Geometry-driven Feature Detection

Changchang Wu

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2011

Approved by:

Marc Pollefeys, Advisor

Jan-Michael Frahm, Co-principal Reader

Svetlana Lazebnik, Reader

Jack Snoeyink, Reader

Frank Dellaert, Reader
Abstract

Changchang Wu: Geometry-driven Feature Detection.
(Under the direction of Marc Pollefeys.)

Matching images taken from different viewpoints is a fundamental step for many computer vision applications including 3D reconstruction, scene recognition, virtual reality, robot localization, etc. The typical approaches detect feature keypoints based on local properties to achieve robustness to viewpoint changes, and establish correspondences between keypoints to recover the 3D geometry or determine the similarity between images. The complexity of perspective distortion challenges the detection of viewpoint invariant features; the lack of 3D geometric information about local features makes their matching inefficient.

In this thesis, I explore feature detection based on 3D geometric information for improved projective invariance. The main novel research contributions of this thesis are as follows. First, I give a projective invariant feature detection method that exploits 3D structures recovered from simple stereo matching. By leveraging the rich geometric information of the detected features, I present an efficient 3D matching algorithm to handle large viewpoint changes. Second, I propose a compact high-level feature detector that robustly extracts repetitive structures in urban scenes, which allows efficient wide-baseline matching. I further introduce a novel single-view reconstruction approach to recover the 3D dense geometry of the repetition-based features.
I would like to dedicate this thesis to my parents and my wife.
Acknowledgments

I would like to thank my advisor, Marc Pollefeys, for supporting me over the years, and for inspiring and guiding my research. I would also like to thank my co-advisor Jan-Michael Frahm for his outstanding mentorship and for being a friend who is always supportive with my work and life. Svetlana Lazebnik gave me many valuable insights to my research which brightened my vision of the big picture. Jack Snoeyink’s insights always help me to locate the critical places that I need to improve, and he gave valuable help on my writing skills. I am grateful to have Frank Dellaert in my committee, and he has been very supportive with constructive insights.

I would like to thank my many friends and colleagues at UNC-Chapel Hill and at ETH-Zürich with whom I have had the pleasure of working over the years. These include Xiaowei Li, Li Guan, Ram Krishan Kumar, Rahul Raguram, David Gallup, Brian Clipp, Christopher Zach, Friedrich Fraundorfer, Megha Pandy, Sudipta Sinha, Seon Joo Kim, Roland Angst, Gabriel Brostow and all the members of the Computer Vision Group at UNC-Chapel Hill and Computer Vision and Geometry Lab at ETH-Zürich.

I would like to thank my friends at UNC-Chapel Hill, Ping Lu, Chen Wang, Xiaoxiao Liu, Hao Xu, Liangjun Zhang, Yuanxin Liu, Xueyi Wang, Jun Huan, Feng Pan, Qi Zhang, and many others, for giving me memorable experiences outside of my office during my PhD years.

Finally, I would like to thank my wife Li Li for being the love and happiness of my life, and for giving me the courage and motivation to finish this thesis.
# Table of Contents

List of Figures ................................................................. x

1 Introduction ................................................................. 1
   1.1 Motivation .............................................................. 1
   1.2 Thesis Statement ..................................................... 3
   1.3 Contribution .......................................................... 3
   1.4 Overview ............................................................... 4

2 Background ................................................................. 6
   2.1 Invariance Based on Local Statistics ............................. 10
   2.2 Invariance Based on Regions ..................................... 14
   2.3 Invariance Based on 3D Scene Geometry ......................... 16
   2.4 Invariance Based on Repetition ................................. 18
      2.4.1 Repetition Detection .......................................... 18
      2.4.2 Repetition-based Features .................................. 22
   2.5 Evaluation ............................................................ 24

3 Viewpoint-Invariant Patch (VIP) Feature ............................ 26
   3.1 Large Scale Reconstruction ....................................... 26
   3.2 Viewpoint-Invariant Patch ........................................ 29
      3.2.1 The VIP Detection ............................................ 30
      3.2.2 Dimensions of a VIP Feature .............................. 31
      3.2.3 Efficient VIP Detection ..................................... 32
List of Figures

1.1 Products using image matching to organize photo collections............ 2

2.1 Structure from Motion (SfM) using invariant features..................... 7

2.2 Feature matching example........................................... 8

2.3 Feature Description.................................................... 9

2.4 Challenges in invariant feature detection and matching.................. 10

2.5 Detection of DoG keypoints........................................... 12

2.6 Harris-Affine region features....................................... 13

2.7 Maximum stable extremal regions.................................... 15

2.8 3D Geometry used to generate projective normalized views............. 17

2.9 Method for detecting repetition...................................... 20

2.10 Repetition patterns difficult for existing detection methods........... 21

2.11 Repetition lattice used as scene recognition........................... 23

3.1 Correspondence of two Viewpoint Invariant Patches(VIPs)................. 29

3.2 The VIPs detected on a 3D model..................................... 31

3.3 VIPs detected on dominant planes.................................... 33

3.4 Texture warping on planar structures.................................. 33

3.5 Wall images for repeatability evaluation................................ 34

3.6 Number of inliers under projective transformations...................... 35

3.7 Re-detection rate of features under projective transformations.......... 35

3.8 Three pairs of point matches given by one VIP correspondence........... 37

3.9 Hierarchical Hypothesis Test (HHT).................................. 38
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.10 Example of distributions of scale and rotation in 3D matching.</td>
<td>39</td>
</tr>
<tr>
<td>3.11 Scene 1: Matching 3D models with $45^\circ$ viewing direction change.</td>
<td>42</td>
</tr>
<tr>
<td>3.12 Scene 2: Matching 3D models with very small overlap.</td>
<td>42</td>
</tr>
<tr>
<td>3.13 Scene 3: Matching 3D models from camera paths crossing at $90^\circ$.</td>
<td>42</td>
</tr>
<tr>
<td>3.14 VIP-based loop closing experiment.</td>
<td>44</td>
</tr>
<tr>
<td>3.15 3D model reconstruction with VIP-based loop closing</td>
<td>45</td>
</tr>
<tr>
<td>3.16 Pose estimation from SIFT-VIP matches.</td>
<td>47</td>
</tr>
<tr>
<td>3.17 Comparison of SIFT matching and SIFT-VIP Matching.</td>
<td>48</td>
</tr>
<tr>
<td>3.18 Illustration of SIFT-VIP match.</td>
<td>49</td>
</tr>
<tr>
<td>3.19 Mobile vision based localization.</td>
<td>50</td>
</tr>
<tr>
<td>3.20 3D model search evaluation.</td>
<td>51</td>
</tr>
<tr>
<td>3.21 Examples of from the 3D model retrieval.</td>
<td>52</td>
</tr>
<tr>
<td>3.22 Loop detection in city-scale data.</td>
<td>53</td>
</tr>
<tr>
<td>3.23 Missing geometric information of typical image features.</td>
<td>54</td>
</tr>
<tr>
<td>4.1 Overview of the repetition analysis in this thesis.</td>
<td>57</td>
</tr>
<tr>
<td>4.2 Example of the proposed repetition detection.</td>
<td>58</td>
</tr>
<tr>
<td>4.3 Drawbacks of local image features.</td>
<td>59</td>
</tr>
<tr>
<td>4.4 Example of vanishing point refinement.</td>
<td>65</td>
</tr>
<tr>
<td>4.5 Example of the detected repeating features and symmetry axes.</td>
<td>66</td>
</tr>
<tr>
<td>4.6 The proposed similarity and quality measurement.</td>
<td>69</td>
</tr>
<tr>
<td>4.7 Repetition region initialization and propagation.</td>
<td>71</td>
</tr>
<tr>
<td>4.8 Example of repetition region decomposition.</td>
<td>73</td>
</tr>
<tr>
<td>4.9 Examples of repetition detection.</td>
<td>74</td>
</tr>
<tr>
<td>4.10 Example of single-feature image retrieval</td>
<td>77</td>
</tr>
</tbody>
</table>
4.11 Evaluation by single-feature image retrieval. .......................... 78

5.1 Repetition intervals give strong clues of 3D geometry. ...................... 81
5.2 Dense reconstruction from repetition. ........................................ 81
5.3 Images used in comparison of dense reconstruction. ......................... 94
5.4 Comparison of different repetition terms. .................................... 95
5.5 Dense reconstruction examples. .................................................. 97
5.6 Ortho-rectified images generated based single view reconstruction. .... 98
5.7 Example of ortho-rectified images. ............................................. 98

6.1 Possible improvements to repetition detection. .............................. 101
6.2 Viewpoint invariant structure based on vertical vanishing points. ......... 103
6.3 Reflectively symmetric features. ................................................ 103
Chapter 1

Introduction

Our eyes constantly obtain information about the world we live in, while our brain is able to efficiently process massive amount of visual data, understand the complicated three dimensional structures of our world, and recognize the scenes we have seen before even when they were observed from different viewpoints. A fundamental problem of computer vision is to match images taken from different viewpoints in order to recover the three dimensional structure or recognizing scenes. This thesis falls in the long line of work to detect invariant image keypoint for the matching task, and focuses mainly on developing invariant keypoints to handle the complicated matching scenarios through the interpretation of underlying geometric information within images, particularly, 3D dense geometry and repetition patterns.

1.1 Motivation

With the wide usage of various digital imaging devices, in our daily life, security, documentation, for military, and many other places, vast amounts of image data are being generated every second, which drives the need for efficient organization of the data through appearance-based matching. Particularly, 3D reconstruction from photo collections or video sequences produces not only compact representations of scenes, but also the geometric relationship between different viewpoints and the 3D structure. The
reconstructed 3D structure can be used for simulation or virtual reality, for example, virtual flight training or remote medical surgery. The recovered motion enables us to navigate photo collections according to their geometric relationship, which has been developed as a public feature in products like Google Maps and PhotoSynth. In addition to user-tags and Geo-locations from user input or imaging devices, appearance-based matching provides an alternative to automatically find possible image matches. For instance, Google Goggles allows a user to search by simply taking a photo with a cell-phone, while the servers will do the matching at the backend. Online-shopping websites can now show products with similar photos to give shoppers an additional way to select products.

Over the past decades, many techniques have been proposed to evaluate the similarity of images and recover 3D geometry from images. Typically, statistics of either global appearance or local interesting keypoints are extracted as a compact representation of images. While global statistics are not robust to viewpoint changes due to the complexity of the view geometry, many local interesting points have been proposed to compensate local perspective distortion, particularly in the family of similarity transformations and affine transformations, and thus achieve more robust features. However, local invariance to viewpoint changes is still limited because of the higher complexity of local perspective
distortions. In terms of feature matching, the larger quantity of local features and the massiveness of the image collections pose additional challenges on the computation cost. The ambiguity in repetitive structures, especially in urban scenes, causes additional problems to the precision of matching.

This thesis contributes algorithms to deal with the challenges in the feature matching problem including large viewpoint changes, repetitive structures, and the large data scale. They will be necessary steps toward the goal of efficient matching of large complicated image datasets, for example, all the images on the Internet, or all the maps.

1.2 Thesis Statement

By exploiting 3D scene geometry and repetition in image(s), projective invariant feature detection can be achieved along with stronger discriminative power, and the geometry-driven features can be used to further improve the accuracy and efficiency of image matching and 3D geometric reconstruction.

1.3 Contribution

My thesis makes the following main contributions.

The Viewpoint Invariant Patch (VIP) feature based on 3D dense geometry.

I introduce a novel class of viewpoint independent local features for robust registration under widely different viewpoints. This approach leverages local shape information to detect interesting keypoints and extract invariant feature descriptors. The novel features are designed to be invariant to 3D camera motion and a single feature correspondence can uniquely define the 3D similarity transformation between two scenes. A novel algorithm is then developed to use the rich information of the new features for efficient image registration, 3D scene alignment and large scale scene reconstruction.
Repetition-based high-level region feature.

I present a robust and efficient framework to analyze large repetitive structures on architectural facades, which essentially extracts features that possibly have high-level meanings. A novel quality measure is proposed to evaluate how image patches fit a repetition interval, based on which an accurate repetition boundary detection is developed. The algorithm is particularly efficient by evaluating repetition and symmetry with adaptiveness to the scale of repetitions. The detected repeating elements are used as high-level compact features for efficient image matching, which provide a new way to evaluate the repeatability of repetition detection.

Repetition-based single-view dense reconstruction.

I propose a dense reconstruction algorithm to recover the 3D details of repeating elements, which extends the repetition-based feature detection. The single-view reconstruction is formulated as an energy minimization problem over the dense pixel correspondences. In order to obtain dense repetitive structures, I develop a novel repetition-based energy term to penalize the 3D structure inconsistency between the dynamically corresponded pixel pairs. The proposed framework is able to enforce high-order geometric constraints in the image domain.

1.4 Overview

The remainder of the dissertation is organized as follows.

Chapter 2 presents the background and fundamental technologies required for invariant feature detection and feature matching. The different families of local invariant features are discussed based on the type of image content interpretation used in feature detection, including region information, 3D scene geometry, and repetition.

Chapter 3 describes a new viewpoint invariant patch (VIP) feature and an efficient 3D registration algorithm to use the proposed feature. Experiments compare the robustness
of VIP with state-of-art image features. The approach is extended to registration of images to 3D models, and efficient search of large scale 3D models datasets.

Chapter 4 introduces a new framework to jointly analyze repetition and symmetry in urban scenes. The robustness of the detection algorithm is evaluated on large datasets. The detected repeating elements are used in image retrieval to evaluate the detection repeatability.

Chapter 5 describes a novel repetition-based single view reconstruction method that recovers the 3D geometry for repeating elements. The new optimization frame automatically balances between photometric consistency, local smoothness and high-order geometric constraints.

Chapter 6 summarizes the contributions, concludes the thesis, and discusses potential improvements and future work.
Chapter 2

Background

Matching images taken from different viewpoints is important for 3D reconstruction and scene recognition, which usually need to deal with arbitrarily taken images (e.g. Figure 2.1). Instead of a direct match of all pixels, state-of-art methods typically first establish the correspondences between some distinctive interest points, then reconstruct camera parameters and scene structures based on projective geometry. To successfully match images with different perspective distortions, interest points need to be extracted such that they are invariant (robust) to viewpoint changes. Hence such interest points are also known as invariant features.

Invariant feature detection is the class of techniques to extract distinctive points, corners, regions, etc, with local properties that are approximately invariant to the projective transformations involved in image formation. Under relatively small viewpoint changes (e.g. 30 degrees), projective transformations at many scene points can often be locally approximated by similarity transformations or affine transformations. By extracting a large number of interest points that satisfy an similarity-invariant property or affine-invariant property, some portion of the extracted points will be invariant to similarity transformations or affine transformations respectively. Actually, similarity-invariant and affine-invariant features satisfy the needs of many matching tasks, thus most of the existing invariant features theoretically target similarity invariance or affine
Figure 2.1: Structure from Motion (SfM) using invariant features. Images taken from different viewpoints are related by their invariant features, from which cameras and 3D model can be recovered based on projective constraints. (best viewed in color)

Invariance. Beyond them, it gets harder to achieve invariance to general projective transformations due to the dimensionality reduction occurring during the projection from 3D scene geometry to 2D images. In the last decade, many invariant feature detectors have been developed for wide-baseline matching and recognition. Figure 2.2 shows examples of similarity-invariant features.

To obtain an invariant description of features for matching, each invariant feature is associated with a surrounding image region that is consistent with its local structure and local projective transformation, so that corresponding features in different images can be transformed to the same normalized view for description. Typically, scale invariant features use scale-dependent circular regions and affine-invariant features use ellipses or parallelograms [Mikolajczyk et al. (2005)]. By normalizing the features according to their different spatial extents and local dominant orientations, invariant feature descriptors are constructed from the normalized views (illustrated in Figure 2.3.) Another research focus is to develop effective feature descriptors. In particular, the descriptor invented by Lowe (2004) for Scale Invariant Feature Transform (SIFT) is one of the most popular descriptors, and uses a 4x4 grid of Gaussian weighted 8-direction gradient histogram. While many descriptors use image gradients histograms, Forssen and Lowe (2007) developed a descriptor to use the shape of detected regions to gain robustness.
Figure 2.2: Feature matching example. SIFT (Scale Invariant Feature Transform [Lowe (2004)]) features are shown on top of the two images on the left. Each feature is displayed as an arrow such that its direction represents feature orientation and its length represents feature size. The right image shows the resulting feature correspondences, which are mostly correct. This pair of images is hard due to relatively large viewpoint change, so there are not so many feature matches. (best viewed in color)

to illumination changes. A performance comparison of many feature descriptors has been presented by Mikolajczyk and Schmid (2005). Winder and Brown (2007) further tried to learn possible good representations of feature descriptors from a training patch database.

Invariant features enable feature-based image matching. With the normalized feature descriptors, feature correspondences between two images can be found by nearest neighbor search in the descriptor space (e.g. Figure 2.2). An additionally often-adopted constraint is a threshold on the ratio between the smallest distance and the second smallest distance, which has been proven effective for filtering the matches that have low probability of being correct [Lowe (2004)]. Typically, the comparison of large image
datasets is time-consuming because the number of features in an image is often large. In feature-based recognition, a vocabulary tree approach is often used by quantizing the descriptors to integers and using them as visual keywords to index images [Nistér and Stewénius (2006)]. Although some accuracy is sacrificed, efficiency can be achieved by replacing direct matching of descriptors with fast indexing-based search.

This thesis addresses the following challenges in feature-based matching.

- **Large Viewpoint Changes.** The changes of local appearances can be too hard to model and the common features detected in different images do not match. For example, the two images in Figure 2.4(a) have a viewing direction difference of 90 degrees.

- **Repetitive Structures**, which is common in urban scenes (e.g. Figure 2.4(b)). Local feature detection results in many similar features. The resulting ambiguous feature matching can easily lead to incorrect image matching.
Figure 2.4: Challenges in invariant feature detection and matching. (a) Large viewpoint changes make it hard to achieve invariance. (b) Repetitive structures make feature matching prone to errors.

- **Large-scale Matching.** When matching with too many images, the large number of local features becomes a problem. Although a speedup can be achieved by indexing with feature-based visual words, the matching results can often be wrong due to the lack of semantic meanings.

The next few sections will discuss some features related to the proposed features of this thesis. Some methods achieve invariance by processing neighboring pixels without interpretation of the underlying structure; other methods achieve improve invariance with the guidance of high-level information, for example, regions, 3D geometry, and repetition, etc. In Section 2.1, 2.2, 2.3 and 2.4, the related feature detectors will be categorized and reviewed based on whether or not and how they exploit the high-level image interpretations.

**2.1 Invariance Based on Local Statistics**

Characteristic scale or affine shape can be defined for an interest point solely based on local statistics without information or interpretation about local 3D structures. Invariance to similarity transformation and affine transformation can then be achieved by normalizing the characteristic shapes of the features.

**Invariance to similarity transformations** is achieved by finding the local characteristic scale and the local dominant orientation. Lindeberg (1993, 1994) studied in
depth the feature scale selection problem based on the Gaussian scale space, and proposed the scale selection at maxima and minima of functions based on combinations of scale-normalized derivatives [Lindeberg (1998)]. These local statistics include normalized Laplace-of-Gaussian (LoG), Difference-of-Gaussian (DoG), and Determinant of Hessian matrix (DoH). Such detectors are also known as blob detectors since they respond strongly to the structures that are darker or brighter than their surroundings. As illustrated in Figure 2.5, Lowe (2004) designed Scale Invariant Transform (SIFT) to use DoG function for keypoint selection, which exploits the constructed Gaussian space to achieve efficiency. To improve the accuracy of feature localization, SIFT applies quadratic interpolation to obtain sub-pixel locations and sub-level feature scales. The Speed Up Robust Feature (SURF) proposed by Bay et al. (2006) detects features by finding DoH maxima. SURF particularly uses integral images [Viola and Jones (2001)] for fast convolution and feature descriptor generation. Improving on the Harris corner [Harris and Stephens (1988)], Mikolajczyk and Schmid (2004) proposed a method that first detects scale-adapted Harris corner points, and then searches for local maxima of Laplace-of-Gaussian (LoG) function to determine the scale of the interest points.

**Invariance to affine transformations** is another heavily investigated area of invariant features. Based on the local characteristic scale of an interest point, a circular region can be upgraded to affine shape according to its local second moment matrix [Lindeberg and Garding (1994)], which is also known as affine blob detection. Harris-affine regions [Mikolajczyk and Schmid (2004)] starts from regions defined by scale-invariant Harris keypoints, and refines affine shapes based on the two eigenvalues of their second moment matrix. Kadir et al. (2004) first define a characteristic scale for an interest point by searching for entropy maximum over scales, and then refine the shape in affine space to obtain salient regions. Tuytelaars and Van Gool (2004) proposed an edge-based region to define characteristic parallelograms for corners structures. From the family of parallelograms defined from a corner and any two points on its two edges, the one with
Figure 2.5: Detection of DoG keypoints [Lowe (2004)]. The left image illustrates the construction of DoG scale space by subtracting Gaussian scale space pyramid. Images are downsampled by 2 when scale doubles. Maxima and minima of the DoG images are detected by comparing a pixel (marked with X) to its 26 neighbors in 3x3 regions at the current and adjacent scales (marked with circles). (Best viewed in color)

extrema of some local statistics is chosen to be the characteristic parallelogram.

Since the shapes of the defined surrounding regions are covariant with the underlying geometry, they are also called affine covariant regions [Mikolajczyk et al. (2005)]. As illustrated in Figure 2.3, the normalized view of a feature can be obtained by transforming ever ellipse to a circle or every parallelogram to a square, then feature descriptors can be constructed in the same way as scale invariant features. Compared to scale-invariant features, affine-invariant features have theoretically stronger invariance to projective transformations, but scale-invariant SIFT often outperforms affine-invariant regions in wide-baseline matching, probably because SIFT has more accurate localization. As a consequence, SIFT has been chosen by many reconstruction systems [Snavely et al. (2006); Li et al. (2008); Agarwal et al. (2009); Frahm et al. (2010)].

Invariance to projective transformations and non-rigid transformation requires us to model more complicated warping than that of similarity transformations or
affine transformations. Ling and Jacobs (2005) developed a geodesic intensity histogram (GIH), a normalized joint distribution of geodesic distance and intensity. The GIH descriptors are invariant to deformation if the scale of the image does not change. Since the geodesic for description is based on the intensity ordering of pixels, GIH is sensitive to noise. GIH is also unable to handle scale changes due to the fixed parameter of the geodesic distance interval. Vedaldi and Soatto (2005) presented a theoretical framework for detecting viewpoint invariant features on non-planar surfaces. They demonstrated the detection of 3D corners by using combinations of multiple affine transformations. Unfortunately, this method has limited applications for wide-baseline matching, because most local structures are locally near planar and violate their assumptions. The self-occlusion problem of non-planar surfaces poses additional challenges.

Obviously both scale-invariant and affine-invariant features have limited robustness to projective transformations since they are based on simplification of projective transformations. Under large viewpoint changes (e.g. Figure 2.4(a)), the corresponding appearance changes of keypoints cannot all be handled, as a result, it may be hard to find enough good matches to recover 3D geometry or recognize scenes. On the other hand, local projective-invariant feature detection is also not mature enough for wide practical use, essentially due to the high dimensionality of projective space. This thesis will focus on the interplay of geometry interpretation and feature detection.
2.2 Invariance Based on Regions

Interpretation of local structures provides additional information for achieving viewpoint invariance. Region extraction or image segmentation groups similar image pixels to regions, and pixels of a region have high probability of belonging to some continuous 3D surfaces. The transformations of such regions can often be approximated by affine transformation, and affine invariance can be achieved though region normalization.

Features can be directly defined by fitting affine shapes to regions. Researchers have been exploring different region extraction methods for feature detection. The Maximum Stable Extremal Region (MSER) method by Matas et al. (2002) thresholds images at different intensity levels to get extremal regions, whose interior pixels are all darker or brighter than the surrounding exterior pixels. It keeps the extremal regions that are maximally stable by comparing with extremal regions of neighboring intensity levels. Ellipses are then fitted to the detected stable regions, and the invariant descriptors are obtained from normalized image patches. To get smaller structures, Perdoch et al. (2007) extended MSER to require regions being only locally maximum stable. Based on watershed segmentation, Deng et al. (2007) use the principle curvature to detect stable regions in scale space with accurate boundaries. Intensity-based regions start from intensity extrema, and find the extremal positions around them along each direction to define a region [Tuytelaars and Van Gool (2004)].

Good region extraction provides a medium-level interpretation of scene geometry, and this information can improve the affine-invariance of feature detection. Mikolajczyk et al. (2005) conducted experiments to compare the performance of many affine covariant regions detectors, where MSER shows better repeatability and stronger invariance to viewpoint changes than other affine-invariant features including salient regions, Harris affine regions, etc. Without proper handling of scales by Gaussian scale space, MSER does have less robustness to scale changes than Hessian-affine regions. To overcome this weakness, Forssen and Lowe (2007) developed an extension of MSER to detect stable
regions in Gaussian scale space.

Texture regions can be used as an affine frame for further invariant feature detection. Schaffalitzky and Zisserman (2001) use texons [Malik et al. (1999)] to segment images into regions that have different texture patterns. Affine normalization is then applied to each texture region, and initial matches can be obtained by texon matching. Further feature detection and matching are then done using local features in the normalized frames.

Similar to the affine invariant regions based on local statistics, region-based affine invariant features have limited robustness to viewpoint changes and limited accuracy of region localization. Because the region extraction in MSER relies on the intensity order of pixels, the region boundaries can be easily affected by noise or illumination changes, and so are the accuracy of the corresponding affine shapes.

Regions can also be defined according to high-level information like repetition. This thesis will particularly explore feature detection based on high-level regions from repetition, and Section 2.4 will discuss the related work.
2.3 Invariance Based on 3D Scene Geometry

Beyond regions, higher level structures, particular local 3D geometry, provide more information on the local projective transformation than that of regions, and allows us to achieve stronger viewpoint invariance.

Many approaches can obtain 3D local geometry efficiently. By using Vanishing Point (VP) detection, simplified planar structures can be reconstructed from a single image along with camera calibration [Caprile and Torre (1990); Cipolla et al. (1999)]. Even when only two vanishing points are detected, a plane can be recovered based on the calibration or the (partially) known camera intrinsics. Digital photos often have EXIF data that can provide relatively good calibrations [Snively et al. (2006); Li et al. (2008); Agarwal et al. (2009); Frahm et al. (2010)]. While considering multiple images or video sequences, 3D structure can also be extracted by stereo reconstruction of those images taken from similar viewpoints. Recently, 3D reconstruction has achieved real time speed by using parallel computation on Graphic Processing Units (GPU) [Gallup et al. (2007)], which enables efficient use of 3D local reconstruction for improved feature detection to handle large viewpoint changes. Depth sensors, such as laser scanners and Time-of-Flight (ToF) cameras, can provide distance data on top of image pixels without any additional computation cost.

3D geometry and camera parameters together provide the projective transformation from 3D space to the image domain. Because the local geometry is consistent in different images, it allows us to generate normalized views by projecting images to tangent planes in 3D space. Koeser and Koch (2007) use depth maps from depth sensors to generate projective invariant normal view for the MSER features in the original images. The local image texture is projected onto a sphere, and the 3D size of a feature is selected according to standard deviation. Rothganger et al. (2006) match MSER features from multiple images, and use sparse 3D reconstruction and projective constraints to generate normalized views of features.
Figure 2.8: 3D Geometry used to generate projective normalized view of features. (a) Koeser and Koch (2007) use a dense depth map. The left image shows one MSER in the image and the depth map; The right image shows the normal view of the feature. (b) Rothganger et al. (2006) use sparse geometry and projective constraint. The left image shows an image with MSER regions, and the right image shows the final 3D features. (best viewed in color)

The straightforward idea here is to recover 3D features that are perpendicular to local surface normals and have some characteristic 3D sizes. Both Koeser and Koch (2007) and Rothganger et al. (2006) use MSER as initial keypoint detector and find 3D features based on it. Koeser and Koch (2007) use standard deviation to determine feature size and it is not accurate under illumination changes. Rothganger et al. (2006) try to find a normalized patch that is consistent with matched features from different cameras, which is not only complicated but also sensitive to errors in the initial detection. The disadvantage is that the normalized views are based on the initial regions, which have limited localization accuracy under large viewpoint changes. This thesis will take a step forward to drop the dependency on the initial feature detection to improve the invariance and accuracy of geometry-based features.

An alternative approach that shares a similar spirit is to re-project an image with a set of possible transformations and combine their normalized features to get more projective invariant features. Morel and Yu (2009) apply a set of affine transformations to globally normalize the image based on camera intrinsics, and then detect SIFT in each normalized frame. This method is able to successfully match images where standard SIFT fails due to large viewpoint changes. Although the additional frames provide more projective invariant features, it also gives more matching outliers. The sampling of the
space of possible transformations also depends on the choice of the parameters.

2.4 Invariance Based on Repetition

High level information such as repetition and symmetry captures the geometric relationship between different parts of an image. As shown in Figure 2.9(a), the repetition captures the 3D information in the slightly different appearance of each element. Similar to region-based features, the repeating elements of repetition define a set of high-level regions. Therefore, repetition-based invariance can combine the power of both 3D geometry and region. This thesis sees repetition detection as a special feature detector and this section discusses some existing repetition detection methods.

Repetition and symmetry are useful in real applications because: 1) They are pervasive not only in urban scenes but also in nature; 2) Many repetitive structures and symmetric structures correspond to interesting high-level semantics. One of the common drawbacks of the detectors in previous sections is that their features do not correspond to any semantic meaningful structures. The existence of repetition and symmetry allows us to find high-level regions that possibly have a richer semantic meaning (e.g. windows, doors). Finding such regions potentially allows many applications such as fast scene recognition.

As a starting point, this thesis only considers urban scenes because urban applications are more interesting. In addition, the repetitions in urban scenes are also regular and they can be processed efficiently. The strong evidences of vanishing points in urban scenes allow image rectification and simplify the detection.

2.4.1 Repetition Detection

The general symmetry concept includes translational symmetry, reflective symmetry, rotational symmetry and their combinations, in which the geometric relationship between
a structure and its symmetric counterpart can be modeled by translation, reflection, rotation, and their combinations respectively. This thesis will normally refer to translational symmetry as repetition and reflective symmetry simply as symmetry if not specified otherwise.

In urban scenes, there are clearly two dominant repetition scenarios: the cases that have repetition in only one direction and the cases that have repetition in a 2D grid. Liu and Collins (1998) classified the possible combinations of symmetries for the two scenarios into 7 frieze groups and 17 wallpaper groups respectively. In this thesis, only a subset of these symmetry groups will be considered, due to the fact that urban structures often do not have reflective symmetry along vertical direction or rotational symmetry. Hence this thesis considers only translational symmetry in horizontal/vertical direction, and reflective symmetry along the horizontal direction.

Given the importance of repetition in urban scenes, several repetition and symmetry detection methods have been developed. Repetition can be analyzed at three different levels: local interest points, dense per-pixel, and model-based structures, which leads the three different types of detection methods as follows:

**Sparse Detection.** This type of methods infer possible repetition based on the matching of local features within a single image, and extract groups of repeated features by growing from random samples of repeated features [Leung and Malik (1996); Schindler et al. (2008); Schaffalitzky and Zisserman (1999); Liu et al. (2004b); Wenzel et al. (2008); Park et al. (2008, 2009)]. In particular, Liu et al. (2004b) and Park et al. (2008, 2009) model the geometry of repetition as a near regular texture (NRT) model, an affine-transformed regular grid that allows small shifts at each node as well as some grid deformations.

These methods do not require rectified images for a 2D repetition grid, because the sampled four features correspond to the hypothesis of a vanishing point pair for rectification. But, in the case of 1D repetition, such methods only work in rectified
Sparse detection is also not able to define optimal regions of repeating elements, because the local feature can be at any location of a repeating element. For example in Figure 2.9(a), the optimal grid should go thought the center of the squares, but the local features forming the grid are not. For large repeating elements (e.g. Figure 2.10), this category of methods generates multiple lattices when using different sets of local features in the images, while they are not able to combine them for optimal region positions.

**Dense Detection.** Dense analysis of repetition finds dense repetitive regions by analyzing all image pixels (limited in affine space). Liu et al. (2004a) determine the boundary of repeating elements by maximizing the local symmetries of repeating elements, including reflective, rotational, and translational symmetry. A correlation-based scoring is used to evaluate the global symmetry. The center of repeating elements are extracted at the position that show maximum reflective symmetry or rotational symmetry [Liu et al. (2004a)]. Figure 2.9(b) shows an result of this method.

Since this method defines the boundaries with a 2D affine grid, it may not be reliable for finding the boundaries along vertical direction in urban scenes, because reflective symmetry along vertical direction and rotation symmetry often does not exist. For example, the two images in Figure 2.10(a) have multiple repeating elements along one
Figure 2.10: Example of common repetition patterns that cannot be handled by existing methods. Both images have repetition along only one direction. (a) Multiple elements whose vertical boundaries need to be found. (b) Depth change of 3D repetitive structures. (Best viewed in color)

direction. It does work for the skyscraper images like Figure 2.11 [Schindler et al. (2008)], which has a dense grid of small windows that are rotationally symmetric. This method also assumes any good matching as part of the repetition, which has ambiguity between real repetition and homogeneous region.

Another limitation of this type of method for real urban scenes is due to its model of strict regular grid and the use of correlation. Finding peaks through correlation can be sensitive to depth change and noises. Real urban repetition often have 3D depth changes from indentation or columns (e.g. Figure 2.10(b)). Therefore a more robust matching function should be used. This thesis will use a scale-adaptive SIFT-based matching to achieve robustness to those local details as well as noise. Sparse detection from local features does not have this problem because it does not evaluate dense matching.

**Model-based Detection.** Model-based repetition detection makes additional assumptions about the shape of the repeating elements and searches for those specific structures in rectified images. Korah and Rasmussen (2007, 2008) assume the repeating elements to be rectangular and uses a Markov Chain Monte Carlo method to extracts them based on the edge segments. Korah and Rasmussen (2007) use to group rectangular windows.
Müller et al. (2007) use a grammar database that defines various rectangular shapes. They use dense mutual information evaluation to obtain repetition intervals, and find different shapes of structures based on edge segments. Their assumptions are quite strong and often not completely valid in urban scenes, for example, curved structures are common.

The limitations of the above repetition detection are rooted in their strong assumptions. In order to handle more general repetitions, it is important to keep the assumptions as loose as possible. Specifically, this thesis will only employ an assumption of weak reflective symmetry on the shape of the repeating elements, and the proposed detection algorithm is able to handle a wider range of repetition scenarios including large repeating elements that repeat along only one direction.

### 2.4.2 Repetition-based Features

Although repetition detection has been explored extensively, few work has used repetition in recognition or matching. One reason, I believe, is that real world repetition is complicated, and different methods use different assumptions and work only in special cases. Only by using as few assumptions as possible and solving more general cases, can repetition-based features be truly useful.

Like all other features, to define repetition-based, the boundaries for repeating elements must be selected with strong characteristic properties, for example, at local maximum of some function. Otherwise, the boundaries from different images will be different, and the features will not match. Existing techniques have various limitations. Sparse detection is not able to define region for 1D repetition, and not able to give optimal unique lattice for 2D repetition. Model-based methods have the advantage of finding correct boundaries but requiring the modeling of various structures, thus they are not too general. Dense-detection works for the cases that have reflective symmetry along vertical direction or rotational symmetry.
Based on the model of Liu et al. (2004a), Schindler et al. (2008) normalize all rectangles in its detected 2D repetition grid to square, learn a median representation called *motif* for each repetition grid, and match them to register images to database. The detection works in this scenario because the windows of skyscrapers are small and rotational symmetric. However the same method does work for 1D repetition like Figure 2.10(a).

To overcome the limitations of current methods, this thesis proposes a new method to determine the boundaries based repetition along only one direction. Similar to the idea of Liu et al. (2004a), the boundaries along horizontal direction are obtained by maximizing reflective symmetry. For the boundary detection along the direction without repetition, a novel repetition quality function is introduced to evaluate repetition for each repetition interval with a suppression of integer multiples of repetition.

An interesting issue in repetition-based feature is whether to use a single description for a group of repeating elements or use a separate descriptor for each element. Schindler et al. (2008) compute a combined representation descriptor based on median tiles to capture the common appearance. Another scenario this thesis will explore is to compute a descriptor for each repeating element in a repetition region. The local irregularities inside each element are potentially discriminative for scene recognition by differentiating
2.5 Evaluation

Given multiple images of a same scene taken with different viewing conditions, a good invariant feature detector should produce a high percentage of repeatedly detected features for its practical application in wide-baseline matching or recognition. Repeatability is affected by the robustness of the feature selection criterion as well as the accuracy of feature localization. First, the same features corresponding a common structure should be robustness re-detected under different viewing conditions. Second they should be assigned to correct spatial extents that are consistent with the geometric transformations, so that invariant feature description can be retrieved.

Many work uses a repeatability score to evaluate the performance of feature detectors [Schmid et al. (2000); Mikolajczyk et al. (2005)]. Given a pair of images, the repeatability score is defined as the ratio between the number of correct correspondences and the smaller of number of regions in the common area of the image pairs. The correctness of feature correspondence is defined according the overlapping of the spatial extent by using a known homography transformation to relate the image pair. Mikolajczyk et al. (2005) give a thorough evaluation many affine invariant features w.r.t. viewpoint changes, scale change, image blur, JPEG compression, lightening change. Similarly, Kadir et al. (2004) use the ratio of number of correct region matches and the number of expected region matches from a database.

\[
\text{repeatability} = \frac{\# \text{ correct correspondences}}{\# \text{ features in common area}} \tag{2.1}
\]

Another evaluation method is by the correctness of direct feature matching. Compared to the evaluation of repeatability, this evaluation is closer to real application because its feature correspondence does not require any knowledge about the ground
truth image warping. Again, the correctness of a feature match is determined by the overlapping according to the known transformations. The precision and recall measure for feature matching is defined as Equation 2.2. Typically, precision and recall are all depending on parameters, and precisions increases and recall drops when stronger parameters are used in matching. Mikolajczyk and Schmid (2005) use 1-precision and recall to evaluate different feature descriptors with different matching strategy and parameters. The same method can also be used to evaluate feature detectors.

\[
\text{precision} = \frac{\# \text{ correct matches}}{\# \text{ matches}} \quad \text{recall} = \frac{\# \text{ correct matches}}{\# \text{ correct correspondences}} \quad (2.2)
\]

Features can also be evaluated by their performance in other applications. For example, image retrieval performance of features can indicate their goodness for recognition tasks.

In this thesis, I'll also evaluate the performance of my proposed features by their repeatability and their performance in scene matching.
Chapter 3

Viewpoint-Invariant Patch (VIP) Feature

The perspective distortions under different viewpoints makes wide-baseline image matching challenging. Under the assumption of local planarity, the local transformations from one image to the other are projective transformations decided by the camera parameters and the 3D structures. Such a projective transformation can often be locally approximated by a similarity transformation or an affine transformation, but the simplified feature detection is essentially limited to relatively small viewpoint changes. This chapter proposes a new Viewpoint Invariant Patch (VIP) feature by exploiting scene geometry to handle the challenges posed by large viewpoint changes.

3.1 Large Scale Reconstruction

Large viewpoint changes are common problems in large scale reconstruction. Basically, every pair of matched images gives some constraints to the 3D reconstruction, and it is likely that certain key constraints rely on the matching of views taken from very different viewpoints. In recent years, there have been significant research efforts in fast large-scale 3D scene reconstruction from video [Pollefeys et al. (2008)] and from Internet photo collections [Agarwal et al. (2009); Frahm et al. (2010)], where the speedup is achieved by exploiting parallelism of compute clusters or Graphics Processing Units (GPU). However, it is still impossible to easily generate full city-scale reconstruction
purely by structure from motion (SfM). Reconstruction from video sequences is a differential technique which accumulates errors over time due to its differential nature. In urban modeling for example, a video’s path often crosses at intersections where the viewing direction differs by about $90^\circ$ (see Figure ??(a)), a single reconstruction often generates un-aligned double representations due to the errors. To avoid accumulated errors, the reconstruction system must recognize previously reconstructed scene parts and enforce correct alignment constraints to the reconstruction. In the case of Internet photo collections, the reconstruction is often biased to popular viewpoints where most of the photos are taken, and failures in matching different viewpoints may result in disjointed models instead of a single complete model.

Large-scale reconstruction naturally will generate multiple 3D models for many reasons: 1) The data might be too big for a single reconstruction task. 2) A single reconstruction can be inaccurate for having too much accumulated errors. 3) It is unable to match some images that have very different viewpoints or different viewing conditions. Consequently, to tackle the large scale reconstruction problem, we need to not only find matches from image subsets or video segments that are taken from different viewpoints, but also to align the 3D reconstructions from different image subsets.

State-of-art invariant features can handle a certain range of viewpoint changes, and relatively complete reconstruction can be achieved by feature matching, given enough photos to cover the viewpoints in between the largely different viewpoints and to provide necessary links/constraints to connect different parts of a 3D scene. Many feature detectors have been developed for robust wide-baseline matching in the last decades. It is clear that improving the robustness of features to viewpoint changes will improve the completeness of the 3D reconstruction. One of the most popular local invariant feature is Lowe’s SIFT keypoints [Lowe (2004)], which is invariant to scale changes and rotation. SIFT basically approximates the local projective transformation by similarity transformation. Affine covariant features go beyond to achieve invariance to affine
transformations [Mikolajczyk et al. (2005)]. Obviously, these approximations are limited because the full projective transformation is more complicated. The feature introduced in this thesis goes beyond affine invariance to robustness to projective transformations.

Large-scale reconstruction requires the alignment of 3D models from multiple sub-reconstructions. Similarly to image matching, the 3D point matches can be hypothesized from image feature matching. Li et al. (2008) use these feature matches to estimate 3D similarity transformations between separate reconstructions and merge them to generate larger reconstructions.

A related problem is the alignment of 3D laser scan models, which typically relies more on the geometry than on the image content. The typical alignment algorithm is Iterative Closest Point (ICP) [Zhang (1994)], which computes the alignment by iteratively minimizing the sum of distances between closest points. As an application in SfM, Zhao et al. (2005) applies ICP to align 3D point clouds from SfM to 3D sensors data and refine camera poses. The initial approximate scene alignment for ICP can also be achieved by matching geometric features such as spin images [Johnson et al. (1999)]. Vanden Wyngaerd et al. (1999) proposed an invariant geometric feature by matching bitangent curve pairs from images, and Vanden Wyngaerd and Van Gool (2002) then extended the work to symmetric characteristics of surface patches. The matching of these geometric features generates relative precise initialization for ICP. On the contrary to the aforementioned methods, King et al. (2005) employs SIFT keypoints in original images to find the initial alignment for 3D laser scan models.

As a hybrid method to utilize both image and geometry, the proposed feature in this chapter achieves improved viewpoint invariance. The detected features are described based on image content for feature matching and associated with characteristic local 3D geometries to enable efficient 3D model alignment.
Figure 3.1: Two corresponding VIPs. The purple and green view frustums are original camera poses. Red view frustums are viewpoint normalized cameras. Lower left and right show patches in the original images while center patches are the ortho-textures for the feature rotationally aligned to the dominant gradient direction.

3.2 Viewpoint-Invariant Patch

Although image features have limited invariance to viewpoint changes, 3D reconstruction can be achieved for images subsets that share similar viewpoints. The limitation of pure image based features is due to the missing of 3D geometry and the approximation of the underlying projective transformation. This thesis proposes a Viewpoint Invariant Patch (VIP) feature to exploit the partial 3D reconstructions to guide feature detection for improved invariance. As illustrated in Figure 3.1, orthogonal views of same 3D points on different 3D models will be the same when scale and orientation are properly selected.

Sharing the similar spirit, Koeser and Koch (2007) obtain depth maps from depth sensors and generate projectively invariant views for MSER features detected in the original images. The local image texture is projected onto a sphere, and the 3D size of a feature is selected according to standard deviation. Rothganger et al. (2006) match MSER features from multiple images, and generate normalized views of features based on sparse 3D reconstructions and projective constraints. In comparison, I apply more robust DoG filtering technique to select invariant keypoints and their scales in 3D. The
determination of 3D feature shape during the detection stage avoids the error-prone refitting of feature shapes.

VIPs are essentially projective invariant image features extracted from multiple images, a set of images whose dense geometry can be relatively easily reconstructed. Since the detection is independent of the original camera viewpoint, VIPs can be used to robustly and efficiently align 3D models of the same scene from video taken from significantly different viewpoints. The remainder of this chapter will discuss the proposed invariant feature technique and its applications in 3D matching and location recognition.

### 3.2.1 The VIP Detection

In VIP detection, the original images are undistorted to generate fronto views according to local scene planes or on local planar approximations of the scene. Conceptually, for every point on the surface we can estimate the local tangent plane’s normal and generate a texture patch by orthogonal projection onto the plane. Within the local ortho-texture patch, a keypoint is generated if the point corresponds to a local DoG Extremum in scale space. The keypoint orientation in the tangent plane is then extracted from local gradients and a SIFT descriptor is generated accordingly. Using the tangent plane avoids the poor repeatability of interest point detection under projective transformations seen in many popular feature detectors [Mikolajczyk et al. (2005)].

VIP detection is basically a process that factors out the original viewpoints. Similar to the normalization of image patches according to scale and orientation performed in SIFT and normalization according to ellipsoid in affine covariant feature detectors, VIP detection can be seen as a Viewpoint Normalization. The VIP detection for each 3D point can be divided into the following two steps:

1. **Project the textured 3D model** into a virtual orthographic camera with viewing direction parallel to the local tangential plane’s normal. The resolution of the orthographic camera is selected to keep the original pixels as much as possible.
2. **Extract the VIP descriptor** from the orthographic patch projection. Invariance to scale changes is achieved by normalizing the patch according to local ortho-texture scale. Feature selection is done by DoG local extrema suppression like SIFT [Lowe (2004)]. VIP orientation in 3D is found based on the dominant gradient direction in the ortho-texture patch.

Figure 3.1 demonstrates the effect of viewpoint normalization. The patches shown as squares are the normalized image patches. The normalized image patches of a matched pair are similar despite significantly different original images due to the largely different viewing directions.

### 3.2.2 Dimensions of a VIP Feature

The geometric dimensions of VIPs are obtained by transforming the keypoints in ortho-texture patches from their virtual camera coordinate systems to world coordinate systems of the 3D models. A VIP feature is fully described by \((x, \sigma, n, d, s)\) where

- \(x\) is its 3D position,
- \(\sigma\) is the patch size in 3D,
- \(n\) is the surface normal at this location,
• $d$ is texture’s dominant orientation as a vector in 3D, and
• $s$ is the SIFT descriptor that describes the viewpoint-normalized patch.

Here I work with the dense models that are reconstructed from video sequences by using SfM and multi-view stereo methods, but VIP detection is equally applicable to textured 3D models obtained using LIDAR or other sensors. Given two reconstructed models of a same scene, we can find their alignment, by matching the SIFT-descriptors of as well as the geometric information of the VIPs.

### 3.2.3 Efficient VIP Detection

In general planar patch detection needs to be executed for every pixel of the image to compute the ortho-textures. Each pixel $(x, y)$ together with the camera center $C$ defines a ray, which is intersected with the local 3D scene geometry. The point of intersection is the corresponding 3D point of the feature. From this point and its spatial neighbors we can get an (approximate) tangential plane at the point, which for planar regions coincides with the local plane. The extracted plane can then be used to select VIP keypoints and compute their description with respect to this local tangent plane. This method is generally valid for any scene.

VIP detection for a set of points that have same surface normal (either on a same plane or on different planes) can be efficiently processed in a single pass. Considering these VIPs, the image coordinate transformations between them are simply 2D scaling and translation. If we project all these points onto a single plane and detect image features in its orthogonal view, the original VIPs can be recovered by first transforming from the plane, to their local coordinate frames, and then to the world coordinate system.

Figure 3.3 shows an example of detecting VIPs on dominant planes. The planes here compensate for the noise in the reconstructed model, and improve VIP localization. Figure 3.4 shows an example of a viewpoint normalized facade.
Figure 3.3: VIPs detected on dominant planes.

Figure 3.4: Original image (left) and its normalized patch (right) based on local plane fitting of local 3D structure. The approximated warping allows robust descriptor generation at most locations, which are locally approximately planar. Indeed, it is not truly viewpoint invariant for structures where the surface normals are inconsistent with the plane (such as the indentations and staircases).

The accuracy of VIP detection is indeed limited by the quality of dense reconstruction we can recover. First, 3D models typically have many errors especially for homogeneous regions, but this can be partially handled by the proposed plane fitting. Second, even perfect reconstructions do not have all the details that are seen from a different viewpoint. Third, the ortho-rectification is based on approximated planar structures that does not handle nonrigid transformations, for example, the indentations in Figure 3.4.
3.2.4 Evaluation of Repeatability

To measure the repeatability of VIP features, I conduct an experiment similar to the evaluation done by Mikolajczyk et al. (2005). This experiment is done jointly with Xiaowei Li and Brian Clipp.

We use the wall dataset shown from Mikolajczyk et al. (2005) for the evaluation experiment. As shown in Figure 3.5, this dataset is a sequence of images of a brick wall taken with increasing angles between the optical axis and the wall’s normal. Each of the images of the wall has a known homography to the first image, which was taken with image plane fronto-parallel to the wall. Using this homography we extract a region of overlap between the first image and each other image. We extract features in this area of overlap and evaluate two performance measures: the number of inlier correspondences and the re-detection rate, for a number of feature detectors. The number of inliers is the number of feature correspondences which fit the known homography. Re-detection rate is the ratio of inlier correspondences found in these overlapping regions to the number of features found in the fronto-parallel view. The number of inliers is shown in Figure 3.6 and the re-detection rate is shown in Figure 3.7.

Figure 3.6 demonstrates that VIP matching generates a significantly larger number of inliers over the wide range of angles than the other detectors. The other detectors we compare to are SIFT [Lowe (2004)], Harris-Affine [Mikolajczyk and Schmid (2004)], Hessian-Affine [Mikolajczyk and Schmid (2004)], Intensity Based Region (IBR) [Tuytelaars and Van Gool (2004)], Edge Based Region (EBR) [Tuytelaars and Van Gool (2004)], and Maximally Stable Extremal Region (MSER) [Matas et al. (2002)]. The
proposed VIP feature also has a significantly higher re-detection rate than the other
detectors as seen in Figure 3.7. This high re-detection rate demonstrates the power of
geometry-guided feature detection on ortho-rectified views. Even under large viewpoint
changes which often result in projective transformations between images that affine
transformations cannot accurately approximate, VIP performs well.

3.3 3D Model Matching with VIPs

The previous section addressed the VIP detection problem, and any 3D model can now
be represented as a set of VIPs. This section discusses how to apply VIPs to match and
align two 3D models. In fact, VIP-based matching is not limited to 3D model alignment, Section 3.6 will show how to utilize the 3D model matching as a basic function in loop detection for large scale structure from motion and 3D model based location recognition.

Like other feature based matching, putative VIP matches can be obtained by matching the feature descriptors, for example, by the nearest neighbor matching. After obtaining all the putative matches between two 3D scenes, it is then possible to find an optimized scene transformation from the 3D transformation hypotheses given by VIP correspondences. Since VIPs are viewpoint invariant, given a correct camera matrix and 3D structure, we can expect the similarity between correct matches to be more accurate than a transformation derived from viewpoint dependent matching techniques.

The rich geometric dimensions of VIP feature allow recovery of the 3D similarity transformation between two scenes from a single match. The ratio of the scales of two VIPs expresses the relative scale between the 3D scenes; the normals and orientations of the VIP pair determine the relative rotation. The translation between the scenes can be obtained from relative feature positions after compensating the scaling and rotation. The scaling and rotation needed to bring corresponding VIPs into alignment is uniform for all the correct correspondences.

A hierarchical method is proposed in this section to estimate the 3D similarity transformation between two scenes from putative VIP matches. Each single VIP correspondence delivers a unique 3D similarity transformation, and so hypothesized matches can be tested efficiently. Furthermore, the rotation and scaling components of the similarity transformation are same for all inlier VIP matches and they can be tested separately and efficiently with a voting scheme.

### 3.3.1 3D Similarity from a Single VIP Match

The scaling, rotation and translation of a 3D similarity transformation can be fully decided by six constraints from 3 pairs of 3D point matches. As illustrated in Figure 3.8,
a single VIP correspondence gives exactly 3 pairs of point matches, which determines a unique 3D similarity transformation. Specifically, given a VIP correspondence of \((x_1, \sigma_1, n_1, d_1, s_1)\) and \((x_2, \sigma_2, n_2, d_2, s_2)\), the scaling between them is given by \(\sigma_s = \frac{\sigma_1}{\sigma_2}\). The rotation between them satisfies \((n_1, d_1, d_1 \times n_1)R_s = (n_2, d_2, d_2 \times n_2)\). The translation between them is \(T_s = x_1 - \sigma_s R_s x_2\). A 3D similarity transformation can be formed from the three components as \((\sigma_s R_s, T_s)\).

Figure 3.8: Three pairs of point matches given by one VIP correspondence can uniquely determine a 3D similarity transformation.

### 3.3.2 Hierarchical Hypothesis Test

The scaling, rotation and translation of a VIP correspondence is covariant with the global 3D similarity transformation, and the local feature scale change and rotation are the same as the global scaling and rotation. The proposed Hierarchical Hypothesis Test (HHT) solves these components separately and hierarchically, which increases alignment accuracy and reduces the search space for the correct similarity transformation. The initial version of the matching algorithm was done jointly with Xiaowei Li, and I then improve this algorithm with a guided feature matching.

As illustrated by Figure 3.9, the 3D similarity estimation in this chapter is done hierarchically in three steps starting from a set of putative VIP correspondences. First, each VIP correspondence is scored by the number of other VIP correspondences that support its scaling. All VIP correspondences which are inliers to the VIP correspondence
Putative Feature Match

Find the most supported scale and inliers

Find the most supported rotation and inliers

Find the most supported translation and inliers

Refine Transformation from all inliers

Use guided match to find more matches

Scale-guided Feature Match

Scale/rotation-guided Feature Match

Figure 3.9: Hierarchical Hypothesis Test (HHT). The left half of the figure illustrates the proposed hierarchical hypothesis test that finds scale, orientation, and translation in order. The right half of the figure shows an improvement to the algorithm by re-matching features with the guiding transformation constraints from the scaling and rotation.

with most support are selected to calculate a mean scaling and outliers are removed from the putative set. Second, the same process is repeated with scoring based on support for each correspondence’s rotation and the putative set is again pruned of outliers. Third, the same process is repeated scoring according to translation to determine the final set of inlier VIP correspondences. A nonlinear optimization is applied to find the scaling, rotation, and translation using all of the remaining inliers.

To handle the case where there are not enough feature matches, I extend the HHT algorithm by searching for additional matches with the intermediate partial 3D similarity transformation, which is illustrated in Figure 3.9. This works particularly for urban scenes because there are many regular repetitive structures and correct scales and rotations can often be hypothesized from incorrect matches.

Figure 3.10 shows that the distributions of scale ratios and rotations from single putative VIP correspondences have enough reliable hypotheses for the estimation. Similarly, the experiments also demonstrate that, in the translation step, the inlier match
Figure 3.10: Example of distributions of scale ratios and rotations from a set of low inlier-ratio putative VIP matches. The log scale ratio and rotation are measured relatively to the final transformation, so 0 correspond to the final result. The scaling and rotation can be accurately extracted from their separate histograms. In the rotation histogram, each sphere center corresponds to a 3-vec rotation, and the size of the sphere corresponds to the number of supports the rotation gathers.

set filtered by the scale and rotations steps has only a small portion of outliers.

It is worth noting that in the 3D matching experiments all possible hypotheses are exhaustively tested, which is efficient because each VIP correspondence generates one hypothesis and the whole sample space is linear w.r.t. the number of putative VIP matches. The method described above can be easily extended to a RANSAC scheme by checking only a small set of hypotheses. It is known that the RANSAC requires $N = \frac{\log(1-p)}{\log(1-(1-e)^s)}$ random samples to get at least one inlier sample free of outliers with probability $p$, where $e$ is ratio of outliers and $s$ is number of matches to establish a hypothesis [Hartley and Zisserman (2000)]. In the VIP case, $s = 1$, so that $N = \log_e (1 - p)$. For example, when the outlier ratio is 90%, 44 random samples are enough to get at least one inlier match with probability 99%. This leads to an even more efficient estimation of 3D similarity transformations. However, especially in wide baseline matching where a high outlier ratio is expected, an exhaustive test of all transformation hypotheses is the most reliable
and it is still efficient.

### 3.3.3 3D Model Matching Experiments

In this subsection, I apply VIP-based 3D matching on several stereo reconstruction models to demonstrate reliable surface alignments. The 3D alignment can serve as a merging tool in large-scale reconstruction to create a complete model after reconstructing different parts of a scene separately.

I choose three urban scenes for the experimental evaluation of VIP matching. For each scene, two image sequences were collected to have different camera viewpoints and camera paths, and two corresponding sets of depth maps are reconstructed by using SfM and stereo. Camera positions were defined relative to the pose of the first camera in each sequence. The first scene, shown in Figure 3.11, consists of two facades of a building reconstructed from two different sets of cameras with significantly different viewing directions (about 45°). The cameras moved along a path around the building. One can observe reconstruction errors due to the occlusion of the trees in front of the building. An offset was added to the second scene model for visualization of the matching VIPs. The red lines connect all of the inlier correspondences. Rotation and scaling have been corrected using transformations calculated using VIPs in this visualization. HHT determines 214 inliers out of 2085 putative matches. The number of putative matches is high because putative matches are generated between all features in each of the models.

Figure 3.12 shows the second scene consisting of two local scene models, with camera paths that intersect at an angle of 45 degrees. The overlapping region is a small part of the combined models, and it is seen from very different viewpoints in the two videos. Experiments show that the proposed 3D model alignment method can reliably detect the small common surface and align the two models.

Figure 3.13 shows the third scene where two models reconstructed from camera paths that cross at an almost 90° angle. There is a significant difference in viewing directions
between the two sets of cameras. The experiment shows that we can match VIP features reconstructed from widely different viewpoints.

Table 3.1 shows the quantitative results of HHT. Note that scale and rotation verification remove a significant portion of the outliers. For evaluation I measure the distances of the matched points after the first 3 stages and after the nonlinear refinement. To measure the quality of surface alignment, I check the point distances between the overlapping parts of the models. The models are reconstructed with scale matching the real building and so the error is given in meters. The statistics in Table 3.1 demonstrate the performance of HHT.

<table>
<thead>
<tr>
<th>Scenes</th>
<th>#1, Figure 3.11</th>
<th>#2, Figure 3.12</th>
<th>#3, Figure 3.13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing direction change</td>
<td>45°</td>
<td>45°</td>
<td>90°</td>
</tr>
<tr>
<td># of putative matches</td>
<td>2085</td>
<td>494</td>
<td>236</td>
</tr>
<tr>
<td># of inlier matches</td>
<td>1224/654/214</td>
<td>141/42/38</td>
<td>133/108/101</td>
</tr>
<tr>
<td>Error after first 3 stages</td>
<td>0.0288</td>
<td>0.0230</td>
<td>0.114</td>
</tr>
<tr>
<td>Error after nonlinear ref.</td>
<td>0.0128</td>
<td>0.018</td>
<td>0.0499</td>
</tr>
<tr>
<td>Surface alignment error</td>
<td>0.434</td>
<td>0.135</td>
<td>0.629</td>
</tr>
</tbody>
</table>

Table 3.1: HHT details in three experiments. The fourth row shows the number of inliers in each stage of HHT. The alignment errors are median errors in meter. The larger surface alignment errors, compared with the feature alignment errors, are due to the large amount of noises in stereo reconstructions.

Additionally I compare VIP-based alignment with the alignment using SIFT feature and MSER feature. For SIFT and MSER, the 2D feature locations are projected to the 3D model surfaces to get 3D points. The putative match generation for them is the same as the VIP matching since they are all described by SIFT descriptors. Then a Least-square method [Umeyama (1991)] and RANSAC are applied to evaluate the 3D similarity transformation between the point matches. Table 3.2 shows the comparison between SIFT, MSER and VIP. The results show that VIP can handle large viewpoint changes for which SIFT and MSER do not work.
Figure 3.11: Scene 1: Matching 3D models with 45° viewing direction change.

Figure 3.12: Scene 2: Matching 3D models with very small overlap.

Figure 3.13: Scene 3: Matching 3D models from camera paths crossing at 90°.
<table>
<thead>
<tr>
<th>Scene 1</th>
<th>SIFT</th>
<th>MSER</th>
<th>VIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Feature(M1/M2)</td>
<td>8717/12244</td>
<td>2254/3410</td>
<td>5947/5553</td>
</tr>
<tr>
<td>#Putative Matches</td>
<td>1600</td>
<td>420</td>
<td>2085</td>
</tr>
<tr>
<td>#Inlier Matches</td>
<td>176</td>
<td>22</td>
<td>214</td>
</tr>
<tr>
<td>Successful</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scene 2</th>
<th>SIFT</th>
<th>MSER</th>
<th>VIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Feature (M1/M2)</td>
<td>12951/16664</td>
<td>4071/5024</td>
<td>9015/4828</td>
</tr>
<tr>
<td>#Putative</td>
<td>641</td>
<td>67</td>
<td>278</td>
</tr>
<tr>
<td>#Inlier</td>
<td>11</td>
<td>0</td>
<td>203</td>
</tr>
<tr>
<td>Successful</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scene 3</th>
<th>SIFT</th>
<th>MSER</th>
<th>VIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Feature (M1/M2)</td>
<td>2363/733</td>
<td>1128/342</td>
<td>2713/1804</td>
</tr>
<tr>
<td>#Putative</td>
<td>131</td>
<td>0</td>
<td>90</td>
</tr>
<tr>
<td>#Inlier</td>
<td>12</td>
<td>0</td>
<td>61</td>
</tr>
<tr>
<td>Successful</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 3.2: Comparison with SIFT and MSER in the 3 scenes. SIFT and MSER work in the first scene but fail in the other two. It can be seen that VIP also gives the highest rate of inlier number to feature number in the first one. It is worth noting that VIP work in scene where there is a 90 degree viewing direction change.

### 3.4 Loop Closing with VIPs

The advantages of VIP for wide baseline matching are perhaps best demonstrated by a large scale reconstruction. Incremental reconstruction of video sequence often has accumulated error, such that two reconstructions of the same place seen twice are not aligned correctly. By finding the transformation that compensates the misalignment and enforcing the transformation, the misalignment errors in 3D reconstruction can be corrected. This experiment is conducted together with Brian Clipp.

We collect video of a building with footprint approximately 37 by 16 meters where the camera’s path completes the loop by crossing at an angle of approximately 90 degrees. Image matching across this wide angle using previous methods is difficult or impossible. However, using VIP patches we are able to complete the loop, generating an accurate 3D point model of the building. The general loop closing actually requires detecting loop hypotheses in advance before loop closing, I will discuss it in Section 3.6.
Figure 3.14: (a) The VIP matches found in camera path circling building. VIPs are only extracted between frames where KLT features could not be tracked in between. Note the matching features at either end of the sequence where the loop completes. (b) The first and the last frame in video circling building.

Our reconstruction is done in three steps. First we estimate the camera path using SfM with KLT [Lucas and Kanade (1981)] feature measurements, bundle adjusting the result. Figure 3.15(a) shows the scene points and camera path before applying the VIP correspondences. Then we extract VIP correspondences between key frames of the initial reconstruction and recover the 3D similarity transformation needed for compensating the accumulated error, which is shown in Figure 3.14. Using the VIP correspondences we compensate for this error by linearly distributing it through the camera poses in the sequence and the 3D feature positions. We add the VIP features to the set of 3D features and measurements from the first bundle adjustment and ran the bundle adjustment again. Figure 3.15(b) shows the final model with the completed loop, where the double representation problem is resolved.

3.5 Pose Estimation with VIPs

This section proposes an algorithm to register an image to a 3D model by matching the SIFT features in the image with the VIP features on the 3D model and find out the camera pose. This algorithm can be used either as a single step of incremental
To match SIFT features in an image with VIP features on a 3D model, the SIFT features need to be fronto-parallel (or close to) to the VIP features in the model. This holds true only for a fraction of features whose plane normals are accidentally parallel to the camera viewpoint. Consider such a match of VIP feature with size $S$ and a SIFT feature with scale $s$. The distance $d$ between the camera center and the plane can be computed from the scale ratio of the matched features:

$$d = f \frac{S}{s},$$  \hspace{1cm} (3.1)$$

where the focal length $f$ of the camera can be taken from the EXIF data of the image or from camera calibration. The rotation around the principal axis be recovered by aligning
the dominant gradient direction of the image patch to the VIP orientation. Estimating from a single match is not reliable, but more accurate camera pose can be obtained from a set of putative SIFT-VIP feature matches. Figure 3.16(b) shows an example of camera pose estimated from SIFT-VIP matches.

For all other features the corresponding image areas need to be warped to their orthogonal views, so that they approximately match the canonical form of the VIP features. For each VIP feature on the 3D model the corresponding image region in the image is determined by projecting the VIP region onto the image plane. Next a homography transform $H$ is chosen to warp the image region to the canonical form of the VIP feature with

$$H = R + \frac{1}{d}TN^T,$$  

(3.2)

where $R$ and $T$ are rotation and translation from the estimated camera pose to the virtual camera of the VIP feature and $N$ is the normal vector of the VIP plane in the coordinate system of the estimated camera. Finally projectively invariant image feature in the warped image area is found by using DoG detector.

Figure 3.16 and Figure 3.17 show two examples of SIFT-VIP pose estimation, where I obtain a large number of feature matches from an initially small set of matches. Figure 3.17 also presents the comparison with the standard SIFT-SIFT matching. While the standard SIFT feature matching fails to handle the large viewpoint changes, SIFT-VIP matching generates sufficient correct feature matches for camera pose estimation.

Figure 3.18 illustrates the warping of image patches used by SIFT-VIP matching. In this experiment, the initial 105 SIFT-VIP matches are mostly on one plane, and they are extended to 223 on the 3 scene planes. Figure 3.18(b) shows several examples of the correspondences between warped SIFT patches and orthographic VIP patches. Significant perspective distortions are compensated by VIP-based warping, and robust matching is achieved.

Although the pose estimation algorithm is based on the strong fronto-parallel as-
Figure 3.16: Pose estimation from SIFT-VIP matches. (a) Initial SIFT-VIP matches. As expected, most matches are on the fronto-parallel plane (the left image is query image and the right image is the texture of the 3D model). (b) Camera pose estimated from SIFT-VIP match (red camera). (c) The resulting set of matches established by warping the image areas to fronto-parallel views. The initial set of 17 matches is extended to 92 correct matches. Many matches are established on the other plane, too.

This assumption, the robustness of SIFT descriptor to small viewpoint changes can significantly compensate the inaccuracy in the estimated camera poses. The simple pose estimation algorithm works impressively well.

### 3.6 Location Recognition with VIPs

This section describes a VIP-based location recognition system. VIP features extracted from 3D models are used as visual keywords to index the Geo-locations, and the location and camera pose of query images can be efficiently recovered. The proposed system works for querying of both 3D models and single images by using VIP-VIP and VIP-SIFT match respectively (e.g. Image query illustrated in Figure 3.19). Experiments in this Section are done jointly with Frederich Fraundorfer who provide the vocabulary tree implementation based on the work in Fraundorfer et al. (2007).
Figure 3.17: Comparison of the standard SIFT-SIFT matching and the proposed SIFT-VIP method. (a) SIFT-SIFT matches. Only 10 matches could be found, most of them are mis-matches. (b) Initial 25 SIFT-VIP matches, which are mostly correct. (c) The final 91 extended feature matches. (d) The SIFT-VIP matches in 3D showing the estimated camera pose (red).

3.6.1 3D Model Search System

Large scale recognition system is feasible with keyword-based indexing scheme rather than direct feature matching. Due to the high computation cost, direct matching with all database images works for only small datasets [Robertsone and Cipolla (2004); Shao et al. (2003b); Zhang and Kosecka (2006)]. Schindler et al. (2007) demonstrate a city-scale location recognition system based on vocabulary tree [Nistér and Stewénius (2006)].

The reconstructed 3D models along with their known location information comprise the database for location recognition. Since the 3D models are associated with ground truth Geo-coordinate from GPS, each VIP feature has its own Geo-location, Geo-orientation, and Geo-size. Compared with the typical location recognition system that searches 2D image database for locations by image matching, the 3D models allows us to take advantage of the robust VIP features for better quality, and the efficient
Figure 3.18: Illustration of SIFT-VIP match. (a) SIFT-VIP matches and estimated camera pose for a scene with 3 planes. (b) Examples of warped SIFT patches and orthographic VIP patches. From left to right: Extracted SIFT patch from query images, warped SIFT patch, VIP patch of 3d model. The VIP patches are impressively well aligned to the warped SIFT patches, despite the inaccuracies of the camera pose.

geometric verification for improved speed.

Similarly to the vocabulary tree approach of Nistér and Stewénius (2006), we quantize each feature descriptor a visual word with a hierarchical vocabulary tree, which was trained on Internet image datasets. Each 3D model is represented as a document vector over the visual vocabulary. Given a query model or image, the distances between the query document vector to all document vectors in a database are computed. All the distances are obtained efficiently thanks to the indexing structure of vocabulary tree and the sparseness of document vector. The database images are ranked by the distance, and the images with the smallest distances are reported as matches.

Given the top matches from the visual word scoring, we apply geometric verification by using the methods presented in Section 3.3 and 3.5. The putative feature correspondence can now be efficiently determined simply from visual word indexing. This visual-word-based matching requires only $O(n)$ time where $n$ is the number of features.
The visual word description is designed to be compact for large scale problem. The plain visual word database size is $DB_{inv} = 4fI$, where $f$ is the maximum number of visual words per model and $I$ is the number of models in the database. The factor 4 comes from the use of 4 byte integers to hold the model index where a visual word occurred. If we assume an average of 1000 visual words per model, a database containing 1 million models would only need 4GB of RAM. In addition to visual words we also need to store the 2D coordinates, scale and rotation for the SIFT features and additional 3D coordinates, plane parameters and virtual camera for the VIP features, which still allows to store a huge number of models in the database.

### 3.6.2 3D Model Search Evaluation

This section demonstrates the performance of the proposed 3D model search system. The video data used to generate the 3D model database was acquired by a car driving through a city. Two cameras were mounted on the roof of the car, of which one was pointing straight sideward and the other one was pointing forward in a 45° angle. The fields of view of both cameras do not overlap but as the system is moving over time the captured scene parts will overlap. To retrieve ground truth data for the camera motion
the image acquisition was synchronized with a highly accurate GPS-inertia system. Accordingly we know the location of the camera for each video frame.

We create a 3D model database represented by VIP features from the side camera video, and we then query with the video frames captured by the forward camera, which are represented by SIFT features. The database contains 113 3D models, and we run the query experiments with 2644 images. The top 10 matches from the query are shown in Figure 3.20 and some successful query examples are given in Figure 3.21. It can be seen that success ratio is high and the 3D model search is promising.

Figure 3.20: 3D model search evaluation. The 3D model search results are visualized by plotting lines between frame-to-model matches. The red markers are query camera positions and green markers are the 3D model positions in the database. The top 10 matches within a 10m range are drawn as blue lines. Note that the identical camera paths of the forward and side camera are shifted slightly to make the matching links visible. Each match gives a match hypothesis that is further tested by geometric verification.

As an application of the 3D model search system, we conduct a large-scale loop detection experiment. A sequence of 4895 3D models that has a loop in Chapel Hill, NC, USA is selected for this experiment. As shown in Figure 3.22, 241 tentative matches
Figure 3.21: Examples of from the 3D model retrieval. The left images give the query image from the forward cameras, and the right images are the retrieved 3D models. The last one is an example of cellphone image query.
from voting are found with 4 of them corresponding the ground truth loop, and further geometric verification by 3D model matching is able to verify the loop hypotheses.

Figure 3.22: Loop detection in city-scale data. The camera poses of the image sequence are plotted in green, and the red crosses are the loop hypotheses. The image on the left shows the entire sequence, and the image on the right is a close up view of the right top loop.

3.7 Further Discussions

In this chapter, I propose a novel image-based feature, Viewpoint Invariant Patch (VIP), and investigate the applications of VIP in 3D model matching, pose estimation, location recognition, and loop detection. The evaluation of VIP demonstrates a promising improvement over other image features for robust and accurate 3D model alignment. VIP feature allows scene alignment from only a single VIP-VIP correspondence and pose estimation from a single VIP-SIFT correspondence. The simplicity of VIP along with the independence of different component of 3D transformation empowers a new efficient hierarchical hypothesis test (HHT) method. The comparison of the proposed 3D matching with the state-of-art image features demonstrates its superior robustness to large viewpoint changes in a variety of challenging scenes. The 3D matching serves
as a loop closing tool to find 3D transformation for compensating accumulated errors.

The accuracy of the VIP detection depends on the accuracy of local scale selection and the quality of 3D geometry. In theory, images can always be reprojected to the orthogonal frame based on accurate 3D scene geometry, but 3D scene geometry from stereo reconstruction or other sources often contains a lot of errors. Currently I apply a local plane fitting to extract surface normals and warp textures, and I also discard the features on noisy surfaces. The limitation is that certain types of scenes (such as bushes, trees) do not have any reliable geometry for VIP detection, thus a pure VIP-based system may fail for them. For applications that need to handle such scenarios, VIP should be coupled with other local features for the matching.

VIP detection takes advantage of 3D geometries not only to achieve projective invariance but also to gain discriminative power. Compared to typical image features that normalize different regions to the same shape for description, VIP features are composed with rich geometric information that boosts their discriminability in 3D matching. Figure 3.23 illustrates the ambiguity for typical image features that VIP can deal with.

Although VIP features are only demonstrated on dense reconstruction of video se-
quences, similar concepts apply to 3D geometry recovered by other methods, for example, dense 3D modeling of photo collections [Goesele et al. (2007); Frahm et al. (2010)]. Baatz et al. (2010) unwarp facade images based on simple planar reconstruction from vanishing points, where the features in the rectified images are similar to VIP except that they have slightly less geometric information. Similar to the proposed VIP, Eric R. Smith (2010) detects features based on laser scan model for scene matching.

As discussed when I explain Hierarchical Hypothesis Test (HHT), strong experimental evidences shows that the incorrect putative matches often provide correct scaling and rotation information. This phenomenon can be explained by the pervasive repeated structures in urban scenes, that is, the repetition in urban scenes can give strong guidance to the feature matching problem. In the next chapter, I will show that repetition can also guide feature detection and enable us to extract compact high-level features.
Chapter 4

Repetition Region Feature

Repetitive and symmetric structures pervade urban environments by design. As a key component of urban scene geometry, repetition and symmetry should be exploited in feature detection and 3D reconstruction.

This chapter presents a novel robust and efficient framework to analyze large repetitive structures in urban scenes. A particular contribution of the approach is that it finds the salient boundaries of the repeating elements even when the repetition exists along only one direction. The outline of the approach is as follows. Vanishing point detection and rectification of a perspective image is obtained by maximizing a symmetry measure of both edges and repeated features. Candidate repetition regions are obtained based on the coexistence of feature repetition and symmetry axes. To evaluate the repetition quality of an image patch w.r.t a given repetition interval, I introduce a novel measure that suppresses integer multiples of repetition intervals, and determine the region boundary along the non-repeating direction based on the change of such repetition quality. Experiments demonstrate the robustness and repeatability of the proposed repetition detection. I further utilize such detected repeating elements as features for scene recognition. Chapter 5 will propose a repetition-based dense reconstruction method to recover the 3D geometry for repeating elements.
4.1 Reconstruction with Repetition Presence

Repetition presents ambiguity in feature-matching and reconstruction. However, repetition also gives us the opportunity to achieve a better understanding of image content and better reconstruction. The general concept of symmetry includes translational symmetry, reflective symmetry and rotational symmetry, and their combinations. Here I will refer to translation symmetry as repetition and reflective symmetry simply as symmetry unless otherwise specified.

4.1.1 Repetition Challenges in Reconstruction

The ambiguity caused by repetitive structures needs to be modeled explicitly for accurate reconstruction. As mentioned in the background chapter, feature matching between images is typically done as nearest neighbor search with a ratio test that suppresses ambiguous matching. With the presence of repetitive structures, many similar features will be detected repeatedly. The standard ratio test suppresses the feature matches on the repeating structures. Even if some matches pass the ratio test, they are not guaranteed to correspond to the exact same element, because similar views can be generated for each repeating element. If the ratio test is not enforced, the inlier ratio will be low with many such incorrect matches, which makes geometric reconstruction error-prone and inefficient.

Repetitions can be processed at different stages of 3D reconstruction, including fea-
Figure 4.2: Example of repetitive structures detected by the proposed method. Note that the vertical boundaries are selected automatically to distinguish between the interesting elements and high frequency repetition of the roof.

feature detection and matching, incremental SfM, and further 3D refinement. During the feature detection step, we can specifically detect the repeating regions, and use the repetition region features for matching. For example, Schindler et al. (2008) detects 2D grids of repeating features based on the method of Liu et al. (2004a). Their grid structure then served as feature matching constraints for registering facade images. During the incremental reconstruction stage, for instance, Zach et al. (2010) use a high level loop constraint to detect and correct the false matches causes by repetitive structures. Additionally, 3D reconstruction can be refined by enforcing repetitions.

4.1.2 Features from Repetition

As learned from VIP detection, the knowledge about scene geometry can provide powerful guidance for feature detection. Urban buildings often consist of a hierarchy of repetitions and symmetries at different scales (e.g. Figure 4.2). Particularly, most of the repeating elements on facades (such as doors and windows) are reflectively symmetric by themselves. This chapter will exploit the coexisting repetition and symmetry in urban scenes to develop a new compact feature.

One of the most obvious advantages of repetition is to allow us to extract the elements that are closer to human interpretation than the local features, such as SIFT, MSER,
Figure 4.3: Example of SIFT features to illustrate the common drawback of local features: Local features do not have strong high-level meanings (e.g. door, window, etc), and there are normally a large number of local features.

The verification of repetition enables us to extract regions that belong to high level structures for urban scenes. The comparison between Figure 4.2 and Figure 4.3 illustrates the power of repetition for feature detection. In Figure 4.2, the detected regions correspond to the high-level elements in the design of those buildings. Additionally, the low number of detected features enables efficient scene matching.

The symmetry and repetition patterns together with the appearance of the repeating/symmetric elements provide a strong characterization of the scene. Given that, particularly for urban scenes, the symmetries and repetitions of a scene describe its high-level structure, they can facilitate the wide baseline image matching [Agarwal et al. (2009); Frahm et al. (2010)]. The reliable boundaries of the detected repeating elements and the symmetric structure can be used as compact image features for effective recognition. The known scene symmetries and repetitions allow us to automatically extract the facade grammars [Ripperda and Brenner (2009); Wonka et al. (2003); Müller et al. (2006)] as well as the semantic parsing of the images. Additionally, the known structure of the facades allows to regenerate facades based on their grammar or to compensate for occlusions by replacing occluded parts through their symmetric or repetitive equivalent.

Reliably detecting repetitions and extracting their boundaries is a challenging task.
Even though the images of planar facades can be rectified to a frontal view by using its vanishing points, the appearance of repeating elements may still significantly change, due to reflections and occlusions. In addition, the perspective change for non-planar structures on the facade plane severely distorts the local symmetries. A particularly challenging scenario that draws attention of this thesis is where the large repetitive structures repeat only along the horizontal direction (e.g. Figure 4.2). Homogeneous regions, edges along vanishing directions, and high-frequency repetitions cause additional ambiguities in choosing meaningful boundaries for the repeating elements. To reliably detect such the boundaries, it is necessary to distinguish between regions that belong to different repetition groups (with different repetition intervals).

### 4.1.3 Repetition for Finer Reconstruction

The presence of repetition provides geometric constraints to refine 3D reconstruction. For example, reconstruction errors can be corrected by replicating the correct counterparts from the repeating elements. This can be used either as a constraint in reconstruction or as a post-processing filtering stage. One application is stereo reconstruction of urban scenes with many repetitive 3D structures, where the reconstruction can be improved by enforcing 3D similarity between the repeating elements.

Even for a single image, the perspective distorted repetition and symmetry encodes 3D geometry information to allow single view reconstruction. If the 3D structure is mostly on a plane, the perspective distortion can be modeled by a planar homography, which allows recovery of the simple planar geometry [Hong et al. (2004)]. However when the repetitive structures are not a single plane, a simple homography is no longer sufficient to model all the perspective distortions. The transformations of different 3D points are actually determined by the interplay of the camera and the 3D structure. Chapter 5 will propose a novel technique to model 3D repetition as energy minimization and densely reconstruct 3D non-planar repetitive structures.
4.2 Related Work

Repetitions are usually hypothesized from the matching of local image features, and are often detected as a set of sparse repeated features by growing or tracking from the small sets of initial features towards their immediate spatial neighbors [Leung and Malik (1996); Schindler et al. (2008); Loy and Eklundh (2006); Wenzel et al. (2008); Schaffalitzky and Zisserman (1999); Park et al. (2008)]. Liu et al. (2004a) extract repeating elements by dense pixel matching. Their method determines the boundary of repeating elements by maximizing global symmetries. A limitation of their method is the requirement of a 2D repetition grid, which is not always available in urban environments. Beyond maximizing local symmetries I will separate different repetition groups by evaluating the local repetition quality conditionally for different repetition intervals with a suppression to ambiguous repetition interval.

Model-based repetition detection assumes some shape prior (e.g. rectangular windows) and search for these specific structures. Korah and Rasmussen (2007, 2008) assume the repeating elements to be rectangular and extract them based on the edge segments in the rectified images. Their assumptions are often not completely valid in urban scenes because curved structures are common. Müller et al. (2007) use a semantic-based database to model a set of different repeating structures and search for the patterns that fit the database elements. Indeed such a method can handle many types of repeating structures while equipped with a large database, it is still not general. In this thesis, I will choose a less restrictive assumption by only requiring the repeating elements to be approximately symmetric.

Repetition and symmetry are essentially similar concepts, and they can often be solved in very similar frameworks (e.g. Turina et al. (2001); Wenzel et al. (2008)). The proposed repetition detection also handles both, but in a joint fashion. The coexisting repetition and symmetry are analyzed together to define the boundaries of the repetition regions that are assumed to be weakly symmetric.
4.3 Repetition in Urban Scenes

This section lists some observations that I employ as assumptions in repetition analysis.

1. Dominant repetition(s) are mostly along the strong vanishing point direction(s) with equal 3D spacing. This gives us the opportunity to refine the vanishing point(s) based on repetition;

2. While many existing approaches require repetition in 2D grids, many building facades lack vertical repetitions and symmetries. The proposed approach here is specifically developed to handle this case.

3. Repeating architectural elements typically also exhibit reflective symmetry around vertical axes. Symmetry axes occur at twice the frequency of the repetition, in the middle and in between repeated elements. This provides a simple way to define the vertical boundary between repeating elements (up to a two-fold ambiguity). Observations show this principle is mostly satisfied according. Note that the rectangular structure assumption adopted for example by Korah and Rasmussen (2007) and Müller et al. (2007) is a special case of this assumption.

4. 3D structures repeating at different depths show slightly different steps in an image due to projective geometry, which sometimes even causes self-occlusions.

4.4 Sparse Repetition and Symmetry Detection

This section first introduces an improved method for vanishing point detection, and then discusses the sparse detection in the rectified images.
4.4.1 Vanishing Point Refinement

Accurate vanishing point (VP) detection is important in this framework because the repetitions are assumed to be along vanishing directions. Inaccuracy in VP locations will interfere with the finding of the optimal repetition interval and symmetry axes since the pairwise distances between the matched features change gradually. Here, I first apply the cascaded Hough transform [Tuytelaars et al. (1998)] to compute the vertical and one or more horizontal vanishing points from edge pixels as initialization.

I propose a VP refinement by maximizing the overall symmetry in the entire image using both edges and features. Given a pair of horizontal and vertical vanishing points, $VP_H$ and $VP_V$, a homography $T = T(VP_H, VP_V)$ can be determined to rectify the image. I define the transformation to preserve the original resolution around image center. By matching SIFT [Lowe (2004)] features extracted in the original image along both vanishing directions and keeping the closest matches (closest in the image), three sets of feature pairs can be extracted. Let $R_H$ be the horizontal repetition, $R_S$ be the horizontal symmetry, $R_V$ be the vertical repetition.

Consider a set of point pairs $R \in \{R_H, R_V, R_S\}$ in the original image and a transformation $T$. Let $X^T(R)$ denote the distribution of their horizontal distances after rectification, $Y^T(R)$ the distribution of their rectified vertical distances, and $C^T(R)$ the distribution of the horizontal coordinates of their rectified midpoints. Typically in urban scenes, there exist only a few strong symmetry axes and repetitions intervals. Correspondingly, we expect to see only a few strong peaks in the distribution of $X^T(R_H), Y^T(R_V)$ and $C^T(R_S)$ minimizing the entropies of those distributions. This work optimizes the rectification by minimizing the summed entropy, so that the vanishing directions are better aligned with repetition directions and symmetry axes.

Let $H$ be the entropy function $H(X) = -\sum_x p(x) \log p(x)$. It can be proven that $H(X^T(R_H))$ and $H(Y^T(R_V))$ are invariant to any affine transformations, and $H(C^T(R_S))$ is invariant to transformation in the form of $\begin{pmatrix} a & 0 \\ b & e \end{pmatrix}$. However, such an affine ambiguity
can be resolved by using the point distances in the direction perpendicular to the repetition or the symmetry, $Y^T(R_H)$, $X^T(R_V)$ and $Y^T(R_S)$, because they are only invariant to transformations in the form of \( \begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix} \).

Although entropy captures the uncertainty in distributions, it does not rely on the actual position of the distribution, that is, any distribution $D(x + t)$ has the same entropy for any constant $t$. To guarantee a distribution $D(x) \in \{Y^T(R_H), X^T(R_V), Y^T(R_S)\}$ to be zero-mean, I use entropy of the symmetrized distribution $\hat{H}(D(x)) = H(D(x) + D(-x))$ instead. In addition, the edge information can be incorporated similarly. Let $G_H$ and $G_V$ be the two set of edge segments corresponding to the two vanishing points, $Y^T(G_H)$ and $X^T(G_V)$ are equivalent to case of repetition.

By assuming the different distributions independent of each other, their joint distributions can be ignored and the quality of repetition and symmetry can be measured by the summed entropy as follow,

\[
Q(VP_H, VP_V) = H(X^T(R_H)) + H(Y^T(R_V)) + H(C^T(R_S)) + \hat{H}(Y^T(R_H)) \\
+ \hat{H}(Y^T(R_S)) + \hat{H}(X^T(R_V)) + \hat{H}(Y^T(G_H)) + \hat{H}(X^T(G_V)),
\]

and the vanishing points $VP_H$, $VP_V$ are then recovered at the minimum

\[
(VP_H, VP_V) = \underset{VP_H, VP_V}{\text{argmin}} Q(VP_H, VP_V).
\]

This method still optimizes both vanishing points if vertical repetition $R_V$ is missing because the horizontal symmetry constraints the vertical vanishing points. Liu and Collins (2001) pointed out the potential of using symmetry in rectification, and it was used by Schindler et al. (2008) to rectify facade images. In comparison, the repetition-based refinement in this work can deal with more general cases.

I apply the gradient descent method to solve Eq 4.1. Figure 4.4 gives an example of the VP refinement. Experiments show the VP refinement reduces the drift of the
Figure 4.4: An example of vanishing point refinement. It can be seen that the refinement produces distributions that have stronger peaks and lower entropies.

estimated repetition interval when the initial detection is not accurate enough. I find that the vanishing points from pure edges are also sensitive to small radial distortions, while the refinement proposed here can balance between the edge segments and feature matches to find better vanishing points.

4.4.2 Repetition Intervals and Symmetry Axes

With the detected VPs, the original images are rectified to be fronto-parallel, and afterwards upright SIFT features are extracted, which is similar to the concept of U-SURF [Bay et al. (2006)]. The single fixed orientation for all features is a natural choice given that the rotation is compensated through the rectification. Hence, the feature matching does not suffer under descriptor changes from the erroneous orientation detections. Recently, Baatz et al. (2010) have exploited the upright SIFT for efficient location recognition.
Figure 4.5: An example of detected repeating features and symmetry axes. Only the feature pairs for the strongest repetition interval are displayed. It can be seen that the symmetry axes are repeating at half the interval of the window repetition.

By matching features along the horizontal and vertical directions, histograms of possible horizontal repetition intervals, vertical repetition intervals, and symmetry axes can be obtained from the features pairs. Note that the feature matching for reflective symmetry detection is the mirrored matching approach of Loy and Eklundh (2006). Local maxima are extracted from histograms to get a set of repetition intervals \( \{I\} \) and symmetry axes \( \{\Lambda\} \). In this work, I do not try to recover vertical symmetries since they typically do not show up in urban scenes. I also skip any repetition intervals that are smaller than 30 pixels to focus only on large repetitive structures.

For each repetition interval, the bounding box of their feature matches roughly gives the repetition regions. Unfortunately these regions are often inaccurate, by either missing part of the actual repetition region or including non-repetition regions from bad feature matches. To find the correct region, a dense measurement should be adopted.

Considering the local symmetry axes of repeating elements and the symmetry axes between neighboring repeating elements, these symmetry axes repeat with an interval of half of the structure size. This is an outcome of assumption #3. See Figure 4.5 for an example. Selecting the horizontal boundaries at the position of those symmetry axes maximizes the local symmetry of the repeating elements.
4.5 Dense Detection

This section presents the new method for dense detection of repetition regions. I initialize repetition regions based on the sparse repetition and symmetry, and refine them by dense matching and propagation.

4.5.1 Evaluation of Repetition Quality

In order to define salient boundaries for repeating elements, it is necessary to densely evaluate how well each location fits a given repetition interval. The dense matching should be able to identify non-repeating regions, while being robust to lighting changes and other small variations. In addition, it is also important to suppress spurious support that could come from homogeneous regions and repetitions at higher frequencies (for example, the roof eaves in Figure 4.5 has a repetition interval of \( \frac{1}{5} \) of the window distances). I first match image patches to evaluate the similarity between any two locations. In order to be invariant to scale changes and different rectification, the patch size \( W_I \) is selected proportionally to the repetition interval \( I \). Through some experiments I have determined that \( W_I = \frac{I}{4} \) consistently provides good results.

To provide robustness to small variations and lighting, I compute the SIFT descriptors to evaluate the patch similarity, which is efficiently done on GPU [Wu (2007)]. Given a repetition interval \( I \) and a location \( x \), the distance between the normalized SIFT descriptor at \( x \) and \( x + I \) is denoted as \( D_R(x, I) \). Similarly, the matching distance w.r.t. a symmetry axis \( \Lambda \) is denoted as \( D_S(x, \Lambda) \).

It can be verified that if an element is repeated many times, and if \( I \) is a valid repetition interval \( 2I, 3I, \ldots \) will also be valid. Therefore, we are interested in the smallest valid repetition interval and want to suppress its multiples. It is therefore important to verify that for a repetition interval \( I \), the repetition intervals \( \left\{ \frac{I}{2}, \frac{I}{3}, \ldots \right\} \) are not valid. In fact, it suffices to verify \( \frac{I}{p} \) with \( p \) prime numbers. In practice, verifying the first
few prime numbers is sufficient (up to 7 in this thesis). Notice that this automatically covers the issue of homogeneous regions as those would show repetition for any interval. Inspired by the widely used ratio test in SIFT matching, I choose a set of translations \( T_I = \{0, \pm \frac{I}{2}, \pm \frac{I}{3}, \ldots \} \), compute the set of matching distances for them \( V = \{D(x, I + t) | t \in T_I\} \), and define the following quality function

\[
f(x, I) = \min(\alpha \frac{V(2) + \sigma}{D(x, I) + \sigma}, 1),
\]

where \( V(2) \) is the second smallest distance in \( V \). Parameter \( \alpha \) is for truncating the quality so that the quality function evaluates to 1 when \( D(x, I) \) is significantly smaller than \( V(2) \) (I use \( \alpha = 0.7 \) as typical for the SIFT ratio test [Lowe (2004)]). Adding a small constant \( \sigma \) reduces noise when all distances are small, which can be seen as a variance in the SIFT distance distribution (I use \( \sigma = 0.1 \)). It can be seen that \( f(x, I) > \alpha \) only when \( I \) is a local minimum. Note that the definition applies to either single patch or a patch set.

In feature matching, matches with a small ratio between the smallest and the second smallest distance often have a high probability of being a correct match [Lowe (2004)], and the ratio test filters out ambiguous and poor matches. Similarly, a high \( f(x, I) \) indicates the opposite. The proposed quality function penalizes both noise and ambiguous high-frequency regions (e.g. Figure 4.6).

As the evaluation of single patches is noisy, similarity and quality measures are defined to evaluate repetition for image regions. The distance between two patch sets is defined as its the median distance: \( D_R(X, I) = \text{median}\{D_R(x, I) | x \in X\} \). For the quality function, I use a pre-learned threshold\(^1\) \( T \) to select an inlier patch set \( X_I = \{x | x \in X, D_R(x, I) < T\} \) of an image region \( X \), and use the inlier set to evaluate the repetition quality as \( F(X, I) = f(X_I, I) \). Experiments show that this quality function is robust to

\(^1\) \( T = 0.64 \) learned from the distributions in labeled images is used in this thesis.
Figure 4.6: The proposed similarity and quality measurement. The colored-patches in the left image gives the distance map (The visualization uses $1 - \frac{1}{2}d^2$ to map distance $[0, \sqrt{2}]$ to $[0, 1]$). The colored patches in the right image give the quality map and the curve gives the quality for each row. The distance map shows good matching for the grass, roof eaves and the horizontal edges, but the quality function is able to penalize them. The black lines in the right image give the places where the vertical boundaries are detected.

outliers even for low inlier ratios. In order to avoid unreliable evaluation from noise, the quality measure will be set to 0 when the inlier ratio is less than 20%.

To correctly handle the first and last element of a repetition sequence, the bidirectional distance and quality are measured instead as

$$D^+(X, I) = \min(D(X, I), D(X, -I)),$$
$$f^+(X, I) = \max(f(X, I), f(X, -I)).$$

In the following, when referring to distance map and quality map, I will be talking about $D^+$ and $f^+$ unless specified otherwise. Similar to $F(X, I)$, only inliers $X^+_I = \{x | x \in X, D^+(x, I) < T\}$ are considered while evaluating $f^+$ for a patch set.

4.5.2 Repetition Region Detection

It is a natural choice to select the horizontal boundaries of the repeating elements according to the detected repeating symmetry axes (e.g. Figure 4.5) since such boundaries
**Algorithm 4.1** The Repetition Detection Algorithm

1: Detect vanishing points and rectify image.
2: Find sparse repetitions \( \{I\} \) and symmetry axes \( \{\Lambda\} \) (Section 4.4)
3: for each un-processed repetition interval \( I \) do
4:   Find sets of repeating symmetry axes \( \{\Lambda_I\} \)
5:   while \( \{\Lambda_I\} \) is not empty do
6:     Find a consecutive set of axes with gap \( \frac{I}{2} \) or \( I \)
7:     Initialize region from the symmetry axes (Section 4.5.2)
8:     Propagate the region by matching at interval \( I \)
9:     Find region boundaries and sub-regions. (Section 4.5.2)
10:    Search and analyze vertical repetition.
11:   Find further decompositions of regions. (Section 4.5.3)
12:   Save detected repeating elements
13:   Remove covered symmetry axes from \( \{\Lambda_I\} \)
14: end while
15: Mark repetitions that can be modeled as **processed**
16: end for

generate elements with maximal local symmetry. As illustrated in Figure 4.7, the initial horizontal extent of a repetition region is defined by a group of symmetry axes that have horizontal distances of \( \frac{I}{2} \) or \( I \) with each other. The initial vertical range is chosen to cover the matched feature pairs whose line segments intersect with the symmetry axes.

The full detection of repeating elements is more complicated than that of symmetry because of the larger repetition count; it requires propagation and verification in order to get the full correct regions. Due to perspective change and noise, not all symmetry axes can be perfectly detected from initial feature matching. The initialization in the previous step is likely to miss some parts of the repeating regions. To extend the repeating region horizontally, propagation is taken with steps of \( \pm I \) or \( \pm \frac{I}{2} \) to match a rectangular region of width \( I \) at the desired location. If the inlier ratio for both the left and right \( \frac{I}{2} \) are high enough, the region is extended by the step size. Given the large window sizes, it is actually not necessary and not optimal to match all the pixels. I typically use an adaptive sparse grid of locations instead, like in Figure 4.7.

Without using vertical repetition, the vertical boundaries are selected based on the quality evaluation of scanlines. Basically, the regions that lack salient repetitions for
interval $I$ are excluded by simply setting boundaries where the quality of rows $F^+(X, I)$ drops from 1 to $\alpha$ (e.g. roof eaves and grass in Figure 4.6). With the determined vertical boundaries, multiple repeating elements can be defined by separating the regions at the rows that do not have salient repetition.

After the horizontal repetition analysis, sparse vertical repetition analysis is applied in the detected region, and the boundaries for the vertical repetition are then detected from the vertical repetition quality map in a similar way. The region detection process now decomposes the initial region to sub-regions that have both horizontal and vertical repetition, and sub-regions that have only horizontal repetition.

### 4.5.3 Region Decomposition

As shown in Figure 4.8, one possible mistake of initializing from symmetry axes is the over-grouping of different repeating elements that have the same repetition interval. In this case, the matching distances between neighboring elements will change over the entire horizontal range, and get particularly large at the places where the repetition elements change. I design a continuation score function to evaluate how the repetition continues over a range of 4 times the repetition interval. Similarly for the quality function, the continuation score from $X$ to $X + I$ is defined based on the ratio of
distances as

$$
Cont(X, I) = \min\left(\frac{D(X^+_I - I, I)}{D(X^+_I + I, I) + \sigma}, \frac{D(X^+_I + I, I)}{D(X^+_I, I) + \sigma}\right) + \sigma.
$$

The ratio threshold $\alpha$ used in the repetition quality functions is basically a closeness threshold. For regions with good continuation of repetition, the continuation score should be in $(\alpha, 1/\alpha)$. In places where the repetition pattern changes to something else, there will be much smaller continuation score. The local minima along the horizontal direction that satisfy

$$
Cont(X, I) < \min(Cont(X - I, I), Cont(X + I, I), \alpha)
$$

give the possible locations for separating different repetition elements, and a continuous set of local minima along the vertical direction defines an edge between different repetition elements. To be robust to noise, I choose a region size $I \times I$ to evaluate the continuation scores. Figure 4.8 gives an example of the continuation scores and the resulting decomposition.

## 4.6 Experiments

This section presents the qualitative and quantitative results. I have run all the experiments with the same parameter settings.

### 4.6.1 Qualitative Results

Figure 4.9 shows several examples of repetition detection. It can be seen that the proposed detection algorithm robustly finds salient boundaries for both horizontal and vertical directions. The boundary detection is robust to occlusions, illumination changes, perspective changes, and existence of homogeneous regions and high-frequency repetition regions. As shown in examples 2, 3, 4, 9, 13, 18 of Figure 4.9, the algorithm detects
Figure 4.8: Example of decomposition. The color stripes in the left image shows the continuation score, and the black vertical lines give the detected element boundary edges. The right image shows the final decomposition into four different repetition groups.

vertical boundaries based on the novel quality function (4.2) and correctly generates multiple repetition regions along the vertical direction.

Although the proposed algorithm initializes the regions from symmetry axes, it does not enforce strict symmetry constraints on the detected elements. This allows successful detection of repetitions under large viewpoint changes, where the symmetry is weak (e.g. 4, 9, 14, 16 in Figure 4.9). In such cases, the repeating elements are detected with imperfect symmetries, and the horizontal boundaries may not be optimal. The experiments also show several limitations of the algorithm. In Figure 4.9.3, the repetition from the left tower to the right tower is missing because the repetition interval is much larger than the tower width. The current proportional patch size will not work, unless the ratio is allowed to vary. Figure 4.9.8 does not detect the pure vertical repetition on the right side because the current implementation only looks for vertical repetitions for horizontally repeating elements. There are also small errors in boundary detection like in Figure 4.9.4 where too much is occluded for correct boundary detection. In Figure 4.9.17, the method has detected a wrong repetition due to inaccuracy of the second vanishing point pair.
Figure 4.9: Examples of the detected repeating elements overlaid on top of the original images. (Best viewed in color with $4 \times$ zoom.)
Table 4.1: The detection performance on the ZuBuD dataset

<table>
<thead>
<tr>
<th>Category (see Table 4.2 for examples of each category.)</th>
<th>#</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. No correct detection due to VP detection failure</td>
<td>25</td>
<td>4%</td>
</tr>
<tr>
<td>2. No detection due to other algorithmic limitations</td>
<td>34</td>
<td>5%</td>
</tr>
<tr>
<td>3. Partial Detection (missing major repetitions)</td>
<td>88</td>
<td>12%</td>
</tr>
<tr>
<td>4. Full detection of all major repetitions, with some boundaries errors</td>
<td>67</td>
<td>9%</td>
</tr>
<tr>
<td>5. Full detection of all major repetitions, with good boundaries</td>
<td>509</td>
<td>70%</td>
</tr>
</tbody>
</table>

4.6.2 Quantitative Evaluation

The ZuBuD database [Shao et al. (2003a)] is used here to evaluate the proposed repetition detection. ZuBuD contains 1005 images of 201 buildings in Zürich taken from different viewpoints and illuminations conditions. Firstly, I manually filtered out 282 images that do not have clear repetitions that satisfy the assumptions introduced in Section 4.3 (due to occlusions, curved surface, etc). Figure 4.9.17-19 show four examples from ZuBuD and Table 4.2 gives some examples of the excluded photos and the failure cases. Table 4.1 gives the statistics of detection on the 723 remaining images. It can be seen that the experiments demonstrate a high success rate for both VP detection and repetition detection.

Furthermore, I conducted an image retrieval experiment to evaluate the repeatability of the repetition detection. I selected the 140 buildings that have clear repetitions in at least 4 images. The detection algorithm finds 10096 features in total (each element is counted as one; multiple elements in the vertical direction generate an extra feature as their combination; there are on average 14 features per images). Similarly to the SIFT descriptor, for each repeating element, I generate a 4x4 and a 8x8 gradient orientation histogram grid aligned with repeated elements to get a 128D and a 512D feature descriptor respectively. Particularly, uniform weighting is used instead of Gaussian weighting to give equal importance to each cell. The distance between two features is defined as the smaller of the original descriptor distance, and the distance when one feature is
<table>
<thead>
<tr>
<th>Category</th>
<th>examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>0. Images not considered as having valid repetition.</td>
<td><img src="image1.png" alt="Image 1" /> <img src="image2.png" alt="Image 2" /></td>
</tr>
<tr>
<td>1. Vanishing point detection failure, typically due to bad initialization</td>
<td><img src="image3.png" alt="Image 3" /> <img src="image4.png" alt="Image 4" /></td>
</tr>
<tr>
<td>2. No repetition detected due to other algorithmic limitations</td>
<td><img src="image5.png" alt="Image 5" /> <img src="image6.png" alt="Image 6" /></td>
</tr>
<tr>
<td>3. Partial Detection, with missing major repetitions</td>
<td><img src="image7.png" alt="Image 7" /> <img src="image8.png" alt="Image 8" /></td>
</tr>
<tr>
<td>4. All major repetitions regions are found, but some boundaries are not accurate</td>
<td><img src="image9.png" alt="Image 9" /> <img src="image10.png" alt="Image 10" /></td>
</tr>
</tbody>
</table>

Table 4.2: Examples to illustrate different outcomes of the detection process
shifted by half interval. The distance of a feature to an image is defined as its smallest
distance to all the features in that image. Given a single feature, images can be retrieved
by selecting the closest ones.

In this experiment, a feature-image retrieval is considered correct if the image is
one of the other 4 images of the same building. For comparison, I choose the 10/100
SIFT features that have the largest scales in each image to run the same experiment.
Figure 4.11 shows the retrieval precisions for the first 4 nearest neighbors, where the
proposed detection of repeating elements demonstrates relatively high repeatability. In
the future, it is worthwhile to compare the repetition regions to the category of global
appearance feature descriptors, for example, the GIST descriptor proposed by Oliva and
Torralba (2001).

Compared with with Schindler et al. (2008), there is a significant difference in the
way repeating elements are used. Instead of using a combined representation that cap-
tures the common appearance, this work keeps a separate descriptor for each repeating
element in a repetition region. Since the repetition regions considered in this thesis
have large areas, the elements that belong to a common repetition region may have
small local differences. Keeping a separate feature for each repeating element preserves
the distinctive appearances for scene recognition. In the future work, it would still be
interesting to compute a descriptor for the common appearance, with additional efforts
to fuse the repeating elements for a smoothed median representation.
Figure 4.11: Evaluation by single-feature image retrieval. REP refers to the repeating elements. 8x8 and 4x4 refers to the grid size for feature descriptor. R3 refers to the elements that repeat at least 3 times. R2D refers to the features that belong some 2D repetition grids. It can be seen that the repetition region features achieve better repeatability compared with standard image features. Additionally, features in R2D and R3 have better precision because they are easier to detect. It is worth pointing out that many of the retrieval failures are due to the similar structures (especially windows) on different buildings.

This evaluation can also be viewed as an application of repetition detection to location recognition. Given that the number of features is small, the search for similar images is efficient. This can serve as the initial matching step for large urban datasets. To improve the retrieval accuracy, I believe a more advanced feature descriptor capturing more region details is necessary. For example, the GIST descriptor captures a set of perceptual dimensions (naturalness, openness, roughness, expansion, ruggedness).

4.7 Further Discussion

This chapter proposes a novel method to detect the repeating elements on architectural facades, which can be used as high-level features for efficient recognition.

The proposed repetition detection algorithm recovers accurate boundaries of repetition regions based on a novel quality function that measures how image patches fit a
repetition interval without ambiguities from integer multiples of other repetition intervals. The proposed detection method achieves efficiency by GPU-based scale-adaptive evaluation of repetition and symmetry, and typical images require only 2-4 seconds to complete the full repetition analysis. Experimental evaluation on large datasets of real urban scenes demonstrates the robustness and repeatability of the proposed detection. Compared with existing repetition detections, the new method works particularly well for structures that repeat for only a small number of times or repeat only horizontally, which I believe will be more general for urban scene images.

Although the assumptions in Section 4.3 enable efficient and robust repetition detection in typical urban scenes, they also place limits to the proposed algorithm. Structures in urban scenes can repeat in complicated patterns, for example, they can repeat circularly instead of along a line, and they can repeat with varying repetition intervals or at different depth. To extend the algorithm to more general repetition geometry, higher-order regularity constraints need to be explored. Alternatively, multiple images can be used to first understand the 3D scene geometry and then analyze repetitions with the guidance of 3D information. Another limitation is the assumption of reflective symmetry. The algorithm does generate repetition regions for the non-symmetric structures by taking the positions that show maximal symmetry, but the region boundaries will not be as good as the real symmetric structures. To correctly define region boundaries for such cases, additional priors need to be included, for example, there are likely to be homogeneous regions between two interesting structures.

In addition, the proposed repetition detection method can be used for recognition of facade images. If a database of manually-labeled of repeating elements is available, we can learn the appearance models of the corresponding urban structures. This can lead to compact summarization of scenes structures, and will allow efficient scene recognition.
Chapter 5

Repetition-based Reconstruction

As an extension of the repetition detection method introduced in Chapter 4, this chapter presents a technique to recover the dense geometry for repeating elements. The shape recovery is formulated as dense pixel correspondences within the rectified images. The pixel correspondences are represented by an interval map that tells the distances of each pixel to its matched pixels within the single images. In order to obtain dense repetitive structures, a new repetition constraint is developed, which penalizes the inconsistency between the repetition intervals of the dynamically corresponding pixel pairs. I deploy a graph cut optimization to balance between high-level geometric repetition, photometric consistency and spatial smoothness of the reconstructed scene. Experiments demonstrate accurate recovery of interval maps and dense 3D repetitive structures on a variety of scenes. Furthermore, these experiments demonstrate the robustness to outliers such as structure variations, illumination changes, and occlusions.

5.1 Dense Reconstruction from Repetition

With the knowledge of the repetition regions within an image, dense reconstruction is enabled by considering the repeating elements as multiple views of a single 3D structure, where the non-rigid differences between the repeating elements encode the underlying geometry. As illustrated in Figure 5.1, these different views can be modeled as a set of
Figure 5.1: Repetition intervals give clues of 3D geometry. (a) An example 3D model. (b) Frontal projective view of the 3D model, where the repetition interval of the closer surface is larger than the interval further away. (c) Illustration of the synergy between the single view of a repetitive structure and multiple cameras viewing a single element that have equal baselines. The three images are provided as a courtesy by Li Guan.

virtual cameras with equal distances.

Figure 5.2 gives an overview of the repetition-based single-view dense reconstruction. The basic steps are very similar to the standard two-view stereo, which first recovers disparity maps that represent the pixel correspondences between two images. The pixel correspondences in repetition is actually more complicated and more geometrically constrained. Each pixel can have multiple correspondences within the single image, and the multiple correspondences should all be consistent with their common depth.

Figure 5.2: Overview of repetition-based reconstruction. (a) The detected repeating elements have differences due to the non-planar structure. For example, the distance between the columns is larger than the distance between the upper windows because the columns are closer. (b) The pixel correspondence recovered as an interval map. (c) The dense repetitive structures are recovered consistently despite the varying reflections and occlusions. (best viewed in color)
5.2 Related Work

5.2.1 Reconstruction from Repetition and Symmetry

Sparse reconstruction based on repetition and symmetry has been well studied. Hong et al. (2004) investigate symmetry-based (including translational, reflective, and rotational) methods for recovering camera position and orientation of objects in a scene, as well as some sparse geometry, from a single view. Francois et al. (2003) hallucinate a virtually mirrored viewpoint to model the single view reconstruction as a two view reconstruction.

Based on the sparse reconstruction, several work has further achieved dense reconstruction. Gool et al. (2007) propose an optimization framework that uses the sparse feature matches as control points to recover dense depth maps. Shimshoni et al. (2000) recover symmetric human face models by propagating the correspondences of manually-given pixels pairs based on photometric stereo. Our work proposes a global energy minimization framework to model both repetition and reflective symmetry in the image domain with a simple interval map model.

5.2.2 Markov Random Field Stereo

MRF-based stereo optimization typically uses a data term to enforce photometric consistency between matched pixels, and a smoothness term to penalize the inconsistency of disparities between pixel neighbors [Boykov et al. (2001); Scharstein and Szeliski (2001); Tappen and Freeman (2003)]. Most stereo algorithms enforce consistency only in the traditional pixel neighborhood with a smoothness term, and consistency between non-neighboring pixels is often considered intractable. I propose a novel repetition and symmetry-based energy function that enforces high-level consistency between the disparities of non-neighboring pixels. In particular, this thesis shows that high-level 3D information can be modeled in the image domain by graph cuts.
5.2.3 Symmetric Stereo

Symmetric stereo methods treat all the images equally. Particularly, to recover multiple depth maps that are consistent with each other, the interactions between the depth maps need to be modeled. However, the interactions between pixels in different images are depth-dependent, posing a challenge to an image-based model. Several methods have been proposed to enforce consistency indirectly through visibility and occlusions. Kolmogorov et al. (2003) define an interaction set among multiple images and enforce a hard visibility constraint through graph cuts. Sun et al. (2005) develop a occlusion term to penalize the occluded regions, which indirectly makes depth maps consistent. In this Section, the proposed energy function will directly enforce the consistency between multiple depth maps (and different parts within the depth map).

5.3 Geometry of Repetition in A Rectified Image

This section will briefly discuss the geometric model of the repetition-based reconstruction. This thesis considers the repetition and symmetry along only the horizontal direction, which is typically in man-made environments. Without loss of generality, for one set of 3D repeating structures that have equal spacing in 3D, I assume the following properties for reconstruction:

- The center of the single camera is at $(0, 0, 0)$;
- The repetition step is 1 along direction $(1, 0, 0)^T$.

Consider the camera projection of 3D repetitive structures. Denote the projection matrix as $P = KR[I \mid 0]$, where $K$ is the intrinsic calibration, and $R$ is the camera pose. Denote the homography that transforms the original image to the rectified image as $H$, 

83
it must satisfy

\[ HKR(1, 0, 0)^T \sim (1, 0, 0)^T \]
\[ HKR(0, 1, 0)^T \sim (0, 1, 0)^T \]

\[ \Rightarrow H \text{ can be written as } \begin{bmatrix} a & 0 & b \\ 0 & c & d \\ 0 & 0 & 1 \end{bmatrix} (KR)^{-1}, \]

where \( a, b, c, d \) are decided by the choice of rectification and the calibration matrix \( K \). Given a 3D point \( (X, Y, Z)^T \), the corresponding pixel \( (x, y)^T \) in the rectified image is

\[ (x, y)^T = \left( \frac{aX}{Z} + b, \frac{cY}{Z} + d \right)^T. \] (5.1)

Consider two repeating 3D points \( (X, Y, Z)^T \) and \( (X+1, Y, Z)^T \). Their projections lie on the same scanline \( y = \frac{cY}{Z} + d \) in the rectified image, and the distance between the projections is a function of the depth as follows:

The repetition interval for depth \( Z \) is \( I_Z = \frac{a}{Z} \). (5.2)

This implies the following for the reconstruction: 1) the pixel correspondences only need to be considered within scanlines, 2) the relative depth of the 3D point can be recovered from the image repetition interval if the camera calibration is known. When the complete camera calibration matrix \( K \) is available, the third vanishing point and the camera pose can be estimated. Hence, \( a, b, c, d \) can be obtained from \( H \), and the 3D location of each pixel \( (p_x, p_y) \) can be recovered as

\[ (X_p, Y_p, Z_p)^T = \left( \frac{p_x - b}{I_Z}, \frac{a(p_y - d)}{cI_Z}, \frac{a}{I_Z} \right)^T. \] (5.3)

For uncalibrated cameras, I assume the principal point of the original image at the image center, and recover the focal length and the camera pose by enforcing the orthogonality of vanishing directions, which is similar to Schindler et al. (2008). In case of degeneracy where the vanishing points are almost at infinity, I choose the focal lengths based on
the EXIF header of the JPEG if it is available or based on a typical viewing angle.

In addition to pure repetition, it is possible to model local reflective symmetries for the repetitive structures. Let the range along $(1, 0, 0)^T$ of the first element be $[X_0 - 0.5, X_0 + 0.5]$. The symmetry planes in 3D are $X_i = X_0 + \frac{i}{2}$, $i \in \mathbb{N}$, and their corresponding positions in the rectified image are

$$x = \frac{a(X_0 + \frac{i}{2})}{Z} + b = (X_0 + \frac{i}{2})I_Z + b. \quad (5.4)$$

Hence, the symmetry axes at different depths are transformed to different locations in the rectified image. Besides the depth, symmetry axes are constrained by the global unknown $X_0$, which can be automatically recovered, as shown in Section 5.4. Note that $x = b$ is the vanishing line of any planes that is perpendicular to $(1, 0, 0)$.

### 5.4 Repetition-based Optimization Framework

The repetition detection algorithm applied to an image provides its rectified image and a set of repetition regions $P$. Due to large depth ranges, multiple regions may be detected for the same 3D repetition, and I first merge the overlapping repetition regions that have similar repetition intervals. From the set of feature matches along scanlines in each $P_i$, the set of horizontal feature distances $DX = \{dx\}$ is obtained. I empirically choose the set of possible repetition intervals as $L = \{l \mid l \in \mathbb{N}, |l - \text{mean}(DX)| < 2\sqrt{\text{var}(DX)}\}$. Experiments show that such a chosen range is typically larger than the actual interval range, but the real range $L'$ can be found from a first-pass reconstruction with $L$.

Similarly to the disparity map in two-view stereo, I use a simple **interval map** for the repetition regions in the rectified image. An interval map is a labeling $f = I_Z$ over $P$, such that each pixel $p$ matches either $p - f(p)$ or $p + f(p)^1$. Analogously to stereo, I assume smooth scene depth except for object boundaries. Hence, the interval map is

\[p \pm I\]

\[p_x \pm I, p_y\]. In addition, no pixel outside the region $P$ is considered.
piecewise smooth, too. As a key geometric property of repetitive structures, the interval map should also satisfy that

\[ f(p - f(p)) \text{ and } f(p + f(p)) \text{ are similar to } f(p). \]

Now I propose a new energy function to jointly model photometric appearance similarity, local smoothness and repetition constraints:

\[ E(f) = E_{\text{data}}(f) + E_{\text{smooth}}(f) + E_{\text{repetition}}(f), \tag{5.5} \]

where the data term \( E_{\text{data}} \) penalizes the photometric dissimilarity between corresponding pixels, the smoothness term \( E_{\text{smooth}} \) penalizes the differences of intervals between neighboring pixels, and the repetition term \( E_{\text{repetition}} \) penalizes the differences of intervals between corresponding pixels. The data term, smoothness term and repetition term will be explained in Section 5.4.1, Section 5.4.2 and Section 5.4.3 respectively.

### 5.4.1 Data Term

Since repetition is bidirectional, the data cost should combine the matching cost from the left matching and right matching. Let \( D(p, q) \) be the matching cost of two pixels \( p \) and \( q \), and define the data cost for a pixel \( p \) as

\[
D_f(p) = \begin{cases} 
\text{mean}\{D(p, q) | q = p \pm f(p), q \in P\} & \text{if } p + L_{\text{max}} \in P \text{ or } p - L_{\text{max}} \in P; \\
0 & \text{otherwise}.
\end{cases}
\]

The second case of \( D_f \) ensures that a pixel either has valid costs for all labels or 0 for all labels, which handles the margins of repetition regions. The data term is defined as

\[
E_{\text{data}} = \sum_{p \in P} D_f(p). \tag{5.6}
\]

Given two pixels \( p \) and \( q \), I design the matching cost based on the maximum abso-
lute difference from the three color channels. Two standard techniques are employed to improve the matching cost: I first apply Birchfield-Tomasi (BT) sampling [Birchfield and Tomasi (1998)] to reduce errors caused by the resampling during the image rectification, and the resulting cost is denoted by $D_{BT}(p,q)$. Second, I truncate with $T_D$ the matching cost to improve robustness to occlusion leading to the pairwise cost $D(p,q) = \min(D_{BT}(p,q), T_D)$.

A significant difference between repetition-based reconstruction and multi-view stereo is that repetition-based reconstruction matches the appearances of multiple surfaces within one image, while multi-view stereo matches the appearance of the same surface across multiple images. Hence I aim for more robustness as real scenes often do not have perfect repetition. The cost truncation supports the robustness of large appearance differences of the repeating elements.

### 5.4.2 Smoothness Term

The smoothness cost is chosen as the truncated $L_1$ distance of the repetition intervals of the neighboring pixels. Let $N_P$ be the set of neighboring pixels in the repetition region $P$ for the 4-neighborhood system. The smoothness term is defined as

$$E_{\text{smooth}} = \omega_{\text{smooth}} \sum_{(p,q) \in N_P} V(p,q) = \omega_{\text{smooth}} \sum_{(p,q) \in N_P} \min(T_V, |f(p) - f(q)|),$$

(5.7)

where $\omega_{\text{smooth}}$ is a positive penalty for violating the smoothness constraint, and $T_V$ is the truncation threshold.

### 5.4.3 Repetition Term

I design a novel repetition term to penalize the deviation between different instances $f(p \pm f(p))$ of the same repetition $f(p)$. Treating the different parts of the interval map as the disparity map of the repeating elements, the repetition term essentially provides a
new formulation for symmetric stereo by explicitly enforcing that matching pixels have similar disparities. For convenience of notation, an indicator function as follows is used:

\[
\rho(\text{condition}) = \begin{cases} 
1 & \text{if condition is true} \\
0 & \text{otherwise}
\end{cases}
\]

The repetition term between two pixels only needs to be enforced when their distance is equal to one of their repetition intervals. The key difference between the repetition term and the smoothness term is that the pixel pairs rely on the labels of the pixels, which defines a dynamic neighborhood for each pixel. For convenience, I define a function \( R_f \) to check if two pixels \( p \) and \( q \) should be compared:

\[
R_f(p, q) = \rho(|p_x - q_x| \in \{ f(p), f(q) \}). \tag{5.8}
\]

Additional requirements are considered for enforcing the repetition. As mentioned in the discussion of the data term, the matching of repeating pixels often involves large appearance changes from occlusions, reflections, noise etc. It is not optimal to enforce the repetition consistency without considering the photometric similarity, as the global optimization may try to avoid those matching costs and propagate incorrect intervals. Hence it is natural to loosen the repetition constraint when the two locations significantly deviate in appearance. This is similar in spirit to the idea of reducing the smoothness constraint across image edges. I define a guiding function \( G(p, q) \) to evaluate if the repetition-based consistency should be considered for two pixels:

\[
G_{local}(p, q) = \rho(D_{BT}(p, q) < T_G), \tag{5.9}
\]

where \( T_G \) is a threshold for testing the pixel similarity. Consequently, the repetition cost is applied only between similar pixels, and larger \( T_G \) gives a stronger constraint.

In Section 5.5, \( G_{local} \) is compared to two other choices of the guiding function: 1) No repetition constraint with \( G_{none} = 0 \); 2) A global repetition term \( G_{global} \) that enforces
repetition based on the region decomposition proposed in Section 4.5.3 without considering the photometric similarity. In the following, $G_{\text{local}}$ from Equation (5.9) is always used unless otherwise specified.

The repetition cost of the entire image is given by

$$E_{\text{repetition}} = \omega_{\text{rep}} \sum_{q_y = p_y, R_f(q, q) = 1} G(p, q) \rho(f(p) \neq f(q)),$$  \hspace{1cm} (5.10)

where $\omega_{\text{rep}}$ is a positive penalty for violating repetition consistency. Equation (5.10) uses a dynamic neighborhood for the pixel nodes in the graph, which can not directly be handled by traditional optimization methods. However, the above equation can be rewritten as

$$E_{\text{repetition}} = \omega_{\text{rep}} \sum_{|q_x - p_x| \in L, q_y = p_y} R_f(p, q) G(p, q) \rho(f(p) \neq f(q)).$$  \hspace{1cm} (5.11)

Now the neighborhood is fixed and the standard energy optimization methods are applicable. Since the number of edges for the repetition term is in $O(|P| * |L|)$, one limitation of the proposed approach is its possibly large memory consumption.

In this work, the efficient $\alpha$-expansion graph-cut [Boykov et al. (2001); Boykov and Kolmogorov (2004)] is used to minimize the proposed energy. Kolmogorov and Zabih (2004) proved that $\alpha$-expansion can minimize the class of energy functions that satisfy the regularity constraint $e(\alpha, \alpha) + e(\beta, \gamma) \leq e(\beta, \alpha) + e(\alpha, \gamma)$. I now prove that the repetition term fulfills the regularity constraint.

Given an edge between two pixels $p$ and $q$, let $\omega = \omega_{\text{rep}} * G(p, q)$ and $\delta = |p_x - q_x|$, the repetition cost $e(f(p), f(q))$ of the edge is a function of their labels:

$$e(f(p), f(q)) = \omega R_f(p, q) \rho(f(p) \neq f(q)) = \omega \rho(f(p) = \delta \text{ or } f(q) = \delta) \rho(f(p) \neq f(q)),$$

which is obviously non-negative and symmetric.
Now, consider three labels $\alpha$, $\beta$ and $\gamma$ to prove the regularity property. If the three labels are all different from each other, at most one of them can be equal to $\delta$, and there are four different cases:

**IF $\alpha = \delta$**
\[
e(\beta, \gamma) = 0, \quad e(\beta, \alpha) = \omega, \quad e(\alpha, \gamma) = \omega;
\]

**IF $\beta = \delta$**
\[
e(\beta, \gamma) = \omega, \quad e(\beta, \alpha) = \omega, \quad e(\alpha, \gamma) = 0;
\]

**IF $\gamma = \delta$**
\[
e(\beta, \gamma) = \omega, \quad e(\beta, \alpha) = 0, \quad e(\alpha, \gamma) = \omega;
\]

**Otherwise**
\[
e(\beta, \gamma) = 0, \quad e(\beta, \alpha) = 0, \quad e(\alpha, \gamma) = 0.
\]

Since $e(\alpha, \alpha) \equiv 0$, the inequality below holds true:
\[
e(\alpha, \alpha) + e(\beta, \gamma) = e(\beta, \gamma) \leq e(\beta, \alpha) + e(\alpha, \gamma).
\]

Additionally, if $\alpha = \beta$ or $\alpha = \gamma$, $e(\alpha, \alpha) + e(\beta, \gamma) = e(\beta, \alpha) + e(\alpha, \gamma)$ can be proved by substituting $\beta$ or $\gamma$ respectively by $\alpha$. If $\beta = \gamma$, the inequality is also satisfied because $e(\alpha, \alpha) + e(\beta, \gamma) = 0 \leq e(\beta, \alpha) + e(\alpha, \gamma)$.

It is also worth noting that the choice of repetition cost is non-trivial. My investigations show other choices such as having $e(f(p), f(q))$ proportional to (truncated) $|f(p) - f(q)|$ or penalizing only the non-occluded pixels violate the regularity constraint.

It can be seen that the repetition term is only pseudo-metric (non-negativity, symmetry and triangle inequality), because the identity of indiscernibles (such that $e(\alpha, \beta) = 0$ iff $\alpha = \beta$) is not satisfied.

### 5.4.4 Reflective Symmetry Term

Similarly to the repetition term, any available knowledge about reflective symmetries can be incorporated. To model them it is required to know the symmetry axis $X_0$ in Equation (5.4). The symmetry axis is initially unknown but can be recovered from sparse feature correspondence or alternatively from the dense correspondences. For improved
robustness I opt for the latter and propose a two-step approach:

1. Recover the interval map $f_0$ using only $E(f)$ from Equation (5.5), and locate $X_0$,

2. Refine the interval map by a graphcut that includes the reflective symmetry term:

$$E^+(f) = E(f) + E_{sym}(f).$$

(5.12)

To extract $X_0$ I render an orthogonal view $f'_0$ of the interval map $f_0$ by generating the 3D model from $f_0$ according to Equation (5.3) and reprojecting the 3D model along $(0, 0, 1)^T$. For robustness 3D points that have $f_0(p + f_0(p)) \neq f_0(p)$ or $f_0(p - f_0(p)) \neq f_0(p)$ are excluded. $X_0$ is then chosen from all possible locations in the repeating element to maximize the number of consistent pixels pairs in $f'_0$.

Next I introduce the reflective symmetry term $E_{sym}$. Given two pixels $(p, q)$ that have different labels, we would like to give a penalty if they are at symmetric positions w.r.t. the depth of either $f(p)$ or $f(q)$. First, Equation (5.4) gives the set of symmetry axes for an interval $I_Z$, thus we can check if the two pixels are symmetric w.r.t to one of these symmetry axes. Second, to enforce reflective symmetry only within each element, $|p_x - q_x| < L_{\text{min}}$ is required. Denote the function that tests the two aforementioned conditions as $C(I_Z, p, q)$. The indicator function $S(f, p, q)$ is defined for a pixel pair $(p, q)$ and an interval map $f$ as

$$S(f, p, q) = \max(C(f(p), p, q), C(f(q), p, q)).$$

(5.13)

Enforcing reflective symmetry is harder than enforcing repetition due to the occlusions we often have from oblique viewpoints. For example, in Figure 5.3.c, the right halves of many repeating elements are severely occluded, and it would create new problems if we were to enforce reflective symmetry naively everywhere. In particular, enforcing reflective symmetry on pixels whose symmetric structures are occluded would require us to perturb the occluding pixels. On the contrary, only a few pixels are oc-
cluded in terms of pure repetition. Therefore, it is less reliable to recover depth from reflective symmetry than from repetition, meaning the reflective symmetry should be a weaker constraint than repetition.

I implement the reflective symmetry term as a refinement such that it does not contaminate the pixels $\Lambda(f_0)$ that already satisfy the reflective symmetry in $f_0$:

$$\Lambda(f_0) = \{p \mid \exists q (q_y = p_y, f_0(p) = f_0(q), S(f_0, p, q) = 1)\}.$$ 

The graph edges $\Phi(f_0)$ for enforcing reflective symmetry are chosen to include all possible symmetric pairs but to not have any pixels in $\Lambda(f_0)$:

$$\Phi(f_0) = \{(p, q) \mid p_y = q_y, p, q \notin \Lambda, \exists l \in L(C(l, p, q) = 1)\},$$

The symmetry term for the entire region is then given by

$$E_{sym}(f) = \omega_{sym} \sum_{(p, q) \in \Phi(f_0)} S(f, p, q) \rho(f(p) \neq f(q)), \quad (5.14)$$

where $\omega_{sym}$ is the penalty for violating reflective symmetry.

The proposed symmetry term also satisfies the regularity constraint. Consider the cost $e(f(p), f(q))$ of an edge $(p, q)$, if $e(\beta, \gamma) = \omega_{sym}$, one of $C(\beta, p, q)$ and $C(\gamma, p, q)$ must be 1. As a result, either $e(\beta, \alpha)$ or $e(\alpha, \gamma)$ will be equal to $\omega_{sym}$, and $e(\beta, \gamma) \leq e(\beta, \alpha) + e(\alpha, \gamma)$ is satisfied. Although it seems intuitive to incorporate occlusion information, I find such reflective symmetry terms violating the regularity constraint.

### 5.4.5 The Algorithm

The pipeline I deploy to recover the dense repetitive and symmetric structures is:

1. $\alpha$-expansion graphcut to minimize $E(f)$ (Equation (5.5));
2. Find the refined interval range $L'$ from recovered $f$;
3. $\alpha$-expansion graphcut to minimize $E(f)$ on $L'$;

4. Extract 3D symmetry axis parameter $X_0$ using $f$;

5. $\alpha$-expansion graphcut to minimize $E^+(f)$ (Equation (5.12)).

The refined interval range is extracted by excluding the labels that are assigned to very few pixels. Let $r(l) = |\{p \mid f(p) = l\}|/|P|$, the new range is chosen to be $L' = [\min\{l \mid r(l) \geq r_{\min}\}, \max\{l \mid r(l) \geq r_{\min}\}]$, where $r_{\min} = 1\%$ is used. Although only a small portion of pixels are affected, the filtered interval range improves robustness of the reconstruction. In all the experiments I use a fixed set of parameters $T_D = T_G = 25$, $T_V = 2$, $\omega_{\text{smooth}} = \omega_{\text{rep}} = 10$, and $\omega_{\text{sym}} = 2$ unless explicitly stated.

5.5 Experiments

This section demonstrates some results of the proposed repetition-based reconstruction to show the advantage of the novel optimization framework.

5.5.1 Reconstruction of Interval Maps

Figure 5.4 presents the results on four challenging images shown in Figure 5.3 when using different repetition and symmetry constraints. Firstly, I compare the recovered interval map with three different repetition terms (after step 1-3).

- **No repetition constraint** $G(p,q) = 0$. The standard graph cut optimization is able to recover some correct intervals for the repetition regions, but the results show its sensitivity to noise and outliers in real scenes.

- **Global repetition term** $G_{\text{global}}(p,q)$ enforces repetition everywhere within each repetition region group. Despite the most continuous repetition results, this constraint is error-prone by propagating local errors (e.g. matching with occlusion boundaries). The quality is limited by the accuracy of the initial region segmentation, for example, the triangle structures in Figure 5.4.4 are smoothed away.
• **Local repetition term** $G_{\text{local}}(p,q)$ enforces repetition only for similar pixels. With the local repetition term, it is possible to reconstruct repeating structures even under large occlusions. It specifically handles the repetition of different structures (e.g. Figure 5.4.3 and 5.4.4).

Table 5.1 lists the computation times for the experiment shown in Figure 5.4. It can be seen that enforcing the local repetition term takes three times longer than the standard optimization. For the local repetition term, no edges will be constructed in the graph for the pairs that satisfy $G(p,q) = 0$, and the total number of edges for repetition is $|\{(p,q)|G(p,q) \neq 0, |q_x - p_x| \in L, q_y = p_y\}|$. Consequently, the optimization with the local repetition term runs faster than that with the global constraint term.

Refinement of interval maps by enforcing reflective symmetry is demonstrated in Figure 5.4.d. The 3D structures in these experiments are all reflectively symmetric, but reconstruction with local or global repetition term cannot produce symmetric interval maps. By introducing the reflective symmetry term into the optimization, the errors in the initial reconstructions can be corrected.

### 5.5.2 Repetition-based 3D Reconstruction

3D structures are reconstructed by incorporating the recovered interval map and calibration according to Equation (5.3). Figure 5.2(b), Figure 5.5 and Figure 5.6(a) show high quality 3D models from the repetition-based reconstruction. The camera calibration parameters for examples in Figure 5.5 are recovered based on vanishing points, while
(a) With no repetition term  (b) With global repetition term  (c) With local repetition term  (d) Refinement with reflective symmetry

Figure 5.4: Comparison of different repetition terms. The brighter colors in the interval maps correspond to larger intervals and closer surface, and the gray scale bars on the left give the number of filtered interval labels. The comparison shows the local repetition term reconstructs correct interval maps despite the variations between the repeating elements, while the global repetition term tends to over-smooth and propagate errors. The last column shows the improvement after enforcing reflective symmetry.

the calibrations for Figure 5.2(b) and Figure 5.6(a) are selected according to the EXIF in the images because the recovered vanishing point locations (almost) at infinity are not accurate. If we scale the 3D structure along the $Z$ direction as well as the focal length with the same scaling factor, the resulting image does not change. Such an ambiguity can not be resolved without additional information (e.g. the third vanishing point).

One interesting application of single view reconstruction is to generate ortho-rectified views, an invariant view of 3D structures. Based on the texture synthesis proposed by Efros and Leung (1999), I generate ortho-rectified images by filling the missing pixels from the best copy out of their repeating counterparts when they are visible in other
instances of the repeating element. To reduce the noise in the reconstruction, I first apply a simple fusion of the orthogonal interval map to refine the reconstruction by enforcing smoothness, repetition and reflective symmetry in the 2.5D space of the interval map. Figure 5.6 shows examples of the fused orthogonal interval map and the final synthesized image. In Figure 5.6, the occluded parts of the white windows in the first copy and the last copy are filled in smoothly. More examples are given in Figure 5.7.

Due to the difference between the original viewpoints and the orthogonal projections, there are pixels in the ortho-rectified image for which all the copies are invisible in the original image (compare Figure 5.6 and Figure 5.4.1). Generating truly realistic ortho-rectified images would require a more complicated model, which is beyond the scope of this thesis.

### 5.6 Further Discussion

This chapter proposes a novel framework to reconstruct dense geometries of repeating elements, which add 3D information to the repetition-based features. The single-view reconstruction method enforces geometric repetition/symmetry constraints, photometric consistency and neighbor smoothness in a single unified optimization framework. The power of the new energy minimization framework is demonstrated by accurate and robust dense reconstruction of many scenes. This new approach goes beyond previous
approaches that can only recover sparse geometries.

The single-view reconstruction method requires the repeating elements to show sufficient differences and to have a relatively large repetition interval range in order to recover the detailed 3D geometry. Such large changes can only be achieved when cameras are close to the 3D structure, which is due to the fact that repeating structures on urban scenes do not have large depth changes. Otherwise, the scene can be approximated by a plane and only such a plane can be recovered.

However, the viewpoint changes will not be limited with multiple cameras, which gives us opportunities to enforce repetition constraints in multi-view reconstruction and also recover detailed repetitive structures. In particular, the repetition intervals correspond to parallel planes at different depths, and it is possible to extend repetition-based optimization to plane-sweeping multiview stereo [Gallup et al. (2007)].
Figure 5.6: Ortho-rectified images generated from repetition-based reconstruction. (a) Dense surface Model. (b) Fused orthogonal interval map. (c) Frontal view with missing pixel marked as red. (d) Synthesized ortho-rectified image (best viewed in color).

Figure 5.7: Examples of ortho-rectified images. The three rows give the rectified images, the orthogonal view with missing pixels, and the synthesized orthogonal view respectively. The synthesized images clearly align the symmetry axes at different depths. Some artifacts are shown in Figure (c): the wrong door copied to the last element and the region around the depth discontinuity above the center door.
Chapter 6

Conclusion and Future Work

6.1 Summary

Research in image features is driven by the need to organize and extract information from the explosively growing amount of image data in a widening range of computer vision applications. Features that are theoretically invariant to similarity transformations [Lowe (2004)] or affine transformations [Mikolajczyk et al. (2005)] have been the most popular. Traditional image features, however, do not have a 3D geometric interpretation, which limits their applicability. 3D objects may not produce the same features under large viewpoint changes; the normalized representation of image features is ambiguous for similar structures that have different 3D scales (or aspect ratio for affine-invariant features); an object’s meaningful higher-level structure tends not to be captured by image features.

This thesis explores the guiding power of 3D geometric interpretation in feature detection to gain improved invariance, discriminative power, and high-level representation ability. As demonstrated by the VIP method presented in Chapter 3, projective-invariance is achieved by exploiting stereo-based 3D local geometry and detecting features in orthogonal views, which enables scene matching under large viewpoint changes. The 3D geometric dimensions of the features are naturally recovered during the detection
process, providing additional discriminating power in 3D matching. As demonstrated by many experiments, the proposed viewpoint invariant patch (VIP) features can be used to find the critical matches between significantly different views in large-scale reconstruction.

Following the spirit of geometry-driven feature detection, I propose in Chapter 4 a novel repetition detection method to find features that capture even more high-level structure. On one hand, repetitions give multiple views of their common geometry and approximate planar reconstruction often can be recovered based on vanishing point detection. On the other hand, the regular repetition patterns in urban scenes allow us to extract high-level regions. Therefore, repetition region feature detection can be viewed as a combination of region-based methods and 3D geometry based methods. Additionally, in Chapter 5, I propose a novel repetition-based dense reconstruction method to recover the finer 3D geometry of the repeating elements, which goes beyond the state-of-art 2D repetition detection methods.

Geometry-driven feature detection further allows us to improve the accuracy, completeness, or efficiency of the scene reconstruction. This thesis demonstrates the strength of the VIP features particularly in handling large viewpoint changes and its applications in scene matching, which can improve the completeness of large-scale reconstruction. By capturing important appearance information of urban scenes, the repetition region features are compact representations of scenes. Section 4.6.2 presents a single feature retrieval experiment to show their potential for efficient and accurate scene recognition.

6.2 Future Work

Potential directions for future work include improvement to the proposed features, applications of the proposed features in scene reconstruction, and other extensions of the idea of geometry-driven feature.
6.2.1 Improvements

VIP detection could be extended to handle sparse scene geometry. If loop detection can be done with sparse reconstructions, the possible errors in the camera motions will be corrected before the dense reconstruction, thus the computation time is saved. Given the existence of multiple images, VIP detection should also improve the repeatability of features by incorporating the information from multiple textures. One possibility is to keep the keypoints that respond in multiple textures.

Repetition detection needs be extended to handle the more complicated repetition patterns shown in Figure 6.1, which go beyond the assumptions of the current algorithm. For instance, the detection would be more general if non-planar repetition pattern can be modeled. For boundary detection, the robustness can be improved by testing a serial of decreasing patch scales instead of only one.

![Figure 6.1: Rotational, Irregular, Distant elements](image)

Figure 6.1: It would require to go beyond the current assumptions to handle the complicated repetition. In (a) and (b), the underlying geometry cannot be simplified to a plane. In (c), the current method may fail to detect the distant repeating elements due to its lack of automatic scale selection.

Repeating elements can be matched more accurately with more discriminative feature description. First, it is possible to assign different weights to different parts of a repeating element according to their variance among all repeating elements, so the matching can be more robust to outliers. Second, we can use feature descriptors that capture more appearance details than the simple histograms. Third, it is possible to use the dense 3D geometry recovered from repetition to generate orthogonal views for matching, which
will be more robust to viewpoint changes.

6.2.2 Applications

One application of viewpoint invariant patches (VIPs) is 3D reconstruction of large-scale photo collections. In 3D reconstruction systems, it is often difficult to reconstruct a large scene as a single 3D model because important transitions between different viewpoints can be missing. For example, in the work by Frahm et al. (2010), the automatically selected iconic images are often only at popular viewpoints. Extracting viewpoint-invariant features like VIP can help in the matching between the separate models in order to generate a more complete reconstruction.

The retrieval experiment in Section 4.6.1 can be extended to powerful urban scene recognition based on detected repetition. Like other local features, the repeating elements can be used in similar way for matching and indexing, while the small number of repeating elements can bring superior efficiency. In addition, the automatically extracted repetition layout produces another level of description of the image facades that may also be helpful in distinguishing different scenes. It will be challenging and interesting to explore the application of the repeating elements for large-scale urban recognition, for example, detecting the repeating elements of an entire city and using cell phone images to query this database for location recognition.

6.2.3 Possible Extensions

3D scene geometry and repeating patterns are not the only geometric properties that can guide feature detection. We should exploit other geometric properties when available.

In particular, it is possible to extract the peak structures (e.g. Figure 6.2) that look similar from all directions to handle extremely large viewpoint changes. The detection can be done by first finding the vertical vanishing point and then extracting symmetric corners that point in the vertical vanishing point direction. Then we could extract
standard descriptors (e.g. SIFT) on such peak structures.

Figure 6.2: Rotationally-symmetric structures (marked as red) that look very similar from all different directions can be used to handle extremely large viewpoint changes.

Detection of reflective symmetric regions without the existence of repetition is a more challenging scenario compared with detection of repetition regions. If we can reliably determine the boundaries of such features under viewpoint changes, the extracted regions can serve as more powerful high-level features that handle more general structures not only for urban scenes but also for natural scenes.

Figure 6.3: Reflective symmetric features are pervasive in manmade scenes. Such regions can also serve as high-level features. (Best viewed in color).
Bibliography


