuming we have those points and know they are correct matches.

In practice, however, one faces practical issues when it finally comes to numerical computation according to Equation (1.8). Given raw stereo images, how to find those points to start with, how to pair them up as correspondences, how to account for incorrect correspondences, and what to do with the often inexact point coordinates (due to quantization errors or image noise).

Wide-baseline stereo aims at solving these issues and bridges the gap between knowing the theory and using it in engineering applications. I categorize wide-baseline stereo into solving a sequence of the following three sub-problems based on their dependency on one another: *feature detection*, *feature matching*, and *fundamental matrix estimation*.

Feature detection In wide-baseline stereo, feature detection refers to extracting the points, $\mathbf{x}_i$ ($i \in \{1, 2, ...P\}$), from one image and $\mathbf{x}'_j$ ($j \in \{1, 2, ...Q\}$) from the other. We call the points *features* because they are distinctive in the images and often correspond to salient 3-D structures. The hope is that the *pre-image*[1] of one feature in one image will be distinctive from other perspectives as well, so that a feature detector is likely to pick them out from both images. Each feature is associated with point coordinates, and possibly with other attributes such as the size and shape of the region it occupies. Feature detection methods (or feature detectors) are often coupled with how one defines a feature, and a specific detector is designed to extract its own preferred features. We expect a good detector to repeatably extract features despite changes in perspective and illumination, and

[1] Throughout this thesis, *pre-image* refers to the back-projection of an image point into its real-world 3-D position, i.e., the 3-D point corresponding to the image point.

to exhibit consistent performance across different types of scenes.

Feature matching Feature matching is the subsequent problem of pairing up the features across the stereo images. This step establishes a tentative set of one-to-one correspondences between previously detected features, generating a set $(\mathbf{x}_{i_k}, \mathbf{x}'_{j_k})$ $(k \in \{1, 2, ..., R\}$, $R \leq min(P, Q))$. Each match $(\mathbf{x}_{i_k}, \mathbf{x}'_{j_k})$ is considered as projections of the same 3-D surface point onto two images, and the inference is based purely on how similar the features appear in the images. At this stage, no consideration is given to the global epipolar constraint. Given two sets of features, one from each image, and suppose that a significant number of the features have true matches, a good matching algorithm should identify a large portion of the correct matches while rarely make incorrect matches (or outliers). The choice of a matching method depends on the type of features and the similarity measure one will use.

The feature matching problem is ill-posed since we know little about the scene but need to make the correspondences based on its images — there are many possible ways to pair up the features if we only examine appearance of individual feature patches. Refer to Figure 1.5 to see the ambiguity in the correspondence problem.

When imaging a 3-D scene from different viewpoints, image content can vary substantially. For wide-baseline stereo matching, a major source of difficulty is due to the fact that the unknown surfaces can have complex 3-D geometry and that the placement of cameras can be arbitrary. In many cases, some parts can be visible in one image but occluded in another. Even for regions visible to both views, the shapes of the structures (in the images) will undergo unknown transformations due to viewpoint change. Another source of difficulty comes from the scene's photometric properties. Some scenes, especially urban structures, contain repetitive patterns. In this case, if each local pattern is chosen as a feature point (as is often the case), it is not easy to correctly match them without resorting to the global image layout. Under real-life conditions, changes in image illuminations and shadow can also be challenging factors.

In wide-baseline stereo matching, it is a common practice to post-process the tentative matches to reduce the number of erroneous matches, and then use random-sampling based robust algorithms (e.g., RANSAC [57] etc.) to single out all outliers. Since feature matching has been completed up until this point, we treat both the post-processing and the robust outlier detection as components of the fundamental matrix estimation (next section). Sophistication of the post-processing methods varies from performing simple cross-validation

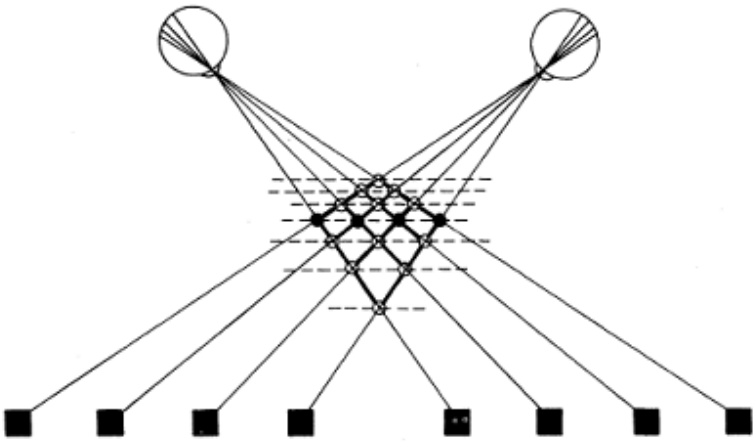


Fig. 1.5 Ambiguity in the correspondence between two retinal projections. In this figure, each of the four points in one eye's view could match any of the four projections in the other eye's view. Of the 16 possible matchings only four are correct (filled circles), while the remaining 12 are 'false targets' (open circles). It is assumed here that the targets (filled squares) correspond to 'matchable' descriptive elements obtained from the left and right images. Without further constraints based on global considerations, such ambiguities cannot be resolved. (Original Fig.1 in Marr and Poggio [110], reproduced with permission from the authors.)

of feature descriptors [114] to solving a full-blown optimization problem with a one-to-one constraint [22].

Fundamental matrix estimation Fundamental matrix estimation is the last step of wide-baseline stereo, where we estimate the fundamental matrix ($\mathbf{F}$) using coordinates of the correspondences established earlier. If the correspondences are correct and their feature locations are precise, one can compute a unique $\mathbf{F}$ by simply solving the set of linear equations (Equation 1.8), as long as one has a sufficient number of correspondences in a non-critical configuration [69].

However, one needs to take into account possible outliers generated by the feature matching algorithm. Besides, one also needs to consider inaccuracies in the point localization, which are often considered as small perturbations from the *exact* positions by independent Gaussian noise. Thus, most methods for estimating $\mathbf{F}$ proceed in two stages:

(1) estimating an initial **F** using a robust method to filter out erroneous matches, and (2) using the matches deemed correct, re-estimating **F** precisely using non-linear optimization [200][111][137]. As mentioned, preceding step (1), there is often a step to reduce the high volumes of outliers by processing the tentative matches [114][22]. As a preprocessing function, this *screening* of tentative matches is an integral component of the fundamental matrix estimation.

Robust methods are designed to deal with estimation problems where some of the data are completely erroneous. In the case of fundamental matrix estimation, random sampling-based approaches (e.g., [57][144]) are typically used to repeatedly draw samples from the data and hypothesize possible fundamental matrices **F**. For each hypothesized **F**, one tests with the rest of the data to see how they fit. The model **F** that fits best with the data is the solution, and the data that do not fit the chosen model are considered to be outliers. The subsequent stage is to find a precise fundamental matrix that best fits the coordinates of remaining matches. Assuming slight inaccuracies in the coordinates as independently and normally distributed, this problem is often cast as the problem of minimizing either an algebraic residual or some geometric distance [107].

A good fundamental matrix estimation procedure should perform well in both of the two stages: in outlier-rejection, it should be able to find all the outliers even if they represent a significant proportion of the input, and it should also compute the **F** that best fits the images.

1.2.5 Closely related areas

A number of other computer vision applications are closely related to wide-baseline stereo matching. *Computational stereo*, in particular, works on stereo images under a more strict camera setup. It directly addresses the depth (or, equivalently, *disparity*) computation problem assuming the stereo system is calibrated [149]. I will describe the characteristics of computational stereo in the first part of this section. Furthermore, wide-baseline stereo belongs to the larger problem of matching two or more 2-D images. Thus, methods for feature detection and feature matching also find use in other image-matching applications, although the different applications may imply different assumptions and require slightly different techniques in their realization. In the second part of this section, I will describe a few of these areas, such as image registration, object tracking, content-based image retrieval,

etc.

Computational stereo Computational stereo refers to computing depth from stereo images whose epipolar geometry has been given. It was inspired by biological vision systems which perceive depth from two slightly different projections of the world onto the retinas of two eyes. Since early research on stereo vision focused exclusively on this setup, when used singularly, the term *stereo* refers to computational stereo.

The geometry of computational stereo makes it simpler to compute depth using trigonometry (cf. Figure 1.6); it is a simplification of general stereo (Figure 1.4). Usually, the input images are assumed to be acquired by horizontally-displaced identical cameras whose image planes are coplanar. Images acquired with more general camera parameters can be post-processed into this configuration by *rectifying* a pair of stereo images once their epipolar geometry is solved [51]. This setup allows one to search for disparities in a one-dimensional space. In Figure 1.6, disparities occur only along the x-axis, thus one needs only to match image points with the same y-coordinates. Typically, the *baseline* (T) is small and the image pair covers similar structures. Consequently, occlusion occurs less often than in a wide-baseline setup, which makes dense matching possible.

The core of computational stereo is to solve the *correspondence* problem. Even though it is reduced to a 1-D search (along the scanline), computational stereo is still ill-posed: without further constraints based on global considerations, there exist ambiguities in deciding which point in the left image should match which in the right image. To make the correspondence tractable, Marr and Poggio [109] propose two rules based on constraints in the physical world: *uniqueness*, which states that "each item from each image may be assigned at most one disparity value"; and *continuity*, which requires that "disparity varies smoothly almost everywhere". These rules comply with regularization theory for solving ill-posed problems [171] and form the foundation for today's computational stereo.

Work in the 80's and early 90's focused on matching a sparse set of locations between the images. The tokens they tried to match were typically lines, curves, points, regions, etc. Due to the limited token-characterizing methods of the time, the correspondence methods use global information and employ sophisticated strategies to resolve matching ambiguity (cf. review papers [8] and [41]). Once one can extract a sparse set of 3-D locations, full 3-D depth can be recovered by interpolation using higher level information. More recently, thanks to developments in global optimization techniques and acceleration in hardware

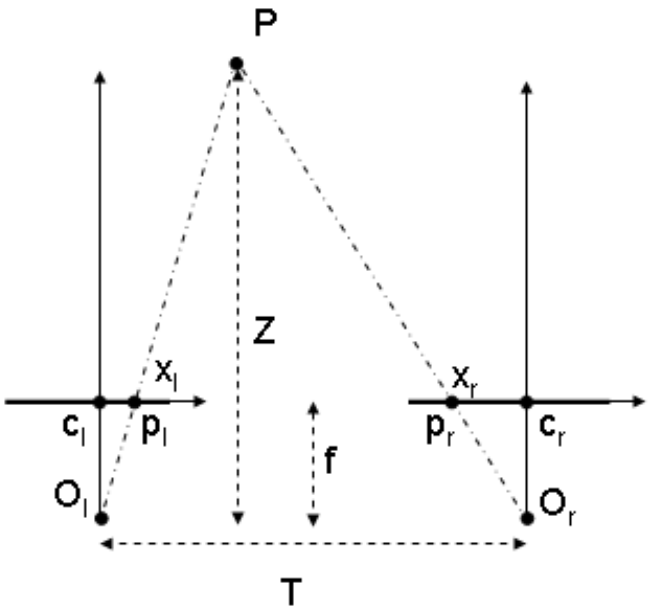


Fig. 1.6 Depth estimation from stereo disparity. This diagram shows a 2-D view of two pinhole cameras, identical and horizontally displaced along x-axis. World point P is imaged p_l and p_r by left and right cameras. Bold lines represent parallel image planes. If the baseline (T), focal length (f), and x-coordinates (x_l, x_r) of points p_l and p_r are known, from trigonometry, the depth of point P to camera center can be computed as $Z = f(T/d)$, where $d = x_r - x_l$ is *disparity*.

computing power, researchers have focused on densely matching all pixels, using area-correlation (cf. review paper [149] and the expanding body of work submitted by authors at the website [150]).

Early stereo matching methods, although they have lost their favor to dense depth computation, represent a rich collection of ideas when it comes to the robust search for correspondence despite ambiguous appearance. I have found some of these ideas very useful for wide-baseline correspondence. I will re-visit this in more detail in Chapter 2 and in Chapter 4.

Other areas In computer vision, a *feature* can be any abstraction of visual information. Thus, feature detection and matching are pervasive in many areas involving visual computing. What features to extract is highly dependent on the task to be performed. In

turn, the features used and the assumptions induced by different applications dictate what matching methods are most appropriate. The awareness of considerations and principles of these areas has greatly influenced and shaped the ideas presented in this thesis. Without trying to be exhaustive, I mention a few of these examples.

Our first example is feature-based image registration [204]. It starts with extracting and matching a set of image locations between the reference image and the target image. The correspondences are then used to estimate the global transform model. In registration, relative depth change of the scene is not as pronounced as in wide-baseline stereo; hence, occlusion typically is not an issue. As a result, one could expect two contiguous blobs in the images to completely overlap each other, though the overlap can be very small and significant variations in the imaging conditions also pose challenge [196].

A second classical problem is object tracking [198]. It also employs the principle of feature detection and temporal feature matching. In this case, adjacent frames differ only slightly, so assumptions about smooth changes are justified. However, there might be dynamic motion involved in the scene, which can be further complicated by partial or complete occlusion. Depending on the task, object tracking uses a variety of features, ranging from primitive geometric shapes (e.g., rectangles, ellipses [30]) to much higher level descriptions (e.g., skeletal models [3] or complex nonrigid shapes [199]).

A third active area of research is content-based image retrieval [37]. Given a query image, the problem becomes finding the best match from a large image database (or video frames). The notion of "best match" is quite evasive here, as it often involves higher level semantic knowledge. Currently, a popular method is to summarize an image through local feature extraction followed by an extra step of feature summarization [53][18][160]. The local features extracted can be either affine invariant features (popular in wide-baseline stereo) or object boundaries (less used in wide-baseline stereo). Here, the step of feature summarization plays an important role. It generates a robust representation of objects by characterizing region shapes and exploiting the spatial relationship of constituent components. These methods robustly characterize objects in a visual image while avoiding confusion arising from non-essential appearance variations. Feature summarization is not considered in wide-baseline stereo, which instead focuses on much more localized regions. However, I find this idea interesting, especially when localized regions are not distinctive by themselves. I will come back to this in more detail in Chapter 4.

1.3 Problem statement and contributions

1.3.1 Objective

This thesis proposes solutions for improving the overall robustness and accuracy of wide-baseline stereo in the setting of 3-D urban environments. Since wide-baseline stereo consists of three modular sub-problems, I approach the three areas separately. This divide-and-conquer strategy makes it easier to obtain an objective assessment of the individual contributions. Furthermore, since each component is also applicable to problems outside wide-baseline stereo, I provide modular tools for a wider range of applications.

Goals for each area Specifically, the goals for the three areas are the following.

1. For feature detection, devise a method that can detect features in urban scenes with better repeatability than existing methods, so as to provide good inputs for the subsequent matching task.

2. For feature matching, present a matching algorithm that can establish more correspondences than current algorithms while keeping false matches rare.

3. For fundamental matrix estimation, given a set of feature correspondences, compute the fundamental matrix more efficiently and more accurately than current methods.

Why urban scenes? I have two reasons for selecting the case of 3-D urban scenes. First, urban environments are where many 3-D vision applications focus. Most human activities happen in urban environments; thus, improving wide-baseline stereo in this environment is crucial for aiding current efforts to build 3-D models of buildings [38][136][42], streets [135], or even cities [1]. With the maturation of theory and algorithms in 3-D reconstruction [134][101][19], research has moved from the lab to solving real-world problems using real imagery, most of which are taken of outdoor urban scenes. Automatic 3-D urban model acquisition will enable deployment of such useful services as virtual tourism [165], navigation, city planning, etc. (e.g., through platforms such as Google Earth or Microsoft Virtual Earth). Second, urban environments still baffle current wide-baseline stereo methods. Currently, most methods perform correspondence by matching some kind of local appearance descriptors (e.g., SIFT descriptors [102]). While sufficient for some

applications, there remain difficulties in many cases, such as matching images of man-made structures with abundant repetitive patterns, matching images with excessive noise, or matching low-resolution images. Refer to the stereo image pair shown in Figure 1.7, the repetitive patterns can easily confuse local feature matching methods, which ignore the larger context of the scene. This is precisely the kind of scene that is commonly encountered in urban environments and calls for a solution.

(a) left (b) right

Fig. 1.7 An example stereo images of an urban scene.

Data sets Images of 3-D urban scenes are the focus for experiments conducted in this thesis. Throughout the validation, I try to use publicly available standard data sets as much as possible. To make up for the lack of data sets containing challenging scenes, I acquired five pairs of stereo images with a digital camera. In the respective chapters, I will point out the sources of the data when they are used in the experiments. At the same time, the novel data used in this thesis is being made available to the public.

1.3.2 Contributions

The major contributions of this thesis are as follows:

1. A new feature detector, the *Structure Guided Salient Region* (*SGSR*) detector, that is effective at detecting regular structures of urban scenes. Meanwhile, compared

with peer methods, it is less affected by unwanted high-frequency patterns regarded as noise.

2. A new representation, the *Salient Feature Graph* (*SFG*), to model the visual appearance of a 3-D scene. At the regional level, the SFG encodes the intrinsic layout of a rigid 3-D scene by preserving its structure under viewpoint changes.

3. A new algorithm, *Context-Consistent Assignment* (*CCA*), to robustly match feature points between widely separated views using both local and contextual feature information.

4. A new procedure to increase the efficiency and accuracy of fundamental matrix estimation by refining feature correspondences. This refinement improves quality of the correspondences by fine-tuning feature localization.

Furthermore, this thesis also contributes the following novel ideas.

1. An extension to saliency detection [79] from greyscale to color images. Proposed a robust implementation for representing probability density function of a region's pixel values with a coarser histogram.

2. A new method to compute similarity between one view of a feature cluster and its image in another view . This method uses a *Neighborhood Transform* to account for discrepancies due to the viewpoint shift. With the *hypothesize-match* procedure, it solves a many-to-many match problem without one-to-one correspondence.

3. An experimental framework that automates the processing and validation of feature extraction, feature matching, and fundamental matrix estimation. Within this framework, a new criterion — *correct ratio vs. (1-precision)* — is proposed for evaluating feature matching performance on real complex 3-D scenes.

1.3.3 Published work

The following publications are related to the work in this thesis.

- Fan, S. and Ferrie, F., (2009), Context-Consistent Stereo Matching, International Conference on Computer Vision Workshop on 3-D Digital Imaging and Modeling, Pages 1694-1701.

- Fan, S. and Brooks, R. and Ferrie, F., (2009), Better Correspondence by Registration, Asian Conference on Computer Vision, Pages 436-447.

- Fan, S. and Ferrie, F., (2008), Structure Guided Salient Region Detector, British Machine Vision Conference, Pages 423-432.

Work in the following publication does not directly appear in this thesis, but is closely related to this research.

- Fan, S. and Ferrie, F., (2010), Photo Hull regularized stereo, Image and Vision Computing, Volume 28, Pages 724-730.

1.4 Structure of the thesis

This thesis is organized as follows. In Chapter 2, I review some background material related to this work. In addition to a high-level review of the problems of *feature detection*, *feature matching*, and *fundamental matrix estimation*, I also review some work in computer vision and pattern recognition related to graph matching techniques, which I use as an effective tools for matching features.

In Chapter 3, I investigate the issue of feature extraction for wide-baseline stereo. I apply the principle that a good feature should be a visually salient point — an image point that catches the eye at the first sight. One notion of "saliency" was introduced by Kadir et al. [79], and shown to be an effective tool for tracking and object recognition. I adapt this notion by also giving proper consideration to image contrast, making it better suited for wide-baseline stereo tasks. I show that, by fusing information theory-based saliency into the feature selection criterion, the resulting features are more likely to correspond to regular structures of our 3-D environments and they are reliably extracted.

In Chapter 4, I investigate the issue of wide-baseline feature correspondence. I apply the principle that local feature correspondence should exploit information contained in feature's neighborhood structure. To this end, I model an image as a graph to establish connections between different parts of the image. Based on this graph, I devise a graph-matching algorithm to find feature correspondences. As a result, I am able to match features by exploiting consistencies of their neighborhood structures — this can be very effective in matching some challenging urban images.

In Chapter 5, I address the question of how to estimate two-view geometry once we have the initial matches. The observation here is that the much needed accuracy of feature localization can be improved once we have feature correspondences. I demonstrate that, by preprocessing the matches, we can compute the fundamental matrix more effectively. This registration-based preprocessing can be carried out before commencing actual fundamental matrix estimation. I show that it can help both in singling out outliers and in precisely computing the fundamental matrix.

I draw overall conclusions and suggest some future research directions in Chapter 6.

Chapter 2

Background

This chapter reviews some background related to this work. Wide-baseline stereo became an active area of research just a little over ten years ago, when the confluence of image processing, pattern recognition and theories of multiple-view geometry made it possible to apply to real images. Nevertheless, its constituent elements — feature detection, feature matching, and fundamental matrix estimation — have long been researched in different contexts. I will review these three elements in Sections 2.1, 2.2, and 2.3. Since I use graph matching techniques in my work, I will also briefly discuss some related work that uses graph theoretic methods (Section 2.4). Each of the above areas is an active research topic of its own. Thus, I do not claim an exhaustive coverage; rather, I try to identify representatives of the key ideas and focus our discussion on those relevant to wide-baseline stereo.

2.1 Feature detection

As mentioned in Section 1.2.5, the concept of a *feature* is very general in computer vision. For the task of matching wide-baseline images, the focus is on the so-called *interest points*, meaning distinct and localizable points in the image, typically having pixel intensity changes *along more than one direction* [152]. Methods for detecting these interest points are called *interest point detectors*, or *feature detectors*.

There are some key properties that a good feature detector should possess. First and foremost, the feature detector should be as independent as possible of changes in the imaging conditions, i.e., the parameters of the camera, camera position relative to the

scene, and the illumination conditions. Second, it should facilitate matching features across images, supposing the corresponding features *are* detected. A popular belief is that the richer the information one can read from the appearance of features, the easier one can match them [153]. This is true for most current methods, which do local appearance-based feature matching depending exclusively on appearance descriptors (e.g., SIFT [102]). In general, having ease of matching is certainly coupled with what matching method is used. Nevertheless, it is always desirable to have a lot of distinctive information contained in a feature. Third, the detector should localize feature positions precisely. The better the localization, the more useful the features are to subsequent tasks, such as camera calibration or 3-D reconstruction.

In early work, *interest point* was used interchangeably with *corners*, which I will discuss in Section 2.1.1. Scale-space theory enables one to analyze features at various scales; in Section 2.1.2, I will discuss a family of approaches that deals with scale and affine-invariance explicitly. In Section 2.1.3, I will discuss some recent approaches that also detect features at various scales, not through scale-space analysis, but by analyzing edges or intensity profiles of image regions. A perspective drastically different from all the above is taken by the work of Kadir et al. [79] [80], who propose to use an information theory-based measure for feature detection. This method detects *Salient Regions* that exhibit entropy extrema in their scale space and at the same time have a large change in probability density function when their scales change. In Chapter 3, I will describe this notion of saliency, and present my extension to it in the context of wide-baseline stereo.

2.1.1 Early features

One major family of feature detectors works by densely analyzing differentials of the greyscale images. Moravec [122] pioneered feature detection by proposing an interest operator to work at multiple resolutions of an image. The Moravec corner detector locates image points that have local maxima of directional pixel intensity changes.

Harris and Stephens [65] later proposed an improvement called the *Harris Corner Detector*, to overcome the Moravec detector's anisotropic response, sensitivity to noise and to edges. The Harris corner detector provides stable corner points by not only distinguishing edges from corners but also computing a measure of corner quality. For an image I, it analyzes the eigenvalues (λ_1, λ_2) of the *structure tensor* (also called Harris matrix, auto-

correlation matrix) at every image location $\mathbf{x}$. The structure tensor ($\mathbf{T}$) at image location $\mathbf{x}$ is a 2×2 matrix comprised of second moment image differentials, averaged within a certain neighborhood around $\mathbf{x}$,

$$\begin{aligned}\mathbf{T}(\mathbf{x}) &= w(\mathbf{x}) * \begin{bmatrix} I_x^2(\mathbf{x}) & I_x I_y(\mathbf{x}) \\ I_x I_y(\mathbf{x}) & I_y^2(\mathbf{x}) \end{bmatrix} \\ &= \begin{bmatrix} \langle I_x^2(\mathbf{x}) \rangle & \langle I_x I_y(\mathbf{x}) \rangle \\ \langle I_x I_y(\mathbf{x}) \rangle & \langle I_y^2(\mathbf{x}) \rangle \end{bmatrix},\end{aligned} \tag{2.1}$$

where I_x and I_y are spatial image gradients, "$*$" is the 2-D convolution operator, $w(\mathbf{x})$ defines a neighborhood window to average over, and $\langle \cdot \rangle$ means the value averaged over $w(\mathbf{x})$.

Harris and Stephens showed that if both eigenvalues of $\mathbf{T}(\mathbf{x})$ are small, then $\mathbf{x}$ is not a feature of interest; if one of them is small and the other is large, then $\mathbf{x}$ is on an edge; if both are large, then a corner is found at $\mathbf{x}$. To avoid explicit eigenvalue decomposition of $\mathbf{T}$, they proposed the *Harris function* (R) to efficiently compute corner quality,

$$\begin{aligned}R &= \lambda_1 \lambda_2 - \kappa(\lambda_1 + \lambda_2)^2 \\ &= \mathbf{Det}(\mathbf{T}) - \kappa \cdot \mathrm{trace}^2(\mathbf{T}),\end{aligned} \tag{2.2}$$

where $\mathbf{Det}(\mathbf{T})$ is the determinant of $\mathbf{T}$, $\mathrm{trace}(\mathbf{T})$ is the trace of $\mathbf{T}$, and κ is a predefined scalar parameter (typically 0.04). A large R signals a prominent corner.

In the same vein, other metrics derived from the structure tensor were used to find corners (e.g., Forstner [58], Shi and Tomasi [158]). Similarly, the Hessian matrix ($\mathbf{H}$) and its entries were also used to locate interest points (e.g., Beaudet [12], Kitchen and Rosenfeld [86], Dreschler and Nagel [44]).

$$\mathbf{H}(\mathbf{x}) = \begin{bmatrix} I_{xx}(\mathbf{x}) & I_{xy}(\mathbf{x}) \\ I_{xy}(\mathbf{x}) & I_{yy}(\mathbf{x}) \end{bmatrix}, \tag{2.3}$$

where I_{xx}, I_{xy} and I_{yy} are second order image derivatives.

There are many other families of approaches apart from those based on intensity differentials. Some corner detectors first extract curves or lines and then search for maximal curvature (e.g., Asada and Brady [5]), inflexion points (e.g., Mokhtarian and Mackworth

[120]), some intersection point (e.g., Horaud [70]), or a combination of these (e.g., Mokhtarian and Suomela [121]). Smith and Brady [164] compared the brightness of each pixel in a circular mask to the center pixel to define an area that has a similar brightness to the center — corners can be detected from the size, centroid and second moment of this area. Reisfeld et al. [139] used the concept of symmetry to detect corners. Rohr [142] recognized corners by fitting parametric models.

2.1.2 Considerations with respect to scale and affine-invariance

Most of the above methods are rotation invariant but do not explicitly handle other types of geometric invariance, such as scale-invariance (detection under magnification or reduction), or affine-invariance (detection under linear distortion by an affine transform). Here I review some methods that address these issues by working on multiple scales of an image.

Dufournaud et al. [45] extended the Harris operator by detecting corners at multiple resolutions of an image I. They then selected corners by unifying them with a scale-normalized measure of "cornerness". Baumberg [10] used a similar idea and further proposed an affine-invariant feature characterization for robust feature matching.

A more principled approach is one backed by research in scale-space analysis [191][87][96]. Theoretical studies [96] [97] have found that filtering an image with the Laplacian-of-Gaussian (LoG) filter of a certain scale can give rise to an optimal response for features at that characteristic scale.

$$LoG(\mathbf{x}, \sigma) = \sigma^2 |L_{xx}(\mathbf{x}, \sigma) + L_{yy}(\mathbf{x}, \sigma)|, \tag{2.4}$$

where $L_{xx}(\mathbf{x}, \sigma)$ and $L_{yy}(\mathbf{x}, \sigma)$ are the second-order derivatives of a Gaussian-smoothed version of I (i.e., $L(\mathbf{x}, \sigma) = G(\mathbf{x}, \sigma) * I(\mathbf{x})$). This has provided the foundation for current methods to detect features at their characteristic scales. This family of methods searches for maxima in 3-D (x, y, and scale σ) representations of an image I, where the σ-dimension is obtained by stacking up images of a certain function of I at increasingly coarse resolutions (cf. Figure 2.1). At coarse resolutions, characteristics of large scales are being analyzed. While they all detect maxima of LoG over scale, different methods select maxima of different functions within the image plane (x- and y-axis), resulting in different structures being singled out.

One function of choice is the Difference-of-Gaussians (DoG) — the scale-space is ob-

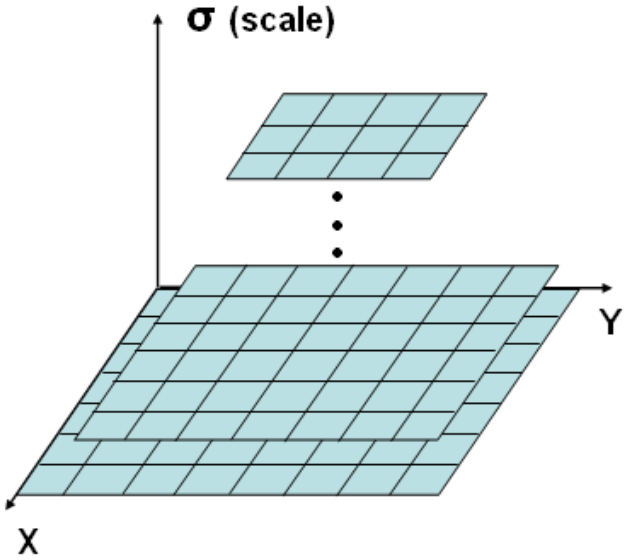


Fig. 2.1 Pyramidal scale-space representation of an image. The base of the pyramid represents the finest resolution of an image. By applying a sequence of combined smoothing and sub-sampling, one generates a pyramid of scale-space images.

tained by stacking images filtered by *DoG* filters at varying scales.

$$DoG(\mathbf{x}, \sigma) = |I(\mathbf{x}) * G(k\sigma) - I(\mathbf{x}) * G(\sigma)|, \tag{2.5}$$

where, $G(\sigma)$ is a Gaussian kernel of standard deviation σ. This method detects maxima of *DoG* over both scale and image-plane location. The work by Crowley and Parker [35] is an early example using the *DoG*, where they built a pyramid representation and detected features by looking for local maxima in their surrounding 3-D cubes. Lowe's Scale-Invariant Feature Transform (SIFT) [102] is an extension of [35]. Lowe built the pyramidal scale-space representation by an efficient *DoG* filtering and showed that selecting *DoG* maxima along the scale-axis is equivalent to the *LoG*-based scale-selection [96]. Moreover, the SIFT was carefully engineered to accurately pinpoint feature locations, to efficiently eliminate edge responses, and to robustly assign feature orientations. The SIFT has proved to be a success in detecting features at varying scales. At the same time, Lowe contributed a highly distinct characterization of feature appearance called the SIFT descriptor. Although SIFT detection was not designed to handle affine changes, the design of the SIFT descriptor with

respect to tolerance of shift makes SIFT features robust against a range of affine distortions.

Inspired by Lindeberg's work [97] on blob detection and automatic scale selection, Mikolajczyk et al. [114] proposed two detectors that are invariant to affine transforms. While using the original formula of *LoG* for optimal scale selection, they used a different function to find structure in the image plane. For locating maxima, they used the Harris function (cf. Equation 2.2) for one detector ("Harris-Affine") and the determinant of the Hessian matrix (cf. Equation 2.3) for another detector ("Hessian-Affine"). Furthermore, to deal with affine distortion, they use the structure tensor (cf. Equation 2.1) to iteratively normalize the point neighborhood. Both detectors achieve the state-of-the-art performance.

2.1.3 Region boundary based methods

Also aimed at wide-baseline tasks are other recent methods that directly analyze image contours or region boundaries. Tuytelaars and Van Gool [181] proposed two detectors: the *intensity extrema-based region* detector analyzes region intensity profiles to find a best-fitting elliptical outline, and the *edge-based region* detector combines Harris corners with edges originating from them to define a stable parallelogram. Matas et al. [111] presented a watershed-like algorithm to extract an intensity induced *maximally stable extremal region (MSER)*. Besides these examples, many researchers used corners, edges, lines or some combination thereof to extract features that are likely to be repeatably detected under wide-baseline conditions [194][195][11]. Jugessur and Dudek proposed PCA-based features [78] that are suitable for appearance-based object recognition.

2.1.4 Summarizing remarks

Corners ([65][158] etc., Section 2.1.1) are usually extracted efficiently and are widely used for video tracking, since frame-rate processing speed is desired and neighboring frames have similar imaging conditions (so that corners do not change much between frames). Methods based on scale-space theory (Section 2.1.2) can deal with wider changes in imaging conditions, thus, they are suitable for wide-baseline conditions. For less severe rotation changes (e.g., $\leqslant 40°$), scale invariance by the SIFT is sufficient for many applications [115]. Methods based on analyzing region boundaries (Section 2.1.3) are also excellent tools for wide-baseline matching tasks; especially popular is the MSER [111] which performs well on various benchmarks [116] and can be extracted efficiently.

The information theory-based saliency measure [79] provides a generic measure for feature detection. But its potential in repeatably detecting features for wide-baseline matching has yet to be seen, partially due to the difficulty in handing viewpoint changes. I will explore the use of this generic formulation by combining it with methods that analyze region boundary intensities.

2.2 Feature matching

As there are many ways of detecting features, there are a myriad ways of matching them. I limit the scope of my discussion to the work that establishes correspondences beginning with no assignments between the two feature sets. I see any attempt to improve tentative matches (either using the global epipolar constraint [173] or through local cross-validation [40][162][22]) as a post-processing step, which can always be used to improve results of any feature matching algorithm. I include a review of relevant early narrow-baseline methods that are also of interest to our wide-baseline matching.

All existing feature matching methods can be categorized into one of three strategies. The first is to use high-level features (e.g., planes), which are often less ambiguous, to reduce ambiguity in low-level feature matching. A second strategy is global-optimization based feature matching, in which the solution corresponds to an optimal overall configuration of the assignments between features on the entire images. Similar to the first, global methods also draw on larger contexts around features, but they seek the optimal solution by minimizing a global energy function instead of using semantical scene objects. The third strategy is to characterize the local appearance of the features as well as possible, and then to match them based only on this characterization. In the following sections, I introduce these three categories and give a non-exhaustive list of examples.

2.2.1 Higher-level feature based matching

Early on, the features to be matched were primarily corners, edges and line segments whose appearance was not distinctively characterized. Thus, researchers often grouped these features together into some higher-level interpretations of the scene, resulting in fewer entities that are easier to distinguish.

Quan and Mohr [138] reduce the search space by perceptually grouping line segments into directional groups, collinear groups, and rays. Pritchett and Zisserman [137] form

groups of line segments and estimate local homographies using parallelograms as measurement regions. These homographies, in turn, guide robust in-plane feature matching. This is one of the earliest works on computing epipolar geometry that explicitly deals with region distortion induced by wide-baseline viewpoint changes. Brown and Lowe [17] group nearby SIFT points to form locally planar image regions which, in turn, are characterized by region descriptors that are invariant to large changes in viewpoint, scale and illumination. Tell and Carlsson [170] create line segments by connecting nearby Harris corners, and then they cyclically match intensity profiles of those line-segments emanating from the corners. These intensity profiles can add more information to the corners and make them more distinct. Johns and Dudek recognize buildings by matching outlines of rooftops of adjacent buildings [76].

Matching of multiple unordered views was attempted by Ferrari et. al. [55], and by Schaffalitzky and Zisserman [148]. The novelty lies in their tracking feature matches across multiple views to "clean up" their initial multiple view matches by removing erroneous matches and adding in new correct matches. In essence, extra views are used as additional information to guide robust feature matching.

2.2.2 Global-optimization based matching

Methods using higher-level features often make assumptions about the scene, say, an abundance of line segments or quadrilaterals, presence of dominant planes, etc. When no such assumptions can be made but at the same time information of local features is too low for unambiguous matching, global optimization is often the method of choice. Global methods find a set of correspondences that give the best overall matching cost while making sure the result is a reasonable interpretation of the scene. Most commonly, it is assumed that one point matches to at most one point (uniqueness) and that features lie on smoothly varying 3-D surfaces (continuity). The latter assumption often implies that neighboring features are physically close to each other and that they share some consistency. Four major global approaches exist: relaxation labeling, hierarchical matching, dynamic programming, and relational-structure matching.

Relaxation labeling [143] is a widely used model to iteratively impose global consistency constraints on multiple matches for the purpose of disambiguation. An early example is the work by Barnard and Thompson [7] to match two sets of sparse features in a narrow-baseline

setting. It uses a relatively small amount of local information for each potential match, and attempts to resolve ambiguities by finding consensuses among subsets of the entire population of matches. The continuity constraint is used in their iterative update. Kim and Aggarwal [85] improved on this approach by including more disambiguating constraints and using more flexible convergence control. Wang [184] extended the approach to match two sets of points that differ by both a translation and a rotation. In the more difficult wide-baseline setting, Zhang et al. [201] used relaxation labeling and pair-wise relationships to robustly match Harris points. They were able to robustly estimate epipolar geometry.

"Coarse-to-fine" hierarchical matching strategies are another way to reduce ambiguity. Marr and Poggio [110], for example, proposed such a scheme inspired by the human visual system. They detect features at both coarse and fine resolutions, and let coarse-level disparities guide the matching at finer levels. Lim and Binford [95] proposed a hierarchical stereo algorithm which starts by matching at the highest level — objects. Results at the object-level are propagated down to lower levels — surface boundaries, junctions, curves and edges. The hierarchical information increases both the speed and accuracy of feature matching, but the extraction of higher level features needs to be robust and consistent.

Dynamic programming is another efficient strategy for globally matching image points [34][63]. Typically, methods in this family assume stereo images are calibrated and exploit ordering constraint on pixels along corresponding image scan-lines. Cox et al. [34] were the first to use dynamic programming in stereo correspondence. They proposed two methods to preserve consistency between scan-lines. One is by imposing, in the cost optimization, several "cohesivity constraints" that minimize the total number of horizontal and/or vertical discontinuities. And the other is by using more than two cameras. Gong and Yang [63] proposed so-called Reliability-based Dynamic Programming, in which they recover depth with more accuracy by multiple dynamic programming passes. More recently, capitalizing on a new formulation that represents an image by a 2-D pixel-tree structure [183], Lei et al. [90] proposed a stereo algorithm that combines dynamic programming with region-based approaches.

Another effective way to reduce ambiguity is through matching relational structures, called *structural stereopsis*. These approaches embed the global information of stereo images into structural graphs and find a globally optimal matching between them. Extending an early formalism of structural image description [156], Boyer and Kak [16] developed theories for probabilistically matching real (noisy) stereo images. They used a skeletal,

or "stick-figure", representation of objects to encode image structures. They defined an inter-primitive distance measure and a relational inconsistency measure to account for similarities between both single primitives and primitive-pairs. They solved the consistent labeling problem using a tree search method. Ayache and Faverjon [6] described stereo images with neighborhood graphs of line segments and matched them through an efficient *prediction and propagation* technique. Li [94] proposed an optimization approach to inexact structure matching that is invariant to arbitrary translations, rotations, and scale changes. Relaxation labeling is used to solve the optimization problem.

2.2.3 Local feature matching

The local methods generally use interest points as features and match them by directly comparing the photometric properties of the regions they occupy. This relies on methods for effectively characterizing photometric appearances of local regions. The seminal work by Schmid and Mohr [152] proposed the use of *differential grayvalue invariants* to characterize the intrinsic appearance of an object. Their method showed great potential in matching objects under scale changes, viewpoint changes and partial occlusion. This sparked a wave of new methods that characterize feature appearance with invariant descriptors. The SIFT [102] descriptor by Lowe is a highly regarded example and has been widely used. The interested reader can learn more about recent descriptors by referring to a survey by Mikolajczyk et al. [115]. Local features other than interest points are also used by some authors, e.g., "line signature" by Wang et al. [185].

On the matching side, for each query feature descriptor in one image, one finds the feature with the closest descriptor from candidates in the other image. The closeness is measured by the Euclidean distance between the descriptor vectors. In recognition, this is often called *nearest-neighbor classification*, thus it is called *nearest-neighbor matching* in the wide-baseline context. To discard features that do not have any true correspondence, Lowe [102] proposed to only keep those matches that have the closest neighbor significantly closer than the closest incorrect match. This method is called *nearest-neighbor distance-ratio matching*: one compares the distance of the nearest neighbor to that of the second-nearest neighbor, and declares that the nearest-neighbor matches the query only if the ratio of the two distances is below a certain threshold (e.g., 0.8).

For matching between two images, it is affordable to exhaustively search for the exact

nearest neighbors. For recognition tasks, for which the descriptors were originally designed, one needs to consider matching on a much larger scale, i.e., against a very large number of candidates. This relies on efficient nearest neighbor indexing schemes [4] and is another major area of research. I will not go into detail on this topic other than to point out that approximate algorithms, such as the Best-Bin-First search by Beis and Lowe [13], are developed to efficiently find the closest match with high probability.

2.2.4 Summarizing remarks

For feature matching, there is no clear-cut categorization of the approaches used. For example, methods using higher-level features often use coarse-to-fine hierarchies. In fact, it is common to see different approaches fused together to obtain better results.

Recently, with many invariant descriptors being proposed (e.g., SIFT, Shape Context [14], GLOH [115], etc.), local methods enjoy superior performance and have dominated many applications [165][1]. Among the various types of local features, interest points are primarily used for matching because they have high information content and are very well localized, giving them advantages over other features such as edges or curves. With the prevalent use of nearest-neighbor matching techniques, local feature matching methods essentially amount to developing new descriptors that can characterize interest points invariant to various imaging condition changes.

Methods using higher level features, such as those using planes [137] or groups of line-segments [138], often need to make strong assumptions about the scene. Thus, they are not widely used in more recent applications. Global optimization based methods are the methods of choice for narrow-baseline stereo [149], but have rarely been successfully used in wide-baseline matching — largely due to a lack of methods for enforcing "smoothness" in the presence of ubiquitous partial occlusion, shape distortion, illumination change, etc. I address this difficulty and show how global optimization techniques can be adapted to match features more robustly. The key idea is a novel embedding of local neighborhood information into the optimization procedure despite highly uncertain variation in neighborhood structure.

2.3 Fundamental matrix estimation

Fundamental matrix estimation is equivalent to the recovery of two-view structure and motion. We will look at the fundamental matrix estimation problem within the larger domain of structure from motion, so that we have a better overview of where it comes from and where it is heading to. Here, the focus is on recovering the camera motion and scene structure, assuming known feature correspondences (which can be obtained by methods in Sections 2.1 and 2.2, but do not have to be "perfect" in matching correctness or localization accuracy).

Early work focused on *what* could be done to recover motion and structure using image sequences. Longuet-Higgins [100] introduced the *essential matrix* to the computer vision community. He showed that the relative placement of two calibrated cameras and the scene structure can be directly computed by solving a set of simple linear equations. Various other works also recovered the motion and structure using features other than points (lines [118][197][98][167], or conic arcs [177][176][108]), or by different problem formulations ([117][141][75][178][71]). The interested reader should refer to the review papers by Aggarwal and Nandhakumar [2], and Huang and Netravali [72]. Two limitations hinder the application of these methods: one is that the assumption of internally-calibrated cameras is hard to meet at times; another is that the noise in the feature correspondences is not addressed.

Beginning in the early 90's, researchers made huge progress in understanding *how* to recover motion and structure effectively. The first major advancement comes from the theory of the fundamental matrix (Luong [105], Faugeras [50][52] and Hartley et al. [67][68]). It made the "calibrated camera" assumption unnecessary and had a major impact on methods to automatically extract 3D geometry: point-matches (or point-matches induced from other forms of features) became predominantly used, and problem formulations focused on computing the optimal fundamental matrix by enforcing its *rank-2* property. The second advancement is in new optimization methods to overcome the noise in point coordinates [186][106][66][23][24][9][169]. These also bring us one step closer to solving practical problems, because they make it possible to find the model that best fits the noise-corrupted data, even though no model can precisely relate all the data. The third advancement is the adoption of robust algorithms ([73][57][144]) to discard grossly incorrect correspondences (also called *outliers*). Thus, outliers produced by feature matching algorithms are

taken care of, making the structure and motion robust enough for practical applications (cf. reviews of some of the methods by Zhang [200] and Torr [173]).

Recently, many researchers have worked towards finding solutions under even more challenging conditions, largely motivated by such images as exemplified by our urban scenes. Some methods make strong assumptions about the scene so as to constrain feature matching to a sub-region of the image and greatly reduce the search space [89][29]. Another family of methods [22][40] focuses on post-processing the initial putative matches, so as to alleviate the burden of subsequent random-sampling based outlier detection. They typically work on a large number of putative matches and reduce them to a smaller set of more reliable matches by using contextual image information. Yet another major family of methods innovates on the robust estimation stage: they typically devise more effective sampling schemes that favor inlier matches by exploiting image information around the point matches. For example, PROSAC [27] and Guided-MLESAC [172] use feature similarity to drastically increase efficiency and robustness, while GroupSAC [125] and SCRAMSAC [147] use the local neighborhood consistency of features. Some methods are even successful at finding the fundamental matrix with 90% outlier contamination. All of the above successes benefit from one common insight: fundamental matrix estimation should use the correspondences *and* their image information, instead of (traditionally) ignoring the latter.

One area that has been overlooked is the localization precision of feature points. Ultimately, fundamental matrix estimation is based on the image coordinates of the localized points, but little attention has been paid to the impact of the localization accuracy on the estimation result. I will examine the possibility of improving fundamental matrix estimation by refining feature localization.

2.4 Graph matching in computer vision

Graphs are widely used to represent structural information in many domains such as networks, information retrieval, knowledge discovery and data mining, and, more relevantly, in computer vision. When used for image representation, typically, the nodes of the graph refer to some regions or features of interest and the edges refer to the structural relationships between objects. We often deal with what are called Attributed Relational Graphs (ARG), since the nodes and edges usually possess some meaningful attributes (e.g., the area of a region, the distance between features, etc.). Graphs are excellent tools for representing

images because they provide a high-level abstraction of the 2-D grid of pixel-values (in the case of 2-D images). Graph representations and their matching algorithms are extensively used in the computer vision community. Instead of attempting a detailed coverage (an excellent review is available in Conte et al. [33]), I will briefly exemplify their application diversity and categorize the computational techniques that they use.

Graph matching applications Due to the expressive power of graphs, many computer vision problems involving image matching can be more easily solved by graph matching. In 2-D image analysis, graphs have been used in object recognition [43], shape recognition [160][159], scene recognition [83][84]), non-rigid registration [26][202], stereo matching [6][94], etc. In video analysis, graphs are used in tracking in the presence of occlusion [62][32], activity recognition [82], etc. Indexing and fast matching of graphs are also used in content-based multimedia retrieval [21].

Graph matching methods From the computational point of view, graph matching is categorized into two broad families. The first contains *exact matching* methods that require a strict correspondence between the two objects being matched or at least between subparts of them. Most of the exact graph matching algorithms are based on some form of tree search with backtracking [182][154]. They incrementally include nodes into the solution, and abandon ("backtracks") a partial candidate, c, as soon as they determine that c cannot possibly be completed to a valid solution. The second family defines *inexact matching* methods, where a matching can occur even if the two graphs being compared are structurally different to some extent. The latter type of matching is more commonly seen in vision applications due to the variability or noise in the construction of ARGs. Since our application also falls into the category of in-exact graph matching, I will discuss these methods in more detail.

Finding the solution to inexact graph matching is usually cast as an optimization problem, the goal being to find the minimum cost of differences between the matched nodes and edges. Two different approaches exist: one is to directly optimize the cost function in terms of graphs, which is inherently a discrete optimization problem. Another is to convert the problem to a continuous problem and solve it using continuous, nonlinear optimization methods.

For most discrete inexact graph matching methods, an explicit model of the possible

errors (i.e., missing nodes, etc.) is defined and each kind of error is assigned a different cost; equivalently, a set of *graph edit operations* (e.g., node insertion, deletion, etc.) is introduced, each assigned a cost. The matching that results in the cheapest error-correcting cost or graph edit cost is the optimal solution. These algorithms are often denoted as *error correcting* [179][193][156][157] or *graph editing* algorithms [47][46][145][99]. Matching is performed with a search based procedure to minimize overall cost of the resulting matches, with the search methods varying from branch and bound [99], to genetic search [123] and Tabu search [187], etc. For practical problems involving more than hundreds of nodes, these methods are hindered by their high combinatorial complexity.

The continuous methods, on the other hand, are very appealing due to their much-reduced computational cost and robustness to noise. Furthermore, they can also be used in exact graph matching settings. Various methodologies are adopted in this domain. One major family of methods pursues some principled statistical measures of graph similarity, with representative early work by Wong and You [192] and Boyer and Kak [16]. Following the work by Christmas et al. [25], Hancock and colleagues push this field towards including more than pairwise relations and using more structural constraints [189][56][190][188][124]. In a second family of methods, Pelillo et al. [132] convert maximal subtree isomorphisms into the maximal cliques problem and solve it using replicator equations. This method is successfully used in the matching of shock graphs [133]. A third popular approach is the *softassign* proposed by Gold and Rangarajan [61], they formulate graph matching as two-way (assignment) constraints and solve it via a deterministic annealing procedure. A notable application is that by Chui where she expands the approach to matching deformable shapes [26]. Fourth, spectral methods [28][104][91] also hold much promise because of their computational efficiency. Being purely structural, current spectral methods can still have the potential to be improved by further incorporating node/edge attributes.

Among these, the softassign formulation enforces a one-to-one constraint in assigning node matches. This is of particular interest to us due to its efficiency in handling large sparse graphs and the ease of incorporating node/edge attributes.

Chapter 3

Structure guided salient region detector

This chapter considers the problem of feature extraction, the first sub-problem of the three-step wide-baseline stereo. Recent methods have been successful at repeatably detecting image features under changes in scale, viewpoint, and, to some extent, illumination. They mainly select patches whose borders have high intensity contrast with surrounding regions [97][10][180][181] [111][114]. Their effectiveness was demonstrated in a recent benchmark paper [116], where MSER [111] and Hessian-Affine [114] detectors are shown to outperform other detectors in repeatably detecting features under various circumstances.

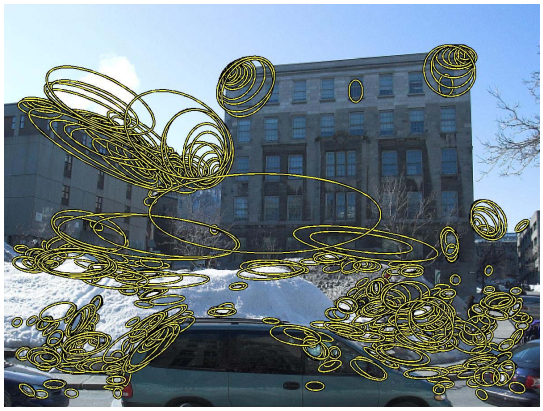
When detecting features in images of real-life 3-D scenes, however, some extra factors come into play. Consider the scene (labeled *J-Scene*) in Figure 3.1 as an example. The *J-Scene* represents a typical outdoor scene with both regular structures (building facades) and many factors that distract the correspondence effort: low-contrast of the facades, occlusions by tree branches and snow banks, small image overlap, etc. Refer to Figure 3.2 for the detected Hessian-Affine and MSER regions. Relying purely on contrast would inevitably miss some low-contrast structures that are obviously eye-catching to humans. At the same time, they are severely hindered by irregular patterns introduced by branches and snow banks.

If we look at the methods more carefully, we may understand why this happens. The Hessian-Affine detector [114] relies on affine normalization over a large neighborhood region. If the local structure is isolated and indeed devoid of abrupt depth change on all sides, the

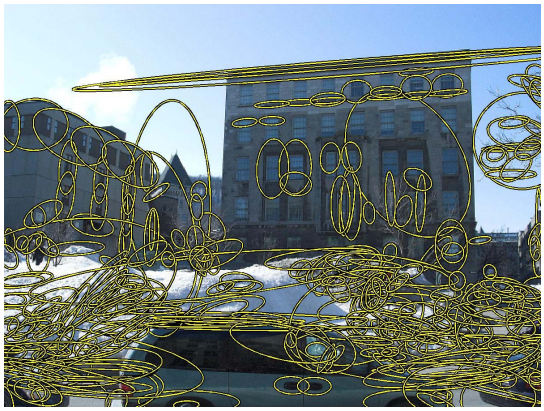
(a) left view

(b) right view

Fig. 3.1 Stereo images of the *J-Scene*



(a) Hessian-Affine regions, left view



(b) MSERs, left view

Fig. 3.2 Features detected on the *J-Scene*

normalization can detect the same scene structure adapted to different viewpoints with elegant affine warps. Patterns on the facades are not neatly isolated, thus, normalization iterations on these regions tend not to converge. That is why no features are detected on the facades. Also, the *J-Scene* is full of various occluders and depth changes. The MSERs are also confused by many irregular (but high-contrast) occluders and fail to pick the more regular (but low-contrast) patterns on the buildings.

We aim at overcoming this limitation by using a visual saliency measure. The rationale

is that the detector should pick visually salient regions, no matter what their greyscale contrast is. For this, we turn to a saliency measure rooted in information theory proposed by Kadir and Brady [79]. They showed that their saliency measure can help detect features (called *Salient Regions*) at their intrinsic scales and that the detected *Salient Regions* contain rich information for recognition tasks. Under our wide-baseline context, we propose to use this tool to single out the salient features that are better for matching stereo images. As a result, we propose what we call the *Structure Guided Salient Region* (SGSR) detector that better suits 3-D urban scenes. We will show its advantages in two respects: (1) repeatability under viewpoint changes using a widely used benchmark (Mikolajczyk et al. [116]), and (2) a real wide-baseline stereo application to 3-D scenes.

The outline of this chapter is as follows. After reviewing the original formulation of entropy-based saliency by Kadir and Brady (Section 3.1), we will describe our Structure Guided Salient Region in detail in Section 3.2. Then, Sections 3.3 and 3.4 evaluate its performance in the cases of planar scenes and 3-D scenes respectively. Finally, we summarize the conclusions to be drawn in Section 3.5.

3.1 Background on the entropy based saliency

According to Kadir and Brady [79], *Salient Regions* are regions that simultaneously hold two properties in scale space: they assume maximal signal complexity and exhibit large self-dissimilarity — both in terms of certain descriptor values. Signal complexity is measured by Shannon entropy of the values inside a region (of a certain scale). Self-dissimilarity is approximated by the change of the values' probability density function (*pdf*) across different scales.

Mathematically, a region's saliency score Y_D is defined as the product of two scalar values, Shannon entropy H_D and self-dissimilarity W_D. D is the set of all values for the chosen feature descriptor. With variable d taking on values in D, function $p(d; s, \mathbf{x})$ describes *pdf* of descriptor values within the circular sample region with scale s located at $\mathbf{x}$. The equations for Y_D, H_D, and W_D are as follows:

$$Y_D(s, \mathbf{x}) = H_D(s, \mathbf{x}) W_D(s, \mathbf{x}), \tag{3.1}$$

$$H_D(s, \mathbf{x}) = -\sum_{d \in D} p(d; s, \mathbf{x}) log(p(d; s, \mathbf{x})), \tag{3.2}$$

$$W_D(s, \mathbf{x}) = \frac{s^2}{2s-1} \sum_{d \in D} |p(d; s, \mathbf{x}) - p(d; s-1, \mathbf{x})|, \tag{3.3}$$

where $s \in R$, $\mathbf{x} \in R^2$, $d \in D$. In practice, pixel greyscale values are used as the descriptor values d, thus, $D = \{0, 1, 2, ..., 255\}$.

The *Salient Regions* detector was later generalized to be invariant to affine transforms induced by viewpoint changes [80]. This invariance is achieved by replacing the circular sampling window with an ellipse. The ellipse is summarized by a vector $\{s, r, \theta\}$, where s is the scale, r is the aspect ratio of the major axis to the minor axis, and θ is the orientation of the major axis. Brute-force searching over the three-parameter space can be very expensive. Therefore, Kadir et al. propose a seeding and local adaptation approach. They start by finding seed regions conforming to the original saliency criterion using circular sampling windows. The seed regions are then locally adapted by searching for optimal s, r and θ values (equivalent to deforming the seed circles to ellipses at their optimal scales), to maximize the regions' saliency measure. This local adaptation method greatly improves efficiency.

There are a few drawbacks to this method. First, the circular sampling window used in the seeding procedure may prefer isotropic structure to anisotropic structure. This bias may contribute to low repeatability scores under viewpoint change. Because a change of viewing angle will skew isotropic structures in one image to anisotropic ones in the other, they do not get an equal chance of being detected. Second, feature locations detected with circular sampling windows will need additional adjustment to fine-tune the center of the deformed region. This positional refinement was not conducted in the original work. Nevertheless, the authors' innovative attempt at introducing information theory into feature detection is in line with human attention mechanisms. We believe this saliency measure may capture more of the regular structures in the scene and be less distracted by noisy patterns that contain little information for matching. By using this property, we hope our method is more likely to repeatably detect features under wide baseline conditions.

3.2 The Structure Guided Salient Region

Based on the above analysis, we propose a different route to salient region detection by seeding with local structure. Similar to Kadir et al.'s method, we also perform a two-step procedure of seeding and local saliency detection. But our seeding makes use of

local intensity structures in the image. After describing our representation of features in Section 3.2.1, we will present our seeding and detection steps in Sections 3.2.2 and 3.2.3 respectively. In Section 3.2.4, we will introduce a new method to estimate *pdf* of feature regions. Besides being more robust to noise and suitable for small patches, this simple method enables us to extract salient regions from *color* images.

3.2.1 Representation of the scale and affine invariant features

We parameterize a scale and affine invariant feature l by $f_l = \{\mathbf{x}_l, s_l, \mathbf{T}_l, \mathbf{v}_l\}$, where $\mathbf{x}_l$ is a 2×1 vector $(x_0, y_0)^T$ signifying the center of the feature region, s_l is a scalar describing the feature's scale, $\mathbf{T}_l$ defines the shape of the image region covered by this feature, and $\mathbf{v}_l$ contains the descriptor values for this feature. Here, $\mathbf{T}_l$ is the structure tensor (as defined in 2.1), represented by a normalized 2×2 symmetric matrix $\begin{pmatrix} A & B \\ B & C \end{pmatrix}$. It is equivalent to representing an elliptical shape by its aspect ratio and orientation, but the tensor representation is more convenient for algebraic computations.

In essence, image feature detection is the estimation of $\{\mathbf{x}_l, s_l, \mathbf{T}_l, \mathbf{v}_l\}$ for all points of interest. Feature matching is the process of establishing correspondences between features from two images by examining the similarity of the feature descriptor values $\mathbf{v}_l$.

3.2.2 Seeding using local structure

The entropy-based saliency theory requires a feature to have a large change of *pdf* over scale, they typically correspond to image blobs that have large intensity variation with respect to their surrounding pixels. Thus, we propose to use these blobs as seeds for saliency detection.

Scale-invariant blob detection techniques can be used to extract blobs. For example, Lindeberg [97] detected blobs by searching for local extrema of Laplacian-of-Gaussian filtered images in scale space. This method detects circular blobs only. For arbitrary blob shapes, one needs an affine-invariant blob detector like the Hessian-Affine detector [116]. Affine-adaptation will need to compute the structure tensor of a region's neighborhood, which is usually much larger than the region itself. For images of 3-D scenes, this large neighborhood is likely to cover surface depth changes, in which case the local neighborhoods are no longer covariant to affine transforms.

Here, we use MSER [111] blobs as our seeds. Since their detection procedure relies solely on image intensity contrast, those with high intensity variation with respect to their surrounding neighbors are preferred over those with low contrast. To capture distinctive points with even minute contrast changes, we lower the "minimum margin" requirement between the inner and the outer regions. This will result in a large collection of regions, many of which may be detected due to noise. Those noisy regions will then be eliminated when their statistical properties are further examined, as described in detail in the next section.

One interesting property of the seeds is that their shape is readily obtained by analyzing the region boundary. Each region l is enclosed by an ellipse, represented by its location $\mathbf{x}_l = (x_0, y_0)^\top$, scale s_l, and structure tensor $\mathbf{T}_l = \begin{pmatrix} A & B \\ B & C \end{pmatrix}$. The ellipse can be defined by a quadratic equation:

$$\Big(\mathbf{x} - \mathbf{x}_l\Big)^\top \mathbf{T}_l \Big(\mathbf{x} - \mathbf{x}_l\Big) = s_l^2. \tag{3.4}$$

3.2.3 Local salient region adaptation

Now that we have obtained the initial set of feature seeds $\mathcal{F} = \{f_1, ...f_N\}$, where $f_l = \{\mathbf{x}_l, s_l, \mathbf{T}_l\}$, $l \in 1, ...N$, we will examine their saliency as defined in Equation (3.1). We will also locally adapt the seeds to choose the position and scale for which they achieve optimal saliency. Since the region boundary already gives a good estimate of the elliptical shape, we will keep it fixed during the optimization. In the adaptation, we will maximize the two criteria, H(region entropy) and W(inter-scale saliency), by alternating *scale saliency selection* steps with *location refinement* steps.

We begin with a *scale saliency selection*. If the initial seed is scale salient (has a local H maximum), it will undergo *location refinement*, otherwise, it will be discarded. For seeds passing the initial scale saliency test, the *location refinement* will end when either maximum H and W are found or the iteration limit is encountered.

Scale saliency selection When choosing the optimal scale of a seed region $f_l = \{\mathbf{x}_l, s_l, \mathbf{T}_l\}$, we look for a local maximum of $H(s_l, \mathbf{x}_l)$ by changing the scale s_l while keeping the location $\mathbf{x}_l$ fixed. If there exists a local maximum at scale s_l', we update this seed's scale to s_l' and proceed to *location refinement*. If no maximum is obtained, this seed is regarded as non-salient and discarded.

Since we have already obtained a rough scale in the seeding step, we can search more efficiently thanks to two simplifications. First, the search range of s'_l can be set to be small. This is in contrast to the original scale-saliency method [80], where a large search space is needed in order to capture all possible Salient Regions. Second, we can stop searching once we encounter the first local maximum H. This is because we are already working in a predefined narrow range of scale and the first characteristic scale is sufficient in defining a tight bound on the interest region.

Location refinement Once the seed's optimal scale is obtained, we maximize the seed's $W(s, \mathbf{x})$ by looking for the nearest neighbor that has a higher $W(s, \mathbf{x})$. Within a certain range, if there is a region at $\mathbf{x}'_l$ that has a larger $W(s, \mathbf{x})$, we *tentatively* move the seed to this position (by updating $\mathbf{x}_l$ with $\mathbf{x}'_l$). This position adjustment will be confirmed if the region also exhibits scale saliency at the new position. If, on the other hand, no neighbor has a better $W(s, \mathbf{x})$, we stop the iteration and take the current $\mathbf{x}_l$ as the optimal position.

3.2.4 Robust histogram estimation and extension to color images

Region intensity histogramming is used for estimating the local *pdf* over the elliptical feature region. For an 8-bit greyscale image, for example, a 256-bin histogram is used to count the number of occurrences of pixels with gray levels from 0 to 255. We find, however, that the region's local intensity histogram is very sensitive to noise. This sensitivity is more evident when the region is small, since only a small number of pixels are used in filling the histogram and small deviations of some of the greylevel values will change the overall histogram significantly.

We tackle this problem by approximating *pdf* of pixel values within a region with a coarser histogram. The approximation comprises applying a Gaussian smoothing on the original intensity histogram (in the case of greyscale images), followed by down-sampling the smoothed histogram to fewer bins. The smoothing window size is related to the down-sampling factor. Here, for greyscale images we use a down-sampling factor of 4 by representing the original 256-bin histogram with a coarser 64-bin histogram. This procedure makes salient region intensity *pdf* estimation more robust to noise.

More importantly, this robust representation enables us to deal with color images, which demands prohibitive computation if using the original formulation of scale saliency due to high dimensional histograms. For example, one would have to work on a histogram of

dimension 16777216 ($256 \times 256 \times 256$) with a normal RGB image. Apart from computational complexity, the traditional representation is also very sensitive to noise. With a down-sampling factor of 16 for RGB color images, we will end up working with 4096-dimensional ($16 \times 16 \times 16$) histograms, which are more efficient to compute and less affected by image noise.

3.3 Performance evaluation on planar scenes

The objective of performance tests on planar scenes is to evaluate the extent to which SGSRs are invariant to viewpoint changes. We use the testing methodology proposed in [116]. In testing performance under viewpoint changes, we ran the SGSR detector on a set of images of the same planar scene (*graffiti*, Figure 3.3 [1]) acquired from different viewpoints. The homographies between the images are given as ground truth.

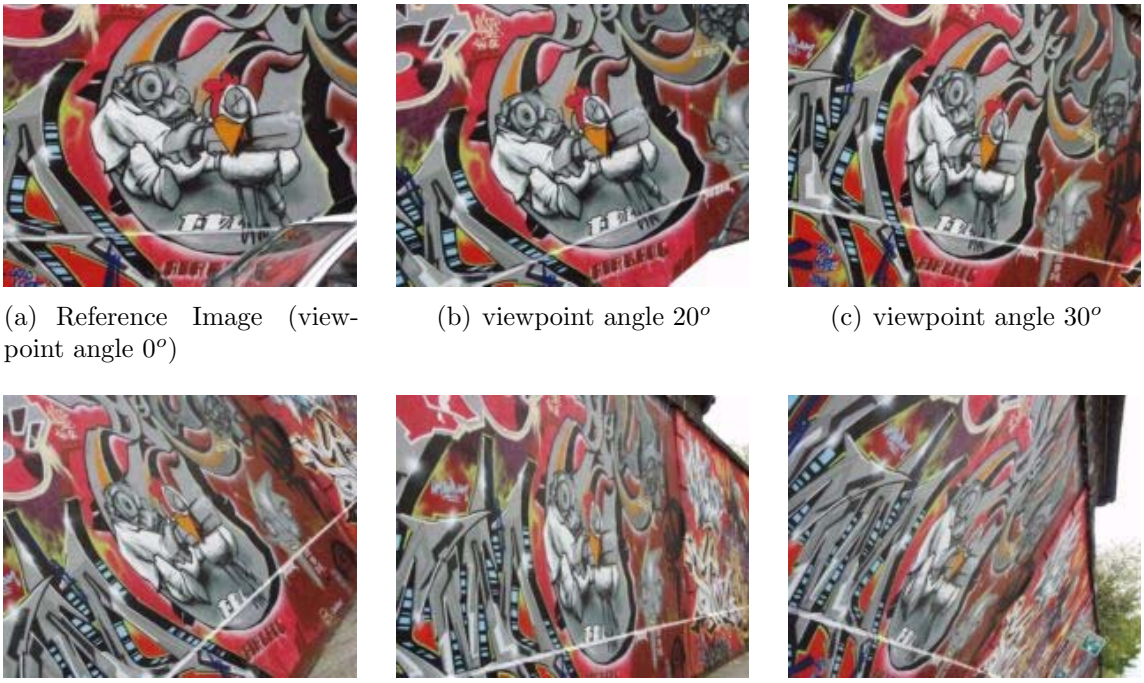
Here, we test SGSR against the state-of-the-art detectors reported in [116]: Hessian-Affine detector, Harris-Affine detector, MSER detector, Intensity Extrema-based Region detector, and Edge-based Region detector. We compare them based on four performance indicators: the number of correspondences, repeatability, the number of correct matches, and the matching score (as defined in [116]):

- *The number of correspondences* is the absolute number of region pairs (between the reference image and the matching image) which are repeatably detected. Two regions are deemed to be repeatably detected if the overlap error ϵ_O is sufficiently small (in this experiment, we choose $\epsilon_O \leqslant 40\%$). The overlap error is defined as the error in the feature areas when the two corresponding regions are converted to a common coordinate frame according to the homography:

$$\epsilon_O = 1 - \frac{R_{\mu_a} \bigcap R_{H^T \mu_b H}}{R_{\mu_a} \bigcup R_{H^T \mu_b H}}, \tag{3.5}$$

  where H is the homography relating the two images, and $(R_{\mu_a} \bigcap R_{H^T \mu_b H})$ and $(R_{\mu_a} \bigcup R_{H^T \mu_b H})$ represent the areas of intersection and union of the regions respectively.

[1] Retrieved from `http://www.robots.ox.ac.uk/~vgg/research/affine/`



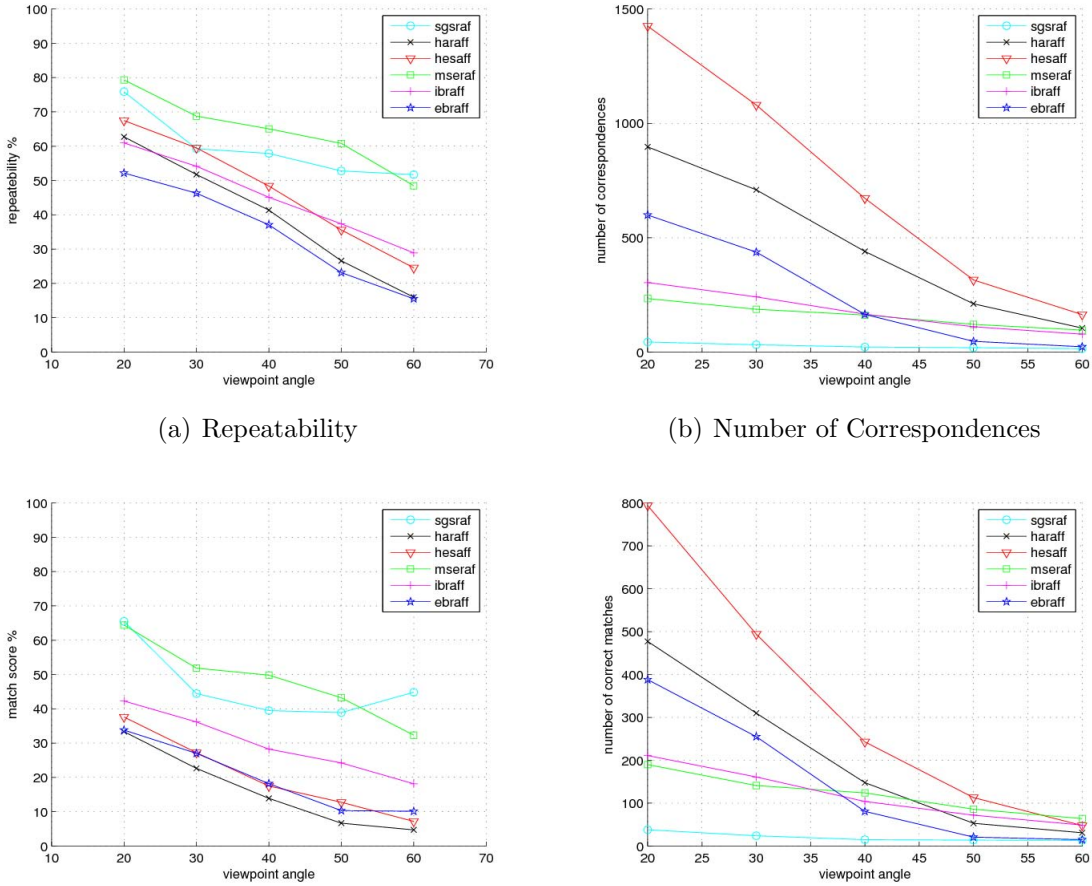
(a) Reference Image (viewpoint angle 0^o) (b) viewpoint angle 20^o (c) viewpoint angle 30^o

(d) viewpoint angle 40^o (e) viewpoint angle 50^o (f) viewpoint angle 60^o

Fig. 3.3 ***Graffiti*** **image set**

- *Repeatability* is the ratio between *the number of correspondences* and the smaller of the number of detected regions in the pair of images.
- *The number of correct matches* is the total number of correct matches among the correspondences. A region correspondence is deemed correct if the overlap error is less than a predefined threshold ($\epsilon_O \leqslant 40\%$). This is the ground truth for correct matches in the *matching score* comparison.
- *The matching score* is meant as an indication of the distinctiveness of features detected by a particular detector. The idea is to see how well the regions can be matched, when all are represented by SIFT descriptors [102]. A match is the nearest neighbour in the descriptor space according to their Euclidean distance. The *matching score* is defined as the ratio between the number of correct matches (obtained using SIFT descriptors) and the smaller number of detected regions in the pair of images. The results are indicative rather than quantitative, since they depend on many factors,

one of which is the type of descriptor that is used in representing the feature.



(a) Repeatability

(b) Number of Correspondences

(c) Matching Score

(d) Number of correct matches

Fig. 3.4 Performance evaluation on planar scene The detectors are compared on the *graffiti* image set; we show the 4 performance measurements of the detectors SGSR (denoted sgsraf), Hessian-Affine detector (hesaff), Harris-Affine detector (haraff), MSER, Intensity extrema-based Region detector (ibraff), and Edge-based Region detector (ebraff).

Comparison results The repeatability comparison results are reported in Figure 3.4(a), showing repeatability as a function of viewpoint change. SGSRs achieve competitive performance for most viewing angles, but rely on a relatively small number of features (Fig-

ure 3.4(b)). The matching scores of SGSRs are close to that of the best performer, MSERs, for smaller viewpoint angle changes, and 10% better than MSERs for a viewpoint change of 60° (Figure 3.4(c)). Again, this is achieved using a much smaller number of features (Figure 3.4(d)).

One unique feature of the SGSR detector is that it achieves competitive results using the most compact set of features. This can be advantageous when applications (such as object or landmark recognition) require a compact representation, as we find that most detectors' performances decline when they are asked to detect a smaller set of repeatable features. It is shown in Figure 21(c) of [116] that most detectors' repeatability falls with decreasing number of features used.

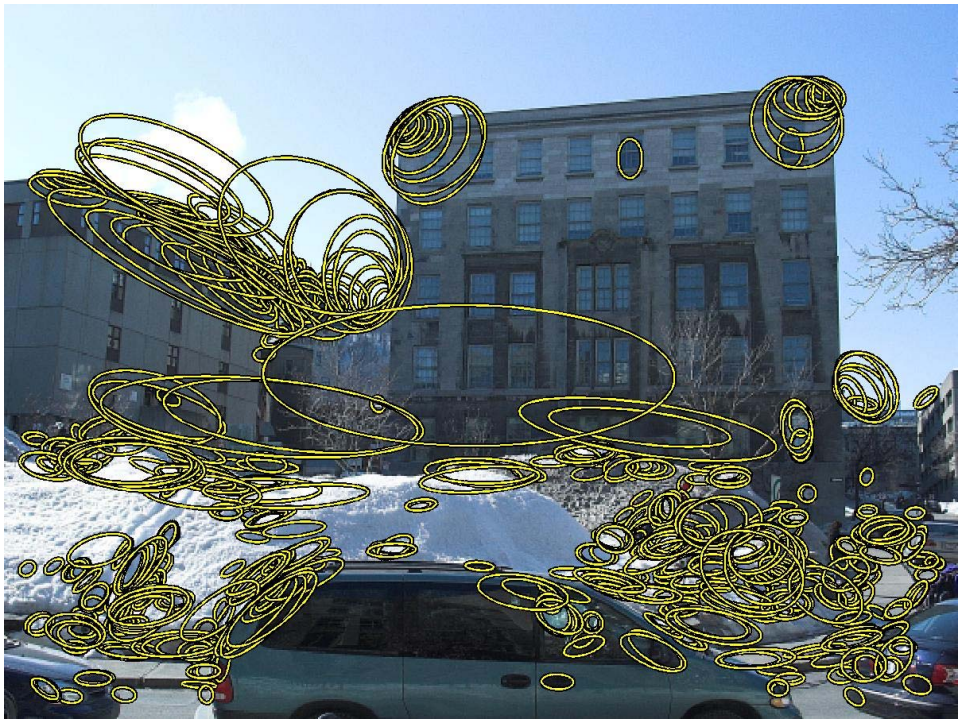
3.4 Performance evaluation on 3-D scenes

The aim here is to measure our method's performance at detecting features in images of 3-D urban scenes for the purpose of wide-baseline matching. We test on real-life images of two 3-D scenes: one is the *J-Scene* (Figure 3.1) and the other is a scene of the ZuBuD data set[1] (Figure 3.8). For each scene, we apply three different feature detectors, Hessian-Affine, MSER, and SGSR. To gauge the quality of features localized by each of the detectors, SIFT descriptors are used as a common basis for matching (procedure given in Section "Feature matching results"). We present our results for both scenes in the Tables 3.1 and 3.2. We will focus our discussion on the *J-Scene* since both results are similar.

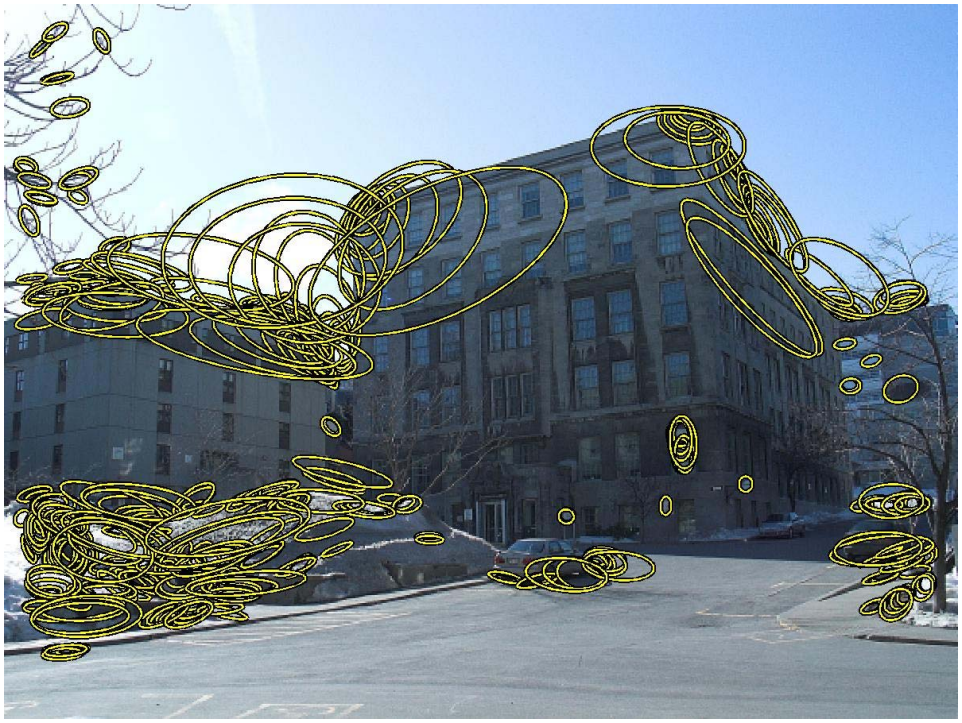
Feature detection results Figures 3.5, 3.6 and 3.7 show the features found on the *J-Scene*. The Hessian-Affine features occur mainly in two places: corners and edges of buildings, where surface discontinuities occur; and snow-banks, which are densely textured and full of noise. In comparison, fewer MSERs occur on building edges and corners and more of them are detected on the building walls. MSERs are also densely detected on the snow-banks and tree branches. The SGSR detector mainly captures blob structures on the building walls and much fewer of them occur in noisy parts of the scene such as snow-banks and tree branches.

The results show that the Hessian-Affine detector failed to detect structures such as windows and bricks on the wall. These blobs are close to each other and create a regular

[1]Retrieved from `http://www.vision.ee.ethz.ch/showroom/zubud/index.en.html`

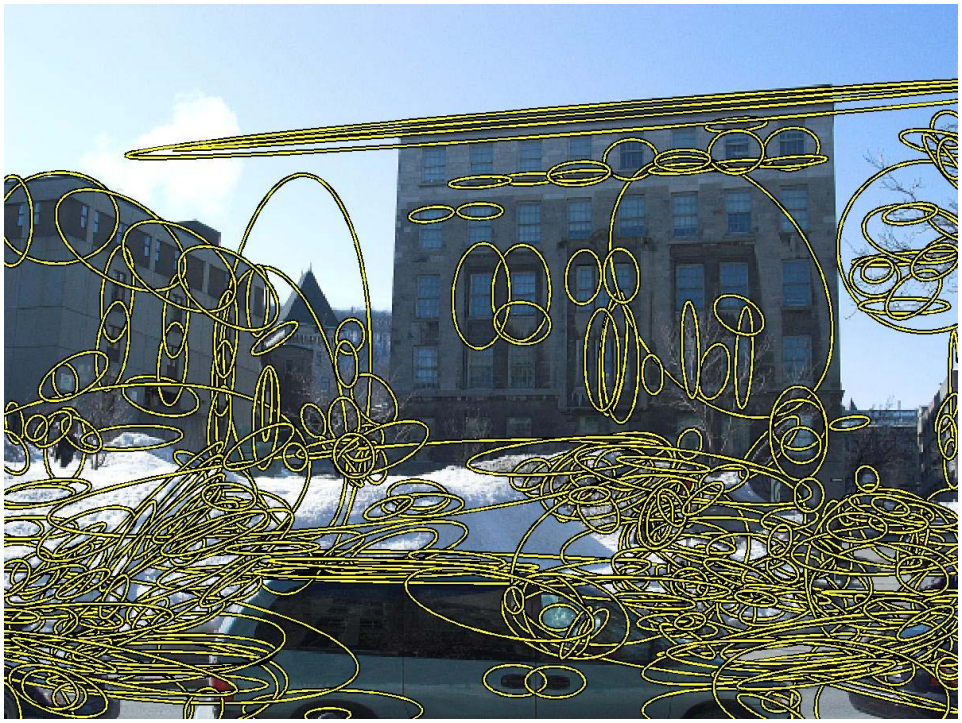


(a) left

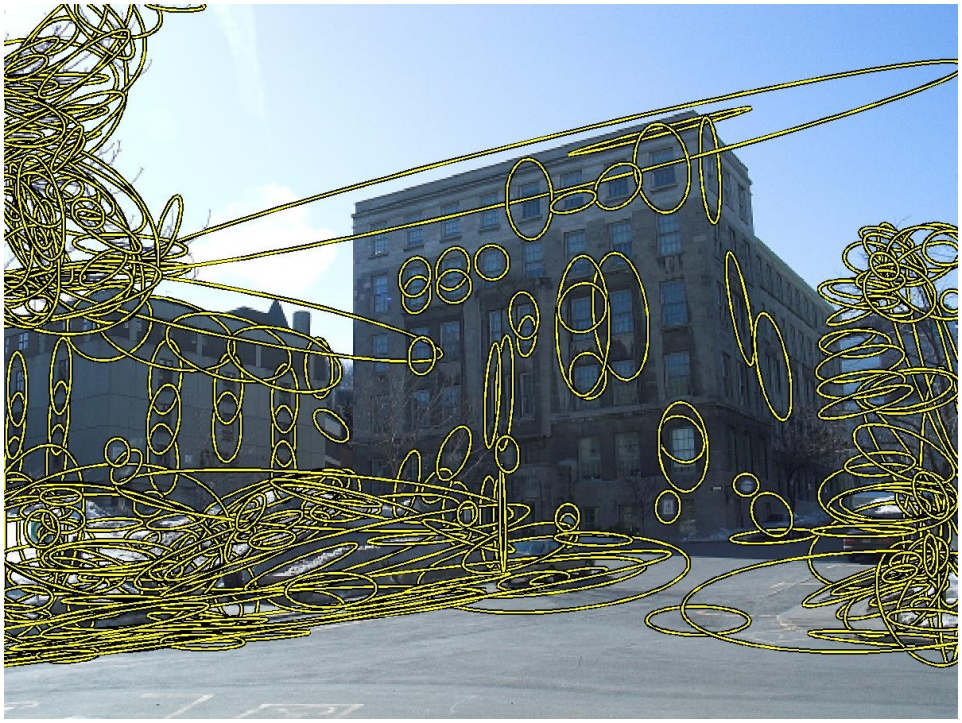


(b) right

Fig. 3.5 Hessian-Affine features detected on the *J-Scene*



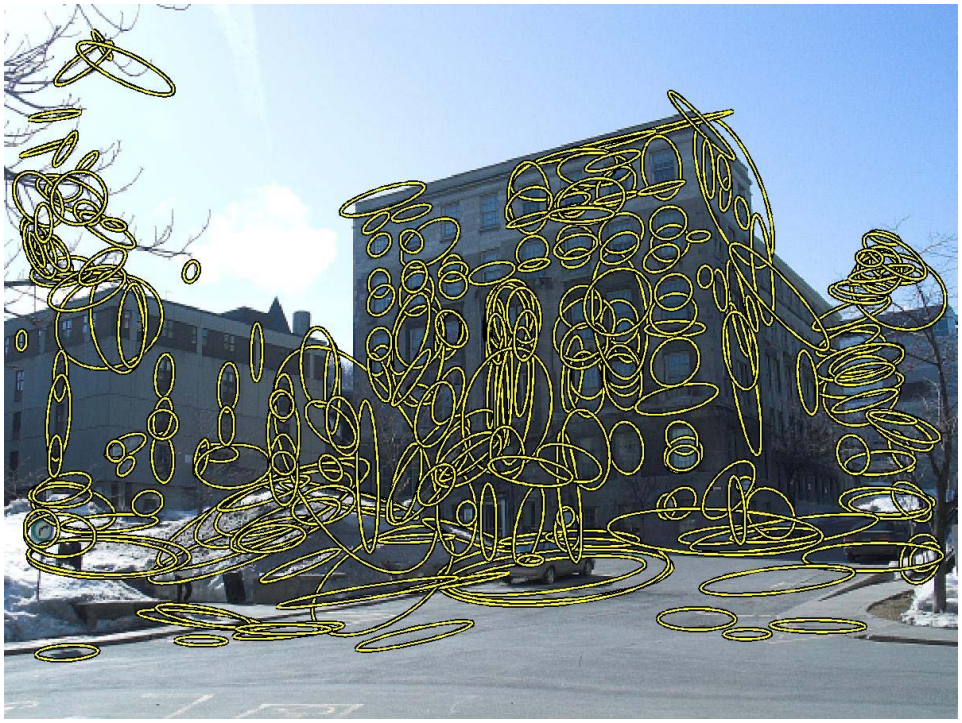
(a) left



(b) right

Fig. 3.6 MSER features detected on the *J-Scene*

(a) left



(b) right

Fig. 3.7 SGSR features detected on the *J-Scene*

repetitive pattern. If we look at the only window detected (on the upper part of the front building in Figure 3.5(a)), it is isolated from its neighbors with distinct intensity. The MSER detector was able to extract some high-contrast blobs, but it also responded positively to many noisy regions. The SGSR detector captures most of the blob patterns on the walls and also discarded many noisy regions.

Feature matching results For each detector, we perform a feature matching experiment with the following procedure. First, the features are normalized to a fixed-sized circular region and their SIFT descriptors are extracted. Second, we obtain the initial set of matches by nearest-neighbour matching in the descriptor space. Finally, outliers are rejected by global consistency checking using RANSAC [57].

For the *J-Scene*, Table 3.1 compares the number of detected features, the number of matched features and the number of outlier matches found by the three detectors. We can see that SGSRs perform best for wide baseline matching of the *J-scene*. In contrast, MSERs and Hessian-Affine regions are poorly matched. Hessian-Affine regions are either not distinctive enough (building corners will have similar SIFT descriptors) or not repeated in the scene (lower part of the images, such as noisy snow-banks and cars). Thus, no correct match is found. Although the MSER detector repeatably captures some high contrast regions such as windows, their SIFT descriptors are not distinctive enough due to large region sizes and different light reflectances of the corresponding window glasses (cf. windows on the side building in Figure 3.1).

Detector	# Features Detected (left-right)	# Total Matches	# Outlier Matches
Hessian-Affine	569-382	2	2
MSER	311-271	4	2
SGSR	266-258	15	2

Table 3.1 Feature matching comparison for the *J-Scene*

We did the same experiment on the *ZuBuD* scene (Figure 3.8). Like the *J-Scene*, the *ZuBuD* scene also has the characteristics of low-contrast structures, large areas of occlusions, and small image overlap. The result in the Table 3.2 echoes our conclusion drawn from the previous data: the SGSR detector excels in being matched in large quantity with few outliers.

(a) left view (b) right view

Fig. 3.8 Stereo images of a *ZuBuD* scene

Detector	# Features Detected (left-right)	# Total Matches	# Outlier Matches
Hessian-Affine	1831-1814	21	3
MSER	585-586	17	5
SGSR	807-597	23	1

Table 3.2 Feature matching comparison for the *ZuBuD* scene

3.5 Conclusions

In this chapter, we presented a novel feature detector that is invariant to scale and viewpoint changes. We used the entropy-based saliency [79] as the measure for selecting salient features in images. This method is different from the original methodology of Kadir et al. both in the initial seeding procedure and in the subsequent local region adaptation, and it is shown to be better suited for wide-baseline tasks in urban environments.

We also introduced a histogram down-sampling procedure to robustly represent greyscale or color pixel-value *pdf* of interest regions. The procedure can smooth out mild intensity variations due to image noise and it can also extend the applicability of entropy-based saliency to color images. The competitive performance of the new feature detector is demonstrated on both planar and 3-D scenes.

Chapter 4

Context-consistent feature matching

After considering feature extraction, we now move on to the second sub-problem of robustly matching features between widely separated perspectives.

Matching image features that correspond to the same 3D location remains a challenge with real-life images. One reason is that the same 3D object will look quite different from two different perspectives. Another reason is that, in the case of images containing man-made structures, periodic elements such as building facades and road grids can confuse matching algorithms. Other difficulties come from occlusion, illumination change, lack of texture, surface reflectance, etc.

Currently, given features extracted from different perspectives, success of matching depends largely on how one characterizes feature appearance, typically using local invariant descriptors (e.g., SIFT [102], GLOH [115]). Relying on those characterizations, feature matching is a matter of pairing up features whose descriptors are most similar [116]. We need to clarify that, although matches obtained by local methods can be post-processed to reject outliers using the epipolar constraint, our emphasis here is how to obtain a set of good *initial* matches, whose quality often decides the success or failure of the entire process. For added robustness, post-processing can always be used afterwards. Common to all local feature matching methods is that they depend purely on local information: a descriptor tells nothing more than the local appearance of a feature; where the feature resides and what its neighbors look like are not considered. On one hand, this focus on local is well-motivated: under wide viewpoint changes, image regions that cover large areas will typically contain abrupt depth changes, which often cause occlusions. Subsequently,

extended regions surrounding features will frequently undergo drastic appearance changes, making it harder to match those regions than to match the smaller features directly. On the other hand, it is this contextual information that we humans are using to be able to successfully relate images that are taken across a wide range of different imaging conditions. Humans can effortlessly extract information about the regional structure of the scene and use that structure to guide the more localized matching process [128].

We believe a mechanism to involve this regional structural information will help to disambiguate between less distinctive candidates. We propose a graph model, called the *Salient Feature Graph (or SFG)*, to embed the regional scene structure. Based on the SFG, we then develop a matching algorithm, *Context-Consistent Assignment (or CCA)*, that uses neighborhood structure in a manner that is unaffected by the afore-mentioned viewpoint changes.

Our method is a shift of paradigm in terms of matching strategies — shifting from the traditional local-based paradigm to the context-driven paradigm. In testing our method, we closely examine results of both approaches. Current literature shows that the leading method to affine-invariantly match features is through the afore-mentioned feature descriptors, i.e., *nearest-neighbor matching* [116]. It is widely used in latest 3-D vision applications [165][166][1]. Thus, we test the two methods on a variety of 3-D urban scenes. In many difficult matching tasks, our method works robustly and shows superior performance.

The remainder of this chapter is organized as follows. Section 4.1 reviews some related work. After describing the Salient Feature Graph modeling in Section 4.2, we explain, in Section 4.3, how to use it to compare similarity between regional structures seen from different perspectives. In Section 4.4, we present our overall feature matching algorithm — *Context-Consistent Assignment*. Section 4.5 presents the data sets as well as the criteria we use to evaluate our method. Experimental results are presented in Section 4.6.

4.1 Background

For more than thirty years [92], matching features between stereo images has been a fundamental problem for computer vision research. Up untill the early 1990s, researchers focused on images taken by cameras that are horizontally placed side by side. Based on this strict configuration, a pair of matched points can tell exact depth of the corresponding scene location by triangulation. Instead of trying to densely match each pixel, early works were

limited, by computer hardware and software of the time, to match a sparse set of image locations. To match features that are less distinctive (corners [122], edges [110], or line segments [112]), various methods were devised to involve image contextual information, such as using a "cooperative" [109] or relaxation algorithm [7], constructing a hierarchy of features [138][95], or building a neighborhood graph of edge segments [6]. These all assume that the binocular cameras are parallel-axis. Thus, matching is greatly simplified to a one-dimensional search problem. Configuration-specific assumptions are commonly used [77], e.g., ordering constraint along equivalent raster lines, similar orientation for line segments, relative position of neighboring features, etc.

Lately, researchers have loosened the requirement for parallel camera axes and matched features between more challenging perspectives. One has to consider candidates at all image locations due to the fact that the epipolar geometry is unknown. With the recent success at characterizing local feature appearances with descriptors [152][102], it has become a standard practice to match features via nearest-neighbor matching of their appearance descriptors (e.g., SIFT).

In addition to the prevalent use of local feature matching, there are some works using similar ideas as ours — directly matching sparse features by using their extended image neighborhood. In matching Harris corners, Zhang et al. [201] define a measure of neighbor-support for the matches and disambiguated matches through relaxation. For each feature, Deng et al. [40] build affine-invariant log-polar elliptical bins to involve regional context information.Their initial "anchor features" are matched exclusively using SIFT descriptors and they disambiguate less distinct matches by accumulating the support-count of SIFT-matches. Similarly, Sidibe et al. [161] employe a simplified relaxation labeling algorithm to match features, also based on the SIFT descriptor. Compared with these methods, our method makes an extra effort in explicitly addressing the issue of affine-invariance due to viewpoint changes. Instead of depending on characterizing local appearance affine-invariantly, we focus on seeking neighborhood consensus by explicitly modeling the affine transform of a feature's extended neighborhood.

4.1.1 Proposed approach

We match features using both local and global information of a image. All parts of the image dynamically interact with each other through iterative graph matching. This interaction

encourages making true matches whose local appearances might have undergone substantial change but have similar neighborhood structures. Mean while, it prevents from pairing up wrong features which locally appear similar only by accident. We pay special attention to high tolerance of image ambiguities due to scenes structure, viewpoint changes, and image quality degradation. For the purpose of context-driven feature matching, we embed image contextual information into an Attributed Relational Graph (ART) called the Salient Feature Graph (SFG), where each node represents a salient feature and edges connect nearby features to provide feature context. To make sure that, for the same 3-D scene, SFGs of different perspectives have consistent graphical structure, the edge-connectivity of the SFG is *designed to be invariant to viewpoint changes*. Utilizing the representational power of the SFG, we propose a new method, *Context-Consistency by Neighborhood Transform*, to examine neighborhood similarities. The *Neighborhood Transform* (NT) is a geometric procedure devised to compensate for affine-distortion of a neighborhood due to viewpoint change, thus facilitating comparison of two neighborhoods using a consistent set of features. We then propose an algorithm, Context-Consistent Assignment (*CCA*), to propagate confidence about feature-matches across the image and obtain an increasing number of matches through successive iterations. Through image-wise graph connections, features from various parts of the image interact dynamically. Thus, each feature contributes globally to the overall matching.

A recent paper by Choi and Kweon [22] contains several elements similar to our work: we both assume that scene surfaces vary continuously and are locally planar, we both measure compatibility between pairs of neighboring correspondences, and we both iteratively aggregate confidence of matches from local neighbors — they match by relaxation optimization and we use *softassign* [61] style iterations. However, there are several important differences. The first and *fundamental difference* is that Choi and Kweon work on a set of given matches to select a subset of good correspondences, while we work on putting together an initial set of matches. Thus, in a wide-baseline stereo pipeline, their procedure occurs *after* our feature matching step. It is a post-processing procedure for feature matching, or, as we stated in Section 1.2.4, a preparative step for the fundamental matrix estimation. As a result, their success depends on this initial set — if some true matches are not in the initial set because of appearance discrepancy, they will not even be considered. Our method, on the other hand, starts by assuming any match is possible and the final matches are determined by aggregation of both local and global information.

Second, our approach draws on all available neighborhood information to judge similarity, not only the neighbors that have been initially matched. Due to the above differences, we also take different routes to the problem formulation and optimization. By introducing confidences to initial matches (including conflicting ones), they solve a constrained optimization problem by relaxation labeling. Each feature is narrowed down, from the initial set of conflicting candidates, to matching with one candidate with high confidence. We formulate feature matching as a graph matching problem and solve it by deterministic annealing optimization [61]. Our method explicitly handles the one-to-one constraint and allows missing nodes/edges. Nevertheless, their method can be an excellent complement to post-process and refine the initial correspondences.

4.2 Salient Feature Graph

This section describes our method to embed contextual image information into a graph model that is suitable for matching features. The rationale of using graphs is that if our graph model can encode intrinsic structure of the visual scene, we will have similar graph models of a scene's images from different viewpoints. Feature matching is thus mapped to a graph matching problem, leading to more reliable results thanks to the relational cues provided by the graphs.

We model an image with an ARG that we call the *Salient Feature Graph (SFG)*. Essentially, the SFG is an undirected ARG where the nodes represent local features and edges connect nearby features in a carefully-designed way. A *SFG* is represented as $G = \{N, E\}$, where N is the set of nodes (G_a) and E the set of edges (G_{ab}) of the Graph G.

For the purpose of wide-baseline stereo matching, we call on two principles in designing the *SFG*. The first is the *Original Node* principle: all nodes carry the original geometric information in the image. This ensures that no geometric information is lost due to generalization of local appearance, as geometry can be a powerful tool in subsequent context-compatibility analysis. The second is the *Invariant Edge* principle: the neighborhood relationship (edge-connected or not) should be preserved regardless of the viewpoint from which the images are taken. This ensures that the features from different images are matched using the same context. Construction of the *SFG* comprises two steps: (1) Summarize the given features using the Original Node principle (optionally, if no feature is provided, detect features before the summarization); (2) Insert *SFG* edges by the Invariant

Edge principle.

4.2.1 Nodes

In principle, any sufficiently repeatable affine feature can be used as our SFG nodes. If no pre-extracted features are provided to our algorithm, we create nodes by detecting Maximally Stable Extremal Regions (MSERs) (Matas et al. [111]), which were found to have good repeatability in many cases. We summarize each node G_a with an attribute vector $\mathbf{G}_a$, containing a feature's photometric and geometric information.

We define $\mathbf{G}_a = \{\mathbf{x}_a, s_a, \rho_a, \theta_a, \mathbf{p}_a\}$, where $\mathbf{x}_a$ is a 2×1 vector $(x_a, y_a)^T$, representing the center of the feature's elliptical region, s_a is a scalar describing the feature's scale, ρ_a is the axis ratio and θ_a the orientation of the ellipse' major axis. Therefore, the radii along major and minor axes of the ellipse are $s_a\sqrt{\rho_a}$ and $s_a/\sqrt{\rho_a}$ respectively. The above values describe G_a's geometric properties and are not used for direct similarity measures. Rather, during the graph matching process, they provide semantic information for context-compatibility analysis. The last component, $\mathbf{p}_a$, contains the photometric information of this feature. We describe it using a low-dimensional version of the spin image descriptor [88] to be tolerant to noise.

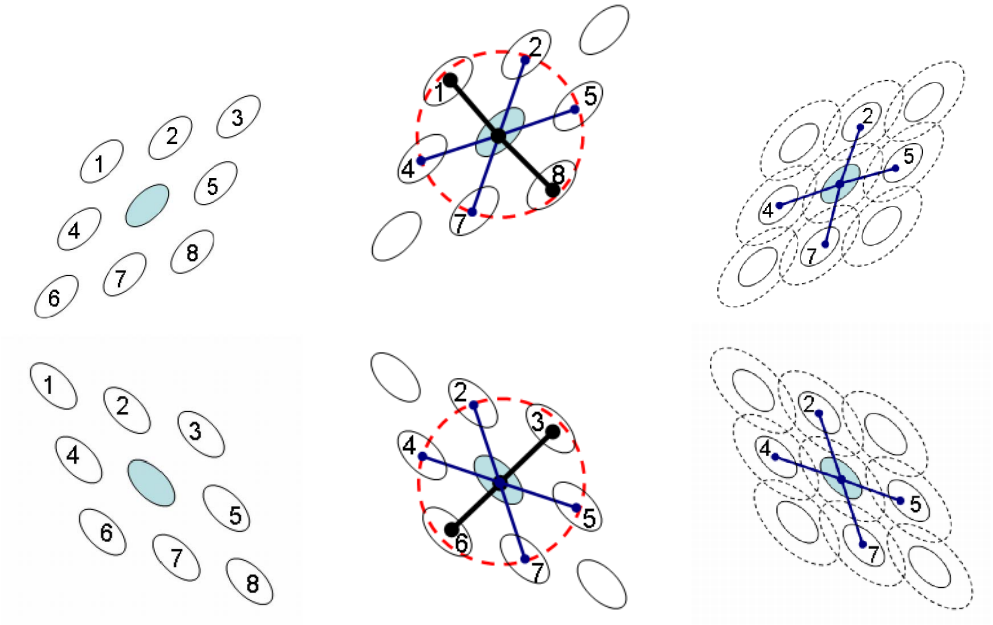
4.2.2 Edges

SFG edges connect spatially nearby features to provide nodes with neighborhood contexts. Naturally, proximity of features is measured by Euclidean distance in the image coordinate space. Some algorithms treat features as points and cluster them using the k-means algorithm [36]. Others also take feature scale into account [53]. None of them complies with the Invariant Edge principle. Actually, one can guarantee to connect features unaffected by viewpoint change only if the scene geometry is fully known, which is equivalent to solving wide-baseline stereo problem.

However, by assuming that surfaces are locally planar and that depth change in the scene is much smaller than its depth from the camera, we can connect neighboring features in a way that approximately satisfies the Invariant Edge principle. Under this commonly used assumption [149][110], the distortion of a local patch induced by viewpoint change can be modeled by an affine transform.

In the following two-step procedure for edge insertion, we try to measure features'

geometric distance affine-invariantly. First, we define a *Range of Influence* for each feature by extending the ellipse diameters K (usually $2 \sim 3$) times longer. Second, we check overlap of the Ranges of Influences between every pair of features G_a and G_b; if they do overlap then we insert an undirected edge G_{ab} between them. This effectively constructs an affine-invariant neighborhood. If two features' Ranges of Influences overlap in one image, their counter-parts in the other image (taken from a different viewpoint) will also overlap, and the opposite also holds true. The Invariant Edge principle is illustrated on Figure 4.1.



(a) Image pair (b) Variable Edge (c) Invariant Edge

Fig. 4.1 The *Invariant Edge* principle. Assume the same regularly spaced co-planar blobs (features) are viewed from two viewpoints, the upper and lower row refer to the left and right image respectively. Suppose we want to identify local neighbors for the central gray blobs in the original images (Figure 4.1(a)). Pure distance-based clustering will create variable edges (in bold) as in Figure 4.1(b), where in the left image edges connecting to nodes 1 and 8 switch in the right image to edges connecting to nodes 3 and 6. Instead, by observing the overlap of the *Ranges of Influence* (in Figure 4.1(c), dotted ellipses enclosing features), we create consistent edges connecting nodes 2, 4, 5, and 7, as in Figure 4.1(c).

According to this principle, a node will consistently select its neighbors that are coplanar with it. For features lying on different planes, they might be accidentally connected in one SFG because they are projected to nearby locations in that image, but they are likely to be disconnected in the other SFG due to their depth difference. Our edge attribute is a binary value of 1 or 0, indicating presence or absence of an edge between two features. For every node G_a, we represent the set of its edge-connected neighbors as $N_a = \{G_b : G_b \in N, G_{ab} = 1\}$. Unlike Part Based Models in recognition, the features' relative geometry is implicitly contained in the nodes rather than explicitly stated in the edges [18].

4.3 Measuring neighborhood similarity using SFG

This section describes our method for measuring similarity of two locally clustered groups of features, or neighborhoods, using the above SFG models. Under the "locally planar surface" assumption, we can counter the distortion due to viewpoint change by applying a linear transform to a neighborhood in the matching view. The resulting neighborhood can then be compared to its counterpart in the reference view. We designed such a geometric transform procedure that is called the *Neighborhood Transform*. Similarity of the resulting neighborhood pairs is what we call the *Context-Consistency*, which we use as an indication of overall similarity between two collections of neighboring features.

Although it might seem overly restrictive for two nearby features to be coplanar, in practice, many surfaces can be approximated as locally planar. Notice that local methods often assume a relatively large planar neighborhood when they describe a feature's appearance, using a region that is several times larger than the detected feature [115] [116]. Furthermore, as will be clear in Section 4.3.2, a departure from this assumption is not penalized in the process of measuring neighborhood similarity. If some neighbors are missing in one SFG or another, which is almost inevitable for the highly variable SFG models, they just miss the opportunity to contribute to the overall scores of the Context-Consistency. Similarly, if the surfaces are not strictly planar, resulting geometric inconsistencies are reflected in a continuously-varying compatibility score — for true matches, support from neighbors on a smoothly varying surface can still be credited, although with a reduced contribution depending on the degree of their non-planarity.

4.3.1 The Neighborhood Transform

We use a procedure called the *Neighborhood Transform* to relate a feature in one image to its potential match in the other image. Conceptually, Neighborhood Transform from G_a to g_i (denoted by $\mathbf{NT}_{ai}$) is the geometric transform that is needed to convert the neighborhood of G_a in the left image to that of g_i in the right image, working in the image coordinate space. Upon comparing similarity of G_a and g_i, we also consider their surrounding contexts by transforming G_a's neighborhood with $\mathbf{NT}_{ai}$. We say G_a is *Context-Consistent* with g_i if the transformed neighborhood has a high similarity with g_i's neighborhood. Higher *Context-Consistency* will give more support to the match contemplated (G_a-to-g_i match).

$\mathbf{NT}_{ai}$ can be derived from the geometry and appearance of the features G_a and g_i. Geometrically, feature $\mathbf{G}_a = \{\mathbf{x}_a, s_a, \rho_a, \theta_a, \mathbf{p}_a\}$ can be viewed as the result of a transform $\mathbf{H}_a$ on the unit circle centered at image origin $(0, 0)$, where $\mathbf{H}_a$ is the concatenation of a scaling by $s_a\sqrt{\rho_a}$ and $s_a/\sqrt{\rho_a}$ in the x and y directions respectively ($\mathbf{D}_a$), a rotation by θ_a ($\mathbf{R}(\theta_a)$), and a translation by $(x_a, y_a)^T$ ($\mathbf{T}_a$), expressed in matrix form:

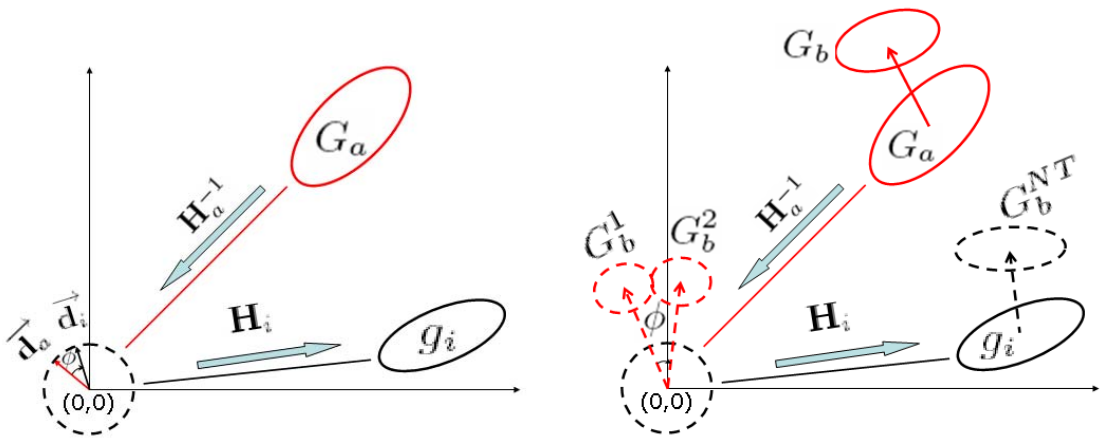
$$\mathbf{H}_a = \mathbf{T}_a\mathbf{R}(\theta_a)\mathbf{D}_a. \tag{4.1}$$

Thus, feature G_a can always be transformed back to the unit circle by $\mathbf{H}_a^{-1}$. If G_a corresponds to g_i, they can be geometrically related by a transform $\mathbf{H}_a^{-1}$ followed by $\mathbf{H}_i$ (i.e., $\mathbf{H}_i\mathbf{H}_a^{-1}$). When taking into account the appearance, however, one has to consider a rotation $\mathbf{R}(\phi)$, which aligns dominant orientations (as defined in [102]) of G_a and g_i when they are normalized to the unit circle.

In summary, *Neighborhood Transform* (from G_a to g_i) can be mathematically defined as follows,

$$\mathbf{NT}_{ai} = \mathbf{H}_i\mathbf{R}(\phi)\mathbf{H}_a^{-1}, \tag{4.2}$$

where $\mathbf{H}_a$ and $\mathbf{H}_i$ are as defined in Equation 4.1, and $\mathbf{R}(\phi)$ as above. See Figure 4.2(a) for a schematic illustration of the $\mathbf{NT}$, and Figure 4.2(b) for the transformation of G_a's neighbor by the $\mathbf{NT}$.



(a) G_a-g_i related by $\mathbf{NT}_{ai}$ (b) Transform G_b with $\mathbf{NT}_{ai}$

Fig. 4.2 Neighborhood Transform (NT). In the diagram, red solid ellipses (labeled with capital letter G) represent features in the left image and black (labeled with lower case letter g) are features in the right. Dashed features are geometrically transformed versions of the originals. In (a), G_a and g_i are related by $\mathbf{NT}_{ai}$, a concatenation of three transforms: a $\mathbf{H}_a^{-1}$-*Transform* on G_a to unit circle, a rotation of the circle by ϕ, and a $\mathbf{H}_i$-*Transform* on the rotated circle. $\overrightarrow{\mathbf{d_a}}$ and $\overrightarrow{\mathbf{d_i}}$ are dominant orientations of $\mathbf{H}_a^{-1}(G_a)$ and $\mathbf{H}_i^{-1}(g_i)$. In (b), by imposing $\mathbf{NT}_{ai}$ on G_a's neighbor G_b, we obtain its **NT**-ed counterpart G_b^{NT}, after going through G_b^1 and G_b^2 intermediately.

4.3.2 Computing Context-Consistency

Once the set of **NT**-ed nodes are obtained by the transform $\mathbf{NT}_{ai}(N_a)$[1], that is $N_a^{NT} = \{G_b^{NT} : G_b \in N_a, G_{ab} = 1\}$[2], we can compute the *Context-Consistency* between G_a and g_i by measuring similarity of the two node sets N_a^{NT} and N_i, denoted by $CC(G_a, g_i)$.

Computing $CC(G_a, g_i)$ can be approached as a many-to-many matching among elements of N_a^{NT} and N_i, followed by accumulating matching scores of the element matches. One useful aspect of the node sets is the geometric relationship among neighbors, which is ignored by conventional many-to-many bipartite matching methods [159]. We propose a procedure called *hypothesize-match* to approximate $CC(G_a, g_i)$. It exploits this inter-feature relationship and also avoids explicit node-matching. Moreover, it avoids estimating

[1] N_a is the set of G_a's edge-connected neighbors, as defined in Section 4.2.2.
[2] $G_{ab} = 1$ implies G_a and G_b are edge-connected.

features' dominant orientations, which is notoriously unstable, especially for blob-like structures [88].

Hypothesize-match Let us first consider the estimation of ϕ, the rotational component of $\mathbf{NT}_{ai}$. In light of the instability of a feature's dominant orientation, we can use orientations of some other vectors to estimate ϕ, as long as they are reliable. This leads us to the vectors that connect centers of neighboring features (vectors $\overrightarrow{\mathbf{ab}}$ and $\overrightarrow{\mathbf{ij}}$ in Figure 4.3). But $\hat{\phi}$ inferred from them will be correct only if the two vectors match. We need the rotation to make the match and we need the match to find out the rotation — a chicken-and-egg problem. This is where the hypothesize-match procedure comes into play.

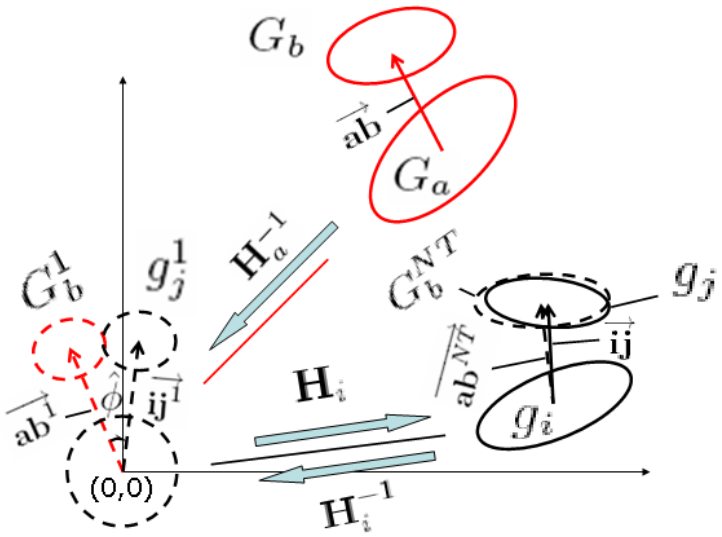
We start by hypothesizing that one neighboring pair (G_a and G_b) matches another pair (g_i and g_j) in the other image. Thus we have the rotation ϕ, and consequently $\mathbf{NT}_{ai}$, to effect the Neighborhood Transform on G_b, which results in G_b^{NT}. The correctness of this hypothesis will be reflected in node similarity between G_b^{NT} and g_j, measured in their geometry (scale s, aspect ratio ρ, and orientation of major axis θ) and appearance (feature descriptor $\mathbf{p}$). This similarity (denoted by $S(G_b^{NT}, g_j)$) will be high for correct hypotheses and low for incorrect ones.

Figure 4.3 illustrates the hypothesize-match and $\hat{\phi}$-estimation. If the hypothesis is correct that G_a and G_b match g_i and g_j, as in Figure 4.3(a), then their $\mathbf{H}^{-1}$-*Transform*-ed counterparts will differ by a rotation only. G_b^1 and $\overrightarrow{\mathbf{ab^1}}$ are obtained by $\mathbf{H}_a^{-1}$-*Transform* on G_b and $\overrightarrow{\mathbf{ab}}$, similarly, g_j^1 and $\overrightarrow{\mathbf{ij^1}}$ by $\mathbf{H}_i^{-1}$-*Transform* on g_j and $\overrightarrow{\mathbf{ij}}$. Thus, the angular difference between $\overrightarrow{\mathbf{ab^1}}$ and $\overrightarrow{\mathbf{ij^1}}$ is our estimated rotation angle $\hat{\phi}$. After applying $\mathbf{H}_i\mathbf{R}(\hat{\phi})$ to G_b^1, we obtain G_b^{NT}, which will have a high similarity to g_j. If, however, our hypothesis is wrong, as in Figure 4.3(b), the angle $\hat{\phi}$ between $\overrightarrow{\mathbf{ab^1}}$ and $\overrightarrow{\mathbf{ij^1}}$ doesn't reflect the true rotation needed for aligning the local patch. Thus, if we apply $\mathbf{H}_i\mathbf{R}(\hat{\phi})$ on G_b^1, the resulting G_b^{NT} will be noticeably different from g_j.

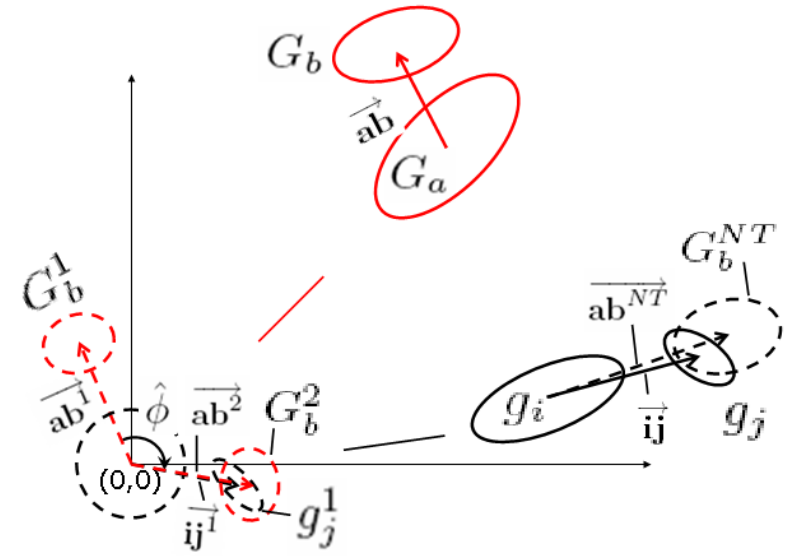
The approximation of $CC(G_a, g_i)$ using hypothesize-match can be computed as

$$CC(G_a, g_i) = \sum_{G_b \in N_a} [\max_{g_j \in N_i} (M_{bj} S(G_b^{NT}, g_j))], \tag{4.3}$$

where, the inner part $(M_{bj} S(G_b^{NT}, g_j))$ is the estimated support for G_a-to-g_i match, by looking at consistency between pairs G_a-G_b and g_i-g_j. It is the node similarity $S(G_b^{NT}, g_j)$ weighted by M_{bj} — the current probability that G_b and g_j match. This is similar to the idea



(a) G_b and g_j are correct match



(b) G_b and g_j are wrong match

Fig. 4.3 Node Similarity Measure by *hypothesize-match*. Node similarity of G_a and g_i will be re-enforced if their neighbors G_b and g_j are compatible, i.e., G_b^{NT} and g_j are similar.

of relaxation labeling, where the matching probability is propagated and updated across local context during each iteration [131]. For each given G_b, we take as the true match the g_j that produces the maximum support, which in turn contributes to our final $CC(G_a, g_i)$.

Because the Neighborhood Transform puts a very strict geometric rule over pair-wise relationships, the hypothesize-match will give rise to high support only for truly correct matches, and it will generate very low — almost negligible — support for wrong matches. Also, the hypothesize-match effectively eliminates the need to explicitly match the node sets N_a^{NT} and N_i, which is not at all a trivial task by itself.

4.4 Matching of *SFG*s

We described how to model an image with a SFG in Section 4.2 and how to compare similarity of certain image neighborhoods using the SFG in Section 4.3. Here, we show how we solve the image matching problem by optimizing graph matching using the developed tools.

Because our SFG models of the visual scene will vary greatly across views and a large portion of the nodes and edges will not be consistent across two views, we are interested in methods to inexactly match ARGs.

4.4.1 Problem formulation

Quadratic assignment formulation

Inspired by Li's [94] constructing and matching of ARGs abstracted from images, Gold and Rangarajan formulated the matching of two ARGs as a two-way assignment problem [61]. Thus, they represent the solution by a 0/1 match matrix M: $M_{ai} = 1$ if node a in graph G corresponds to node i in graph g.

Given two ARGs G and g that have A and I nodes respectively, matching of G and g is formulated as finding the match matrix M such that the following objective function is minimized:

$$E_{arg}(M) \quad = \quad -\frac{1}{2}\sum_{a=1}^{A}\sum_{i=1}^{I}\sum_{b=1}^{A}\sum_{j=1}^{I} M_{ai}M_{bj}\sum_{r=1}^{R} C_{aibj}^{(2,r)} \quad + \quad \alpha\sum_{a=1}^{A}\sum_{i=1}^{I} M_{ai}\sum_{s=1}^{S} C_{ai}^{(1,s)}, \quad (4.4)$$

where, $\sum_{i=1}^{I} M_{ai} \leqslant 1$, $\sum_{a=1}^{A} M_{ai} \leqslant 1$, and $M_{ai} \in \{0, 1\}$. Parameter α controls the weight of the node similarity. In measuring node similarities, a total of S node attribute types are considered. The unary term $C_{ai}^{(1,s)}$ reflects the difference of s'th attribute between nodes G_a and g_i. Their sum is the overall cost of node G_a matching to node g_i. The more similar G_a and g_i are, the smaller the matching cost — thus, the smaller the objective function. In measuring edge similarities, a total of R edge attribute types are considered. The binary term $C_{aibj}^{(2,r)}$ is the difference of the r'th edge attribute between edges G_{ab} and g_{ij}. Their sum integrates the overall compatibility between the edges G_{ab} and g_{ij}. The more compatible G_{ab} and g_{ij} is, the larger the sum — thus, the smaller the objective function. Compatibility between edges G_{ab} and g_{ij} is 0 if either of them does not exist ($G_{ab} = 0$ or $g_{ij} = 0$). In practice, many of the edges do not exist (i.e., sparse graph), thus, the binary compatibility can be computed efficiently. Since the above objective function contains a quadratic cost term, it is a quadratic assignment problem.

Context-Consistent Assignment formulation

As an extension of the above formulation, under the wide-baseline stereo context, we propose what we call the *Context-Consistent Assignment* (*CCA*) formulation. The key difference is that we replace the edge-compatibility term of Equation 4.4 with the Context-Consistency term (Equation 4.3).

For the unary term, we measure node similarity in terms of features' greyscale appearances. The nodes are affine features whose appearance attribute is encoded in the $\mathbf{p}$. Between nodes G_a and g_i, we define the C_{ai} as the symmetric Kullback-Leibler Divergence ($\mathbf{KL_{sym}}$ as defined by Equations 4.5, 4.6) between $\mathbf{p}_a$ and $\mathbf{p}_i$. Thus the closer $\mathbf{p}_a$ to $\mathbf{p}_i$ is, the lower the matching cost C_{ai}. The Kullback-Leibler Divergence is commonly used to measure distance between two distributions and $\mathbf{p}$ is such a function describing spatial distribution of pixel intensity.

$$\mathbf{KL}(\mathbf{p}_a, \mathbf{p}_i) = \int \mathbf{p}_a(x) \mathbf{log} \frac{\mathbf{p}_a(x)}{\mathbf{p}_i(x)} dx. \tag{4.5}$$

$$\mathbf{KL_{sym}}(\mathbf{p}_a, \mathbf{p}_i) = \frac{1}{2}(\mathbf{KL}(\mathbf{p}_a, \mathbf{p}_i) + \mathbf{KL}(\mathbf{p}_i, \mathbf{p}_a)). \tag{4.6}$$

With $\mathbf{p}$ being a low-dimensional descriptor, the C_{ai} both reflects an "approximate" closeness of appearances and can be computed efficiently.

In terms of the binary term, we extend the concept of edges to beyond node-to-node connections. In our formulation, we regard the connections between the central node and all its edge-connected neighbors as a single "virtual edge". Conventionally, for nodes comprised of feature blobs, measuring edge compatibility amounts to comparing neighboring features' geometric relationships. Embedding this relationship into scalar values has been investigated in works such as [18] and [39]. Demirci [39], for example, introduced such attributes as *Scale normalized distance*, *Relative orientation*, *Bearing* and *Scale ratio*. However, these measures are not invariant to viewpoint changes. Instead of comparing individual edges, we go one step further and compare neighbors of G_a against those of g_i all at once, by comparing the two "virtual edges". Between features G_a and g_i, the *Context-Consistency* (Equation 4.3) encodes exactly this similarity between two "virtual edges". Context Consistency redefines the way we measure the binary compatibilities. Neighboring nodes are now compared, instead of the *relationships* of neighboring nodes. This way, the binary term *is* invariant to the affine transformation due to the perspective change.

Based on the above discussion, matching two SFGs is formulated as follows. Given two SFGs G and g, suppose they have A and I nodes respectively, we find the match matrix M such that the following objective function is minimized:

$$E_{SFG}(M) = \sum_{a=1}^{A} \sum_{i=1}^{I} M_{ai}(\alpha C_{ai} - CC(G_a, g_i)), \tag{4.7}$$

where, $\sum_{i=1}^{I} M_{ai} \leqslant 1$, $\sum_{a=1}^{A} M_{ai} \leqslant 1$, and $M_{ai} \in \{0, 1\}$. Parameter α controls weights of the node attribute values. C_{ai} is the Kullback-Leibler Divergence between $\mathbf{p}_a$ and $\mathbf{p}_i$. $CC(G_a, g_i)$ is the *Context-Consistency* between G_a and g_i (as computed by Equation 4.3).

4.4.2 Overall algorithm

The SFG-matching problem is characterized by highly sparse graphs, high noise (missing nodes and edges) and a one-to-one constraint. We approach this assignment problem as inexact graph matching using the softassign algorithm by Gold and Rangarajan [61].

To deal with missing nodes/edges, a slack node is added to each graph, resulting in the augmented match matrix $\hat{M}$ with an extra row and column. By turning the discrete variables $\hat{M}_{ai}$ into continuous values between $[0, 1]$, the discrete optimization problem (Equation 4.7) is solved using a deterministic annealing method, by adjusting a control

parameter during the annealing procedure. The intermediate continuous values $\hat{M}_{ai}$ can be interpreted as the probability of node G_a matching to node g_i.

We can now summarize, in Algorithm 1, the overall *Context-Consistent Assignment* (*CCA)* algorithm. The CCA comprises two major steps. Step 1 is the preparation step and generates a pair of SFG models from the images (as described in Section 4.2). Step 2 performs matching of the SFG pair, i.e., minimizing $E_{SFG}(M)$ of Equation (4.7), as detailed below.

Algorithm 1: Context-Consistent Assignment

Data: stereo images I_{left}, I_{right}
begin SFG modeling
  (optional, Node detection,) Node description by the *Original Node* principle
  Edge Insertion by the *Invariant Edge* principle
  Result: I_{left}− >graph G, I_{right}− >graph g.
end
begin SFG matching: $G \longleftrightarrow g$
  Initialize: β to β_0, $\hat{M}_{ai}$ to $KL(\mathbf{p}_a, \mathbf{p}_i)$, $\mathbf{F}$ to $NULL$, Set of Correspondences SC to $\varnothing$
  repeat A:
    repeat B:
      $Q_{ai} \longleftarrow -\frac{\partial E_{arg}}{\partial M_{ai}} = -\alpha C_{ai} + CC(G_a, g_i)$
      $M^0_{ai} \longleftarrow \exp(\beta Q_{ai})$
      repeat C: Normalizing $\hat{M}$
        Update $\hat{M}$ by row-normalization:
        $\hat{M}^1_{ai} \longleftarrow (\hat{M}^0_{ai})/(\sum_{i=1}^{I+1} \hat{M}^0_{ai})$
        Update $\hat{M}$ by column-normalization:
        $\hat{M}^0_{ai} \longleftarrow (\hat{M}^1_{ai})/(\sum_{a=1}^{A+1} \hat{M}^1_{ai})$
      until $\hat{M}$ *converges, or # of iterations* $> N_1$
    until M *converges, or # of iterations* $> N_0$
    Update SC
    if *Enough correspondences in SC* **then**
      ($\mathbf{F}$,$SC_{inliers}$) $\longleftarrow$ RANSAC(SC)
    $\beta \longleftarrow \beta_{\triangle}\beta$
  until $\beta \geq \beta_f$, *or* $\mathbf{F} \neq NULL$
  Result: Fundamental Matrix $\mathbf{F}$, $SC_{inliers}$.
end

SFG matching is converted into solving a succession of assignment problems by repeating Loop **A** at an increasingly large annealing parameter β. This outer loop starts from initial value $\hat{M}^0$ that is initialized with the Kullback-Leibler Divergence. Within the Loop **A**, each assignment problem (Loop **B**) is formulated as maximizing

$$\sum_{a=1}^{A}\sum_{i=1}^{I} Q_{ai}M_{ai}, \tag{4.8}$$

where

$$Q_{ai} = -\frac{\partial E_{SFG}}{\partial M_{ai}} = -\alpha C_{ai} + CC(G_a, g_i), \tag{4.9}$$

substituting the $CC(G_a, g_i)$ (Equation 4.3) into Equation (4.9), our final Q_{ai} is

$$Q_{ai} = -\alpha C_{ai} + \sum_{G_b \in N_a} [\max_{g_j \in N_i}(M_{bj} S(G_b^{NT}, g_j))]. \tag{4.10}$$

Upon convergence of Loop **B**, the continuation method returns the corresponding globally optimal doubly stochastic matrix M ([61]) for the current value of parameter β. At the end of each Loop **A**, we update our Set of Correspondences by scanning the resulting M and adding the pairs (G_c,g_k) whose matching probability M_{ck} is above a certain threshold (e.g., 0.99). By doing so, we are gradually incrementing our confirmed matches and it provide a mechanism for early termination once wide-baseline matching is established. We declare the matching successful once we have recovered the Fundamental Matrix and obtained a sufficient number of consistent correspondences.

The above is the complete procedure of the *CCA* algorithm. In our experimental validation, we skip the outlier detection component for the sake of fair comparison. This way, we compare both the inliers and the outliers obtained by competing methods.

4.5 Performance evaluation

4.5.1 Data sets

While the proposed method is applicable to general scenes comprised of piece-wise planar surfaces, here we choose to test extensively on 3-D urban scenes.

We experimented on a set of 10 urban image pairs, some of which we also tested on

their sub-sampled and noise-contaminated versions. This data set was compiled from two sources. Some are publicly available image pairs that are commonly used for feature matching validations, and others are acquired by ourselves with a digital camera — in order to create a variety of challenging cases. Figure 4.7 contains the standard test images we used. Among them, (a), (b), and (d) were retrieved from the Oxford Visual Geometry Group's website [1], and (c) was used in the *ICCV 2005 'Where Am I ?' Computer Vision Contest*.

Figures 4.5 and 4.8 contain the more challenging image sets, which are all acquired by ourselves with the exception of the *Valbonne* (also by Oxford). The challenging images typically have large changes in viewpoint angles, small overlap due to occlusions and scale changes, and some contain repetitive patterns. The images range in size from 512×384 to 1024×768 - some of which were obtained by sub-sampling originals. To test how little information is required for a successful matching, we also tested on low-resolution versions of some pairs by sub-sampling images to less than 200 pixels in length and width. Besides, we added Gaussian i.i.d. noise onto images and compared performance of competing matching methods. The results can be interpreted as the algorithms' robustness to image noise. The results are conditioned on the data set, but the range of challenging pairs shown should be suggestive of the broad effectiveness of the *CCA* algorithm.

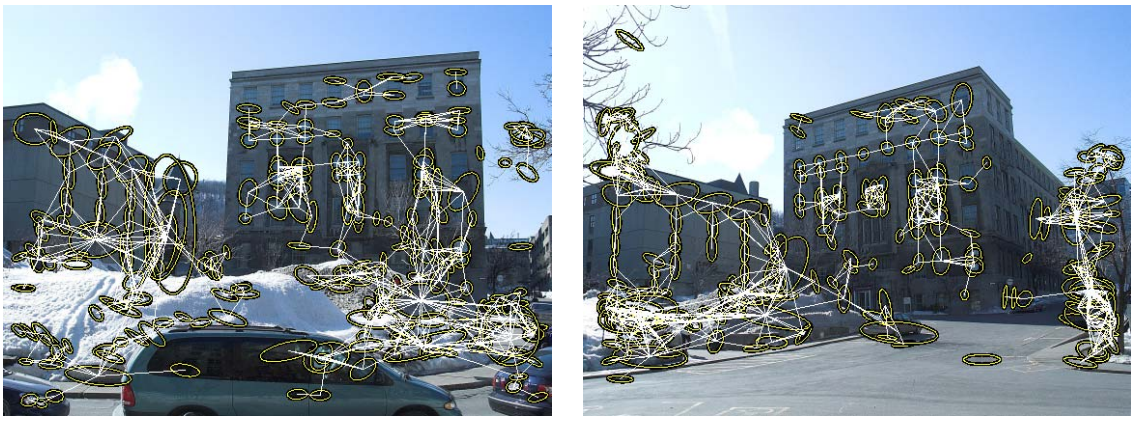
An example SFG model

With the *J-Scene* as an example, we show the constructed SFG models overlayed on the images (Figure 4.4). If we observe the two SFGs carefully, consistency of their nodes varies across different regions. Most of the nodes on the left-side wall are consistent in both views, whereas most of those on the up-front wall are not. Nodes of the lower parts of the images are from totally different objects and serve as distractors for the matching task. With so many inconsistencies and noise, this pair of SFGs is challenging for any regular graph matching algorithm.

4.5.2 Evaluation criteria

Objective evaluation metrics, like the *recall vs. (1-precision)* used in [115], would need ground-truth about correctness of each match — Mikolajczyk and Schmid automated validation against ground-truth by using planar scenes that differ by known homographies.

[1] `http://www.robots.ox.ac.uk/~vgg/data1.html`.



(a) Left (213 Nodes) (b) Right (226 Nodes)

Fig. 4.4 *SFG* models overlayed on the *J-Scene*.

Our experiments, however, use complex 3-D urban scenes with no readily available ground truth. Depending on the difficulty of matching particular scenes, we evaluate the methods with two similar criteria that are conducive to objective observation — *correct ratio vs. (1-precision)* and *outlier percentage*. Both these criteria need ground truth information about whether or not a match is correct, i.e., outlier classification. The *correct ratio vs. (1-precision)* is based on outlier judgement by human verification. If a pair of features are projections of different surface points, we call this match outlier. The later criterion of *outlier percentage* is based on automated outlier classification using the global epipolar constraint. For one pair of matching points, if the distance between one point and the corresponding epipolar line is greater than a certain threshold, the match is declared outlier.

Correct ratio vs. (1-precision) The *correct ratio vs. (1-precision)* is essentially a modified version of the *recall vs. (1-precision)*. The *(1-precision)* is the number of false matches divided by the total match number.

$$1 - precision = \frac{\#false\_matches}{\#correct\_matches + \#false\_matches} \tag{4.11}$$

And we define the *correct ratio* as the ratio of correct matches to a chosen constant C.

$$correct\_ratio = \frac{\#correct\_matches}{C} \tag{4.12}$$

In Equation (4.12), to imitate the *recall*, we use the number of correct matches as the numerator of the fraction. But we have to use a constant C as the denominator, to deal with the fact that the number of overall correct matches (used by the *recall*) is unknown. Nevertheless, the *correct ratio* differs from the *recall* by a constant multiplier. Thus, relative merits of compared methods are the same for both metrics.

The idea for *correct ratio vs. (1-precision)* is to examine the top C matches produced by competing methods and compare their matching accuracy. The quantity C is chosen according to two considerations. C is not too large so that a manual verification of all C individual matches is feasible. Also, matching accuracies of the top C matches can serve as a convincing indicator of competing methods. We choose the C empirically and it is scene-specific. We make sure that at least one method has a high probability of generating outliers after the C^{th} match.

For the *CCA* method, every time new matches are generated by the iterations, we obtain a pair of *correct ratio* and (*1-precision*) values. For the local method (the comparison method described below), each threshold t on the feature descriptor-distance corresponds to one pair of *correct ratio* and (*1-precision*). We generate the curves by following the iterations (for *CCA*) or varying the threshold t (for the local method).

Outlier percentage In other cases, where the images are less challenging and a large number of matches can be easily established (e.g., Figure 4.7), we will not manually label correctness of the matches. Instead, we compare the methods in terms of the numbers of total matches and the percentages of outliers. When counting the number of outliers, we use the epipolar constraint and RANSAC [57] to check global consistency.

4.6 Experimental results

This section presents our experimental results. To illustrate the overall performance, we thoroughly tested our method in two respects. First, we compare CCA feature matching against nearest-neighbor matching of features' SIFT descriptors. This is a side-by-

side evaluation of the two paradigms — context-driven and local-based. At present, the nearest-neighbor matching is the only widely-used method to match features without strong assumptions about the scene structure. Our method also mildly assumes that the scene surfaces are locally-planar, or, equivalently, piecewise-smooth. Use of the SIFT descriptor is based on the recent finding that it leads all descriptors in matching performance [115]. Second, we analyze the contributions of two key components of the CCA algorithm.

Section 4.6.1 deals with the first respect, where we compare the CCA against the nearest-neighbor matching method on a range of images. Under a variety of cases, we measure its ability to establish a sufficient number of matches while maintaining low false-positive rates. The second respect is examined in Section 4.6.2, where we examine the effectiveness of using the *Invariant Edge* for graphical image modeling and using the *Neighborhood Transform* for graph matching.

Except as otherwise noted, both methods match the common set of MSER features [111]. MSERs suit a variety of scenes and were found to be the best feature for the nearest-neighbor matching method [116]. This claim was also echoed by results of our recent experiments [49]. However, we need to stress that the proposed method provides a general framework for matching features irrespective of specific feature types.

The procedure for the CCA follows Algorithm 1, with the termination criterion modified for fairness of comparison. It stops when a prescribed number of total matches is reached (as described in Section 4.5.2). For the nearest-neighbor matching, we detect MSERs (or directly use the given features, if provided), normalize them to fixed-sized circular regions, and extract their SIFT descriptors (using the authors' binaries [116]). We obtain the final matches by nearest-neighbor classification of their SIFT descriptors. For each method, we tuned the parameters manually and kept them fixed for all experiments.

4.6.1 Comparison with the local method

In comparing with the nearest-neighbor matching, we test on images of varying degrees of difficulty. Our first experiment is concerned with images that contain very weak textural information. In this case, we test on hand-picked structures, so that we can better gauge the matching methods themselves free from the factor of feature detection. The subsequent two comparisons work on matching features of standard scenes and more challenging cases respectively.

Matching weakly textured images

Figure 4.5 shows the weakly textured scene (*Wall*) and the blob features we manually selected. Notice that we selected features based on distinctiveness to the eye, rather than intentionally choosing the patches that have correspondences. Running both matching algorithms, we obtain the *correct ratio vs. (1-precision)* curve as shown in Figure 4.6. In the ideal case for this type of plot, values would lie on top of the y-axis. Here, the curves are ragged because we used a small set of samples (top 31 matches were used); also, they are sampled densely — sampling every time a new match is produced. The figure reveals comparative merits of the methods. For the majority of matches obtained by the local method, the *(1-precision)* values vary between 0.4 and 0.5. For the CCA, on the other hand, these values are very low for the top matches (0 for the first few matches). Outlier numbers start to increase after the match number becomes larger. This is desirable since users can have confidence that their top matches by CCA are reasonably reliable.

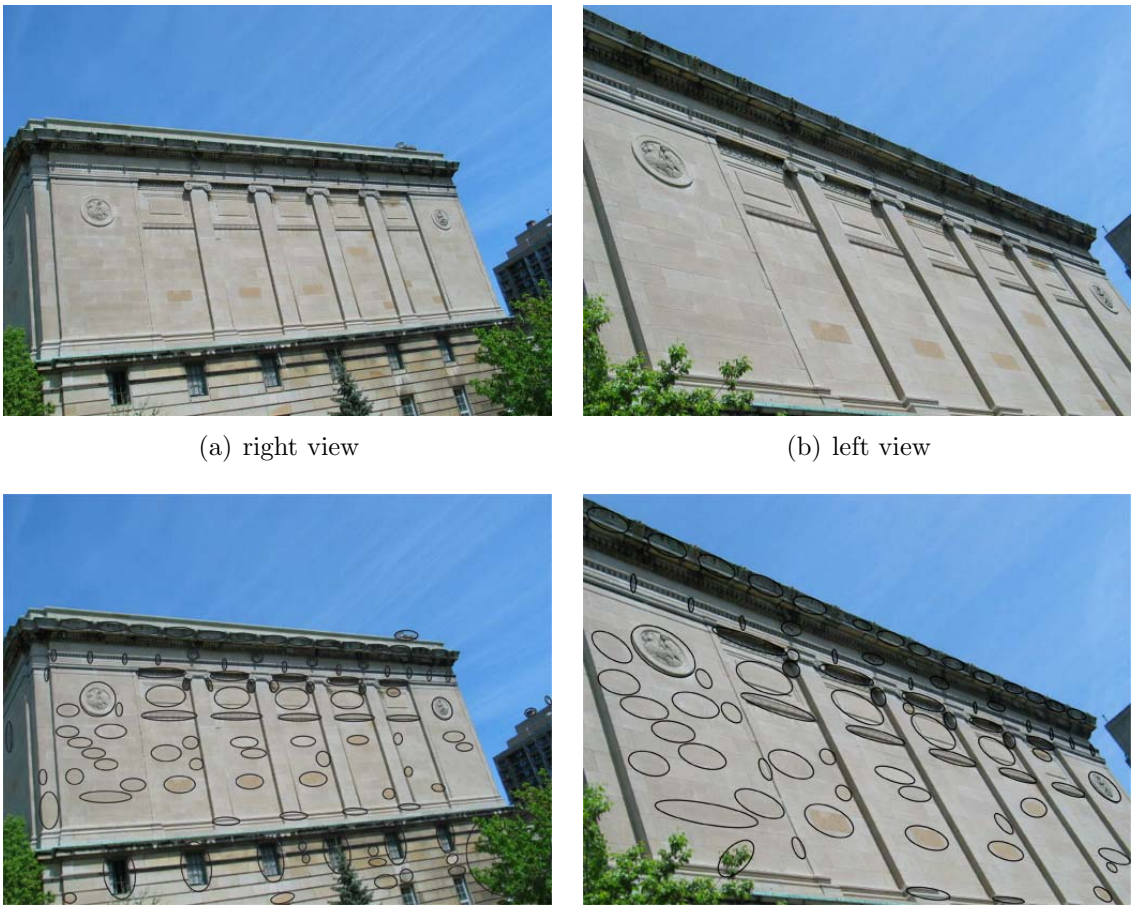
Matching standard images

Figure 4.7 shows the images used for standard scenes. For each image pair, using the *Outlier percentage* criterion, we compare the numbers of total matches and outliers, as well as outlier percentages (Table 4.1). Images of the *House*, *Wadham*, and *Corridor* were taken under ideal conditions and contain many distinct textures, hence, both methods perform well. They obtain large numbers of matches with few outliers, which generally account for less than 10% of the total matches. The *House Array* is slightly more difficult due to changes in viewpoint and scale, and occlusions by the tree. Both perform satisfactorily, although with fewer matches. For images of this category, CCA is comparable to the local method.

Matching difficult images

For more challenging cases, we test three scenarios: (1) images of challenging scenes, (2) low-resolution images, (3) images with additive Gaussian noise.

Challenging scenes Figure 4.8 shows the images of challenging scenes. Each of these scenes contains one or several of the following factors that make the matching difficult:



(a) right view (b) left view

(c) right with features (d) left with features

Fig. 4.5 Image set *Wall* and hand-picked features. (a) and (b) are original images, while (c) and (d) contains our manually outlined features.

repetitive patterns, small image overlap, large viewpoint change, large scale change, occlusions by tree branches, etc. These factors, however, are commonly encountered in our everyday urban environments. The results are shown in Figures 4.9, 4.10, and 4.11. All the results indicate that the CCA is advantageous at handling these difficult scenes. Perhaps the most drastic contrast is in the case of the *Burnside* (Figure 4.9(a)), where the local method completely fails and the CCA works nearly perfectly. The repetitive patterns greatly confuse similarity judgement of the local method, while the CCA is able to satisfactorily reason about similarity by making full use of the neighborhood contexts.

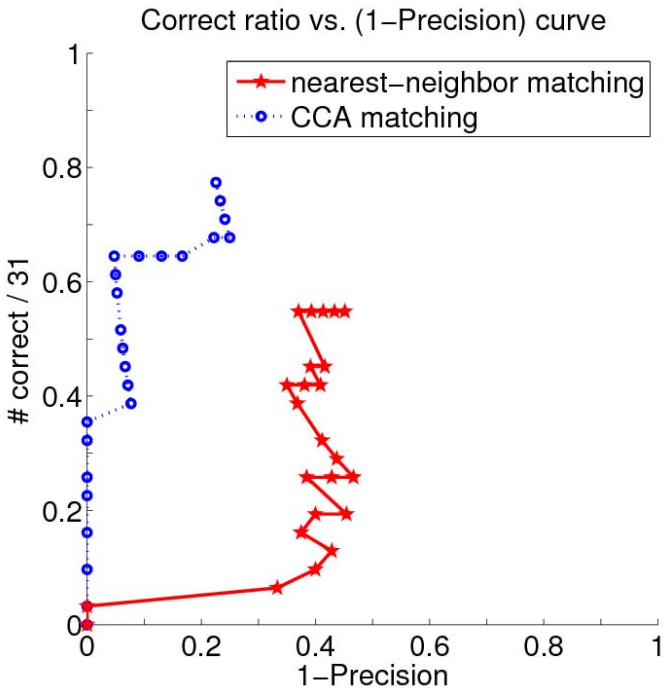


Fig. 4.6 Feature matching result on the *Wall*. The *correct ratio* vs. (*1-precision*) graph is generated according to the top 31 matches obtained by both methods (31 shown as the denominator of the y-axis label).

(a) *House* (b) *Wadham* (c) *House Array* (d) *Corridor*

Fig. 4.7 Image set of standard scenes. These include both outdoor and indoor scenes.

	Matching Methods	*nearest-neighbor*	*CCA*
House	#outliers / #total	**5 / 69**	9 / 76
	Outlier %	**7.2%**	11.8%
Wadham	#outliers / #total	**14 / 155**	17 / 184
	Outlier %	**9.0%**	9.2%
House Array	#outliers / #total	1 / 28	**0 / 32**
	Outlier %	3.6%	**0%**
Corridor	#outliers / #total	12 / 59	**11 / 59**
	Outlier %	20.3%	**18.6%**

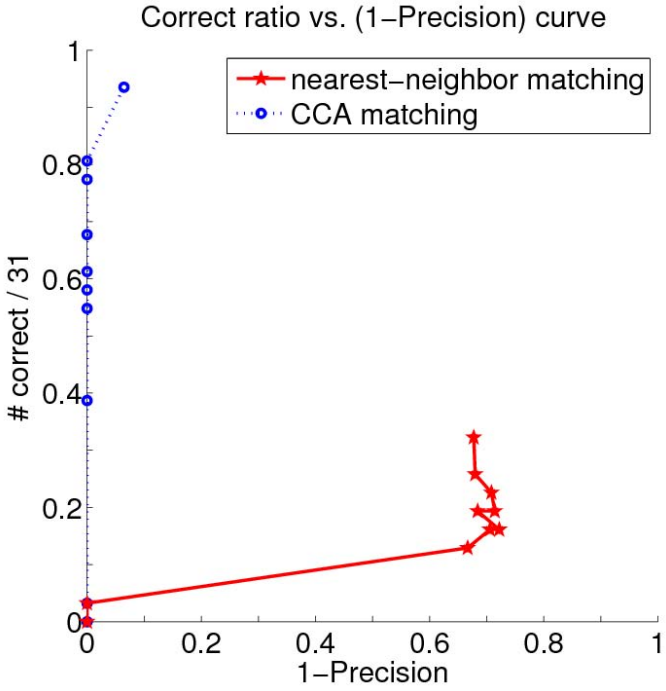
Table 4.1 Feature matching comparison on standard scenes. We compare *CCA* matching against the nearest-neighbor matching side-by-side. Matching error is expressed as outlier percentage (%). The winner is highlighted in **Bold**.

(a) *Burnside* (b) *J-Scene*

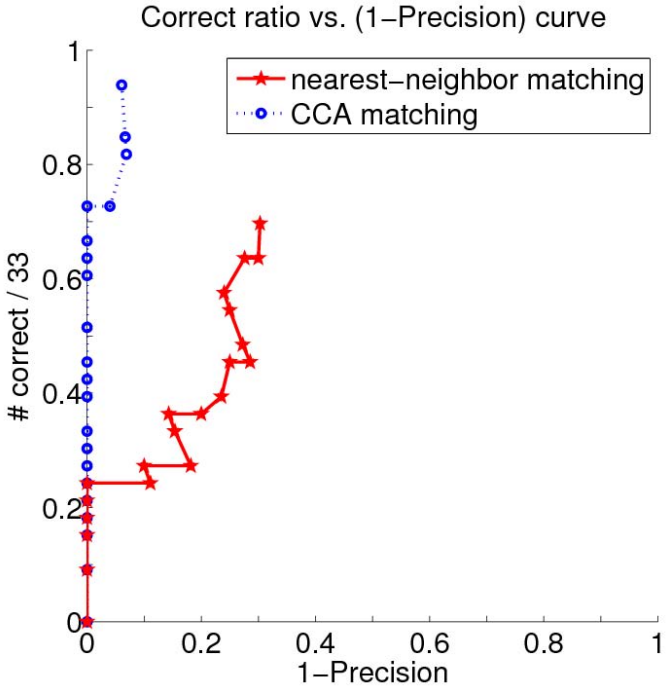
(c) *Valbonne*

(d) *College MTL* (e) *Schulich North*

Fig. 4.8 Image set of challenging scenes.

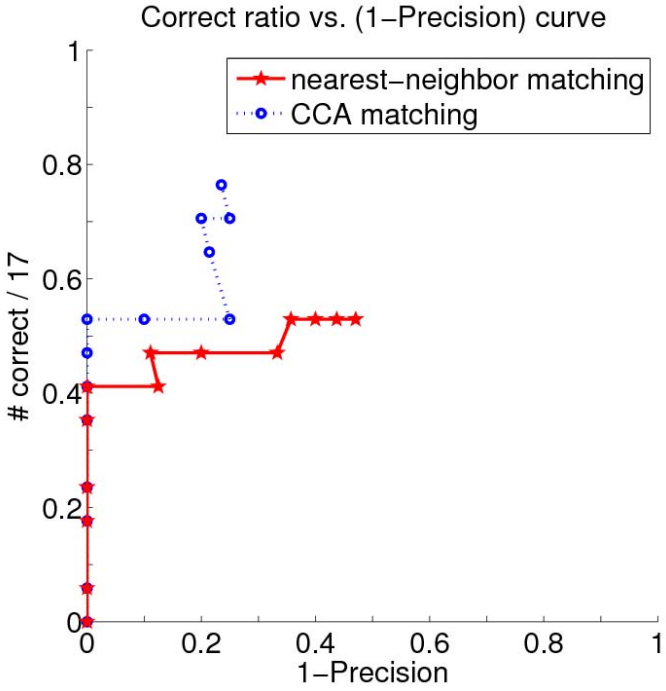


(a) *Burnside*

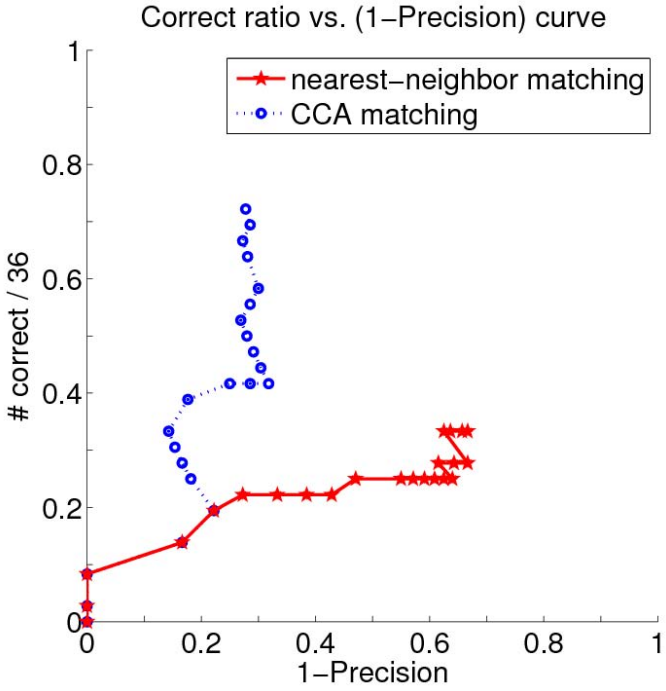


(b) *J-Scene*

Fig. 4.9 Feature matching results for challenging scenes (1).

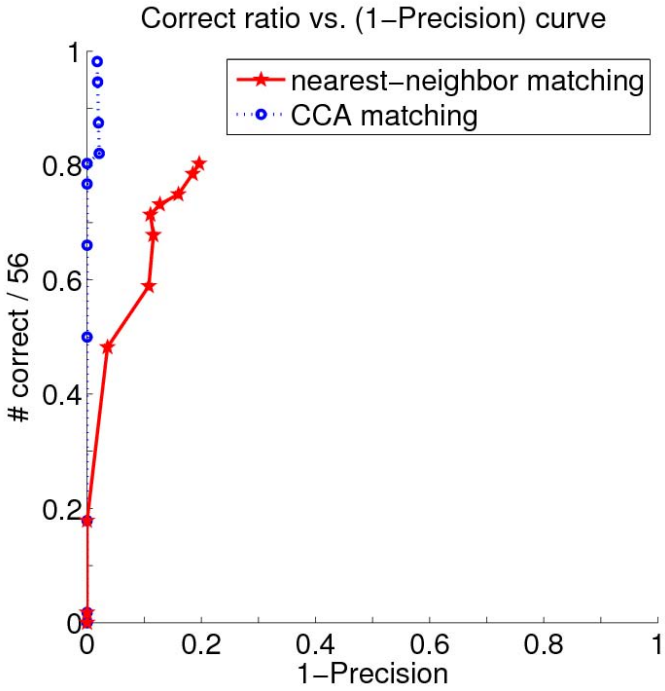


(a) *Valbonne*



(b) *College MTL*

Fig. 4.10 Feature matching results for challenging scenes (2).



(a) *Schulich North*

Fig. 4.11 Feature matching results for challenging scenes (3).

Low resolution images To obtain the low resolution image sets, we sub-sampled the original images, sized 768×576 (*House*) and 568×426 (*J-Scene*), to 192×144 and 220×165 respectively. The matching results are shown in Figure 4.12. As expected, both CCA and local method see a clear decrease in *(1-precision)* after obtaining a dozen or so high quality matches. By comparing the curves for the same scene (*J-Scene*) on different resolutions (Figure 4.12(b) and Figure 4.9(b)), we can appreciate this decline as a result of sub-sampling. While universally affecting all methods, low-resolution inputs have less of a negative effect on CCA than on the local method. Thus, CCA performs more robustly under this challenging scenario.

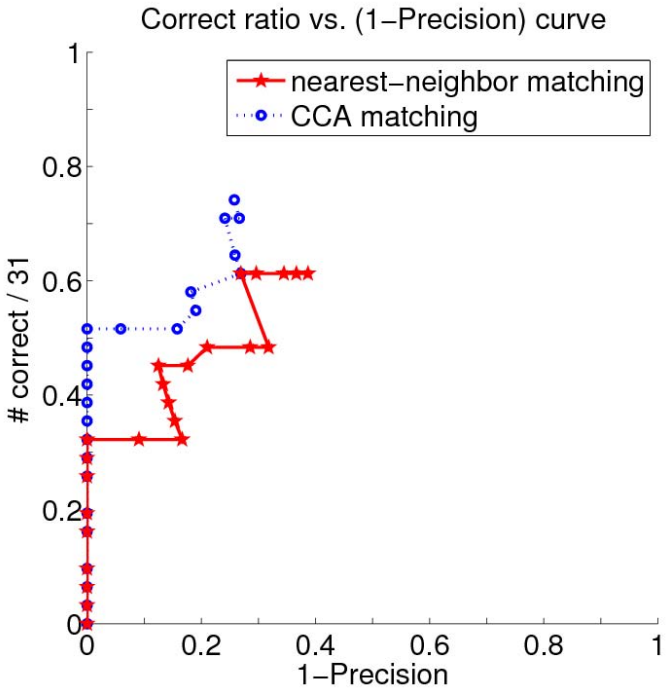
Noisy images To test on noisy images, we added Gaussian white noise with a variance of 0.01 to the originals (*House* and *J-Scene*), and repeated the same protocols. The results are shown in Figure 4.13. In both tests, notice the consistency of CCA in producing both more matches and fewer outliers than the local method. Again, CCA has a clear advantage in matching images contaminated by additive Gaussian noise.

Approximate run time

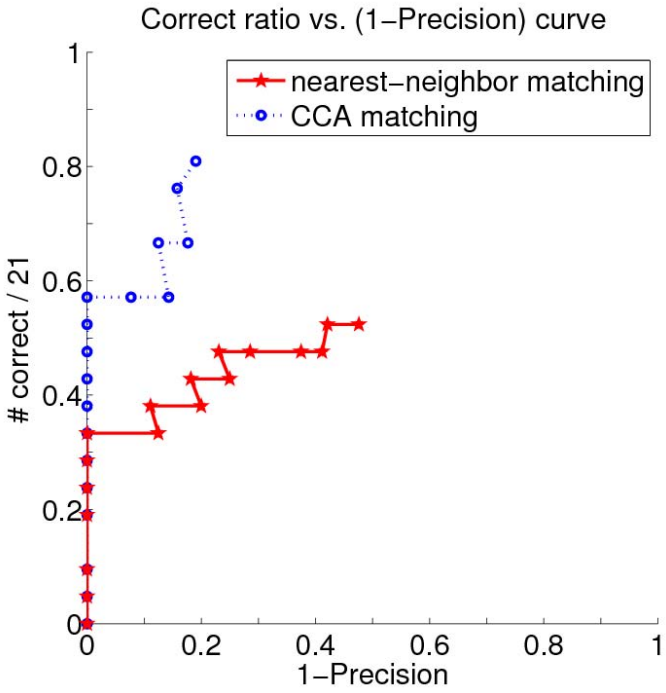
Currently, no attempt was made in code optimization and possible parallelization of the deterministic annealing part of the algorithm. The run-time for the CCA varies with varying numbers of SFG nodes. Running a single core on a 1.8GHz Intel Core Duo laptop processor, run time ranged from 20 seconds to 2 minutes for small (around 100 nodes) to normal-sized (200 to 400 nodes) images. Convergence on the noisy images took about 30 minutes since each SFG has 1500 to 2000 nodes. The local methods, on the other hand, were faster and matched in 5 to 15 seconds. These are approximate numbers and can only serve as an indication of relative computational expense.

4.6.2 Evaluating key components

In the second part of the experiments we test the contribution of the two key contributions: the concept of Invariant Edge for graphical modeling of an image, and the Neighborhood Transform for the matching algorithm. For each case, we compiled one version of the CCA that did not use that component and compared it to the standard CCA procedure. Both tests were performed on the image sets *Burnside* and *Schulich North*.

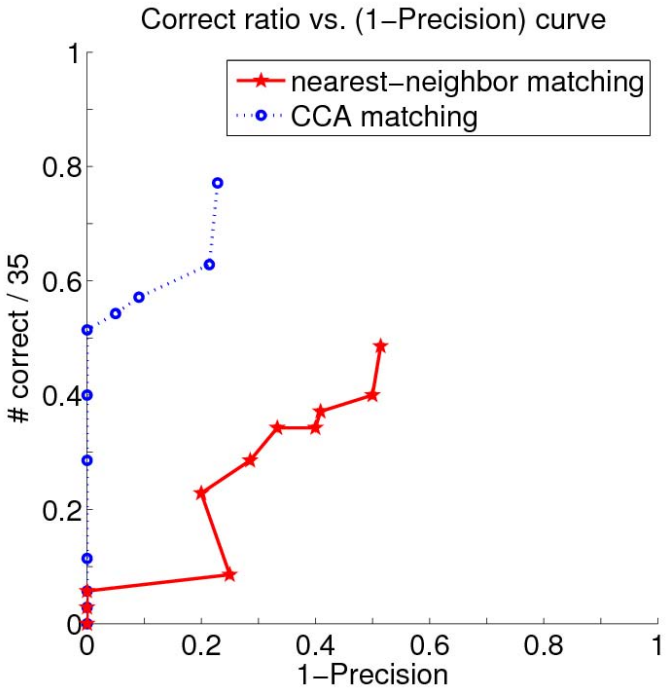


(a) *House*

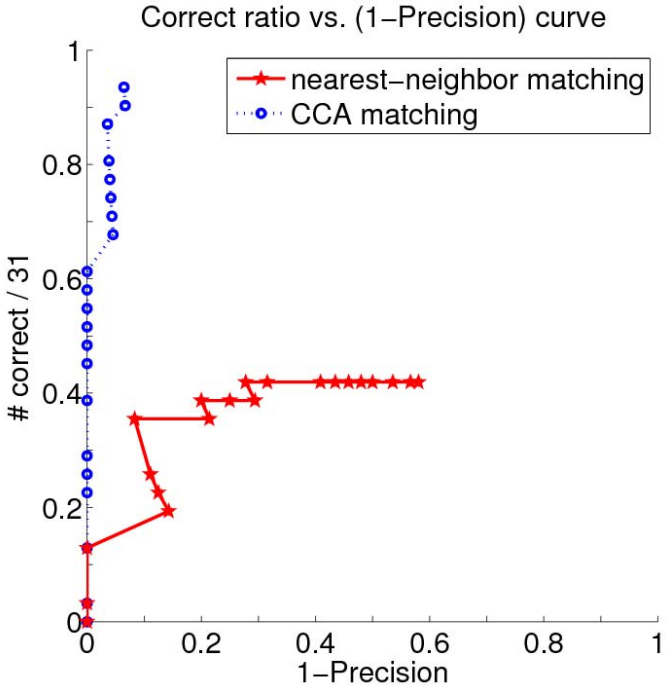


(b) *J-Scene*

Fig. 4.12 Feature matching results for low-resolution images.

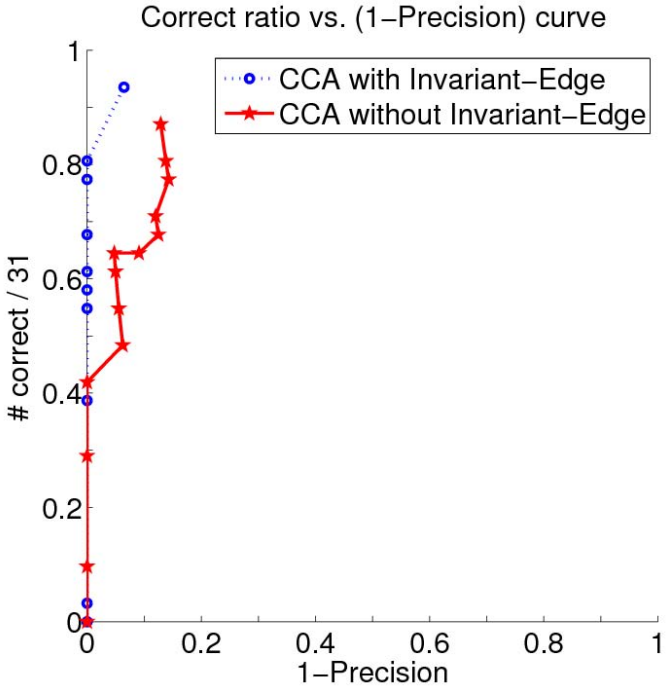


(a) *House*

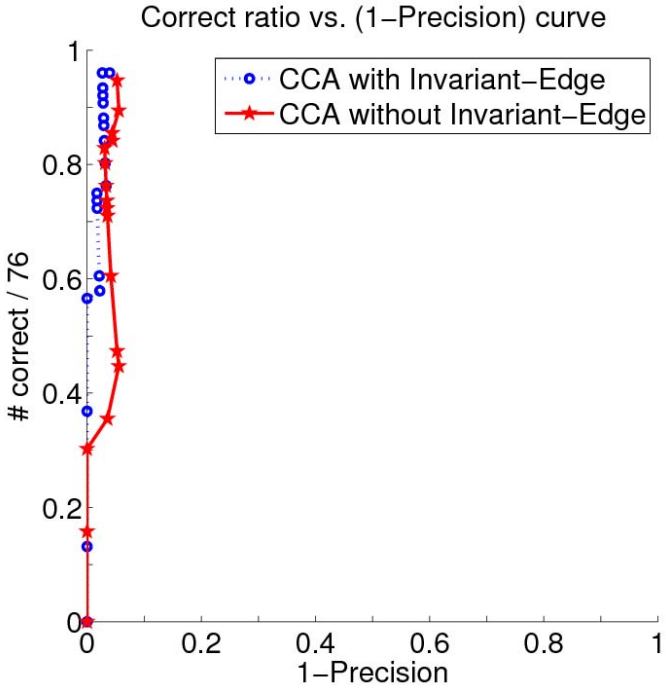


(b) *J-Scene*

Fig. 4.13 Feature matching results for noisy images.



(a) *Burnside*



(b) *Schulich North*

Fig. 4.14 Evaluating contribution of the Invariant-Edge.

Contribution of the Invariant Edge

The opposite of enforcing Invariant Edge is to construct a feature's neighborhood based on Euclidean distance in the image - exactly as illustrated by the Figure 4.1(b). To simulate that situation, we constructed a feature's neighborhood by multiplying its scale by the same factor as the one that was used for the Invariant Edge (Figure 4.1(c)). From the result (Figure 4.14), we can see there is slight but consistent improvement by using the Invariant-Edge for the *SFG* modeling.

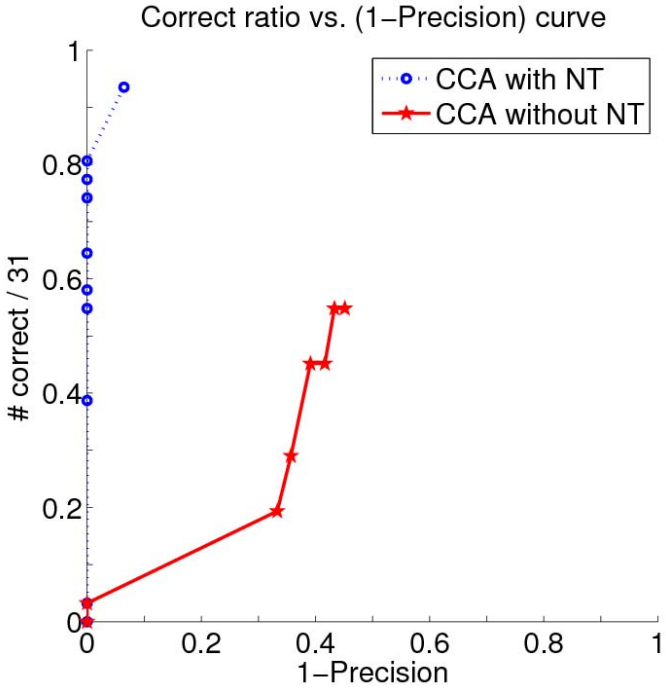
Contribution of the Neighborhood Transform

The Neighborhood Transform is used in comparing similarity between nodes' edge-connected neighbors. Its novelty lies in geometrically transforming the corresponding neighbors before measuring their similarity. We simulated a comparison case by not doing any geometric transformation and comparing the corresponding neighbors directly. The results are shown in Figure 4.15. We can see that the Neighborhood Transform significantly improves the method's robustness.

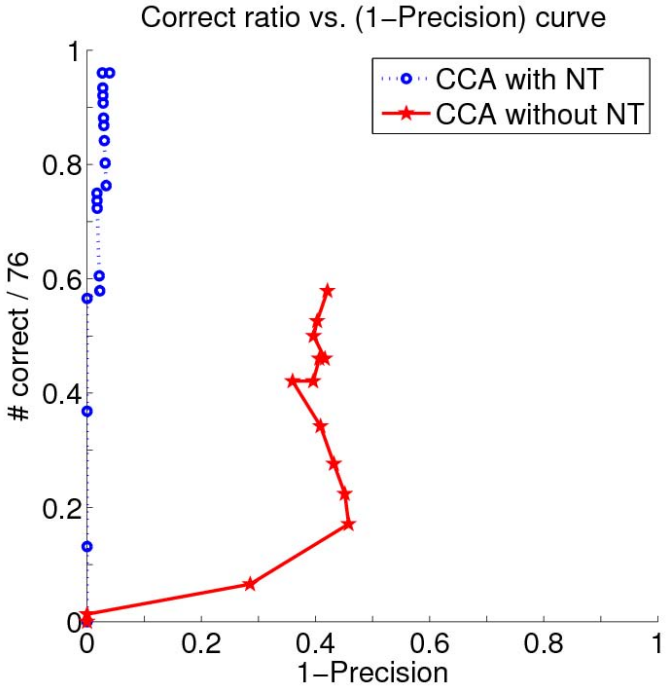
4.7 Conclusions

We have developed a model for image representation and a framework for robustly matching wide-baseline stereo images. The SFG representation encodes intrinsic proximity relationships in 3D scenes; thus, it is able to provide a consistent neighborhood for a context-aware stereo matching algorithm. This model can be useful for other applications where features' geometric layout needs to be exploited, e.g., in pattern recognition, image and video retrieval, etc.

The CCA algorithm performs more robustly and considerably better in cases where ambiguities exist if one distinguishes features solely based on their local appearances. This performance boost is the direct result of the matching strategy. Local descriptors, on which the local methods rely exclusively, are designed to be distinctive and thus are very sensitive to local appearance discrepancies. The CCA algorithm, on the other hand, considers spatial layout of local feature clusters and models their commutation with viewpoint changes. Although our method needs extra computation in optimizing graph matching, it is worthwhile when accuracy and robustness are more important for the required tasks. The key



(a) *Burnside*



(b) *Schulich North*

Fig. 4.15 Evaluating contribution of the Neighborhood Transform (NT).

to its success lies in our use of the Neighborhood Transform to check features' Context Consistency. Further, the hypothesize-match procedure relates the relevant neighbors in the absence of correspondence. Overall, by relying on both local appearance and surrounding neighborhood context, the CCA method can make a more informed decision about correspondence.

The framework of Context-Consistent Assignment can also be used for features other than blobs, such as edges, curves, etc., with appropriate adaptation of the Neighborhood Transform. This idea for measuring Context-Consistency can also be used in other applications where image context needs to be exploited.

Chapter 5

Better correspondence by registration

With the feature correspondences established, e.g., by using the method proposed in Chapter 4, we now consider how to use them to effectively compute the fundamental matrix.

The process of fundamental matrix estimation using feature correspondences is affected by error in two main ways. First, not all matches between features reflect real correspondences between objects in the 3D scene, and it is necessary to filter out the false matches before attempting to estimate the fundamental matrix. Erroneous matches (outliers) are singled out using robust estimation methods such as M-estimator [173] and random sampling algorithms [57][174]. These methods, however, discard the valuable information about correspondence quality contained in the similarity score between the two points, in effect assuming that all matched pairs have an equal chance of being a mismatch. Some recent work [172][27] mitigated this shortcoming by considering this (normally discarded) similarity quality and achieved improved results. The second issue is how to accurately recover the epipolar geometry assuming we are working with inlying feature matches. In practical applications, errors in the position of the matched point centriods are unavoidable. A feature's geometric properties, such as location and shape, are determined by its appearance in a single image. Under wide-baseline conditions, these properties are highly volatile due to factors such as image noise, occlusion, image quantization error, etc. Hence, even correctly corresponding features cannot always be precisely related by the ground truth two-view geometry. This problem is echoed by the recent work of Haja et al. [64]. They showed that different feature detectors exhibit significantly different localization accuracy in position and feature shape. They have also found this positional accuracy is proportional

to feature scale, which agrees with our intuition. Various numerical schemes [81][24] have been proposed for high accuracy fundamental matrix computation, under the assumption that the locational errors of each feature are Gaussian. Also, Georgel et al. [59] implicitly corrected this error by introducing a photometric cost to their pose estimation framework.

We present a method to improve both the robustness and the accuracy of fundamental matrix estimation by advancing both of the above areas. We achieve this by an intensity based alignment of the local patches around each matched feature point. For each of the putative matches, we locally adjust the position and shape of the feature in one image according to the appearance of its counterpart in the other image. Consequently, we will have a better characterization of the feature similarity and the features are better localized towards the image of a common 3D structure. This improved similarity and localization will enable a more effective robust outlier rejection. At the same time, we obtain a more accurate fundamental matrix by directly correcting the source of the inaccuracy: feature location errors.

The remainder of this chapter is organized as follows. Section 5.1 discusses related work on fundamental matrix estimation. After describing our registration-based refinement in Section 5.2.1, we layout the procedure for the improved fundamental matrix estimation in Section 5.2.2. The effectiveness of the proposed correspondence refinement is validated in Section 5.3. This chapter concludes with a discussion of other possible applications and future work.

5.1 Related work

Let us repeat the formulation of the epipolar constraint between two corresponding points using the fundamental matrix (previously introduced as Equation 1.6 in Section 1.2.3).

$$\mathbf{x}'^{\top}\mathbf{F}\mathbf{x} = 0, \tag{5.1}$$

where $\mathbf{x} = (a, b, 1)$, $\mathbf{x}' = (a', b', 1)$ are homogeneous representations of the point image-coordinates, and $\mathbf{F}$ is a 3×3 matrix of rank-2. Most methods for fundamental matrix estimation (abbreviated as $\mathbf{F}$-estimation) proceed in two stages, estimating an initial $\mathbf{F}$ using a robust method to filter out erroneous matches (Robust $\mathbf{F}$-estimation), and then re-estimating $\mathbf{F}$ precisely using the matches deemed correct (Precise $\mathbf{F}$-estimation).

5.1.1 Robust F-estimation

Robust methods are designed to deal with estimation problems where a portion of the data is completely erroneous. Representative works are M-Estimators [173], least median of squares [144], and random sampling approaches (e.g., [57][174]). However, each of these operates under the assumption that each input datum is equally likely to be erroneous. In the image matching problem discussed here, additional information is available to estimate the quality of the matches being used to estimate **F**.

The PROSAC algorithm by Chum and Matas [27] and the Guided-MLESAC algorithm of Tordoff and Murray [172] introduced some domain-specific priors into the random sampling scheme. That is, they incorporated information about the quality of the point matches into the random sampling process. These schemes, which we call *prior-influenced random sampling*, demonstrate significant gain in computational efficiency and robustness. PROSAC is of particular interest because of its mild *not-worse-than-random* assumption and its computational efficiency. The method draws samples on progressively larger subsets consisting of top-ranked correspondences. Their ranking is based on such similarity measures as Euclidean distance of discrete cosine transform (DCT) coefficients [127] or ratio of SIFT distances [102]. The confidence in the solution is guaranteed by a RANSAC-equivalent termination criterion.

5.1.2 Precise F-estimation

Even once a set of correct matches has been selected, the equations implied by Equation (5.1) can not be satisfied exactly due to the noise in the point positions. Precise **F**-estimation is often cast as a minimization problem, minimizing either an algebraic residual or some geometric distance. From the algebraic perspective, it can be solved either linearly by the Orthogonal Least Squares Regression algorithm [173], or by nonlinear iterative estimation methods [15]. When approached as a geometrical minimization problem, the objective function bears some meaningful geometrical distance. It can be either reprojection errors of corresponding points (Golden method [66]), or the perpendicular geometric distances of points to a certain conic (Sampson distance [186]), or the distance of a point to its epipolar line [106].

5.2 Our approach

Robust **F**-estimation methods rely on the possibility of making a clear distinction between outliers and inliers. However, the errors in feature alignment have a tendency to blur this distinction. As these errors increase, all components of the system degrade. First, the similarity scores used to rank points are less reliable, which makes the *prior-influenced random sampling* less effective. Second, the initial **F** estimated by the robust methods are of poorer quality, which leads to more difficulty in distinguishing inliers from outliers. In fact, the inlier/outlier categorization is inherently less reliable, as the errors on inlying matches tend to be larger. Finally the resulting precisely estimated **F** is less accurate, as its accuracy is ultimately determined by the accuracy of the point matches used as input to the minimization algorithm.

We propose to improve point match alignment by local registration. This will produce two immediate results. The first is a more accurate localization of the matched points. This effectively reduces the noise level of the points. The second is a better similarity measure of the matches because of this reduction in position and shape discrepancy.

Robust outlier rejection will benefit from these results. The improved similarity provides a more reliable prior for the *prior-influenced random sampling* schemes. In the meantime, the reduced noise in position will give rise to a stronger positive vote if a correct model is being tested by a random sampling method. Thus, one would expect the inliers to be detected more efficiently and with better success rate. Finally, precise **F**-estimation will also benefit from the improved feature localization.

5.2.1 Localization refinement by registration

Most feature points used in image matching applications achieve a level of transformation invariance by incorporating some transformation information into their description. We parameterize an elliptical feature region, i, by a centroid, $\boldsymbol{x}_{c_i}(x_i, y_i)$, as well as three parameters, a_i, b_i, c_i, describing the location, shape and scale of the ellipse. A correspondence between a pair of points then implies that these elliptical regions correspond to each other. An affine transformation $\boldsymbol{\phi}$ which matches one ellipse onto the other can be computed by determining three or more equivalent points on each ellipse and solving for the affine transformation parameters that map the points from one ellipse to the other.

For each correspondence, our registration-based refinement tries to find the optimal

affine transform $\boldsymbol{\phi_{opt}}$ based on pair-wise appearances and to re-align the corresponding features accordingly. This registration is implemented in two steps, $\boldsymbol{\phi}$-*initialization* and $\boldsymbol{\phi}$-*optimization*.

$\boldsymbol{\phi}$-initialization

This step establishes an initial affine transform $\boldsymbol{\phi_{init}}$ with which to start registration, based on an approximate patch alignment. With each feature ellipse being defined by five parameters $(x_i, y_i, a_i, b_i, c_i)$, one cannot infer a six parameter affine transform $\boldsymbol{\phi}$ between a pair of features without resorting to the use of more information such as image appearances. By mapping bounding rectangles of the two ellipses, we establish an approximate transform $\boldsymbol{\phi_{init}}$ and leave the accurate $\boldsymbol{\phi}$-estimation to the optimization step.

Specifically, the ellipse for each point satisfies the quadratic form

$$[\boldsymbol{x} - \boldsymbol{x_c}]\, A\, [\boldsymbol{x} - \boldsymbol{x_c}] = 1; \quad A = \begin{bmatrix} a & b \\ b & c \end{bmatrix}. \tag{5.2}$$

It is known that the lengths of the semimajor and semiminor axes of the ellipse are given by the square roots of the eigenvalues, $(\lambda_{max}, \lambda_{min})$ of A^{-1}, and the direction of each axis is given by the corresponding eigenvector, $(\boldsymbol{v_{max}}, \boldsymbol{v_{min}})$. The major axis thus intersects the ellipse at $\boldsymbol{p}_{1,2} = \boldsymbol{x}_c \pm \sqrt{\lambda_{max}} \cdot \boldsymbol{v_{max}}$, and the minor axis intersects the ellipse at $\boldsymbol{p}_{3,4} = \boldsymbol{x}_c \pm \sqrt{\lambda_{min}} \cdot \boldsymbol{v_{min}}$.

To simplify initialization, we assume the bounding rectangles of two features i and j correspond to each other. If the length (width) of the bounding rectangle i still maps to length (width) of the bounding rectangle j, these rectangles are related by a *restricted* affine transform with 5 degrees of freedom (dof): a translation (2 dof), a rotation (1 dof), and re-scaling along the length/width of the rectangle (2 dof). The affine transformation parameters, $\boldsymbol{\phi_{init}} = \{\phi_1, ...\phi_6\}$ mapping rectangle i onto rectangle j can then be found by solving the linear equation:

$$\begin{bmatrix} \boldsymbol{p}_{j_1} & \boldsymbol{p}_{j_2} & \boldsymbol{p}_{j_3} & \boldsymbol{p}_{j_4} \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} \phi_1 & \phi_2 & \phi_3 \\ \phi_4 & \phi_5 & \phi_6 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{p}_{i_1} & \boldsymbol{p}_{i_2} & \boldsymbol{p}_{i_3} & \boldsymbol{p}_{i_4} \\ 1 & 1 & 1 & 1 \end{bmatrix}. \tag{5.3}$$

There is a 180 degree ambiguity in the direction of the major axis. We resolve it

assuming that the correct transformation will involve the smaller amount of rotation. This heuristic is in accordance with the way digital photographs are taken — usually, we hold cameras roughly vertical to the floor/ground and we do not make big camera rotation around the optical axis. If camera rotations can be large, this can easily be fixed, in the $\boldsymbol{\phi}$-*optimization* step, by trying both directions.

$\boldsymbol{\phi}$-optimization

We optimize the transformation $\boldsymbol{\phi}$ using direct or intensity based registration. Intensity based registration approaches solve for a smooth transformation between images (or image patches) by maximizing a similarity measure defined on the pixel intensities (cf. , e.g., [119] for details). Specifically, one image, $\boldsymbol{I_{mov}}$, is warped by some parameterized transformation to match the other one, $\boldsymbol{I_{fix}}$, and the similarity is evaluated. The correct registration is considered to be the one that maximizes the similarity between the images. The registration problem is thus expressed as an unconstrained optimization problem

$$\boldsymbol{\phi_{opt}} = argmax_{\boldsymbol{\phi}} S(\boldsymbol{I_{fix}}, \boldsymbol{I_{mov}}(\boldsymbol{W}(\boldsymbol{x}, \boldsymbol{\phi}))). \tag{5.4}$$

In our case, we use the normalized correlation coefficient as the similarity measure S and use the affine transformation ($\boldsymbol{\phi}$ in Equation 5.3) as the warping function $\boldsymbol{W}(\boldsymbol{x}, \boldsymbol{\phi})$. We solve this optimization problem using a trust-region Newton-Raphson optimization method. Details of this type of optimization may be found in [31].

5.2.2 Improved F-estimation

Algorithm 2 gives an outline of our proposed method for automatically computing epipolar geometry between two images. The input to the algorithm is simply the image pair, and the output is the estimated $\mathbf{F}$ and a set of correct matches. The key difference between this algorithm and previous approaches is the *Correspondence refinement* step, where we refine the localizations of each pair of correspondences. However, it is worth pointing out that our approach can be applied to any multiple-view geometry estimation method that follows this basic pattern.

We use the MSERs as our features and their putative correspondences are established by nearest-neighbor matching of their SIFT descriptors [116]. The Sampson distance is used

as the distance function for both the PROSAC consensus testing and the final iterative **F**-estimation. The Sampson measure is found to give adequate accuracy in the **F**-estimation [173][66].

Algorithm 2: Improved algorithm for computing the fundamental matrix.

Data: stereo images I_l, I_r

1 **begin**

1. **Features:** Extract affine invariant features in each image.
2. **Putative correspondences:** Compute a set of feature matches based on similarity of their SIFT Descriptors.
3. **Correspondence refinement:** For each putative match, re-align the position and shape of the feature in one image (I_r) according to match appearances.
4. **PROSAC robust estimation:** Progressively enlarge sample pool, starting with most promising candidates. In the end, PROSAC chooses the **F** with the largest number of inliers.
5. **Non-linear estimation:** Re-estimate **F** from all inliers by minimizing the Sampson cost function.

2 **end**

Result: Fundamental Matrix **F**, the set of inlier matches.

5.3 Experiments

We evaluated our method on three performance metrics. The first is feature localization accuracy, the second is robust outlier rejection, and the third is the accuracy of the final **F** estimated. We experiment on four standard image sets: the *House*, *Corridor*, *Valbonne* and *Shed*. [1] The images are presented in Figure 5.1 and show the estimated epipolar lines. Among them, the *House* and *Corridor* have ground truth **F** and correspondences for our localization accuracy analysis.

[1]The *House*, *Corridor* and *Valbonne* were retrieved from `http://www.robots.ox.ac.uk/~vgg/data1.html`.

(a) House (b) Valbonne (c) Corridor (d) Shed

Fig. 5.1 Image sets with estimated epipolar lines.

5.3.1 Test on feature localization accuracy

It was reported that the MSER [111] is the most accurate in feature localization [64], thus we work on the MSER features to see if further accuracy improvement was indeed achieved.

Error measure

Since our application here is estimating $\mathbf{F}$, we measure the errors of feature locations using the epipolar geometry. We focus on the accuracy of locations only since the shape information was not used in our $\mathbf{F}$-estimation.

We measure the deviation of the matched points with the distance between a point's epipolar line and the matching point in the other image $d(\mathbf{x}'_i, \mathbf{F}\mathbf{x}_i)$, where $d(\mathbf{x}, \mathbf{l})$ is the distance in pixels between a point $\mathbf{x}$ and a line $\mathbf{l}$ (both in homogeneous coordinates). The more precise the matched points are localized, the smaller this distance (or error) is. To ensure that each image receives equal consideration, we examined statistics of the set of errors in both images

$$D = \{d(\mathbf{x}'_i, \mathbf{F}\mathbf{x}_i), d(\mathbf{x}_i, \mathbf{F}^\top \mathbf{x}'_i) | \forall i \in [1, 2, ..., N]\}, \tag{5.5}$$

where N is the number of inlier matches.

Note that although the deviation of a point from its epipolar line doesn't exactly measure its displacement on a 2D image, it can be used as an adequate approximation. By definition, the error $d(\mathbf{x}'_i, \mathbf{F}\mathbf{x}_i)$ measures the deviation of points only in the direction perpendicular to the epipolar lines. A more precise measure is that used in the work by Haja et al. [64], where they examined match localization accuracy by measuring the region overlap errors based on carefully estimated ground truth homographies. However, their overlap measure is very restrictive: one is limited to planar scenes where all features lie on the same plane; while the measure here is suitable for real-life 3D scenes containing complex structures. When the errors in point positions are equally possible in all directions (a condition that is commonly satisfied), the measure is often adequate.

Results on localization accuracy

On the image sets *House* and *Corridor*, we compared the sets of errors D of three point sets using the ground truth fundamental matrix $\mathbf{F}_{truth}$: the *Oxford points*, the *original points*, and the *refined points*. The *Oxford points* are ground truth point matches provided along with the image set. The *original points* are the MSER features selected as inlying matches. And the *refined points* are the inlier subset of our registration-refined MSER features. Both of the above two point sets are selected using the PROSAC algorithm.

Figure 5.2 shows the error of the *refined points* is statistically lower than other point sets on both *House* and *Corridor*. On the *House* dataset, for example, *refined points* have an error median of 0.1 pixel; the values for *original points* and *Oxford points* are 0.14 and 0.25 respectively. Since the ground truth points were hand-selected by visual inspection, their localization could be inaccurate, thus, the errors of the *Oxford points* end up being the largest. This also explains why our registration-based refinement can lead to an even better localization accuracy.

5.3.2 Improvement on robust inlier detection

We applied Algorithm 2 on the image sets and obtained results on robust inlier detection. Table 5.1 shows the average number of inliers detected and samples drawn of different point sets over 100 experiments.

Table 5.1 reveals that refinement can improve robust inlier detection in two ways. First, it encourages more inliers to be detected. This is because some small inaccuracies of feature

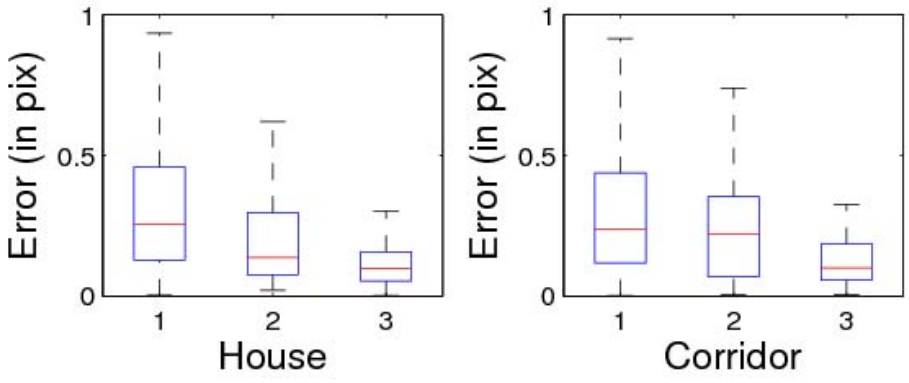


Fig. 5.2 Accuracy comparison result using ground truth fundamental matrix ($\mathbf{F}_{truth}$). For each of *House* and *Corridor*, we show error boxplots of three different point sets, the *Oxford points* (represented by 1 on the x-axes), *original points* (2) and *refined points* (3). Along the y-axis, the units are in pixels. For each point set, the red line in the center of the box is the error median and the upper and lower horizontal lines of the box represent the top quartile and bottom quartile.

Methods	#Total matches	#Inlier matches		#Sample trials	
		original matches	refined matches	original matches	refined matches
House	77	55	62	16	13
Valbonne	26	16	18	32	8
Corridor	66	45	54	31	7
Shed	47	34	35	30	10

Table 5.1 The comparison result of improvement on robust estimation. In the table, we compare the number of inliers detected and the number of samples drawn on the set of original vs. refined matches.

location are improved, thus some of the previously classified outliers are corrected. This correction is particularly important when the overall match count is low - we do not want to lose any matches due to false-negatives. Second, it drastically reduces the number of samples needed to find the correct solution. Using the PROSAC scheme on both point sets, fewer samples are drawn from the *refined points* to detect the inliers. This trend is consistent over all data. Part of the reason is more accurate locations will facilitate identifying the correct model (in this case, the fundamental matrix $\mathbf{F}$) more quickly, i.e., avoid being distracted by minor inaccuracies. Another factor is that the registration provides better

similarity scores for the match ranking of PROSAC. This sample count reduction is more obvious when the inlier percentage is low, as in *Valbonne*, *Shed*, and *Corridor*.

5.3.3 Improvement in F-estimation accuracy

The statistics on the set of distances D (Equation 5.5) are also commonly used in measuring **F**-estimation accuracy [66]. We estimated **F** using the different point sets and then measured the resulting errors. The same procedure, iteratively re-weighted least squares minimization using Sampson distance [173], is used for estimating the **F** on all point sets.

In Figure 5.3, we show the errors of *original points* versus *refined points* on the four images. The *refined points* consistently achieve better accuracy than the *original points*. Between 21% and 67% reduction in error median is achieved by the proposed method. It is worth noting that, in the case of *Valbonne* (cf. plot for *Valbonne* in Figure 5.3), if we were to directly compare the errors here against that presented in Table 3 of [111], we would see no noticeable improvement. The reason is that although both used the *Valbonne* sequence, they used *different* images in the sequence. Also, their set of correspondences were obtained differently: first they estimated a "rough EG", then more and better correspondences were obtained using "guided-matching" with a very narrow threshold. This *narrowing* of threshold effectively ensures that only those better-localized matches be selected. Their "guided-matching" effectively adds more accurate matches and deletes bad ones; while our method works on available matches and makes them better. Both strategies are useful; in practice, one can combine them for a further improved result.

5.3.4 Computing time

Running time of the proposed method is dependent on many factors, e.g., the number of putative matches, accuracy in their original alignment, etc. Our method spends extra time on local patch optimization but needs less time for robust outlier rejection.

Our current implementation of the algorithm is a mixture of Matlab scripts and C++ binaries, and its running time is not particularly informative. However, we have timed the registration components of the algorithm. The local patch optimization consists of an initialization, mainly involving pre-computing image derivatives, which only needs to be done once per image pair, and the optimization of each patch. Running a single core on a 1.8GHz Intel Core Duo processor, the initialization times ranged between 0.32–0.72s

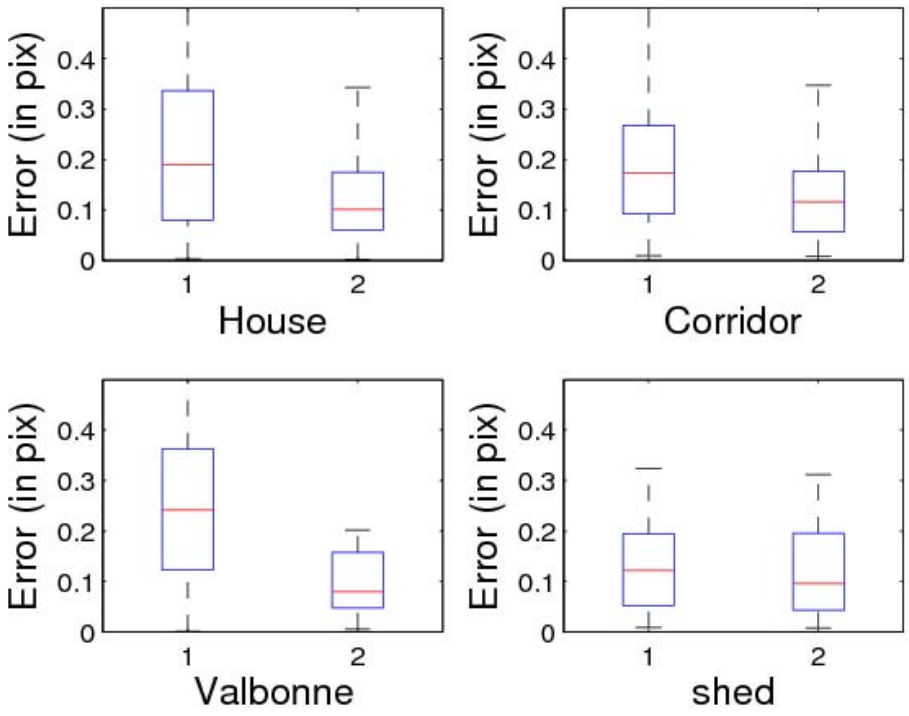


Fig. 5.3 Accuracy comparison result of F-estimation. Error boxplots of different point sets on four image sets. 1 and 2 on the x-axes represent the *original points* and *refined points* respectively. The lower the error (along y-axis) is, the more accurate is the **F**-estimation. The red lines indicate the error medians.

(depending on the size of the image), and the optimization time averaged 11.5ms per patch. Considering that both parts of the algorithm can be easily parallelized, we expect that the processing time of an integrated algorithm could be reduced to a reasonable range.

5.4 Conclusions

We proposed a method for improving the quality of correspondences by patch-wise registration. This registration further improves localization accuracy of feature detectors and produces a better measure of feature similarity. In the context of robust fundamental matrix estimation, our method enables a more effective outlier rejection and obtains a more accurate fundamental matrix.

It is reasonable to expect this improvement in feature localization accuracy since information from both images is utilized, whereas in the feature detection step one decides feature localization based on only a single image. This idea of registration-based correspondence refinement can also be used in other tasks involving multiple-view correspondence, since this gain in localization accuracy is always desirable. The effect of our method on the MSERs implies that the **F**-estimation can always benefit from the refinement even if one uses other features, since most popular features have more room for localization improvement than MSERs [64].

Another gain from this registration that can be explored is the improvement in the local mapping implied by the feature shape parameters. Some approaches, such as [140], have used this shape information to assist in **F**-estimation using only a few feature correspondences. Improving the accuracy of this local mapping could enhance the applicability of this type of approach.

Chapter 6

Conclusions and future research

This thesis deals with the problem of wide-baseline stereo, with an emphasis on applications in 3-D urban environments. I treat the problem of matching widely separated views as solving three inter-connected problems: extracting feature points for matching, establishing one-to-one matching between the points, and using the matches to estimate the two-view epipolar geometry. Solving wide-baseline stereo problems is fundamental to many 3-D computer vision applications. For example, the latest wide-baseline stereo techniques have enabled researchers to recover the structure of 3-D landmarks using a sparse set of images mined from the internet [165][1][203]. Robot navigation also makes use of wide-baseline stereo to reason about the environment's 3-D spatial layout, e.g., autonomous navigation of Mars rovers by Olson and Abi-Rached [130].

Algorithms described in this thesis make wide-baseline stereo more accessible to 3-D vision applications. Real-life applications are often faced with input images that are hard to match, either due to noise of image acquisition systems or due to photometric/geometric properties of 3-D structures. These kinds of challenges are well-handled by our methods, thus, they bring much-needed robustness to 3-D vision applications. In addition, our techniques can also increase accuracy for 3-D applications, owing to the benefits of correspondence refinement. For example, one could obtain more accurate 3-D models if refined correspondences were used for image-based modeling, more photo-realistic texture if used for image-based rendering, or better location estimation in the case of localization and navigation, etc.

Methods developed in this work are also useful to areas other than wide-baseline stereo.

As exemplified in Section 1.2.5, the component problems are widely used in fields such as image registration, object tracking, content-based image retrieval, etc. Thus, I have studied the three problems individually so that the results could be used modularly by many applications. Each problem has been approached from a theoretical perspective, and backed up by extensive experiments. Throughout, I proposed methods that are generally formulated, rather than limited to wide-baseline stereo assumptions. As a result, contributions from this thesis are applicable to a wide range of computer vision and pattern recognition applications.

6.1 Discussion

In Chapter 3, I examined the extraction of features for wide-baseline matching. The notion of entropy-based saliency has been successfully applied in learning and recognition [79][53][54][93], but no effective use has been reported for wide-baseline matching. One explanation might be found in the evaluation reported by Mikolajczyk et al. [116], who found that the Salient Region falls behind other features in the repeatability criterion. I hypothesized that this might be due to localization inaccuracy of the Salient Region detector, which overshadows its advantage in saliency detection. Thus, I proposed to combine the strength of localization by MSER and the saliency detection capability of the entropy-based approach. Experiments showed promising results from this combination. The resulting Structure-Guided Salient Regions are comprised of a subset of MSERs that exhibit the best saliency scores. Those MSERs that are due to unwanted high-frequency patterns are regarded as noise and eliminated by the saliency selection.

With the proposed method comes a much faster run time for saliency detection. The original Salient Region detector needs 2–3 hours to process a mid-sized (e.g., 1024×768) image; while it takes the SGSR detector only a few seconds. By analyzing only the seeds originated from MSERs, saliency examination is far more efficient due to a much reduced set of candidates. As another source for gain in efficiency, we only need to examine a much narrower range of scale for each seed.

The new detector also extends saliency detection from greyscale to color images. This is due to our new procedure for representing the probability density function of pixel values with a histogram at reduced resolutions. Thus, the histogram of a color region can be efficiently processed. Besides, this representation is better suited for small regions and

more robust against image noise.

Chapter 4 dealt with matching features between stereo views. The primary novelty is a new paradigm for feature matching that is driven by image context. This paradigm inherits concepts developed in the stereo vision literature [143][85][6][94][201], and applies them into wide-baseline matching by explicitly dealing with scale and affine-invariance, partial occlusion, and noisy measurements. The experiments show that, in the absence of reliability of region descriptors, the relative layout of spatially proximate features can serve as a strong cue for matching features.

To realize this new paradigm, I proposed to connect features in an image by a graph model called the Salient Feature Graph (SFG). The SFG enjoys the desirable property that its structure is viewpoint-invariant. For a distinct 3-D point, its corresponding features as imaged from various viewpoints will have consistent edge-connected neighbors, assuming the surface is locally planar. This ensures that, when comparing spatial neighborhoods of features from different views, visual information of the same 3-D surface is examined. The above concept is implemented by building the SFG based on a new measure of feature proximity. According to the new proximity measure, each feature is edge-connected to a set of nearby features as its "context", taking into account the geometry of both the central and the peripheral features. As a result, the SFGs of stereo images will likely have a more consistent graph structure — greatly facilitating the subsequent graph matching.

To examine similarity between two feature clusters, I proposed a geometric procedure called Neighborhood Transform. The Neighborhood Transform explicitly deals with the effect of viewpoint changes. Thus, when we examine similarity between features from two perspectives by using their neighboring points as support, it facilitates the comparison. In essence, the Neighborhood Transform geometrically aligns neighbors in different views into the same perspective, so that viewpoint-induced geometric distortion is accounted for.

Along with the Neighborhood Transform, I proposed a procedure called hypothesize-match to compute similarity between two groups of nearby features, solving a many-to-many match [39] without explicit one-to-one correspondence. In practice, despite the SFG structures being designed for viewpoint invariance, those of real stereo images inevitably differ from each other due to image noise, feature detection, or deviation from the local-planarity assumption. Because the hypothesize-match procedure computes similarity between individual neighbors independently, the accumulation of contextual support is resilient to structural inconsistencies between stereo SFG models.

To match features using information globally across the image, I proposed a graph-based optimization algorithm, namely Context-Consistent Assignment (CCA). The CCA adapts the softassign formulation and abstracts the concept of binary edge into the notion of "Virtual Edge". The Virtual Edge can be seen as connecting the central node with all neighboring nodes all at once and similarity between Virtual Edges is reflected by their Context Consistency.

Chapter 5 considered the problem of estimating the fundamental matrix using feature matches. Different from conventional methods, I treated the given matches as raw inputs whose localization *can* be refined. Thus, I proposed a refinement procedure that is based on a patch-wise image registration. Experiments showed that the refinement consistently improves both efficiency and accuracy of the fundamental matrix estimation.

6.2 Future research

With the demonstrated contributions, this work also reveals numerous areas for further investigation.

One area for investigation is the utilization of features' saliency score (cf. Equation 3.1) in feature matching. The quantitative values of feature saliency are computed during the feature detection by entropy-based methods [79][48]. They have not been used in matching features, since many feature detectors do not produce such a quality score. These scores, however, can be used by our context-driven feature matching. For example, we can use the scores as an indicator of confidence in the contextual support (Section 4.3.2) contributed by the individual features. For this purpose, a desirable property for SGSRs is that those with high saliency scores have high likelihoods of being detected when the imaging conditions change. We can gain insight into this property by experimentally testing the correlation between saliency and repeatability, using image sets representing various changes of imaging conditions (e.g., those provided by Mikolajczyk et al. [116]).

A second area for investigation is along the line of matching features using their image context. In this thesis, our algorithm draws context from features' immediate neighbors. The neighbors provide *semi-local* contexts that are topologically connected by the Salient Feature Graph. Meanwhile, studies of the human visual system can provide us with inspiration on alternative ways to use image context. Oliva and Torralba, for example, found that "fast scene recognition does not need to be built on top of the processing of objects,

but can be analyzed in parallel by scene-centered mechanisms" [129]. Similarly, we could build a system that matches images in a global-to-local fashion. Such a method starts with quickly matching several distinct areas of the images, based on a coarse structure of the scene. Subsequently, fine structures (features) can be matched within their circumscribed regions. At each scale-level, this coarse-to-fine approach only needs to distinguish between a few candidates. Thus, the matching can be both fast and robust. The difficulty lies in the extraction and representation of the relatively large "distinct areas". One possible solution might be through a bottom-up construction: we can hierarchically cluster nearby features extracted from fine structures. The statistics or topological structures of the clusters can serve as their representation.

A third area for investigation would be determining a way to integrate the epipolar constraint into the feature matching procedure, just as we humans use spatial layout constraint to quickly match different perspectives of a rigid scene. Feature matching leads to the estimation of epipolar geometry, because the former provides a number of correspondences to fundamental matrix estimation algorithms. However, the global epipolar constraint can effectively reduce the search space of the matching. Thus, it is an accepted practice to augment feature matches with a rough epipolar geometry that are estimated using an initial set of matches [111][66]. In challenging situations, it may become difficult to reliably obtain the initial matches. In this case, a matching algorithm will operate on a much reduced search space if it can make use of the *unknown* epipolar geometry. For example, we might interleave feature matching with computing the fundamental matrix by maintaining an overall matching cost. We could make gradual adjustments on the parameters of the fundamental matrix in the direction that reduces the overall matching cost (i.e., increases overall matches). The initialization and optimization of the fundamental matrix would be the first major theoretical undertaking on the path to making such an improvement happen. In essence, the integrated method seeks to simultaneously solve steps two and three of wide-baseline stereo. It could resolve matching ambiguity in a way that is more intelligent than traditional methods that try to solve each step individually.

As can be seen from the above avenues, the present work hints at a new approach to wide-baseline stereo. By drawing inspiration from mechanisms of the human visual system, future methods could be capable of matching more challenging images — at a speed as fast as the blink of an eye.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski. Building rome in a day. In *Proceedings of International Conference of Computer Vision*, pages 72–79, 2009. 1, 2, 18, 33, 55, 103

[2] J.K. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images: A review. *Proceedings of IEEE*, 76:917–935, 1988. 34

[3] A. Ali and J.K. Aggarwal. Segmentation and recognition of continuous human activity. In *Proceedings of IEEE Workshop on Detection and Recognition of Events in Video*, pages 28–35, 2001. 17

[4] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM*, 45(6):891–923, 1998. 33

[5] H. Asada and M. Brady. The curvature primal sketch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):2–14, January 1986. 25

[6] N.J. Ayache and B. Faverjon. Efficient registration of stereo images by matching graph descriptions of edge segments. *International Journal of Computer Vision*, 1:107–132, 1987. 32, 36, 56, 105

[7] S. T. Barnard and W. B. Thompson. Disparity analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(4):333–340, 1980. 8, 30, 56

[8] S.T. Barnard and M.A. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, December 1982. 15

[9] A.E. Bartoli and P.F. Sturm. Nonlinear estimation of the fundamental matrix with minimal parameters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):426–432, March 2004. 34

[10] A.M. Baumberg. Reliable feature matching across widely separated views. In *Proceedings of Computer Vision and Pattern Recognition*, pages 774–781, 2000. 26, 38

[11] H. Bay, V. Ferrari, and L. Van Gool. Wide-baseline stereo matching with line segments. In *Proceedings of Computer Vision and Pattern Recognition*, pages 329–336, 2005. 28

[12] P.R. Beaudet. Rotationally invariant image operators. In *Proceedings of International Conference on Pattern Recognition*, pages 579–583, 1978. 25

[13] J. S. Beis and D. G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proceedings of Computer Vision and Pattern Recognition*, 1997. 33

[14] S.J. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002. 33

[15] F. L. Bookstein. Fitting conic sections to scattered data. *Computer Graphics and Image Processing*, 9(1):56–71, 1979. 92

[16] K.L. Boyer and A.C. Kak. Structural stereo for 3-d vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):144–166, March 1988. 31, 37

[17] M. Brown and D.G. Lowe. Invariant features from interest point groups. In *Proceedings of British Machine Vision Conference*, pages 253–262, 2002. 30

[18] G. Carneiro and A.D. Jepson. Flexible spatial models for grouping local image features. In *Proceedings of Computer Vision and Pattern Recognition*, pages II: 747–754, 2004. 17, 61, 68

[19] M. Chandraker, S. Agarwal, D. Kriegman, and S. Belongie. Globally optimal affine and metric upgrades in stratified autocalibration. In *Proceedings of International Conference on Computer Vision*, pages 1–8, 2007. 18

[20] P. Chen, F. Dong, C. Zhao, and T. Guan. Augmented reality based on online trifocal tensors estimation using multiple features. *Sensor Review*, 29(3):277–286, 2009. 2

[21] F. Chevalier, J.P. Domenger, J. Benois Pineau, and M. Delest. Retrieval of objects in video by similarity based on graph matching. *Pattern Recognition Letters*, 28:939–949, June 2007. 36

[22] O. Choi and I.S. Kweon. Robust feature point matching by preserving local geometric consistency. *Computer Vision and Image Understanding*, 113(6):726–742, June 2009. 13, 14, 29, 35, 57

[23] W. Chojnacki, M.J. Brooks, A.J. van den Hengel, and D. Gawley. On the fitting of surfaces to data with covariances. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1294–1303, November 2000. 34

[24] W. Chojnacki, M.J. Brooks, A.J. van den Hengel, and D. Gawley. A new constrained parameter estimator for computer vision applications. *Image and Vision Computing*, 22(2):85–91, February 2004. 34, 91

[25] W. J. Christmas, J. Kittler, and M. Petrou. Structural matching in computer vision using probabilistic relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):749–764, 1995. 37

[26] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, February 2003. 36, 37

[27] O. Chum and J. Matas. Matching with PROSAC: Progressive sample consensus. In *Proceedings of Computer Vision and Pattern Recognition*, pages I: 220–226, 2005. 35, 90, 92

[28] F. R. K. Chung. *Spectral Graph Theory*. Number 92 in CBMS Regional Conference Series in Mathematics. American Mathematical Society, February 1997. 37

[29] Y.C. Chung, T. X. Han, and Z. He. Building recognition using sketch-based representations and spectral graph matching. In *Proceedings of International Conference of Computer Vision*, pages 2014–2020, 2009. 35

[30] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–577, May 2003. 17

[31] A. R. Conn, N. I. M. Gould, and P. L. Toint. *Trust-Region Methods*. Society for Industrial and Applied Mathematics and Mathematical Programming Society, 2000. 95

[32] D. Conte, P. Foggia, J.-M. Jolion, and M. Vento. A graph-based, multi-resolution algorithm for tracking objects in presence of occlusions. *Pattern Recognition*, 39(4):562 – 572, 2006. 36

[33] D. Conte, P. Foggia, C. Sansone, and M. Vento. *How and Why Pattern Recognition and Computer Vision Applications Use Graph*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007. 36

[34] I. J. Cox, S. L. Hingorani, and S. B. Rao. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996. 31

[35] J.L. Crowley and A.C. Parker. A representation for shape based on peaks and ridges in the difference of low-pass transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):156–169, March 1984. 27

[36] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Proceedings of ECCV Workshop on Statistical Learning in Computer Vision*, pages 1–22, 2004. 59

[37] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image retrieval: ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2):1–60, April 2008. 17

[38] P.E. Debevec, C.J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, pages 11–20, 1996. 18

[39] M.F. Demirci, A. Shokoufandeh, Y. Keselman, L. Bretzner, and S.J. Dickinson. Object recognition as many-to-many feature matching. *International Journal of Computer Vision*, 69(2):203–222, August 2006. 68, 105

[40] H.L. Deng, E.N. Mortensen, L.G. Shapiro, and T.G. Dietterich. Reinforcement matching using region context. In *Proceedings of CVPR Workshop on Beyond Patches*, page 11, 2006. 29, 35, 56

[41] U. R. Dhond and J. K. Aggarwal. Structure from stereo - a review. *IEEE Transactions on Systems, Man and Cybernetics*, 19(6):1489–1510, 1989. 15

[42] A.R. Dick, P.H.S. Torr, and R. Cipolla. Modelling and interpretation of architecture from several images. *International Journal of Computer Vision*, 60(2):111–134, November 2004. 18

[43] S.J. Dickinson, A.P. Pentland, and A. Rosenfeld. From volumes to views: An approach to 3-d object recognition. *Computer Vision, Graphics, & Image Processing*, 55(2), March 1992. 36

[44] L. Dreschler and H.H. Nagel. Volumetric model and 3d trajectory of a moving car from monocular tv frames sequence of a street scene. *Computer Graphics and Image Processing*, 20(3):199–228, November 1982. 25

[45] Y. Dufournaud, C. Schmid, and R. Horaud. Matching images with different resolutions. In *Proceedings of Computer Vision and Pattern Recognition*, pages 612–618, 2000. 26

[46] M.A. Eshera and K.S. Fu. A graph distance measure for image analysis. *IEEE Transactions on Systems, Man and Cybernetics*, 14(3):398–408, May 1984. 37

[47] M.A. Eshera and K.S. Fu. A similarity measure between attributed relational graphs for image analysis. In *Proceedings of International Conference on Pattern Recognition*, pages 75–77, 1984. 37

[48] S. Fan and F. Ferrie. Structure guided salient region detector. In *Proceedings of British Machine Vision Conference*, pages 423–432, 2008. 106

[49] S. Fan and F. Ferrie. Context-consistent stereo matching. In *Proceedings of ICCV Workshop on 3-D Digital Imaging and Modeling*, pages 1694–1701, 2009. 74

[50] O.D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *Proceedings of European Conference on Computer Vision*, pages 563–578, 1992. 1, 8, 34

[51] O.D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993. 15

[52] O.D. Faugeras, Q.T. Luong, and S.J. Maybank. Camera self-calibration: Theory and experiments. In *Proceedings of European Conference on Computer Vision*, pages 321–334, 1992. 8, 34

[53] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of Computer Vision and Pattern Recognition*, pages II: 264–271, 2003. 17, 59, 104

[54] R. Fergus, P. Perona, A. Zisserman, R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *Proceedings of Computer Vision and Pattern Recognition*, pages I: 380–387, 2005. 104

[55] V. Ferrari, T. Tuytelaars, and L.J. Van Gool. Wide-baseline multiple-view correspondences. In *Proceedings of Computer Vision and Pattern Recognition*, pages I: 718–725, 2003. 30

[56] A.M. Finch, R.C. Wilson, and E.R. Hancock. Matching delaunay graphs. *Pattern Recognition*, 30(1):123–140, January 1997. 37

[57] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 12, 14, 34, 52, 73, 90, 92

[58] W. Forstner. A feature based correspondence algorithm for image matching. In *Proceedings of International Society of Photogrammetry and Remote Sensing Congress*, pages III: 150–166, 1986. 25

[59] P. Georgel, S. Benhimane, and N. Navab. A unified approach combining photometric and geometric information for pose estimation. In *Proceedings of British Machine Vision Conference*, pages 133–142, 2008. 91

[60] T. Goedeme, T. Tuytelaars, and L.J. Van Gool. Fast wide baseline matching for visual navigation. In *Proceedings of Computer Vision and Pattern Recognition*, pages I: 24–29, 2004. 2

[61] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:377–388, April 1996. 37, 57, 58, 66, 68, 70

[62] C. Gomila and F. Meyer. Graph-based object tracking. In *Proceedings of International Conference on Image Processing*, pages II: 41–44, 2003. 36

[63] M. Gong and Y. H. Yang. Fast stereo matching using reliability-based dynamic programming and consistency constraints. In *Proceedings of International Conference on Computer Vision*, pages 610–617, 2003. 31

[64] A. Haja, B. Jahne, and S. Abraham. Localization accuracy of region detectors. In *Proceedings of Computer Vision and Pattern Recognition*, pages 1–8, June 2008. 90, 97, 98, 102

[65] C. Harris and M.J. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–152, 1988. 24, 28

[66] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 8, 9, 11, 34, 92, 96, 100, 107

[67] R.I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *Proceedings of European Conference on Computer Vision*, pages 579–587, 1992. 1, 8, 34

[68] R.I. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proceedings of Computer Vision and Pattern Recognition*, pages 761–764, 1992. 8, 34

[69] R.I. Hartley and F. Kahl. Critical configurations for projective reconstruction from multiple views. *International Journal of Computer Vision*, 71(1):5–47, January 2007. 13

[70] R. Horaud, F. Veilon, and T. Skordas. Finding geometric and relational structures in an image. In *Proceedings of European Conference on Computer Vision*, pages 374–384, 1990. 26

[71] T.S. Huang and O.D. Faugeras. Some properties of the e matrix in two-view motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(12):1310–1312, December 1989. 34

[72] T.S. Huang and A.N. Netravali. Motion and structure from feature correspondences: A review. *Proceedings of IEEE*, 82(2):252–268, February 1994. 8, 34

[73] P.J. Huber. *Robust Statistics*. Wiley, 1981. 34

[74] T. Jebara, A. Azarbayejani, and A.P. Pentland. 3d structure from 2d motion. *IEEE signal processing magazine*, 16(3):66–84, May 1999. 8

[75] C. Jerian and R.C. Jain. Polynomial methods for structure from motion. In *Proceedings of International Conference on Computer Vision*, pages 197–206, 1988. 34

[76] D. Johns and G. Dudek. Urban position estimation from one dimensional visual cues. In *The 3rd Canadian Conference on Computer and Robot Vision*, page 22, 2006. 30

[77] G.A. Jones. Constraint, optimization, and hierarchy: Reviewing stereoscopic correspondence of complex features. *Computer Vision and Image Understanding*, 65(1):57–78, January 1997. 56

[78] D. Jugessur and G. Dudek. Local appearance for robust object recognition. In *Proceedings of Computer Vision and Pattern Recognition*, pages I: 834–839, 2000. 28

[79] T. Kadir and M. Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45(2):83–105, November 2001. 20, 21, 24, 29, 40, 53, 104, 106

[80] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *Proceedings of European Conference on Computer Vision*, pages 228–241, 2004. 24, 41, 44

[81] K. Kanatani and Y. Sugaya. High accuracy fundamental matrix computation and its performance evaluation. *IEICE Transactions on Information and Systems*, E90-D(2):579–585, 2007. 91

[82] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In *Proceedings of International Conference on Computer Vision*, pages 1–8, 2007. 36

[83] S. Kim and I.S. Kweon. Robust model-based scene interpretation by multilayered context information. *Computer Vision and Image Understanding*, 105(3):167–187, March 2007. 36

[84] S. Kim and I.S. Kweon. Simultaneous place and object recognition using collaborative context information. *Image and Vision Computing*, 27(6):824–833, May 2009. 36

[85] Y. Kim and J. Aggarwal. Positioning three-dimensional objects using stereo images. *IEEE Journal of Robotics and Automation*, 3(4):361–373, August 1987. 31, 105

[86] L. Kitchen and A. Rosenfeld. Gray level corner detection. *Pattern Recognition Letters*, 1(2):95–102, December 1982. 25

[87] J.J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984. 26

[88] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, August 2005. 59, 64

[89] J. A. Lee, K. C. Yow, and A. Y. S. Chia. Robust matching of building facades under large viewpoint changes. In *Proceedings of International Conference of Computer Vision*, pages 1258–1264, 2009. 35

[90] C. Lei, J. Selzer, and Y.H. Yang. Region-tree based stereo using dynamic programming optimization. In *Proceedings of Computer Vision and Pattern Recognition*, pages II: 2378–2385, 2006. 31

[91] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *International Journal of Computer Vision*, pages II: 1482–1489, 2005. 37

[92] M.D. Levine, D.A. O'Handley, and G.M. Yagi. Computer determination of depth maps. *Computer Graphics and Image Processing*, 2(2):131–150, October 1973. 8, 55

[93] F.F. Li and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proceedings of Computer Vision and Pattern Recognition*, pages II: 524–531, 2005. 104

[94] S. Z. Li. Matching: Invariant to translations, rotations and scale changes. *Pattern Recognition*, 25(6):583 – 594, 1992. 2, 32, 36, 66, 105

[95] H.S. Lim and Binford T.O. Stereo correspondence: A hierarchical approach. In *Proceedings of DARPA Image Understanding Workshop*, pages 234–241, 1987. 31, 56

[96] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer, 1994. 26, 27

[97] T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998. 26, 28, 38, 42

[98] Y.C. Liu and T.S. Huang. Estimation of rigid body motion using straight line correspondences. In *Proceedings of IEEE Workshop on Motion*, pages 47–52, 1986. 34

[99] J. Llados, E. Marti, and J.J. Villanueva. Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1137–1143, October 2001. 37

[100] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981. 8, 34

[101] M.I.A. Lourakis and A.A. Argyros. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical report, Institute of Computer Science - FORTH, Heraklion, Crete, Greece, Retrieved Dec. 2nd, 2009 from website `http://www.ics.forth.gr/~lourakis/sba`. 18

[102] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 2, 18, 24, 27, 32, 46, 54, 56, 62, 92

[103] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision (darpa). In *Proceedings of the 1981 DARPA Image Understanding Workshop*, pages 121–130, April 1981. 2

[104] B. Luo and E.R. Hancock. Structural graph matching using the em algorithm and singular value decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1120–1136, October 2001. 37

[105] Q.-T. Luong. *Matrice fondamentale et auto-calibration en vision par ordinateur*. PhD thesis, Universite de Paris-Sud, Orsay., 1992. 8, 34

[106] Q.T. Luong, R. Deriche, O.D. Faugeras, and T. Papadopoulo. On determining the fundamental matrix: Analysis of different methods and experimental results. In *INRIA Technical Report 1894*, 1993. 34, 92

[107] Q.T. Luong and O.D. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *Proceedings of International Conference of Computer Vision*, 17(1):43–75, January 1996. 2, 14

[108] S.D. Ma. Conics-based stereo, motion estimation and pose determination. *International Journal of Computer Vision*, 10(1):7–25, February 1993. 34

[109] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194(4262):283–287, 1976. 2, 8, 15, 56

[110] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London - Biological Sciences*, 204(1156):301–328, 1979. 13, 31, 56, 59

[111] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, September 2004. 2, 14, 28, 38, 43, 59, 74, 97, 100, 107

[112] G. Medioni and R. Nevatia. Matching images using linear features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):675–685, 1984. 56

[113] Microsoft. Photosynth homepage. *http://photosynth.net/*, Retrieved Dec. 2nd, 2009. 1

[114] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, October 2004. 2, 13, 14, 28, 38

[115] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, October 2005. 28, 32, 33, 54, 61, 71, 74

[116] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005. 28, 38, 40, 42, 45, 48, 54, 55, 61, 74, 95, 104, 106

[117] A. Mitiche and J.K. Aggarwal. A computational analysis of time-varying images. *Handbook of Pattern Recognition and Image Processing*, pages 311–332, 1986. 34

[118] A. Mitiche, S. Seida, and J.K. Aggarwal. Line-based computation of structure and motion using angular invariance. In *Proceedings of IEEE Workshop on Motion*, pages 175–180, 1986. 34

[119] J. Modersitzki. *Numerical Methods for Image Registration*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2004. 95

[120] F. Mokhtarian and A.K. Mackworth. Scale based description and recognition of planar curves and two-dimensional shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):34–43, January 1986. 26

[121] F. Mokhtarian and R. Suomela. Robust image corner detection through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1376–1381, December 1998. 26

[122] H. Moravec. Towards automatic visual obstacle avoidance. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 584–587, 1977. 24, 56

[123] R. Myers and E.R. Hancock. Least-commitment graph matching with genetic algorithms. *Pattern Recognition*, 34(2):375–394, February 2001. 37

[124] R. Myers, R.C. Wilson, and E.R. Hancock. Bayesian graph edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6):628–635, June 2000. 37

[125] K. Ni, H. Jin, and F. Dellaert. Groupsac: Efficient consensus in the presence of groupings. In *Proceedings of International Conference of Computer Vision*, pages 2193–2200, 2009. 35

[126] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004. 8

[127] S. Obdrzalek and J. Matas. Image retrieval using local compact DCT-based representation. In *German Association for Pattern Recognition Annual Conference*, pages 490–497, 2003. 92

[128] A. Oliva. Gist of the scene. *In the Encyclopedia of Neurobiology of Attention*, pages 251–256, 2005. 55

[129] A. Oliva and A. Torralba. Building the gist of a scene: the role of global image features in recognition. *Progress in Brain Research*, 155:23–36, 2006. 107

[130] C.F. Olson and H. Abi-Rached. Wide-baseline stereo experiments in natural terrain. In *Proceedings of International Conference on Advanced Robotics*, pages 376–383, July 2005. 103

[131] P. Parent and S.W. Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8):823–839, August 1989. 66

[132] M. Pelillo. Replicator equations, maximal cliques, and graph isomorphism. *Neural Computation*, 11(9):1933–1955, 1999. 37

[133] M. Pelillo, K. Siddiqi, and S.W. Zucker. Matching hierarchical structures using association graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1105–1120, 1999. 37

[134] M. Pollefeys, R. Koch, and L.J. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proceedings of International Conference on Computer Vision*, pages 90–95, 1998. 1, 18

[135] M. Pollefeys, D. Nistr, J. . Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S. . Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewnius, R. Yang, G. Welch, and H. Towles. Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision*, 78(2-3):143–167, 2008. 2, 18

[136] M. Pollefeys, L.J. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232, September 2004. 1, 2, 8, 18

[137] P. Pritchett and A. Zisserman. Wide baseline stereo matching. In *International Journal of Computer Vision*, pages 754–760, 1998. 2, 7, 14, 29, 33

[138] L. Quan and R. Mohr. Matching perspective images using geometric constraints and perceptual grouping. In *Proceedings of International Conference on Computer Vision*, pages 679–684, 1988. 2, 29, 33, 56

[139] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Context-free attentional operators: The generalized symmetry transform. *International Journal of Computer Vision*, 14(2):119–130, March 1995. 26

[140] F. Riggi, M. Toews, and T. Arbel. Fundamental matrix estimation via TIP - transfer of invariant parameters. In *Proceedings of International Conference on Pattern Recognition*, pages 21–24, Hong Kong, August 2006. 102

[141] J.W. Roach and J.K. Aggarwal. Determining the movement of objects from a sequence of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(6):554–562, November 1980. 34

[142] K. Rohr. Recognizing corners by fitting parametric models. *International Journal of Computer Vision*, 9(3):213–230, 1992. 26

[143] A. Rosenfeld, R. A. Hummel, and S. W. Zucker. Scene labeling by relaxation operations. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-6(6):420–433, 1976. 2, 30, 105

[144] P.J. Rousseeuw. *Robust Regression and Outlier Detection*. Wiley, 1987. 14, 34, 92

[145] A. Sanfeliu and K.S. Fu. Distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-13(3):353–362, 1983. 37

[146] T. Sato, M. Kanbara, N. Yokoya, and H. Takemura. Dense 3-d reconstruction of an outdoor scene by hundreds-baseline stereo using a hand-held video camera. *International Journal of Computer Vision*, 47(1-3):119–129, 2002. 2

[147] T. Sattler, B. Leibe, and L. Kobbelt. Scramsac: Improving ransac's efficiency with a spatial consistency filter. In *Proceedings of International Conference of Computer Vision*, pages 2090–2097, 2009. 35

[148] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In *Proceedings of European Conference on Computer Vision*, pages 414–431, London, UK, 2002. 30

[149] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002. 14, 16, 33, 59

[150] D. Scharstein and R. Szeliski. Middlebury stereo vision page. *http://vision.middlebury.edu/stereo/*, Retrieved Dec. 1st, 2009. 16

[151] G. Schindler, P. Krishnamurthy, R. Lublinerman, Y.X. Liu, and F. Dellaert. Detecting and matching repeated patterns for automatic geo-tagging in urban environments. In *Proceedings of Computer Vision and Pattern Recognition*, pages 1–7, 2008. 2

[152] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, May 1997. 2, 23, 32, 56

[153] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172, June 2000. 24

[154] D. C. Schmidt and L. E. Druffel. A fast backtracking algorithm to test directed graphs for isomorphism using distance matrices. *Journal of the Association for Computer Machinery*, 23(3):433–445, 1976. 36

[155] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of Computer Vision and Pattern Recognition*, pages 519–528, 2006. 2

[156] L. G. Shapiro and R. M. Haralick. Structural descriptions and inexact matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3(5):504–519, 1981. 31, 37

[157] L. G. Shapiro and R. M. Haralick. Metric for comparing relational descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7(1):90–94, 1985. 37

[158] J. Shi and C. Tomasi. Good features to track. In *Proceedings of Computer Vision and Pattern Recognition*, pages 593–600, 1994. 25, 28

[159] A. Shokoufandeh, L. Bretzner, D. Macrini, M.F. Demirci, C. Jonsson, and S.J. Dickinson. The representation and matching of categorical shape. *Computer Vision and Image Understanding*, 103(2):139–154, August 2006. 36, 63

[160] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. In *International Journal of Computer Vision*, pages 13–32, 1999. 17, 36

[161] D. Sidibe, P. Montesinos, and S. Janaqi. Fast and robust image matching using contextual information and relaxation. In *Proceedings of 2nd International Conference on Computer Vision Theory and Applications*, page 6875, 2007. 56

[162] D. Sidibe, P. Montesinos, and S. Janaqi. Matching local invariant features with contextual information: An experimental evaluation. *Electronic Letters on Computer Vision and Image Analysis*, 7(1):26–39, November 2008. 29

[163] S. N. Sinha, J. . Frahm, M. Pollefeys, and Y. Genc. Feature tracking and matching in video using programmable graphics hardware. *Machine Vision and Applications*, pages 1–11, 2007. 2

[164] S.M. Smith and J.M. Brady. SUSAN: A new approach to low-level image-processing. *International Journal of Computer Vision*, 23(1):45–78, May 1997. 26

[165] N. Snavely, S.M. Seitz, and R. Szeliski. Photo tourism: Exploring image collections in 3d. *ACM Transactions on Graphics*, 25:835–846, 2006. 2, 18, 33, 55, 103

[166] N. Snavely, S.M. Seitz, and R. Szeliski. Skeletal graphs for efficient structure from motion. In *Proceedings of Computer Vision and Pattern Recognition*, pages 1–8, 2008. 55

[167] M.E. Spetsakis and Y. Aloimonos. Closed form solution to the structure from motion problem from line correspondences. In *AAAI Conference on Artificial Intelligence*, pages 738–743, 1987. 34

[168] C. Strecha, R. Fransens, and L. Van Gool. Wide-baseline stereo from multiple views: A probabilistic account. In *Proceedings of Computer Vision and Pattern Recognition*, pages I: 552–559, 2004. 8

[169] Y. Sugaya and K. Kanatani. High accuracy computation of rank-constrained fundamental matrix. In *Proceedings of British Machine Vision Conference*, pages 282–291, 2007. 34

[170] D. Tell and S. Carlsson. Combining appearance and topology for wide baseline matching. In *Proceedings of European Conference on Computer Vision*, page I: 68 ff., 2002. 2, 30

[171] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-Posed Problems*. V.H. Winston, Washington, 1977. 15

[172] B.J. Tordoff and D.W. Murray. Guided-MLESAC: Faster image transform estimation by using matching priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1523–1535, Oct. 2005. 35, 90, 92

[173] P.H.S. Torr and D.W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24(3):271–300, October 1997. 29, 35, 90, 92, 96, 100

[174] P.H.S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, April 2000. 90, 92

[175] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998. 8

[176] R.Y. Tsai. 3-d inference from the motion parallax of a conic arc and a point in two perspective views. In *Proceedings of International Joint Conference on Artificial Intelligence*, pages 1038–1042, 1983. 34

[177] R.Y. Tsai. Estimating 3-d motion parameters and object surface structures from the image motion of curved edges. In *Proceedings of Computer Vision and Pattern Recognition*, pages 259–266, 1983. 34

[178] R.Y. Tsai and T.S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(1):13–27, January 1984. 34

[179] W.H. Tsai and K.S. Fu. Subgraph error-correcting isomorphisms for syntatic pattern recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 13(1):48–62, January 1983. 37

[180] T. Tuytelaars and L. Van Gool. Wide baseline stereo matching based on local, affinely invariant regions. In *Proceedings of British Machine Vision Conference*, pages 412–425, 2000. 2, 7, 38

[181] T. Tuytelaars and L. Van Gool. Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85, August 2004. 28, 38

[182] J.R. Ullmann. An algorithm for subgraph isomorphism. *Journal of the Association for Computer Machinery*, 23(1):31–42, January 1976. 36

[183] O. Veksler. Stereo correspondence by dynamic programming on a tree. In *Proceedings of Computer Vision and Pattern Recognition*, pages II: 384–390, 2005. 31

[184] C.Y. Wang, H. Sun, S. Yada, and A. Rosenfeld. Some experiments in relaxation image matching using corner features. *Pattern Recognition*, 16(2):167–182, 1983. 31

[185] L. Wang, U. Neumann, and S. You. Wide-baseline image matching using line signatures. In *Proceedings of International Conference of Computer Vision*, pages 1311–1318, 2009. 32

[186] J.Y. Weng, T.S. Huang, and N. Ahuja. Motion and structure from two perspective views: Algorithms, error analysis, and error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):451–476, May 1989. 34, 92

[187] M.L. Williams, R.C. Wilson, and E.R. Hancock. Deterministic search for relational graph matching. *Pattern Recognition*, 32(7):1255–1271, July 1999. 37

[188] R.C. Wilson, A.D.J. Cross, and E.R. Hancock. Structural matching with active triangulations. *Computer Vision and Image Understanding*, 72(1):21–38, October 1998. 37

[189] R.C. Wilson and E.R. Hancock. A bayesian compatibility model for graph matching. *Pattern Recognition Letters*, 17(3):263–276, March 1996. 37

[190] R.C. Wilson and E.R. Hancock. Structural matching by discrete relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):634–648, June 1997. 37

[191] A.P. Witkin. Scale-space filtering. In *International Joint Conference on Artificial Intelligence*, pages 1019–1022, 1983. 26

[192] A.K.C. Wong and M. You. Entropy and distance of random graphs with application to structural pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(5):599–609, September 1985. 37

[193] A.K.C. Wong, M. You, and S.C. Chan. An algorithm for graph optimal monomorphism. *IEEE Transactions on Systems, Man and Cybernetics*, 20(3):628–638, May/Jun 1990. 37

[194] J.J. Xiao and M. Shah. Two-frame wide baseline matching. In *International Journal of Computer Vision*, pages 603–609, 2003. 28

[195] J. Xie and H.T. Tsui. Wide baseline stereo matching by corner-edge-regions. In *Proceedings of International Conference on Image Analysis and Recognition*, pages 713–720, 2004. 28

[196] G. Yang, C.V. Stewart, M. Sofka, and C. L. Tsai. Registration of challenging image pairs: Initialization, estimation, and decision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(11):1973–1989, Nov. 2007. 17

[197] B.L. Yen and T.S. Huang. Determining 3-d motion and structure of a rigid body using straight line correspondence. In *Image Sequence Processing and Dynamic Scene Analysis*, pages 365–394, 1983. 34

[198] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 38(4 2006):13, 2006. 17

[199] A. Yilmaz, X. Li, and M. Shah. Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1531–1536, November 2004. 17

[200] Z.Y. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–195, March 1998. 2, 14, 35

[201] Z.Y. Zhang, R. Deriche, O. D. Faugeras, and L. Quan. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1-2):87–119, 1995. 31, 56, 105

[202] Y.F. Zheng and D. Doermann. Robust point matching for nonrigid shapes by preserving local neighborhood structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):643–649, April 2006. 36

[203] Y.T. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.S. Chua, and H. Neven. Tour the world: Building a web-scale landmark recognition engine. In *Proceedings of Computer Vision and Pattern Recognition*, pages 1085–1092, 2009. 103

[204] B. Zitova and J. Flusser. Image registration methods: A survey. *Image and Vision Computing*, 21(11):977–1000, October 2003. 17