

REPETITION DETECTION AND SHAPE RECONSTRUCTION IN RELIEF IMAGES

Thesis submitted in partial fulfillment
of the requirements for the degree of

MS by Research
in
Computer Science

by

Harshit Agrawal
200802013

harshit.agrawal@research.iiit.ac.in



Center for Visual Information Technology
International Institute of Information Technology
Hyderabad - 500 032, INDIA
December 2013

Copyright © Harshit Agrawal, 2013
All Rights Reserved

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled “Repetition Detection and Shape Reconstruction in Relief Images” by Harshit Agrawal, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: Prof. Anoop Namboodiri

To My Family.

Acknowledgments

First of all, I would like to extend my sincere gratitude towards my advisor, Dr. Anoop M. Namboodiri who have been a constant support and motivation through out the duration of my work. I thank him for the insightful discussions related to the topic and beyond work. He is an inspiration and has always motivated me to think about the bigger goals while giving utter significance to the minute details. I thank him for the freedom, he gave me to study my topic of interest in my own way without any force of regular meetings. His influence led to my personal and professional development, shaping my attitude towards research and life, in general. I would also like to thank the other professors at CVIT; Prof. P.J. Narayanan, Prof. C.V. Jawahar and Prof. Jayanthi Sivaswamy for their encouraging and motivational presence and creating a wonderful environment in the lab that makes learning fun.

Being in CVIT was fun with the fellow dual-degree students Jay, Ankit, Chandrashekhar, Mihir, Aditya, Madhulika, Nitish and my seniors Abhinav, Vinay, Siddharth K., Harshit S., Siddharth Ch., Atif, Akhil, Shrikant, Rohan, Parikshit, Shashank and so many others whom I could not acknowledge here. I would like to thank Anand, Vijay, Siddharth Choudhary, Srijan for the helpful discussions that provided me deeper understanding of the field and come up with better ideas to solve the problems. My stay at IIIT-H could not have been so awesome without my friends, Sandeep, Yash, Tushar, Hemant, Aman, Akshay, Hardeep, Sarvesh, Nachiket, Saket, Arpit, Ajay, Ishan, Deepika and so many others. I thank them for their constant support and the belief they shown in me. This journey could not have been possible without them.

I owe this thesis to my family who has supported me throughout, encouraged me when things were not right and were always there when I needed them. I thank my parents for their utter belief in me and their unbounded patience. I thank my sister for her unconditional love and belief in my abilities during my entire stay at IIIT-H. Many thanks to everyone who affected my life in any way, and wasn't acknowledged personally above.

Abstract

Relief carving is very popular sculpting technique that is being used for decoration and depicting stories and scenes from ancient times till today. With time, many of the ancient cultural heritage artifacts are getting damaged and one of the important methods that aids preservation and study is to capture them digitally.

Reliefs carvings have certain specific attributes that makes them different from regular sculptures, which can be exploited in different computer vision tasks. Repetitive patterns are one such frequently occurring phenomenon in reliefs. Algorithms for detection of repeating patterns in images often assume that the repetition is regular and highly similar across the instances. Approximate repetitions are also of interest in many domains such as hand carved sculptures, wall decorations, groups of natural objects, etc. Detection of such repetitive structures can help in applications such as image retrieval, image inpainting and 3D reconstruction. In this work, we look at a specific class of approximate repetitions: those in images of hand carved relief structures. We present a robust hierarchical method for detecting such repetitions. Given a single relief panel image, our algorithm finds dense matches of local features across the image at various scales. The matching features are then grouped based on their geometric configuration to find repeating elements. We also propose a method to group the repeating elements to segment the repetitive patterns in an image. In relief images, foreground and background have nearly the same texture, and matching of a single feature would not provide reliable evidence of repetition. Our grouping algorithm integrates evidences of repetition to reliably find repeating patterns. Input image is processed on a multi-resolution pyramid to effectively detect all possible repetitions at different scales. Our method has been tested on images with large varieties of complex repetitive patterns and the qualitative results show the robustness of our approach.

Reconstructing geometric models of relief carvings are also of great importance in preserving heritage artifacts, digitally. In case of reliefs, using laser scanners and structured lighting techniques is not always feasible or are very expensive given the uncontrolled environment. Single image shape from shading is an under-constrained problem that tries to solve for the surface normals given the intensity image. Various constraints are used to make the problem tractable. To avoid the uncontrolled lighting, we use a pair of images with and without the flash and compute an image under a known illumination. This image is used as an input to the shape reconstruction algorithms. We present techniques that try to reconstruct the shape from relief images using the prior information learned from examples. We learn the variations in geometric shape corresponding to image appearances under different lighting condi-

tions using sparse representations. Given a new image, we estimate the most appropriate shape that will result in the given appearance under the specified lighting conditions. We integrate the prior with the normals computed from reflectance equation in a MAP framework. We test our approach on relief images and compare them with the state-of-the-art shape from shading algorithms.

Contents

Chapter	Page
1 Introduction	1
1.1 What are Reliefs ?	1
1.2 Motivation	2
1.2.1 Repetitive Patterns in Reliefs	3
1.2.2 Shape Reconstruction from Images	4
1.3 Contribution of the Thesis	5
1.4 Outline of the Thesis	6
1.5 Summary	6
2 Literature Review	8
2.1 Patch Matching Techniques	8
2.1.1 Tree-based Methods	9
2.1.2 PatchMatch	9
2.2 Repetition Detection	10
2.2.1 Detection in Regular repetitive patterns	11
2.3 Shape Reconstruction	12
2.3.1 Multi-View Stereo Reconstruction	12
2.3.2 Structured-lighting Techniques	13
2.3.3 Shape Reconstruction from Single Image	14
2.4 Summary	14
3 Detection and Segmentation of Approximate Repetitive Patterns in Relief Images	16
3.1 Introduction	16
3.2 Observations and Assumptions	17
3.3 Pairwise Matching	19
3.3.1 Pairwise Feature Matching	19
3.3.2 Pairwise Patch Matching	20
3.4 Grouping Patches	20
3.5 Detection and Segmentation of Repetitive Patterns	23
3.6 Dataset Collection and Annotation	24
3.7 Results and Discussions	26
3.8 Conclusion	29
3.9 Summary	31

4	Shape Reconstruction from a Single Relief Image	32
4.1	Introduction	32
4.2	The Proposed Approach	34
4.3	Shape from Shading	35
4.4	Image under known Illumination	35
4.5	Learning the priors for relief image	36
4.5.1	Dictionary Learning	37
4.5.2	Sparse Coding	37
4.5.3	Shape Recovery using relief priors	38
4.6	Datasets	38
4.7	Experiments and Results	39
4.7.1	Quantitative Evaluation	39
4.7.2	Qualitative Evaluation	40
4.8	Conclusion	43
4.9	Summary	43
5	Conclusions and Future Work	44
	Bibliography	47

List of Figures

Figure	Page
<p>1.1 Figure shows an example of reliefs from around the world and were created in different time scales. (a), (c) and (e) are examples of <i>high-reliefs</i>, as the structures are raised high above the background planes. (b) and (f) shows examples of <i>bas-reliefs</i>. (d) is an example of <i>sunken-relief</i>.</p>	2
<p>1.2 A comparison between the repetitive patterns in building facades and reliefs. In (a), the repeating elements are well-defined and are regularly repeating in the image space. Also, the repeating elements can be easily distinguished from the background plane. In contrast, (b) does not have well-defined repeating instances. The repetitions are not exactly similar and are repeating irregularly in the image space. The foreground and background of the relief have very similar surface properties. Hence, most of the algorithms outputs highly unsatisfactory results in case of relief images.</p>	3
<p>1.3 Figure shows example of relief images collected from different places or walls a larger cultural heritage site. A complete site may have been presumably built with similar stones and the story narrated by these reliefs often relates the style of them to each other.</p>	5
<p>2.1 Three stages of the PatchMatch Algorithm [8]. (a) All the patches are initialized randomly. (b) A patch searches the matches of its neighbors and propagates good matches. For example, the blue patch in (b) checks above/green and left/red neighbors. (c) The patch searches randomly for improvements in concentric neighborhoods. (Image credit: [8])</p>	9
<p>3.1 Example reliefs with approximate repetitive patterns.</p>	17
<p>3.2 (a) Dense sift features after removal of false matches (b) Pairwise sift matching, blue features are pairwise matches of yellow (c) Pairwise patch matching, green patches are correct match of yellow patch and red patches are false matches (d) Connectivity graph at various scales in the multi-resolution pyramid (e) Score Image for the corresponding connectivity graph in (d). (f) Segmented image regions after merging all score images with distinct labels for each region. (g) These labels are merged and grouped to produce the final output of our algorithm with color-coded convex hulls of repetitive patterns. . .</p>	18
<p>3.3 Example of an approximately repeating structure.</p>	23

3.4 Example annotations of repetitive patterns in different types of relief images. In each pair of image, left to right, original and the annotated image. The two top left annotations show an example of approximate repetitions. In first example, head of a horse is bent down and hence excluded from the repetition. Similarly, trunk of elephants are not in repetition. Each image was closely analyzed to find all the approximate repetitive patterns. 25

3.5 Each row shows the result of detection and segmentation on varieties of reliefs images. (a) shows detection in case of approximate repetitions. (b) Elephant’s back is wrongly detected as a horse’s repetition. (c) is an accurate detection. (d) Red pattern is a false positive due to repeating sky. (e) Detected multiple repetitions with irregular intervals. (f) As parts of the statue is repeating, they are identified as different repeating elements. 27

3.6 Example of robustness of our detection and segmentation algorithm in presence of multiple repetitive patterns and significant projective skews. Original image(top), Annotated image(middle), Our Result(bottom). 28

3.7 Example failure cases. (a) Large projective skews, (b),(d) close repetitive patterns, and (c) partial detection due to occlusions. 29

3.8 Example of our detection performance on Facades and NRT images. 30

4.1 *Shape Reconstruction from single relief image (Depth maps are shown with psuedo-color visualization, red is near and blue is far). (a) Complete exemplar dataset consists of only 7 relief images, (b) Original Relief Image, (c) Depth Map obtained by SfS of Tsai et al. [55], (d) Depth Map obtained by Barron and Malik [10], (e) Depth Map obtained by our approach. The depth map obtained from (c) is noisy. We learn shape priors for reliefs to improve the shape reconstruction. Note that we recover overall geometry as well as details of face, legs and the left part of relief that is not recovered in (d). Results of [10] were poor for color images, so we used gray scale image to obtain (d).* 33

4.2 Sample of dictionary elements learned from exemplar images and their surface gradients. (a) elements from image appearance. (b) and (c) elements from surface gradients in x and y-direction respectively. 35

4.3 Example of Pure Flash image computation. (a) Flash+Ambient Image **F** (b) Ambient Image **A** (c) Pure Flash Image **PF** 37

4.4 Comparison between two variants of signal construction discussed in Sec. 4.7. (a) Original Image, (b) Depth map using Pixel-wise Signal Construction, (c) Depth map using Patch-wise Signal Construction. Note that, (b) captures finer details where as (c) is more smoothed shape. 39

4.5 Qualitative results of our approach on relief images collected from various sources. In each instance, three images are the original image, our pixel-wise and patch-wise results, respectively. All these results are computed using the same dictionary learned on the exemplar relief images. The results shows robustness of our approach in presence of ambient illumination along with point light sources. 41

4.6 Depth Maps obtained from our approach on Human body poses dataset [26]. Each instance is shown as original image, results of pixel-wise and patch-wise approaches as depth map respectively. The dictionary was learned using a set of 12 exemplar images. The depth variation of both the legs are correctly estimated in (a) and (b), and the depths of head in (c) and (d). Note that pixel-wise method is able to recover the depth variation of feet. 42

4.7 Example of failure cases. (a) Significant errors in shape reconstruction due to strong sunlight and cast shadows. (b) Incorrect shape reconstruction because of the violation of lambertian reflectance assumption. (c) Non-uniform albedo results an error in overall shape recovery. 42

List of Tables

Table		Page
3.1	Repetition detection and segmentation performance of our algorithm	28
4.1	Quantitative Results as Average Mean Squared Error for Reliefs and Human body poses dataset.	40

Chapter 1

Introduction

1.1 What are Reliefs ?

The term *relief* is derived from the Latin word *relevo*, to raise. Relief is a very fascinating sculpting technique that is used for enhancing the aesthetic beauty of architectural buildings all over the world. The intent behind creating a sculpture in relief is to give an impression that the sculpted material has been raised above the background plane. Reliefs are chiselled out of a flat surface of stone or wood thereby lowering the background field and the unsculpted parts seemingly raised. The chiselling away of the background requires considerable amount of time and also good artistic creativity. This technique of creating a relief saves forming the rear of a subject, and are also less fragile. It can be more securely fixed than other round sculpture, especially a standing sculpture where the bottom parts like ankle are potential weak points.

Reliefs are common throughout the world on the walls of the places of worship, palaces, public buildings and parks. Along with enhancing the ambiance of the place, a sequence of several sections of reliefs may represent extended narratives. Unlike the free-standing sculpture, reliefs are more suitable for depicting complicated subjects with multiple figures and much more active poses, such as battles, dance forms, etc. Most of the reliefs can be roughly classified into the following three broad categories.

- *Bas-relief* or *low-relief* is a projecting shape with a shallow overall depth. In *low-reliefs*, when seen from the front the small variations in depth gives an impression of three dimensional shape, and if seen from the sides it makes little sense.
- In *High-relief*, considerable depth variations are present across the sculpture. Prominent elements of the composition, especially heads and limbs, often projects out of the background plain with sufficient depths and sometimes even cut off from the background plain.
- A lesser prominent relief type is *sunk* or *sunken* relief largely restricted to the art of ancient Egypt. In these types of reliefs, the image is made by cutting the reliefs in a flat surface. So, the relief is a *low-relief*, but set with a sunken area shaped round the figures, so that the relief never rise beyond the original flat surface.



(a) Relief of *Herodotus* in Louvre palace, Paris



(b) Relief from the temple of *Hatshepsut* in Egypt showing Egyptian soldiers



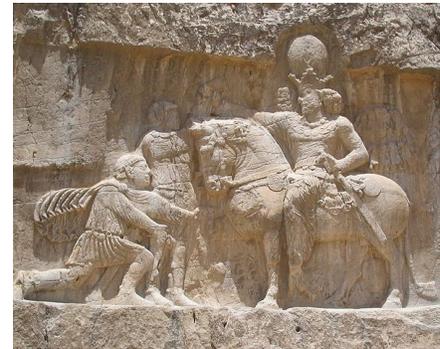
(c) *Sri Senpaga Vinayagar* temple, Singapore



(d) Relief of *King Akhenaton*



(e) Relief of a chariot with Hindu Gods and Goddess in Hampi, India



(f) A bas relief sculpture at Naqsh-e Rostam, Iran, depicting the triumph of Shapur I over the Roman Emperor Valerian.

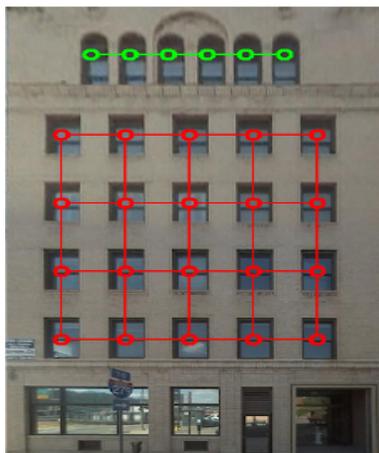
Figure 1.1 Figure shows an example of reliefs from around the world and were created in different time scales. (a), (c) and (e) are examples of *high-reliefs*, as the structures are raised high above the background planes. (b) and (f) shows examples of *bas-reliefs*. (d) is an example of *sunken-relief*.

1.2 Motivation

Reliefs are used as a form of decoration and a medium to depict stories and incidents from ancient times. With time many of these structures get weathered down and parts of the reliefs may get broken or damaged. Hence, it is of great importance to preserve these heritage symbols digitally. Several events [1, 2] are organized that focuses on restoration, preservation, protection, documentation and presentation, and massive digitization of the cultural heritage contents by promoting interdisciplinary research on cutting edge technologies. Numerous projects have been funded all over the world to capture the glory of the world's heritages to both showcase and preserve for future generations. The work in this thesis provides generic solutions that are developed to tackle specific challenges faced due to relief images. Better solutions to the below problems can assist in digitally preserving these cultural heritage sites.

1.2.1 Repetitive Patterns in Reliefs

In man-made environments, structures with repetitive patterns are often employed as they are aesthetically pleasing. The problem of repetition detection is to find similar instances in repetitive patterns from images. In reliefs, structures appear in different repetitive patterns. Repetitions, in themselves, have redundancy as all the structures have some common properties like shape, style, texture, etc. The information extracted by detecting repetitions can be used as prior in segmentation and reconstruction of elements of the structures. It can also provide valuable information in case of partial occlusions. Repetitive patterns can also assist in single view reconstruction of the scene [76]. Robust representation of repetitive patterns can be used as an invariant descriptor to identify semantically relevant objects to match across images [16].



(a) An image of a building facade where translational symmetry patterns are detected with the method proposed by Zhao et al. [81]. (Image credits: [81])



(b) Relief image of a wall from heritage site in India. (Image credits: Flickr)

Figure 1.2 A comparison between the repetitive patterns in building facades and reliefs. In (a), the repeating elements are well-defined and are regularly repeating in the image space. Also, the repeating elements can be easily distinguished from the background plane. In contrast, (b) does not have well-defined repeating instances. The repetitions are not exactly similar and are repeating irregularly in the image space. The foreground and background of the relief have very similar surface properties. Hence, most of the algorithms outputs highly unsatisfactory results in case of relief images.

Repetitive patterns are also very common in building facades and urban scenes. Many algorithms have been developed for detecting repetitions and translational symmetry [39, 41, 42, 44, 49], but a fully automatic method for detecting approximate repetitions in complex real images like reliefs is still a very challenging task. A repetitive pattern can have both uniformly spaced repetitions or irregularly spaced repetitions. We consider a generic notion of repetitive pattern as repetition of a structure or an object. In urban facades, the structural elements like windows and doors usually have regularity in repetition,

for instance in Figure 1.2(a). The regularity can always be used as a reliable cue for detecting such repetitive patterns. Different algorithms [36, 49, 53, 81] have been developed that exploits the regularity of such repetitions. Unlike facades, reliefs have irregular repetitions with changes in the appearances. Perspective view of facade images can be rectified using vanishing points to get the fronto-parallel view [49]. It is often difficult to extract robust vanishing points in reliefs due to absence of linear structures (see Fig. 1.2(b)).

1.2.2 Shape Reconstruction from Images

With the advents in computer vision and computer graphics, one can browse different places and structures in three dimensional views by capturing multiple images. The reconstructed 3D model can be viewed as a point cloud or as textured mesh model with complete freedom to play with the camera parameters. Many multi-view stereo algorithms [66, 23, 56] have been developed over the past few years that takes a sufficient number of 2D images of the structure and outputs a 3D dimensional point cloud representation with camera parameters. The 3D point cloud can then be used to create a mesh model that can be used for the purpose of displaying in the interactive 3D browsers or applications.

Large scale 3D reconstruction has been the back-bone of many projects that aim at digitally preserving and protecting the cultural heritages of the world. Along with that, virtual walk-through projects have been sponsored for showcasing the important cultural heritages to the future generations. In multi-view stereo reconstruction techniques, each real world point must be visible in at least three images to allow the complete coverage of the structure. Techniques used for 3D reconstruction have their own limitations, especially for large scale usage. Highly accurate systems such as laser scanners are extremely expensive for use by common man, whereas multi-view stereo methods require large number of images. Other methods also make similar tradeoffs between cost, ease of use and accuracy.

Shape reconstruction from a single image is an ill-posed problems that has been solved by using various monocular cues, constraints and assumptions. The particular way of construction of reliefs can provide us useful cues that can be exploited to reconstruct the shape from a single image, which is the primary focus of our work. As the relief is made up of a single stone, the color and surface reflectance is almost uniform across the relief. It is often the case with ancient reliefs which are weathered down with time. In reliefs, the sculpted portion that forms the background of the structure is usually a plain surface. It is due to this sculpted background plain, the beautiful carvings gives a perception of 3D dimensional shape, especially in the *low-reliefs*. This can be a useful cue in reconstructing shape from a single relief image. Also, other than *high-reliefs*, the shape variation across a relief is mostly small and continuous.

In ancient times, reliefs were constantly deployed while constructing huge palaces, temples, cathedrals, and even kingdoms. Logically, the reliefs present at these huge sites should share very similar properties amongst them. This can be due to the same type of stones, similar carving styles or many different factors. This can be a very advantageous cue for reconstructing shape from relief images. The surface properties like surface reflectance, texture, color are almost similar for all the reliefs at that place. So, if we can capture the properties of some exemplar reliefs then, they can be used as a prior



(a) An exemplar collection of the bas-reliefs of Rouen Cathedral, Rouen, France.



(b) A set of images captured from different temples of Hampi, a cultural heritage site in India.

Figure 1.3 Figure shows example of relief images collected from different places or walls a larger cultural heritage site. A complete site may have been presumably built with similar stones and the story narrated by these reliefs often relates the style of them to each other.

knowledge for other reliefs from the same place. Figure 1.3 shows such an example from important world heritage sites. We can see the surface texture, color and reflectance are very similar among various images captured for the same site. Apart from the surface properties, the carvings on the reliefs are also similar from an artistic point of view. They usually have pretty similar shape variations and it generally should depend on the proximity of the two structures. If they are closer to each other then it is more likely for them to share similar properties and shape variations.

1.3 Contribution of the Thesis

Reliefs forms a major part of the world’s important cultural heritage and the work in this thesis helps in preserving them by developing methods that work well, especially in case of reliefs. The thesis presents our previously published works on Detection and Segmentation of approximate repetitive patterns in relief images [5] and Shape reconstruction from a single relief image (See Sec. 5). The main contributions of the thesis are :

- Repetitive patterns are very prevalent in reliefs. Detecting repetitive patterns have been a long standing problem in the field of computer vision. In this work, we detect and segment repetitive patterns in a relief image automatically. Repetitive patterns in reliefs are very different from those in building facades and other grid-like repetitions. It is highly unlikely to find exactly same instances repeating in a relief. So there is a trade-off between accuracy and robustness. The proposed method is flexible enough to be tuned between robustness and accuracy by tweaking very few parameters as per the requirements. Repetitive patterns with irregularity are captured unlike

the other methods that require a grid-like regular repetitions. Also, the approach is independent of the number of repeating instances. It can also detect repetitive patterns with only two repeating instances. The hierarchical approach does not presume anything about the repeating instances. In reliefs, it is very difficult to identify the repeating element automatically. Also, surface properties like color and texture are mostly similar for both the foreground and the background. Along with the detected patterns, the method also outputs a *rep-field* with correspondence amongst the matching instances, that can further be exploited in other applications. The proposed method appropriately detects multiple repetitive patterns with different frequencies at different positions in the image. The detected patterns are also assigned a score that describes the matching confidence of the repeating instances. The repeating patterns are segmented into individual elements that can be used in applications like image retrieval, shape reconstruction, etc.

- Single-view reconstruction is an ill-posed and a challenging problem. Digitally preserving the cultural heritage is of great importance to the human society. 3D shape representation is an important step in showcasing the cultural heritage in digital form. Many of the present 3D reconstruction techniques either require expensive instruments or a careful and controlled experimental setup with time consuming processing in presence of an expert in the field. Our goal is to develop a least expensive and easy to use method that improves the accuracy of shape reconstruction from images. The proposed method requires only a single input image of relief and it outputs the corresponding depth map. The data-driven approach learns a relief specific shape priors using an exemplar set of relief images and corresponding depth maps.

1.4 Outline of the Thesis

The thesis is organized as follows. The next chapter looks at the previous works in patch matching techniques, detecting repetitive patterns and 3D shape reconstruction from 2D images. In chapter 3, we describe the proposed approach to detect and segment approximate repetitive patterns in reliefs. We show our performance on facades and regular texture images. In chapter 4, we discuss our shape reconstruction algorithm from a single 2D relief image. We conclude the thesis with our contributions and existing problems with the methods and the possible future works in these directions.

1.5 Summary

With time many of our ancient heritage artifacts are getting damaged and are vanishing due to factors such as harsh weather, neglect and act of nature. Hence, it is of great importance to digitally protect and preserve these artifacts, so that our future generations can virtually explore them when the sites are physically vanished. Reliefs have a very important role in enhancing the aesthetics of any architectural artifact and they are constantly deployed to narrate stories and scenes from ancient times. Structures

in reliefs are often seen to form repetitive patterns across the relief. However, in most cases, repetitive instances are only approximately or partially similar to each other. Hence, detection of approximately repeating patterns in reliefs becomes a more challenging task than in building facades and near regular textures where the repetitive instances are predefined and the repetitions are often exactly similar to each other. Virtual 3D models of reliefs also aids in digitally preserving these ancient cultural heritages. Recovering 3D shape using a single image is a long standing problem. Reliefs are constructed in a particular way by carving out a single stone to give an impression that the sculpted part is raised above the surface. This particular way of construction provides us with many useful cues that be exploited in reconstructing the shape from a single relief image. Reliefs cover a large part of the cultural heritage sites and we can significantly contribute in digitally preserving these important heritage sites, if we are able to solve these challenging problems.

Chapter 2

Literature Review

In this chapter, we build the background for detecting approximate repetitive patterns in reliefs. We review the literature for symmetry detection and repetition detection in facades. We also discuss the various techniques used for reconstructing shapes of scenes and objects from single and multiple images. Patch-matching is at the heart of many computer vision tasks. To make our patch-matching algorithm in Sec 3.3 more clear, we give an overview of various types of patch-matching approach used to solve computer vision tasks, especially, detection of symmetric and repetitive patterns.

2.1 Patch Matching Techniques

Repetition detection has the most important role of feature matching as when we want to identify similar instances of an object, an accurate and robust technique is required to solve the problem reliably. Feature matching in multiple images is well studied in the computer vision society. Problems like image mosaicing, multi-view stereo reconstruction, image registration etc, involve matching of similar features in multiple images. Generally, these are overlapping images of a scene or an object, so when a robust and discriminative feature is used on these images, we get highly reliable feature matches. However for repetition detection, we have to find similar instances in a single image.

Several nearest-neighbor(NN) algorithms have been developed for finding similar patches between images. Different applications have different requirements from the patch-matching algorithms. There is always a trade-off between speed, accuracy and robustness. As an evidence, there are a lot of parameters which can be tuned to get the best results for an application. Speed and accuracy are generally the desired properties of patch-matching algorithm. Using general-purpose NN algorithms [67, 77] is not the most efficient solution as they would not take advantage of the structures and properties specific to images. So, patch-matching is generally posed as a search problem in a high-dimensional data space.

2.1.1 Tree-based Methods

Tree-based methods organize data in a hierarchical structures. *kd*-trees [28, 35] generally perform splitting along a data dimension, usually one with the maximum variance. Node divisions in *kd*-trees are always axis-aligned, regardless of the data-distribution. A memory efficient and faster method was an extension to *kd*-trees, Gaussian-*kd*-tree structure proposed by Adams *et al.* [3]. It combines effects of storing more than one element per node and making local rearrangements during insertions. PCA trees [38, 67] try to relax the axis-aligned assumption of *kd*-trees by performing a Principal Component Analysis (PCA) [30] to reduce dimensionality of data. Other methods such as Ball Trees [50] and Vantage Point Trees (*vp*-trees) [77] partition the data points based on some metric defined on pairs of points. Kumar *et al.* [37] perform several optimizations on tree based methods and claim that the *vp*-tree structure, which is not well-known in the vision community, gives the best overall performance for finding similar image patches.

2.1.2 PatchMatch

PatchMatch [8] is a dense and global ANN(Approximate Nearest Neighbor) method which performs a randomized, cooperative hill climbing search and calculates dense nearest neighbor matches quickly. It relies on the coherence between patches of an image for speedup. That is, if we find a pair of similar patches, in two images, then their neighbors in the image plane are also likely to be similar. The algorithm has three main components, illustrated in Figure 2.1.

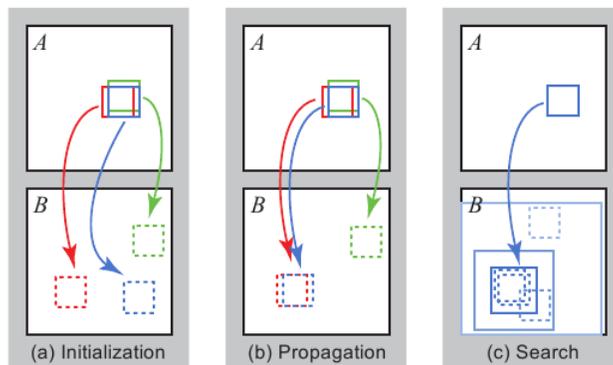


Figure 2.1 Three stages of the PatchMatch Algorithm [8]. (a) All the patches are initialized randomly. (b) A patch searches the matches of its neighbors and propagates good matches. For example, the blue patch in (b) checks above/green and left/red neighbors. (c) The patch searches randomly for improvements in concentric neighborhoods. (Image credit: [8])

1. **Initialization:** The nearest-neighbor field (NNF) is initially filled with either random offsets or some prior information. Some applications use the guesses upscaled from coarser pyramid levels

to achieve a good initial guess and escape from local minima. After the initialization, the next two steps are performed iteratively.

2. **Propagation:** In each iteration, matches of a patch at (x, y) are improved using known matches of $(x - 1, y)$ and $(x, y - 1)$, translated by one pixel to get the corresponding match. In even iterations, the scan-line processing is reversed and matches of $(x + 1, y)$ and $(x, y + 1)$ are checked for improving the match of the patch at (x, y) .
3. **Random Search:** In each iteration, the matches of a patch at (x, y) are further improved by testing a sequence of candidate offsets picked uniformly at random with an exponentially decreasing distance from the current match.

PatchMatch was generalized to include K nearest matches and searching for various rotations and scales [9]. PatchMatch is an order of magnitude faster than previous approaches and enables several interactive applications [8]. However, it is not very accurate as one would hope. It requires more iterations for achieving higher accuracy.

Unlike PatchMatch, our Pairwise Matching finds feature matches in a single image itself. As discussed earlier, for a patch in image A, PatchMatch search for a matching patch in image B, whereas in our problem we have to find multiple occurrences of shapes and structures present in a single image. To improve the correspondences, they use the offset values of the neighboring pixels. In contrast, we start with confident pairwise matches, found independently and then impose the neighboring constraint while doing the hierarchical grouping. Also, our pairwise matching is different from the image retrieval in the sense that we are not using a Bag-of-words representation of features [65]. We require strong matches between features and in a bag-of-words representation, many features will be denoted by a representative visual word which will increase the number of false matches among the features.

2.2 Repetition Detection

Repetition detection in images is a long standing problem and till date researchers in computer vision have made significant progress towards detection of symmetries and repetitions not only in images but also in 3D data. The problem of repetition detection is of detecting similar instances of an object, shape or an artifact occurring at multiple places in 2D images. Researchers have looked at this problem from various point of views and solved them using certain assumptions about the object under consideration, the repetitive pattern, or even about the 2D image.

Leung and Malik [39] proposed a simple window matching scheme followed by grouping of patterns for finding repeating scene elements. Their approach consisted of detecting interesting elements in the image, matching these candidate interesting elements to their neighbors and finding the affine transformation between them. The size of the interesting elements were chosen randomly, so they grow the elements to form more distinctive units and finally, grouping the elements.

Schaffalitzky and Zisserman [61] proposed a RANSAC-based grouping method for imaged scenes, which repeat on a plane in the scene. They compute interesting elements and find associations among them by fitting a homography. RANSAC is then used to find the best hypothesis for the repetitive pattern. Their approach is focused on regular grid-like patterns. Loy and Eklundh [44] presented an efficient method for grouping feature points based on their underlying symmetry and also characterizing the symmetries present in the image. They extract rotationally invariant features like SIFT [43], and then finds matches for features that are bilaterally and rotationally symmetric to this feature. They group these symmetric features to find symmetric constellations and thereby computing symmetries in the image plane.

2.2.1 Detection in Regular repetitive patterns

Repetitive patterns have been very popular in architectural styles of the urban areas. Most of the buildings in these days consists of repeating patterns where windows, doors, arches, etc forms the repeating elements. Along with their primary purpose, they contribute to the aesthetics of the buildings. Most of these patterns are constructed in a grid-like manner which makes them interesting to study the structural modeling. Significant amount of research has been done on detecting the repetitions in patterns which repeat at regularly spaced intervals.

Lin *et al.* [41] proposed a method to extract periodicity of a regular texture based on simple autocorrelation functions. Detecting repetitive patterns and symmetry in facades have been of keen interest in the research community. Facades are typically characterized by perpendicular regularities in horizontal and vertical directions. Repetition detection in facades can contribute in structural modeling, exploiting the regularities and also enable us to infer compact descriptions of the repeated structures. Wenzel *et al.* [73] have proposed a method for detecting repeated structures in facade images. They begin by detecting the dominant symmetries and then use clustering of feature pairs to detect the repeating structures in the image. Using the strong assumptions of structured construction in facades, Korah and Rasmussen [36] developed a probabilistic framework using Markov Random Field modeling and Markov Chain Monte Carlo (MCMC) optimization to explicitly recognize and group rectangular structures that appear in a grid-like pattern.

Unlike relief images, facades images have sufficient number of sets of parallel lines both in horizontal and vertical directions. So, facades can be accurately rectified to fronto-parallel views. It helps in simplifying the problem by removing the complexity of projective distortions while working on a solution. Most of the methods used for repetition detection in facades either assume the input as a fronto-parallel facade or apply a pre-processing to rectify the image before initiating the process.

Müller *et al.* [49] proposed an image-based method for automatically deriving 3D models of high visual quality from single facade images of arbitrary resolutions. They subdivide the facades texture in a top-down manner into elements such as floors, tiles, windows, and doors. They combine the procedural modeling pipeline of shape grammars with image analysis to derive a meaningful hierarchical facade

subdivision. Park *et al.* [53] developed an algorithm using an efficient Mean-shift belief propagation for 2D lattice detection on near-regular textures.

Recently, significant amount of work has been done on detecting repetitive patterns in facades. Wu *et al.* [75] developed an approach to detect large repetitive structures with salient boundaries in facades. They assume reflective symmetry in the architectural structures and use this to localize vertical boundaries between repeating elements. Their algorithm depends on accurate vanishing point detection for rectifying the image to a fronto-parallel view. Zhao and Quan [81] describe a robust and efficient method for detecting translational symmetries in fronto-parallel view of facade images using joint spatial and transformation space detection. Recently, Zhao *et al.* [82] developed a robust and dense per-pixel translational symmetry detection and segmentation in facade images. Their algorithm considers most of the limitations of earlier approaches. They create translational maps in horizontal and vertical directions. For segmentation of a repeating element, they have used a learning based approach to classify each pixel as either a wall pixel or a non-wall pixel. Cai and Basui [15] proposed a region growing image segmentation algorithm for detecting and grouping higher level repetitive patterns. Their algorithm begins with manual marking of a region of repetition and then they iterate between growing and refinement steps.

As we discussed, many algorithms have been developed for detecting repetitions and translational symmetry, but a fully automatic method for detecting approximate repetitions in complex real images like reliefs is still a very challenging task. Our robust hierarchical method proceeds from lower to higher level features, thus effectively detects structures with partial repetitions. We consider a generic notion of repetitive pattern as repetition of a structure or an object. A repetitive pattern can have both uniformly spaced repetitions or irregularly spaced repetitions. Our approach is also completely independent of the positioning of the repeating instances.

2.3 Shape Reconstruction

Recovering the 3D world from 2D images has long been an active research area in computer vision and graphics. In computer graphics, the aim is to acquire accurate geometry information of the scene with detailed appearance information (bi-directional reflectance distributive functions, texture maps, etc) suitable for high-quality rendering in a traditional graphics pipeline. In contrast, computer vision prominently deals with recovery of the 3D shape from 2D images of the scene. The techniques and methods used for reconstructing 3D shape largely depends on the specific problem to be solved. We discuss the popular scenarios and approaches used to solve the problem shape reconstruction.

2.3.1 Multi-View Stereo Reconstruction

The goal of multi-view stereo reconstruction algorithms is to reconstruct a complete 3D geometric model of an object by using multiple images preferably from different viewpoints such that the object is completely covered by the images. According to the recent survey provided by Seitz *et al.* [62],

state-of-the-art MVS algorithms have shown accurate reconstruction performance even for a set of low resolution images. Reconstruction performances of various kinds of techniques may vary depending upon the object or scene under observation.

According to [23], these techniques can be roughly classified into four groups in terms of the underlying object models: Bounding box based approaches [56, 71, 72, 64] require the object or scene to be contained within a known bounding box. They are also referred to as *Voxel-based* approach and their accuracy is limited by the resolution of the voxel grid. Algorithms based on *deformable polygonal meshes* [27, 78, 22] demand a good starting point to initialize the corresponding optimization process, which limits their applicability. Approaches based on *multiple depth maps* [24, 68, 13] are more flexible, but require fusing individual depth maps into a single 3D model. *Patch-based* methods [40, 25] represent scene surfaces by collections of small patches (or surfels). They are simple and effective, and can also be used for the visualization purposes. It requires the point cloud to be converted into a mesh model that is more suitable for image-based modeling applications. Along with their survey, Seitz *et al.* provided high quality datasets that can be used for evaluating the performance of multi-view stereo reconstruction algorithms. They also provided the ground truth for each dataset acquired via laser scanning process to be used as a baseline for the evaluation.

2.3.2 Structured-lighting Techniques

An accurate method to recover the 3D shape of an object is to use a *3D Scanners*. They are devices that analyzes a real-world object or scene to collect data on its shape and appearance that can be used to construct three dimensional models. These devices are build using many different techniques each associated with its own limitations, advantages and costs. Structured-lighting is one of those many techniques that is actively used to reconstruct the 3D models because of its speed and accuracy. In general, Structured-light 3D scanners project a pattern of light on the subject and analyze the deformation of the pattern on the subject. The experimental setup usually consists of a projector that projects light patterns on the object and a camera, offset slightly from the projector, captures the shape of the pattern on the object. We give a very brief overview of the different methods used for structured-lighting 3D scanning.

Most of the structured-lighting setup includes a calibrated pair of projector and a camera. The methods differ in the structured-lighting pattern being projected onto the object surface. Various techniques use either discrete coding where the pattern presents a digital profile or a continuous coding where the pattern has a continuous variation in intensity. Spatial multiplexing of *De Bruijn-based* striped pattern are popular in the literature [12, 59, 48]. In spatial multiplexing, the codeword for a specific location is identified by the surrounding points. Methods using *non-formal* codings [20, 19, 70] are non-orthodox codification where specific patterns are created to fulfill some particular requirements. Time multiplexing methods [33, 69, 60, 79] are based on the codeword created by the successive projection of the patterns onto the object surface. Recent survey by Salvi *et al.* [58] provide a much detailed classification of the state-of-the-art structured-lighting 3D reconstruction techniques being used in the literature.

2.3.3 Shape Reconstruction from Single Image

Humans have an innate capability of judging depth from single monocular images. This is done using a combination of various monocular cues such as texture variations, texture gradients, occlusion, known object sizes, haze, defocus, etc [47, 74, 14]. For instance, if there is an occlusion between two objects, we can know which object is closer to us. Shading is another cue that enables us to perceive the smooth depth variations. Shape from shading (also referred as SfS) is a long standing problem and is still a very active area of research. Thorough and complete surveys of early work can be found in [80]. Durou *et al.* [17] surveyed recent works on numerical methods for SfS. Most of the works have popular assumptions such as Lambertian reflectance, single distant point light source, orthographic projection, and constant uniform albedo. These assumptions limits the use of SfS to very controlled and ideal experimental setups. Recent works have relaxed a few of these assumptions. Oxholm and Nishino [51] present a framework to jointly estimate the shape and reflectance of an object from single image under a known natural illumination. Similar works on shape recovery under natural illumination are Huang and Smith [31] and Jhonson and Adelson [34].

Apart from SfS approaches, researchers have examined the relationship between the shading or appearance and the shape variations in local neighborhoods [57]. Freeman *et al.* [21] presented a graphical model framework incorporating patch-based priors. In [26], database consisting of objects of highly similar class like faces, body poses etc., were used to recover the shape for a new query image of the same class. Apart from matching image appearances, they have given higher probability to patches lying in similar regions of the example images, which is possible due to the class specific database. Huang *et al.* [32] presented a generalized patch-based approach where they learn the prior probabilities for a given image patch using a database of spherical geometric primitives and their appearances. These priors are then incorporated in a variational shape from shading formulation. Panagopoulos *et al.* [52] proposed a data-driven approach that learns a dictionary of geometric primitives and their appearances. The dictionary is used to learn a small set of hypotheses about the local 3D structure for the given image to get an initial guess that is then regularized by an MRF optimization layer.

2.4 Summary

In this chapter, we reviewed the previous works done in the field of repetition detection and shape reconstruction. In computer vision, patch matching is the epicenter for a large variety of problems. We looked at the evolution of techniques used for patch matching. It started with general Nearest-Neighbor algorithms where there is always a trade-off between speed, accuracy and robustness. Later, tree-based methods were used where the data is organized in hierarchical structure that reduces search space. We also discussed in brief about the recent Patch-Match technique [9]. It is a randomized and cooperative hill climbing search approach that efficiently finds approximate nearest neighbor patches between two images. We looked at the background of repetition detection and various techniques used to solve the problem. Most of the methods have strong assumptions that make them unsuitable for use in reliefs.

Shape reconstruction is vast field in computer vision and is approached in a variety of ways. We briefly discussed different techniques used for shape reconstruction from multiple and single images. In the next two chapters, we discuss in detail, our approach for repetition detection and shape reconstruction from a single relief image.

Chapter 3

Detection and Segmentation of Approximate Repetitive Patterns in Relief Images

In this chapter, we describe our method for detecting approximately repeating patterns in relief images. The method outputs color-coded segmented regions corresponding to different repetition groups. As discussed in previous chapters, repetitive patterns in reliefs are very different from those in building facades and other regular repetitions. Our approach is both robust and accurate in detecting the repetitive patterns in reliefs. To test our method, we collected relief images with different repetitive patterns from various freely available web sources and also from a cultural heritage site *Hampi*, in India. We also test our algorithm on building facade images and near-regular texture images. The quantitative and qualitative results depicts the robustness and accuracy of our approach.

3.1 Introduction

Repetitive patterns are present in various structures and shapes of the world at many different scales and forms. We propose a robust method to detect the approximately repeating structures in reliefs. The hierarchical method proceeds from lower to higher level features, thus robustly detects structures with partial similarity. Humans inherently recognize symmetries and repetitive patterns in objects and images. Detection of repetitions for humans happen at multiple levels of detail. At a coarse level, we may use the overall texture of a scene or part of it to find possible repetitions. We then analyze the parts of objects within it and their arrangement to find objects that repeat. If those pieces or objects are found in a similar configuration elsewhere, we identify this object as a repetition. The motivation for the design of our algorithm is very similar to this. We begin by finding reliable matches for individual components in an image. These individual matches are then verified and grouped together to get regions with possible repetitions. These grouped matches are then used to detect different repetitive patterns and elements. The detected repetitive patterns are appropriately segmented and each individual instance in the pattern is represented by a color-coded convex hulls. Now, we discuss our approach in more detail along with the observations about the repetitive patterns in reliefs.



Figure 3.1 Example reliefs with approximate repetitive patterns.

3.2 Observations and Assumptions

Repetitive patterns in reliefs have certain interesting characteristics that are different from the typical patterns found in the urban facades. This section lists some of the observations and assumptions that lead to the design of our repetition detection algorithm.

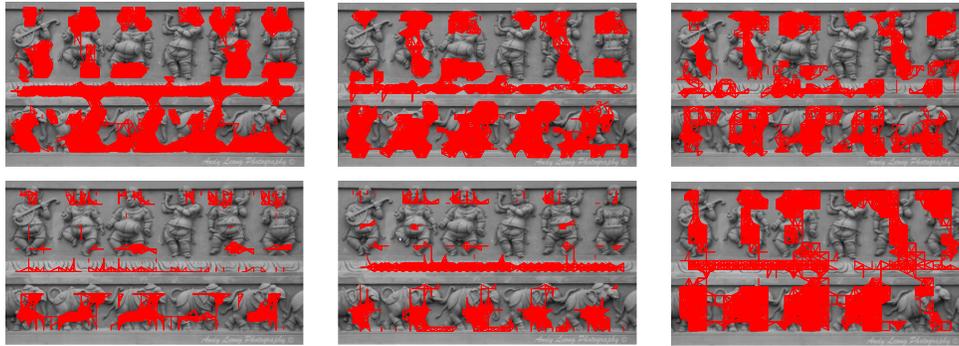
1. In general, repetitions in reliefs do not occur in a regular grid or at regular interval. Repetitive patterns often appear non-uniformly in an image (see Fig. 3.1(b) & 3.1(a)). Our algorithm is independent of factors including the number of repetitions, repetition interval and repetition direction.
2. Occasionally, in reliefs it is difficult to define repetitive patterns where each object is a single unit. Repeating elements generally have some variance in appearance from other repetitions (see Fig. 3.1(c)). Elements can have partial repetitions and we achieve robust detections for such reliefs with our algorithm.
3. It is often difficult to apply traditional rectification techniques on reliefs as detecting the vanishing line is not robust enough in images such as Fig. 3.1(a). Our algorithm is not constrained to any requirement of input image to be fronto-parallel or rectified. Detection is robust to significant image skews.



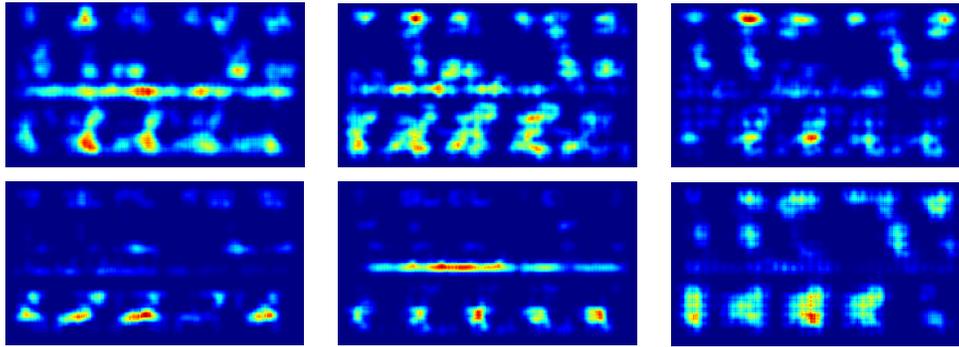
(a) Dense Feature Extraction

(b) Pairwise Feature Matching

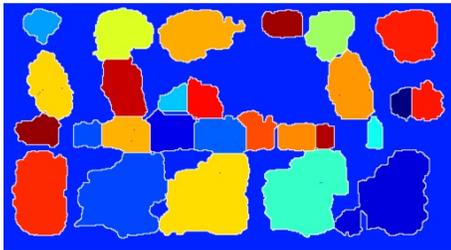
(c) Pairwise Patch Matching



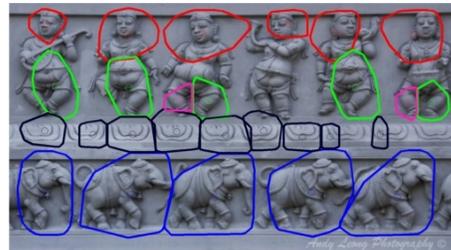
(d) Connectivity graph at different scales



(e) Score images for corresponding connectivity graphs (blue with lower scores and red with higher scores).



(f) Segmented image regions.



(g) Detected repetitive patterns.

Figure 3.2 (a) Dense sift features after removal of false matches (b) Pairwise sift matching, blue features are pairwise matches of yellow (c) Pairwise patch matching, green patches are correct match of yellow patch and red patches are false matches (d) Connectivity graph at various scales in the multi-resolution pyramid (e) Score Image for the corresponding connectivity graph in (d). (f) Segmented image regions after merging all score images with distinct labels for each region. (g) These labels are merged and grouped to produce the final output of our algorithm with color-coded convex hulls of repetitive patterns.

3.3 Pairwise Matching

As a brief overview, our algorithm initiates by building a multi-resolution pyramid of the given input image. Fig. 3.2 shows the complete pipeline of our algorithm. We process each scale independently. At each scale, we extract feature descriptors and a list of matching features are found for each feature. We then consider the features in image patches and search for similar patches using our similarity score function. These pairwise patch-matches give us a confidence score for matching image patches. Confidence scores are then used to identify the regions that are repeating using watershed segmentation of the confidence score image. We begin by presenting our framework for finding pairwise patch correspondences, then describe the detection and segmentation of repetitive patterns.

3.3.1 Pairwise Feature Matching

In order to cover the entire image, we have extracted SIFT descriptors using sift interest points and dense SIFT descriptors with a fixed step size (10 pixels, in our experiments). We convert the SIFT descriptors to ROOT SIFT. This has been shown to work well on many computer vision tasks [7]. We denote the set of SIFT descriptors \mathcal{S}_n for the image at current level n . For each $s_i \in \mathcal{S}_n$, k nearest neighbors are found using efficient implementation of kd-tree data structure. We consider s_j as a reliable match for s_i if their scales and orientations are very similar. For a pair of sift descriptors, we define a similarity score as:

$$SS(s_i, s_j) = 1 - \frac{sd(s_i, s_j) + od(s_i, s_j) + dd(s_i, s_j)}{3}, \quad (3.1)$$

where $sd(s_i, s_j)$ is absolute difference between the scales, $od(s_i, s_j)$ is absolute difference between the orientations and $dd(s_i, s_j)$ is the L1 norm of difference vector of their descriptors, each of them normalized between 0 and 1. We consider the sift pair as a strong match if $SS(s_i, s_j)$ is more than a threshold sim_thresh ($= 0.7$ in our experiments). We discard the pair if any of sd , od , dd exceeds the individual threshold (nearly $sim_thresh/3$).

Remove false matches: Trivial repetitions in images like the background plane or repetitions with very small interval are considered insignificant. To remove false matches for sift feature, we find its $num_neighbor$ nearest neighbors in sift descriptor space. We use the following false match removal algorithm to filter the set of SIFT descriptors. We remove a feature s_i from \mathcal{S}_n if:

- No nearest neighbor has similarity score greater than sim_thresh , else check the following conditions.
- At least two nearest neighbor has spatial euclidean distance less than the minimum repetition interval allowed or,
- At least one nearest neighbor are spatially very close to s_i .

The minimum repetition interval is chosen as the patch size of the current scale. This step removes a large number of insignificant features, which increases the efficiency of the algorithm. All the remaining

features in \mathbf{S}_n have strong and reliable matches along with their corresponding similarity scores. We denote the j^{th} strong match of s_i as $sift_match_{ij}$ and score with $sift_score_{ij}$.

Images with reliefs are noisy due to their textures. Although strict thresholds are used for similarity scores, we further want to increase the reliability of matches. We use a higher level matching after sift feature matching to further remove the remaining false matches. In next step of our algorithm, we introduce a higher level feature matching technique that boosts the robustness and reliability of the feature matches found in this step.

3.3.2 Pairwise Patch Matching

Matching a set of features has proved to be more reliable than matching a single feature. Next higher level of SIFT matching is matching a set of SIFT features spatially close to each other in the image space. We consider a set of overlapping image patches $\mathbf{P} = \{p_c\}$, where each patch p_c is of size $\tau_n \times \tau_n$ ($\tau_1 \sim 15$) is centered at a regular interval in the image. Let $\mathbf{s}_c = \{s_i\}$ be the set of SIFT descriptors lying within the image patch p_c . We present our patch-match algorithm for finding matches for each patch in \mathbf{P} .

For each image patch $p_c \in \mathbf{P}$, find all the patches that can possibly match p_c . To find the possible patches we use $sift_match_{ij}$ for all $s_i \in \mathbf{s}_c$. Each possible patch is centered such that the spatial position of s_i and s_j are same in their corresponding patches. We define $matching_score(p_i, p_j)$ using the spatial configuration of matching sift features in p_i and p_j . To provide flexibility, we divide each patch in 2×2 cells and use soft-binning for each matching sift feature. We find the distance of each matching sift feature in the patch to the centers of the 4 cells and then give a weight $w_i = \frac{1}{1+d_i}$, where d_i is spatial euclidean distance from the center of i^{th} cell. After considering all the matching sift features, we will get a pair of vectors v_i and v_j of size 4×1 . For a pair of patches, the $matching_distance$ is defined as the L1 norm of the difference vector of v_i and v_j . So the $matching_score = 1 - matching_distance$. Two patches have a strong match if their matching score is greater than a threshold (0.7 in our experiments).

After this step, $patch_match(p_c)$ stores position of all the matching patches of p_c along with their matching scores. The patch-match algorithm proposed above will further remove features that are not matched in groups. We remove from \mathbf{P} all the patches that do not have any strongly matching patch. The correctness of retained features are increased after the patch-match algorithm.

3.4 Grouping Patches

The pairwise patch matches found in the previous section must be grouped together to accomplish our goal of identifying repetitive patterns. Before grouping patches, we want to remove overlapping patches that have strong matches to the same patch. Repetitions in reliefs are not identical and hence there could be multiple places at which the same matching patch is centered. To find a single patch out of multiple overlapping patches, we follow a variant of the non-maximum suppression technique.

Algorithm 2 Patch Grouping Algorithm

// All considered patches $\in \mathbf{P}$

```
1: for all  $p_c \in \mathbf{P}$  do
2:    $neighbor(p_c) \leftarrow$  neighboring patches of  $p_c$ 
3:   for all  $p_i \in neighbor(p_c)$  do
4:     for all  $pair \in$ 
        $(patch\_match(p_c), patch\_match(p_i))$  do
5:       if  $config\_matches(\{p_c, p_i\}, \{pair\})$  then
6:         join centers of  $p_c$  and  $p_i$ 
7:         join centers of patches in  $pair$ 
8:       end if
9:     end for
10:  end for
11:  if  $p_c$  is not joined to any patch then
12:     $patch\_match \leftarrow patch\_match - p_c$ 
13:  end if
14: end for
```

In reliefs, the individual repetitive unit can have small variations in the shape and structure. So the neighborhood property is defined to have the trade-off between robustness and accuracy. Consider two pairs of patches (p_a, p_b) and (p_{am}, p_{bm}) where p_a and p_b are neighboring patches, p_{am} and p_{bm} are their matching patches respectively. We consider two criteria for the pairs of patches to have a matching configuration. First, spatial distance between the neighboring patches and second, relative spatial arrangement of the neighboring patches in the image space. We have kept a relaxed threshold for both the criteria providing robustness to the grouping, where as the correctness is already been verified at each level before this step. If the absolute difference of the $dist(p_a, p_b)$ and $dist(p_{am}, p_{bm})$ ($dist(\cdot, \cdot)$ is euclidean distance) is less than a threshold (~ 5 pixels) then it satisfy the first criterion. For spatial orientation, we find the angle made by the line joining the centers of each patch with the horizontal axes. If the absolute difference in the angles is less then a threshold ($\sim 35^\circ$ to 45°), then it satisfy the second criterion. We say that $config_matches(\{p_a, p_b\}, \{p_{am}, p_{bm}\})$ is true if both the criteria are satisfied. After joining the center of the grouped patches, we get a connectivity graph (Fig. 3.2(d)) in the image space where nodes are the patch centers and the edges denotes that patches have high chance of belonging to a single repetitive unit.

Merging Results of all Scales: A region of the graph with high connectivity among the neighboring patches corresponds to a strongly repeating region but we cannot guarantee that a region with low or no connectivity is not a repeating unit. All the computation is done at each scale of multi-resolution pyramid. Each scale could possibly detect different patches. To ensure completeness, we need to merge the output of all the scales. We have outputs in the form of matching patches and the connectivity graphs at each scale.

To merge the results from different scales, we create a score image from the connectivity graph at each scale. Score image has a score (between 0 and 1) at each pixel and the score denotes the confidence

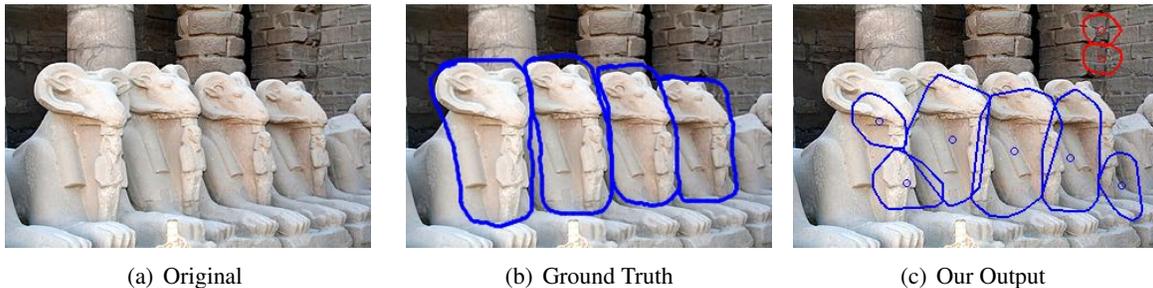


Figure 3.3 Example of an approximately repeating structure.

of that pixel belonging to a repetitive pattern. While joining the patches in Algorithm 2, we removed all the patches that are not joined to any neighboring patch. So all the patches in $patch_match(p_c)$ is joined with valid neighboring patch. We create a $score_patch(p_c)$ of size $\tau_n \times \tau_n$ with each value as the maximum score among all the matches $patch_match(p_c)$ of a patch p_c . A Gaussian mask with $\sigma_n = \frac{\tau_n}{3}$ is applied to give more weight to the center of the patch. The $score_patch(p_c)$ for p_c is added in the score image at the corresponding place. We add the score patches for all $p_c \in \mathbf{P}$ to the score image at the current level. After repeating this for all scales, we will have a score image at all the scales. Then, each image is mapped to the image at the highest scale of the pyramid. For each pixel, we take the maximum score among all the score images. We also merge $patch_match(p_c)$ for all patches at all scales to the $patch_match(p_c)$ of the highest scale. After merging the results at all scales, we create a repetition field rep_field in the image space where $rep_field(x, y)$ stores the $patch_match(p_c)$ information along with the matching scores for patch p_c which is centered at (x, y) .

3.5 Detection and Segmentation of Repetitive Patterns

After grouping patches in the above section, we have the following information available for the given input image. Score image in which each pixel gives the confidence of belonging to a repetitive pattern. For each pixel in the image space, rep_field stores all the matching pixel positions and scores. To detect the repetitive patterns we segment the score image using watershed segmentation technique [46]. We have no prior knowledge about the number of repetitive elements in the image, hence we have to use an unsupervised approach. Watershed segmentation is an automatic segmentation that considers a topographic representation of the image intensity. Intuitively, an infinite source of water is kept at the lowest basin and then watersheds or dams are formed to prevent water flow from one catchment basin to another.

In our score image, the matches with high matching score will have high intensity values and an absence of a match with zero intensity value. If we invert the score image then region with high matching score corresponds to catchment basins and hence watershed algorithm can be effectively used. Water-

shed algorithm often gives an over-segmented output. Inverted score image will form large number of catchment basins as an evident of max operation while merging all the scales. Pre-processing of the image with smoothing filters were also over-segmenting the image. So we first converted the gray-scale score image into a binary image by keeping a low threshold (between 0.01 to 0.15) and then convert it back to a gray scale image using euclidean distance transform of the inverted image. This reduces the over-segmentation to large extent and outputs larger segments with unique labels corresponding to possible repetitive regions. We develop a label merging algorithm to get the correct repetitive regions. Each watershed pixel separates two regions with labels l_1 and l_2 . Given the repetition field *rep_field*, we can say that if no pixel in l_1 has a repeating pixel in l_2 , then both l_1 and l_2 must constitute the same label. If more than β ($\beta = 10$, in our experiments) watershed pixels satisfy the constraint then we merge l_1 and l_2 into one label.

After appropriate merging of the regions, we get possible repetitive regions associated with unique labels. To detect the repetitive pattern, we need to find correspondences between the repeating element. For each pixel in a label, we find the labels corresponding to its matching pixels using *rep_field*. Large number of correspondences between two labels implies that both the label belongs to same repetitive pattern. After finding the pairwise correspondences, we group the regions to get the correct repetitive patterns with the repetitive elements.

3.6 Dataset Collection and Annotation

As per our knowledge, there is no such dataset available in the research community that can be used to evaluate the performance of repetition detection algorithms in relief images. In order to test the robustness and accuracy of approach we collected reliefs images from various sources on web like flickr, Google Images, etc. Along with them, we also collected reliefs from the ruins Vijayanagara Empire located in *Hampi*, a village in northern Karnataka state, India. Repetitive patterns in reliefs are present in abundance in the ruins. Most of the repetitive patterns we found have repetitive instances with approximate similarity and this provided us challenging test images that is used to evaluate our performance. The approximate repetitive patterns were present in different scales and at various positions on the walls.

To capture the images, we used a simple consumer digital camera. We did not use any other special device. The images were captured in various environmental conditions like strong sunlight, shadow and cloudy weather. We wanted to reduce the projective skews in the images but there was no strong constraint on that. In next section, we have discussed the performance of our approach under projectively skewed images. For collecting relief images with repetitive patterns from web, we searched using various keywords on image sharing website like Flickr.com, Google.com. The web sources provided us reliefs with various repetitive patterns from around the world. By including all the images, we built a varied and challenging collection of relief images with both exact and approximate repetitive patterns that is then used to evaluate the performance of our approach. Other than reliefs, we used images from



Figure 3.4 Example annotations of repetitive patterns in different types of relief images. In each pair of image, left to right, original and the annotated image. The two top left annotations show an example of approximate repetitions. In first example, head of a horse is bent down and hence excluded from the repetition. Similarly, trunk of elephants are not in repetition. Each image was closely analyzed to find all the approximate repetitive patterns.

two freely available datasets. We have used some of the facade images from ZuBuD database [63]. The regular texture images are taken from PSU Normal-near regular texture images.

Annotation: As the repetitive patterns have approximately similar instances, annotating the collection of images is a very difficult task. Each relief image needs to be annotated manually using a simple image editor tool. After collecting the above set of reliefs with repetitive patterns, we visually analyzed each image. For a single relief image, we start grouping visually similar objects into a single repetitive pattern. To mark a single instance of a repetitive pattern we draw a closed boundary around the object. All the occurrences of an instance in a repetitive pattern are marked using the same color. While annotating, we denote an object to be a region with the largest area repeating in a single repetitive pattern (see Fig. 3.4). We carefully mark the boundaries of the object and search the complete image for each repetitive pattern. We repeat the above process for all the repetitive pattern in an image. We only annotate a repetitive pattern if there are more than a single occurrence of the repetitive instances.

All repetitive patterns are marked by color coded closed boundaries. These closed regions can also be used for segmenting the repetitive instances out from the relief image. For automatic evaluations, these closed regions can be assigned labels. All the repeating instances that belong to a single repetitive pattern can be grouped and assigned to a single label. Fig. 3.4 shows a sample of the relief images from our collection with their corresponding annotations.

3.7 Results and Discussions

We have tested our method on a PC with 2 GHz CPU and 4GB RAM. The Matlab implementation of our method has 3.8min average run time for a typical 500x500 image with 12 to 15 levels in the pyramid. We tested our algorithm on a collection of various images as discussed in Sec. 3.6. We have shown results on representative images from the collection of reliefs, facades and NRT images.

Our algorithm gives correct detection results for reliefs with highly similar repetitions in almost all the images. When the repetition is approximate (see Fig. 3.3), the algorithm robustly detects the repetitive pattern. In reliefs, only particular parts of the object may repeat. In those cases, algorithm properly segments parts that belong to the repetitive pattern such as in Figures 3.5(a) and 3.5(e), where the heads are approximately repeating. Figures 3.5(b) and 3.5(c) show the robustness of our approach for irregular repetitions. In Fig. 3.5(b), the partial elephant is grouped with the horse pattern due to the matching back, which has same appearance to horse’s back. Fig. 3.5(d) shows detection of multiple irregular patterns. The red patterns are a result of matches between sky patches. In Fig. 3.5(f), our algorithm detected parts of the repetitive element as different pattern because of the absence of matching patches in those regions.

We prepared the ground truth results for all the images by drawing an approximate border around each repetitive elements. We evaluate our detection and segmentation algorithm using accuracy and recall measures. The segmentation performance is explained mainly by accuracy and the detection performance is explained by recall measure. We call any detected region a true positive (TP) if it belongs to the correct repetitive pattern. If a non-repeating region is assigned to a repetitive pattern, then we call it a false positive (FP), similarly a segmented region is false negative (FN) if it does not belong to correct repetitive pattern. Similar evaluation criteria is used by Basui and Cai [15]. Table 3.1 shows the detection and segmentation performance of our algorithm on the collection of images. Figure 3.8 shows result of our algorithm on facade and NRT images.

Limitations - Our algorithm has some limitations for robust repetition detections. Our algorithm is not robust to occlusions and large projective skews because of insufficient pairwise matches. In Fig. 3.7(c), the top-left window is partially detected due to occlusion by a tree. In Fig. 3.6(a), robust repetitions are detected, but for relatively large projective skew like in Fig. 3.7(a), our algorithm fails to find all the reliable matches. In images where the repetitive patterns are very close to each other for example Fig. 3.7(b) and 3.7(d), our segmentation algorithm incorrectly merge multiple elements or segments one element into two parts but still we can get satisfactory segmentations by tuning the threshold values.

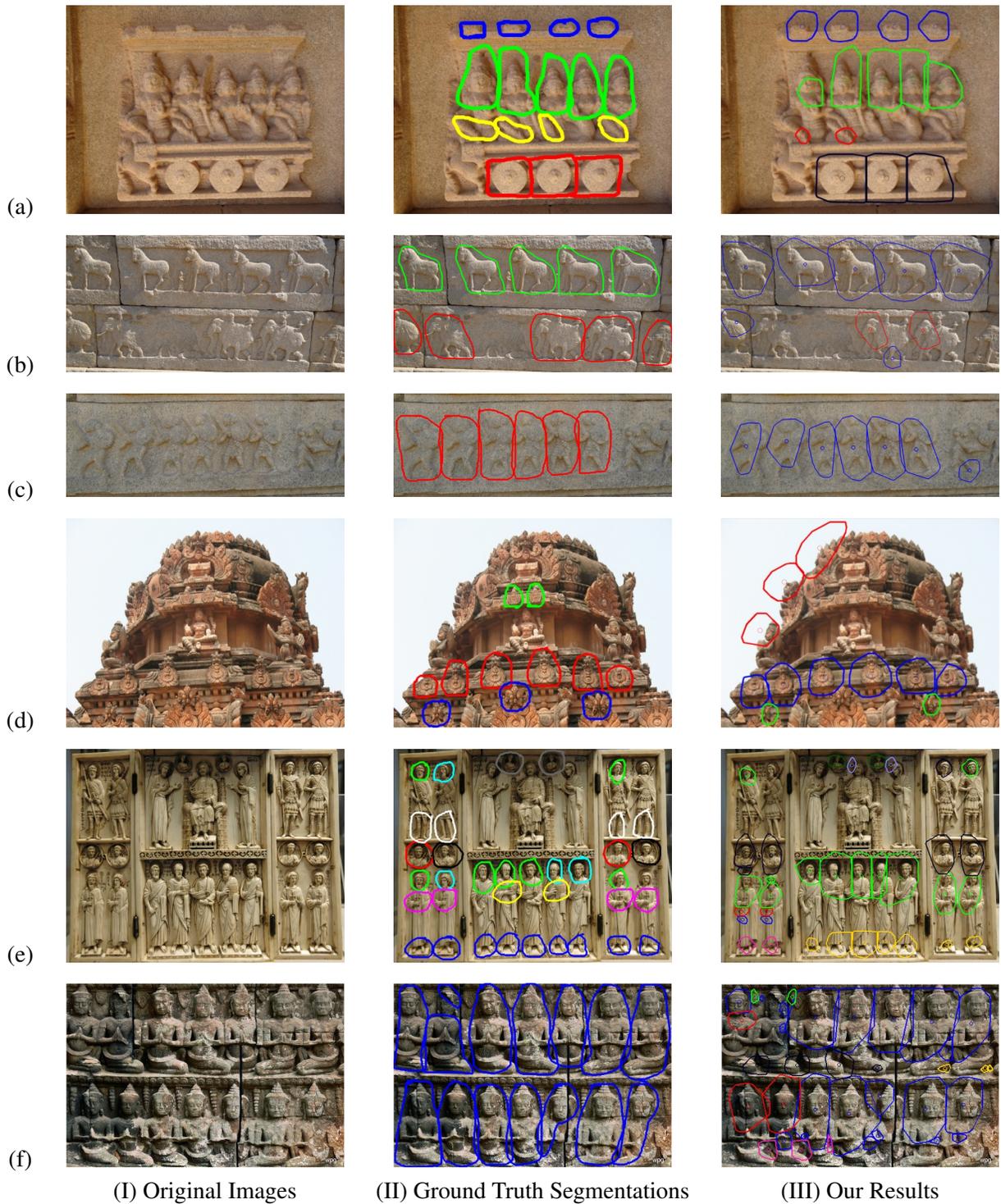


Figure 3.5 Each row shows the result of detection and segmentation on varieties of reliefs images. (a) shows detection in case of approximate repetitions. (b) Elephant's back is wrongly detected as a horse's repetition. (c) is an accurate detection. (d) Red pattern is a false positive due to repeating sky. (e) Detected multiple repetitions with irregular intervals. (f) As parts of the statue is repeating, they are identified as different repeating elements.

Table 3.1 Repetition detection and segmentation performance of our algorithm

Image Type	#Images	Avg. Accuracy	Avg. Recall
Reliefs	53	89.66%	79.77%
facades	22	85.3%	80.1%
normal-NRT	13	88.1%	58.3%

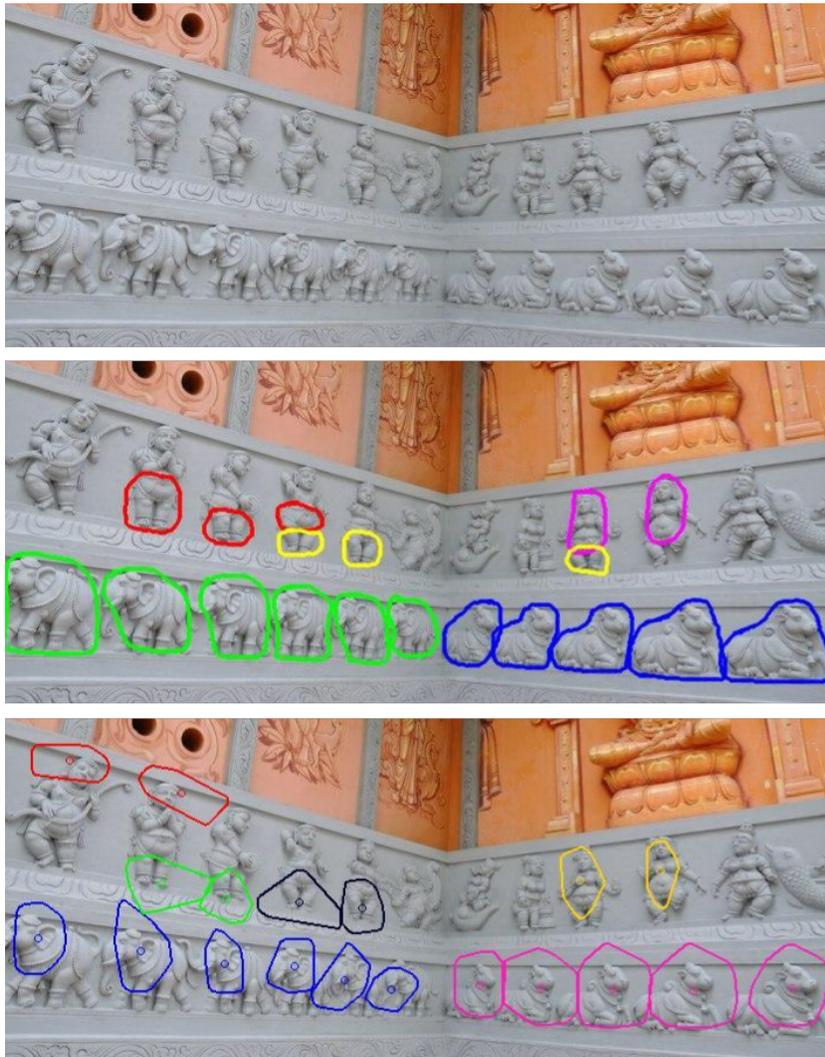


Figure 3.6 Example of robustness of our detection and segmentation algorithm in presence of multiple repetitive patterns and significant projective skews. Original image(top), Annotated image(middle), Our Result(bottom).

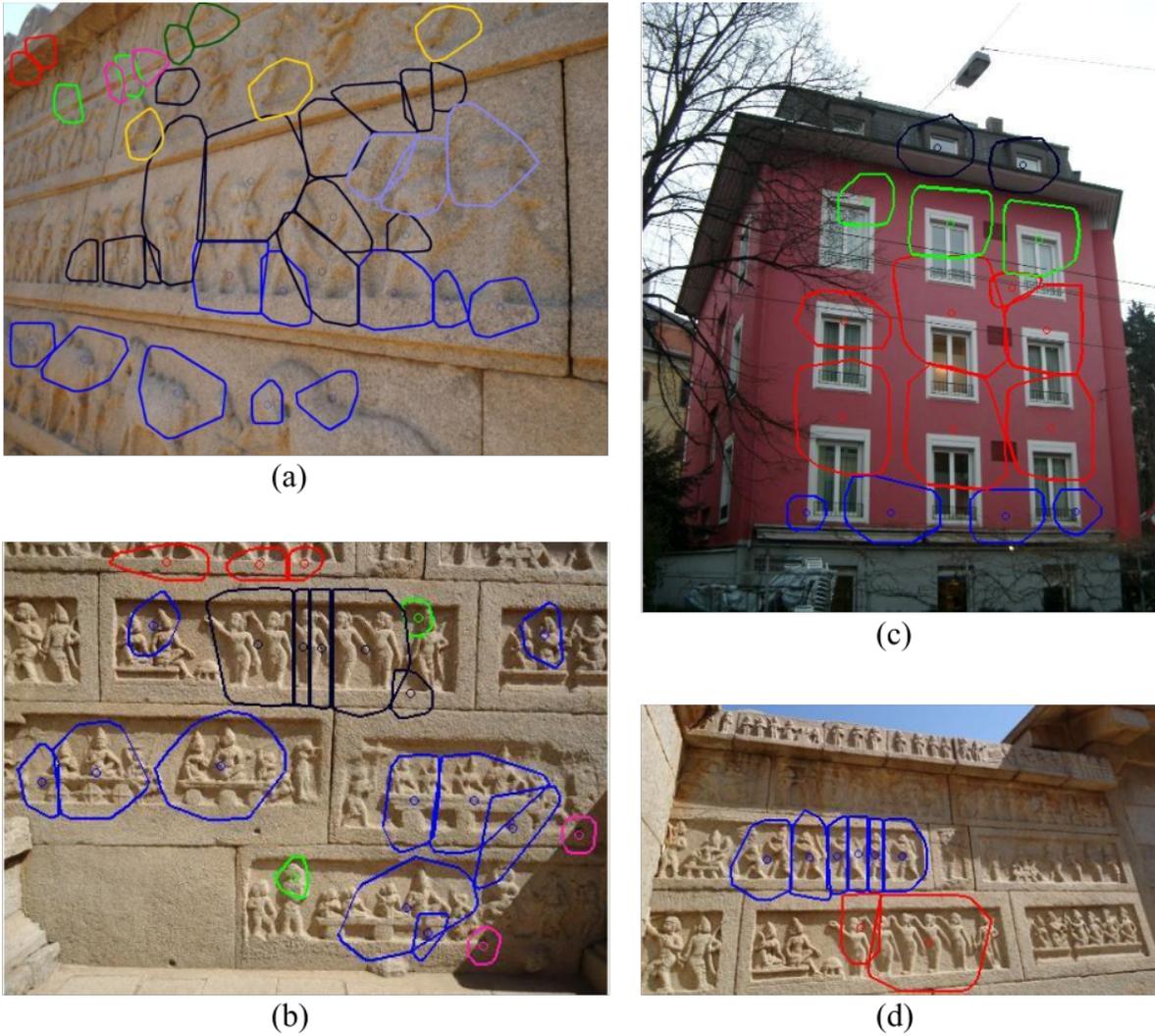


Figure 3.7 Example failure cases. (a) Large projective skews, (b),(d) close repetitive patterns, and (c) partial detection due to occlusions.

3.8 Conclusion

We proposed a robust and efficient method for detecting approximate repetitions in relief images. Our algorithm outputs labeled segmentation of the repetitive patterns by computing a convex hull of the repeating elements. We have evaluated our algorithm on images with various types of repetitions. The robustness of the algorithm is also tested on facade and near-regular-texture images. Our algorithm outputs good results for repetitions with large texture variations. We allows small changes in scale and shapes to be matched for the same repetitive pattern. Our algorithm works well for irregular and low count repetitions.



Figure 3.8 Example of our detection performance on Facades and NRT images.

3.9 Summary

In this chapter, we proposed a robust hierarchical approach for detection of approximate repetitive patterns in relief images. We process the image at multiple scales of a image multi-resolution pyramid. At each scale, we find dense features in a hierarchical order, starting from the lowest level features to higher level features. We then merge the results from all the scales. The merged results are then processed to detect and segment the multiple repetitive patterns in the image. We also compute a dense pixel-wise *rep_field* that stores the corresponding positions of the other instances in the repetitive pattern. The existing works on repetition detection does not outputs good results on relief images due to the factors such as - approximate repetitive patterns, irregular repetitions, similar background and foreground, unknown repetitive instances, projective skews. Quantitative and qualitative results show that our approach is robust and accurate in detecting the approximate repetitive patterns in reliefs. We also show results on building facades images and near-regular texture images.

Chapter 4

Shape Reconstruction from a Single Relief Image

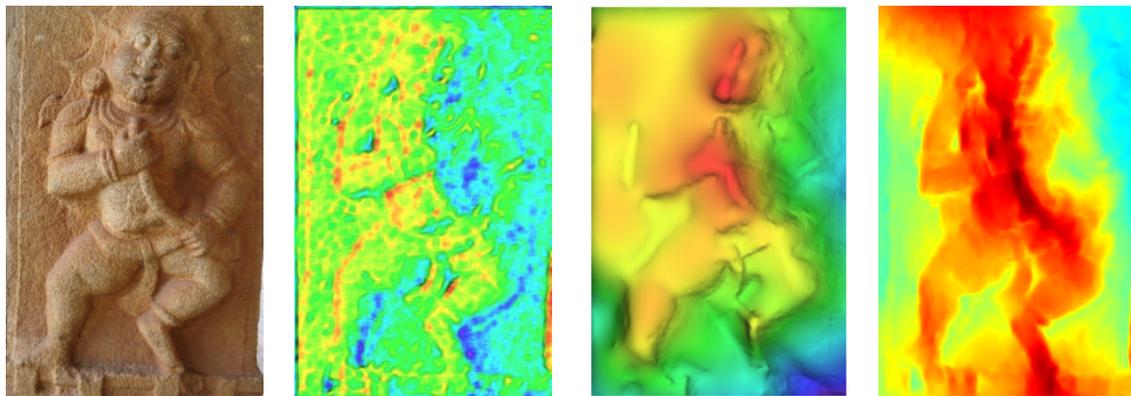
We describe our method to reconstruct depth map from a single relief image. Our approach is a data-driven approach that takes in a single relief image and computes the appropriate depth map for the image. Single view reconstruction is being done using various cues and assumptions about the illumination, surface reflectance of the object and image projection models. Using a single view shape from shading method directly on relief images do not produce satisfactory results. We use an exemplar dataset of relief images and their corresponding depth maps to learn relief specific shape priors. These shape priors are then integrated to the shape from shading results using a MAP framework to produce results of better quality. Our shape priors for reliefs are reliable and can be used as an initial estimate in other shape reconstruction algorithms. We evaluate our approach on various kinds of reliefs collected from different sources. Along with reliefs, we present experiments on Human Body Poses dataset [26]. We also show the comparison of our method to the state-of-the-art methods.

4.1 Introduction

Shape reconstruction from a single image is an ill-posed problem. Various simplifying assumptions and regularization constraints are used to solve for surface normals from a single image. Still it needs strong prior knowledge about the scene or the object under consideration. An effective prior arises from the fact that surface normals at occluding contours lie in the image plane [51], [10]. However, it is very difficult to correctly detect the occluding contours in a relief, making it ineffective. In this work, we aim to learn a relief specific prior from a set of sample images and their corresponding depth maps. We note that humans perceive shape from a single image by not only estimating the properties of the environment but also by a higher level recognition. In other words, prior knowledge from previously seen instances improves the reconstruction for us. In our approach, we encode the prior knowledge in a non-parametric way using a training database of reliefs. Shape reconstruction using exemplar database has been shown to work well in many highly similar class specific objects or shapes [11], [26] and in photometric stereo [29].



(a)



(b)

(c)

(d)

(e)

Figure 4.1 Shape Reconstruction from single relief image (Depth maps are shown with psuedo-color visualization, red is near and blue is far). (a) Complete exemplar dataset consists of only 7 relief images, (b) Original Relief Image, (c) Depth Map obtained by SfS of Tsai et al. [55], (d) Depth Map obtained by Barron and Malik [10], (e) Depth Map obtained by our approach. The depth map obtained from (c) is noisy. We learn shape priors for reliefs to improve the shape reconstruction. Note that we recover overall geometry as well as details of face, legs and the left part of relief that is not recovered in (d). Results of [10] were poor for color images, so we used gray scale image to obtain (d).

As mentioned above a variety of techniques are available for reconstruction of 3D structures including relief carvings. However, each of them has its own limitations, especially for large scale usage.

Highly accurate systems such as laser scanners are extremely expensive for use by common man. At the other end of the spectrum, multi-view stereo methods can work with images from a consumer digital camera, but require large number of images. Other methods also make similar trade-offs between cost, ease of use and accuracy. Our goal is to come up with an easy to use and least expensive method that improves the accuracy of reconstruction of relief carvings.

The most effective approach to recover shape from a single image of an object with very limited depth variation is to use the classical shape from shading (or SfS) with appropriate constraints to recover the normals. A depth map is then inferred from the computed normal map. However, the approach assumes that we know the lighting direction or, in some cases, that it has a single frontal light source at infinity (parallel light rays). These assumptions hold good for objects images under controlled conditions in a laboratory. However, images of reliefs acquired in real world are illuminated by a complex illumination from the environment and is rarely frontal. As described later, we overcome this challenge using a simple modification to the imaging process using consumer cameras without the need of any additional hardware. We assume Lambertian surface reflectance model and orthographic image projection. These assumptions are valid for reliefs as the surface of reliefs are very rough and also the shape variation is very small as compared to the distance between the camera position and reliefs. The albedo, however, is not constant across the images, but is considerably uniform as the relief is made up of a single stone. We test our approach on both synthetic and real datasets. Our approach shows improvements over the shape from shading methods and is able to capture both overall shape and finer details.

Even with mostly uniform albedo of the carved reliefs, the SfS results in highly noisy depth map (see Fig. 4.1). As mentioned before we use a relief specific prior that significantly improves the results and we learn them from sparse coding of sample relief images. Sparse representations of image patches are widely used for many computer vision applications like color image denoising [18], demosaicing and image inpainting [45]. We solve the single image shape reconstruction problem by incorporating prior knowledge of relief surfaces. These priors are obtained by representing the query image signals as a sparse linear combination of basis signals learned from an exemplar set of images. Reliefs provide two important priors: i) the height variation across a relief is small and continuous especially in low reliefs and, ii) the overall shape of the relief is a flat plane with surface variation above the plane (see Fig. 4.1). Learning the relationship between the image appearance and the corresponding shape patches inherently reduces ambiguities caused by looking at an individual pixel. By using sparse representations of image patches, we are able to capture the correlation between the image appearance and local shape variations. We now describe our method in detail along with experiments on both real dataset of reliefs and synthetic dataset of human body poses.

4.2 The Proposed Approach

The core of the proposed approach involves two independent processes for estimating the shape from a relief image. The first one is based on recovery of normals using the lambertian reflection laws.

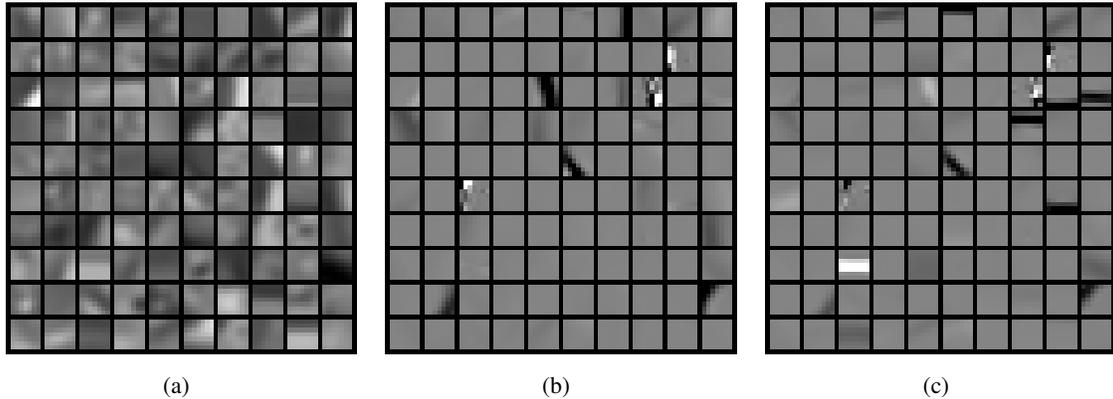


Figure 4.2 Sample of dictionary elements learned from exemplar images and their surface gradients. (a) elements from image appearance. (b) and (c) elements from surface gradients in x and y-direction respectively.

Independently, we use the prior distribution of image patches from other relief images to estimate local geometric shapes. This is computed using a sparse coded representation over a relief dictionary, and serves as our relief prior for the normals. We convert surface normals to surface gradients. A MAP framework is introduced to integrate the results of the two estimates.

We also present results on a synthetic dataset of body poses, in addition to a real dataset of relief images collected from ancient heritage sites. As we see from the results, our algorithm is able to capture the overall shape and local geometric shapes, and the approach can be extended to work with objects other than reliefs.

4.3 Shape from Shading

Most shape from shading algorithm assumes lambertian reflectance, distant point light source, orthographic projection and a constant known albedo. Tsai and Shah [55] proposed a SfS algorithm with linear approximations that was one of the better performing algorithms in the survey by Zhang *et al.* [80]. It is a local approach where they apply discrete approximation of the gradients first, and then linearize the reflectance function in terms of the depth directly, instead of the gradients. Their approach performed well for real images, but is sensitive to noise in the intensity image. We combine their approach on the relief images along with a modified imaging process, and then improve the results by using the relief priors learned from sparse representation of query image patches.

4.4 Image under known Illumination

As mentioned above, the illumination of real-world relief carvings is often complex due to the environment and cannot be assumed to be a single distant light source. This can be simplified if we can

capture two images of the same relief from approximately the same point of view, one with and the other without a flash. Flash photography is popularly used for various vision tasks such as ambient image denoising, detail transfer from flash to ambient, white balancing, red-eye correction, etc. [54]. We use it to get an image of the reliefs under a known illuminant, the camera flash in this case. We acquire two images of the object using a tripod to ensure the pixel alignment. It avoids the problem of image registration which is not the focus of our work.

Let \mathbf{A} be the ambient light image and \mathbf{F} the image using flash. We apply gamma correction on \mathbf{A} and \mathbf{F} to bring both the images in the same linear space. We keep the focus, aperture and ISO settings same for both images. If Δt_A and Δt_F are the exposure times for the ambient and flash images respectively, then we compute the pure flash image \mathbf{PF} as shown below

$$\mathbf{PF} = \mathbf{F} - \mathbf{A} \frac{\Delta t_F}{\Delta t_A} \quad (4.1)$$

Fig. 4.3 shows an example of computing pure flash image. If the distance between the camera position and the reliefs is large enough, we do not consider the angular and radial illumination falloff of the flash. However, these quantities can be easily learned and integrated from a single image of a white plane using the camera and the flash. Given a pair of flash and non-flash images, complex natural illumination can be simplified in the above manner to improve the accuracy and robustness of the SfS process.

In spite of the illumination correction, the SfS results are often noisy due to violations of the pure lambertian reflectance and uniform albedo assumptions of the object. Another common problem is a smooth but significant deviation from planarity of the relief base, often caused by minor illumination fall-offs. We now look into the process of computing the shape prior for the image to overcome some of these problems.

4.5 Learning the priors for relief image

Our approach is similar in principle to the recent work from Panagopoulos *et al.* [52]. They proposed a data-driven approach that learns a dictionary of geometric primitives and their appearances. The dictionary is used to learn a small set of hypotheses about the local 3D structure for the given image to get an initial guess that is then regularized by an MRF optimization layer. In our approach, we learn the relief priors using an overcomplete dictionary with a composite signal of image appearance, surface gradients, and light source direction. To reconstruct the geometry of a given image, we sample the image densely at each pixel and for a patch around this pixel, we reconstruct a signal from the learned dictionary using a sparse linear combination of the basis signals. We use the sparse representations in learning the correlation between the image appearances and the corresponding shape variation.

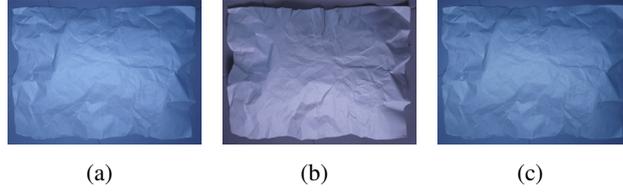


Figure 4.3 Example of Pure Flash image computation. (a) Flash+Ambient Image \mathbf{F} (b) Ambient Image \mathbf{A} (c) Pure Flash Image \mathbf{PF}

4.5.1 Dictionary Learning

For each instance in the exemplar set, we know the gray scale image appearance I_k , surface gradients P_k and Q_k in x and y directions respectively, and the light source direction S_k . A signal $w \in \mathbb{R}^d$ in the dictionary encodes the correlation between the appearance, surface gradients and light source direction. We represent the intensity in image appearance by a square patch p densely sampled at each pixel of I_k and surface gradients at that pixel by z_x and z_y . Each signal w is then constructed by concatenating p , S_k , z_x and z_y . Given the densely sampled signals in each instance, we learn the overcomplete dictionary as follows:

$$\{D, \alpha_i\} = \arg \min_{D, \alpha_i} \|w_i - D\alpha_i\|_2 \quad \text{s.t.} \quad \|\alpha_i\|_0 < L \quad (4.2)$$

where D is the dictionary, w_i are the signals, α_i are the sparse representation of signals, and the constant L ($L = 3$, in our experiments) defines the required sparsity level.

For basis learning, we use the K-SVD algorithm presented in [6]. We learn the basis dictionary $D \in \mathbb{R}^{d \times n}$ where n is number of basis signals, such that each signal is represented by a few basis element. Fig. 4.2(a) shows some of the learned dictionary patches for the relief images dataset.

4.5.2 Sparse Coding

Once the basis is learnt, any query signal $\mathbf{q} \in \mathbb{R}^d$ can be decomposed sparsely over the basis and can be reconstructed i.e.,

$$\mathbf{q} \approx D\alpha \quad \text{s.t.} \quad \|\alpha\|_0 \ll L, \quad (4.3)$$

where α is the sparse representation of the signal and $\|\cdot\|_0$ is l_0 pseudo-norm, which gives a measure of number of non-zero entries in a vector.

For any given image, the surface gradients are unknown. We form query signals $\mathbf{q} \in \mathbb{R}^d$ sampled densely at each pixel, with their gradient values set to zero. To represent this incomplete signal from the learned overcomplete basis, we mask the dictionary D such that the surface gradients signals are set to zero. We use the Orthogonal Matching Pursuit(OMP) technique to learn the α such that the query signal \mathbf{q} is sparsely reconstructed from the basis signals. The learned α is then used to recover the corresponding surface gradient values for each pixel in the image.

4.5.3 Shape Recovery using relief priors

Given an image of a relief carving, we have now computed a shape prior and a noisy normal field from SfS. We pose the integration as a maximum-a-posteriori (MAP) estimation problem from these quantities. To achieve this, we convert normals to surface gradients and compute the most likely surface gradients \hat{G} at each pixel of the image, given the observation G_s , the gradients computed from SfS. This may be written as:

$$\hat{G} = \arg \max_G p(G|G_s) = \arg \max_G p(G_s|G)p(G|G_p)$$

where G_p is the learned surface gradient priors. Note that the denominator in the Bayes formulation is not relevant for computation of $\arg \max$. The two densities, $p(G_s|G)$, and $p(G|G_p)$ models the error probabilities in the SfS and prior computations respectively. The two are estimated from ground truths of the training samples. Assuming normal distributions, the minimization has a closed form solution of the form: $\hat{G} = \alpha G_p + (1 - \alpha)G_s$. α is given by $\sigma_s^2/(\sigma_s^2 + \sigma_p^2)$, where σ_s^2 and σ_p^2 are the variances of the SfS and prior depth error distributions. The surface gradients thus obtained are integrated by affine transformation of gradients using diffusion tensors [4].

4.6 Datasets

To test the accuracy and robustness of our approach, we collected relief images from various sources. We searched on image sharing websites for relief images and choose appropriate images that satisfy our assumptions about the input image and are also challenging in terms of fine shape variation and unknown uncontrolled environment conditions. To create a varied and challenging dataset we collected images ranging from *low-reliefs* to *medium* and *high reliefs*. For images that are collected from the web source, we do not have the corresponding ground truth depth maps. Hence, these images can only be used for qualitative evaluations.

Apart from the web sources, we also collected images from the cultural heritage site *Hampi*, in India (refer Sec. 3.6). The ruins of the Vijayanagara empire has large number of reliefs at various places in *Hampi*. Some are partially damaged and distorted where as some are still intact. The ground truth depth map for a relief image should consist of a depth value for each pixel of the image. To create the ground truth dataset for relief images, Shape from structured lighting (refer Sec. 2.3.2) was not feasible given the uncontrolled environmental conditions like sunlight that would not allow the structured-lighting pattern to be captured by the camera. Structured-lighting techniques are more feasible in a laboratory like controlled environment. Looking at the recent improvements in the robustness and accuracy of Multi-view Stereo techniques (refer Sec. 2.3.1), we decided to use them in creating the ground truth dataset for the relief images we collected from *Hampi*. For each relief image, we collected sufficient number of images such that each point on the relief is covered by atleast three images. Then we used bundler [66] followed by dense reconstruction using PMVS [23]. The dense point cloud obtained by PMVS is then back projected on the image and a gaussian interpolation is used to achieve pixel wise

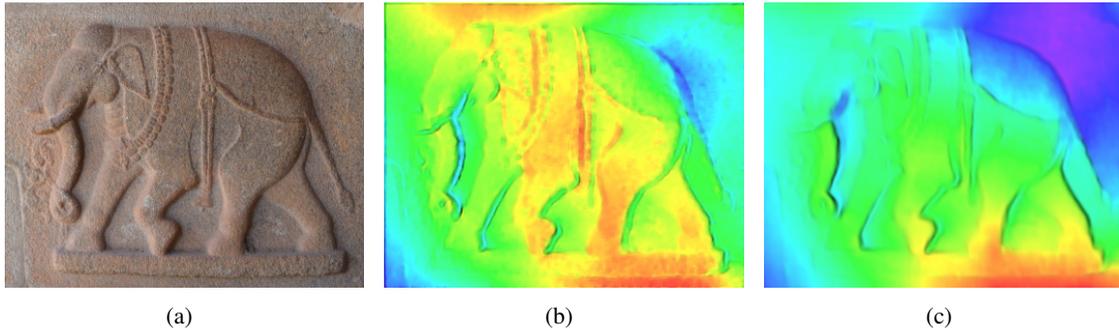


Figure 4.4 Comparison between two variants of signal construction discussed in Sec. 4.7. (a) Original Image, (b) Depth map using Pixel-wise Signal Construction, (c) Depth map using Patch-wise Signal Construction. Note that, (b) captures finer details where as (c) is more smoothed shape.

depth correspondence. The exemplar set of relief images consists of 7 images with different lighting directions (see Fig. 4.1). The albedo is mostly uniform across the images with minor variations.

We also test our approach on synthetic dataset of Human body poses [26]. Human body poses dataset consists of 12 images with corresponding depth maps. We use a subset of these images to form the exemplar dataset. In the next section, we discuss the experiments and analyze our performance on these datasets.

4.7 Experiments and Results

We test our approach on a real dataset on relief images and a synthetic dataset of Human body poses. Along with the signal representation discussed in Sec. 4.5, we also test our approach with a modified signal representation. We learn the relation between a local image appearance patch and the surface gradient patches. So, each query signal at a pixel estimates the surface gradients for a patch centered at that pixel. We refer this as patch-wise approach, and the former as pixel-wise approach in accordance with the type of surface gradient learning. Fig. 4.2(b) and 4.2(c) shows the learned dictionary patches for surface gradients in x and y directions respectively. The patch-wise signal will have the following effects on prior learning. (i) Query signals is more incomplete in patch-wise, so the sparse representation may be less accurate. (ii) As each pixel will find surface gradients for a patch, the overall shape will become more smoothed and it may remove the finer geometric details. Fig. 4.4 shows the comparison between the two methods of signal construction.

4.7.1 Quantitative Evaluation

We use the exemplar dataset to evaluate our algorithm quantitatively. We learn the overcomplete dictionary using all the images except the test image. We compare our approach to other shape from shading approaches. Choosing a good error metric is important for quantitative evaluation. Comparing

	Reliefs	Human Body Poses
Tsai <i>et al.</i> [55]	0.03422	0.02817
Barron and Malik [10]	0.01868	0.01811
Our approach (Patch wise)	0.02278	0.01337
Our approach (Pixel wise)	0.02212	0.01412

Table 4.1 Quantitative Results as Average Mean Squared Error for Reliefs and Human body poses dataset.

absolute depth values is not an appropriate way of comparing the approaches. We choose our shape evaluation metric as :

$$N - MSE(\hat{N}, N^*) = \frac{1}{n} \sum_{x,y} \arccos(\hat{N}_{x,y} \cdot N^*_{x,y})^2 \quad (4.4)$$

This is the mean squared error between the angle the normal field \hat{N} of our estimated shape and the normal field N^* of the ground-truth shape. This error metric is invariant to shifts in depth Z . Table 4.1 shows the quantitative results as average N - MSE for both the datasets. Our approach significantly improves upon the SfS results of Tsai *et al.* [55]. All the results were computed using a very small exemplar set consisting and we believe that performance of our approach should improve by using a larger representative exemplar dataset.

4.7.2 Qualitative Evaluation

In Fig. 4.5, we show results on a variety of relief images using our pixel-wise and patch-wise approaches. These images were chosen such that they satisfy the approximate uniform albedo and single distant point light source assumptions. Still, the reliefs have some ambient illumination and small albedo variations. Our pixel-wise approach performs better in 4.5(a), 4.5(b) and 4.5(f). We are able to recover the overall shape of the relief and also local shape variations 4.5(c), 4.5(d) and 4.5(e). Note that, our technique performs well in case of different lighting directions and the results can further be improved with the availability of images with and without flash as discussed in Sec. 4.4. Fig. 4.6 shows our result on body poses dataset. We correctly recovers the difference in hand positions in 4.6(d) and legs positions in 4.6(a), 4.6(b).

Failure Cases: Although our approach works under various lighting directions, certain conditions significantly hampers our performance. Our approach fails in case of cast shadows and harsh lighting conditions. We can incorporate the illumination problem with a pair of flash and non-flash images, as discussed in Sec. 4.4. Also our approach does not output correct shape in presence of large albedo variations. Fig. 4.7 shows examples for these cases.

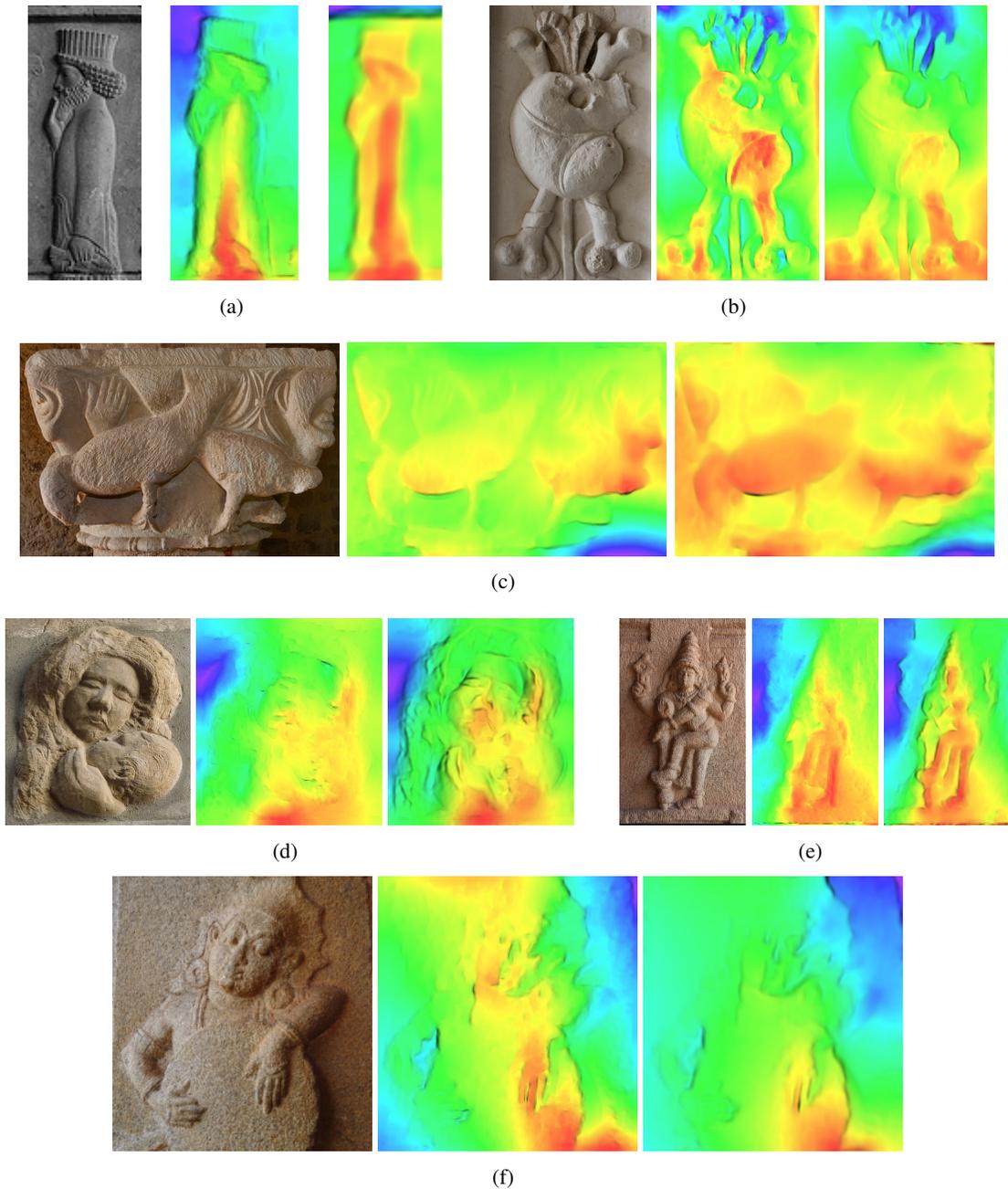


Figure 4.5 Qualitative results of our approach on relief images collected from various sources. In each instance, three images are the original image, our pixel-wise and patch-wise results, respectively. All these results are computed using the same dictionary learned on the exemplar relief images. The results shows robustness of our approach in presence of ambient illumination along with point light sources.

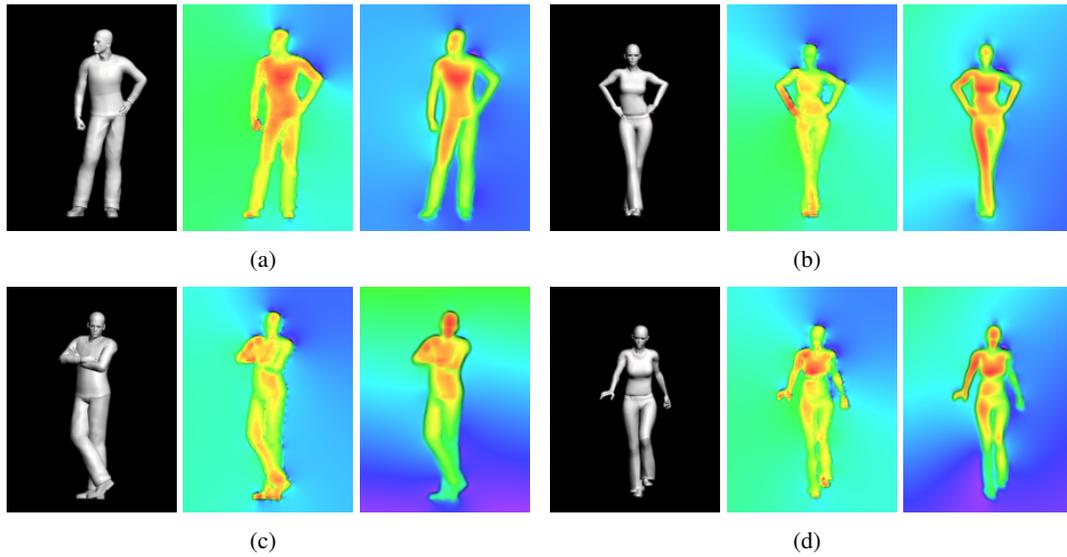


Figure 4.6 Depth Maps obtained from our approach on Human body poses dataset [26]. Each instance is shown as original image, results of pixel-wise and patch-wise approaches as depth map respectively. The dictionary was learned using a set of 12 exemplar images. The depth variation of both the legs are correctly estimated in (a) and (b), and the depths of head in (c) and (d). Note that pixel-wise method is able to recover the depth variation of feet.

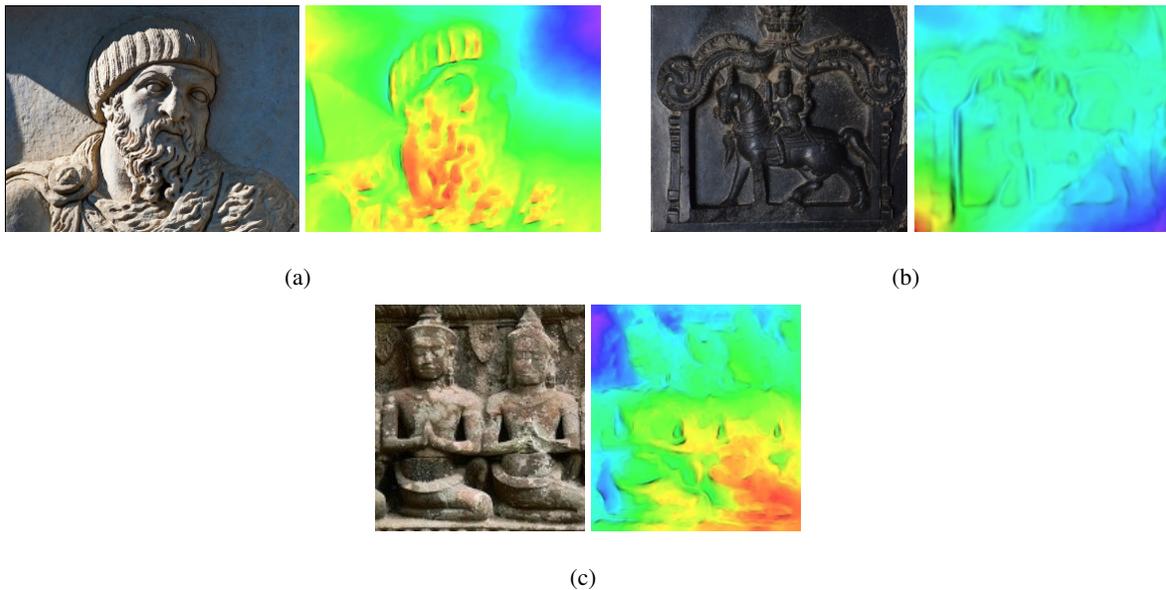


Figure 4.7 Example of failure cases. (a) Significant errors in shape reconstruction due to strong sunlight and cast shadows. (b) Incorrect shape reconstruction because of the violation of lambertian reflectance assumption. (c) Non-uniform albedo results an error in overall shape recovery.

4.8 Conclusion

We solve the shape recovery problem using a single view of relief surfaces. Reconstructing shape from relief images is a challenging task because of the uncontrolled illumination environment, so, using laser scanners or structured lighting is not always feasible. We solve the problem in two independent steps. We estimate the surface gradients using the SfS technique. The obtained gradients are noisy given the strong assumptions of SfS. We use a set of exemplar images with their corresponding shapes to learn relief specific priors. The correlation between the local image appearance and the geometric shape is learned using sparse representation technique. We remove the unnecessary complex illumination using a pair images with and without the flash. It gives us the relief image under a known illumination. After learning the relief priors, we recover the most appropriate shape by integrating the relief priors using a MAP framework. Our approach is tested on both synthetic and real datasets and result shows that our approach is able to recover both overall geometric model and local shape variations.

4.9 Summary

In this chapter, we discussed our method for reconstructing shape from a single relief image. We used sparse representation techniques to learn relief specific shape priors. The particular way of construction of reliefs provide us with useful cues that can be exploited for reconstructing shape from a single image. We learned the correlation between the image appearances patches and shape variation patches using an exemplar set of images with the corresponding shape variation. Shape priors learned by our approach can also be used as an initial depth estimate or as a depth prior for the given image. Most of the previous approaches for solving single view shape reconstruction problem does not work properly in case of reliefs because of the noisy surface texture. Qualitative and quantitative results show that our approach is able to recover the shape variation both at a coarse level and a finer level.

Chapter 5

Conclusions and Future Work

In the first part of the thesis, we presented a robust approach for detection and segmentation of approximate repetitive patterns in relief images. Given the nature of the reliefs, this problem is more challenging problem than detection of repetitive pattern in building facades or near regular textures. We approached the problem by a simple intuition of how a human detect repetitive patterns in daily life. Our hierarchical approach begins at the lowest level feature matching where we prune large number of features resulting in a smaller search space for the next level of feature matching. After matching the feature at the highest level, we group the features to find the repetitive patterns. To capture various repetitions, we process the input relief image on a multi resolution pyramid. To test our approach, we collected challenging relief images from various sources. Qualitative and quantitative results these images show the robustness and accuracy performance of our approach. Other approaches for detection repetitive patterns have strong assumptions about the given image and hence, performs poorly on the challenging relief images. Our approach is robust in detecting and segmenting approximate repetitive patterns as shown in the Sec. 3.7. Our approach is also able to detect repetitive patterns in facades and near regular texture images.

In the second part of the thesis, we presented an efficient and robust approach for shape reconstruction from a single relief image. To judge the 3D shape of an object, we humans also use a higher level recognition along with estimating various parameters like surface reflectance, illumination, etc. We followed this intuition to come up with a data driven approach for reconstructing depth map from a single relief image. Estimating the surface normals for a single pixel does not provide reliable observations given the real images of reliefs. We learn shape priors using a generic exemplar set of relief images and their corresponding depth maps. Our data driven approach learns a dictionary to correlate the image appearance patch with the corresponding shape variation. For a given relief image, we find an appropriate shape for each patch in the image using this learned dictionary. The relief priors learned by this are then integrated to noisy shape estimated using SfS technique. To test our approach, we collected various relief images captured in different conditions from various sources. The quantitative and qualitative results discussed in Sec. 4.7 shows the robustness and accuracy performance of our approach. The relief

priors learned from our approach can also be used as initial estimate or depth prior by other approaches. We are able to capture both coarse and fine shape variations in the relief images.

The work in this thesis solves interesting and challenging problems for relief images. Reliefs are different from other objects in many ways as we discussed in the previous chapters. We proposed a novel and robust method for detecting approximate repetitive patterns in reliefs. Significant amount of work has been done in detecting similar repetitive patterns in facades and near regular texture images, but very little efforts are made to solve the problem of detecting approximate repetitive patterns. Our work opens up ample opportunities in future that can be explored in this field. We see a possibility of a multi-core algorithm for detection of approximate repetitive patterns. As in our approach, we process each scale independently and then later merge the results from each scale. The results of pairwise correspondences detected by our algorithm can be exploited in various computer vision applications. The repetitions can also help in reconstructing partially damaged structures using region growing and graphical model techniques. The detected objects can possibly be used to describe and retrieve similar objects from a large database of images.

Shape reconstruction a single image is a very classical problem and we approached the problem in a data-driven manner. We used sparse representation technique to find the correlation between the image appearance patch and the corresponding shape variation patch. Repetition detection can also be exploited in shape reconstruction from a single image. If we know that there are two approximately similar instances are present in a repetitive pattern, and if we know the shape variation of one of the instances, then the pixel wise correspondences provided by our repetition detection algorithm can be used to infer the shape variation in the other repeating instance. This problem can be explored by modeling for relief surfaces with non-lambertian reflectance. Further, we believe that along with the shape variation, we can also learn the illumination model for the given image in an iterative way where we can keep one of the parameters as known and then find the most likely values for the other parameter. One could also extend the applicability of our approach for generic objects that roughly satisfy the assumptions.

Related Publications

- Harshit Agrawal and Anoop M. Namboodiri, “Detection and Segmentation of Approximate Repetitive Patterns in Relief Images”, in Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP), 2012, Article No. 46.
- Harshit Agrawal and Anoop M. Namboodiri, “Shape Reconstruction from Single Relief Image”, in Proceedings of Second IAPR Asian Conference on Pattern Recognition (ACPR), 2013.

Bibliography

- [1] Digital Heritage, International Congress, 2013. <http://www.digitalheritage2013.org/>.
- [2] International conference on cultural heritage, 2014. <http://www.culturalheritage2014.eu/>.
- [3] A. Adams, N. Gelfand, J. Dolson, and M. Levoy. Gaussian kd-trees for fast high-dimensional filtering. *ACM Transactions on Graphics (TOG)*, 28(3):21, 2009.
- [4] A. Agrawal, R. Raskar, and R. Chellappa. What is the range of surface reconstructions from a gradient field? In *Computer Vision–ECCV 2006*, pages 578–591. Springer, 2006.
- [5] H. Agrawal and A. M. Namboodiri. Detection and segmentation of approximate repetitive patterns in relief images. In *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP '12*, pages 46:1–46:8, New York, NY, USA, 2012. ACM.
- [6] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 54(11):4311, 2006.
- [7] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2911–2918. IEEE, 2012.
- [8] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics-TOG*, 28(3):24, 2009.
- [9] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized patchmatch correspondence algorithm. In *Computer Vision–ECCV 2010*, pages 29–43. Springer, 2010.
- [10] J. T. Barron and J. Malik. Color constancy, intrinsic images, and shape estimation. In *Computer Vision–ECCV 2012*, pages 57–70. Springer, 2012.
- [11] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques, SIGGRAPH '99*, pages 187–194, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [12] K. L. Boyer and A. C. Kak. Color-encoded structured light for rapid active ranging. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (1):14–28, 1987.
- [13] D. Bradley, T. Boubekeur, and W. Heidrich. Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

- [14] I. Bühlhoff, H. Bühlhoff, and P. Sinha. Top-down influences on stereoscopic depth-perception. *Nature neuroscience*, 1(3):254–257, 1998.
- [15] Y. Cai and G. Baciú. Higher level segmentation: Detecting and grouping of invariant repetitive patterns. pages 694–701, 2012.
- [16] P. Doubek, J. Matas, M. Perdoch, and O. Chum. Image matching and retrieval by repetitive patterns. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 3195–3198. IEEE, 2010.
- [17] J.-D. Durou, M. Falcone, and M. Sagona. Numerical methods for shape-from-shading: A new survey with benchmarks. *Computer Vision and Image Understanding*, 109(1):22–43, 2008.
- [18] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15(12):3736–3745, 2006.
- [19] P. Fichtler and P. Eisert. Adaptive colour classification for structured light systems. *Computer Vision, IET*, 3(2):49–59, 2009.
- [20] F. Forster. A high-resolution and high accuracy real-time 3d sensor based on structured light. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pages 208–215. IEEE, 2006.
- [21] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International journal of computer vision*, 40(1):25–47, 2000.
- [22] Y. Furukawa and J. Ponce. Carved visual hulls for image-based modeling. In *Computer Vision–ECCV 2006*, pages 564–577. Springer, 2006.
- [23] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8):1362–1376, 2010.
- [24] M. Goesele, B. Curless, and S. M. Seitz. Multi-view stereo revisited. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2402–2409. IEEE, 2006.
- [25] M. Habbecke and L. Kobbelt. Iterative multi-view plane fitting. In *Proc. Vision, Modeling, and Visualization Conference*, pages 73–80, 2006.
- [26] T. Hassner and R. Basri. Example based 3d reconstruction from single 2d images. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW’06. Conference on*, pages 15–15. IEEE, 2006.
- [27] C. Hernández Esteban and F. Schmitt. Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3):367–392, 2004.
- [28] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340. ACM, 2001.
- [29] A. Hertzmann and S. M. Seitz. Example-based photometric stereo: Shape reconstruction with general, varying brdfs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1254–1264, 2005.
- [30] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

- [31] R. Huang and W. A. Smith. Shape-from-shading under complex natural illumination. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 13–16. IEEE, 2011.
- [32] X. Huang, J. Gao, L. Wang, and R. Yang. Exemplar-based shape from shading. In *3-D Digital Imaging and Modeling, 2007. 3DIM'07. Sixth International Conference on*, pages 349–356. IEEE, 2007.
- [33] I. Ishii, K. Yamamoto, T. Tsuji, et al. High-speed 3d image acquisition using coded structured light projection. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 925–930. IEEE, 2007.
- [34] M. K. Johnson and E. H. Adelson. Shape estimation in natural illumination. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2553–2560. IEEE, 2011.
- [35] J. Kopf, C.-W. Fu, D. Cohen-Or, O. Deussen, D. Lischinski, and T.-T. Wong. Solid texture synthesis from 2d exemplars. *ACM Transactions on Graphics (TOG)*, 26(3):2, 2007.
- [36] T. Korah and C. Rasmussen. 2d lattice extraction from structured environments. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 2, pages II–61. IEEE, 2007.
- [37] N. Kumar, L. Zhang, and S. Nayar. What is a good nearest neighbors algorithm for finding similar patches in images? In *Computer Vision–ECCV 2008*, pages 364–378. Springer, 2008.
- [38] S. Lefebvre and H. Hoppe. Parallel controllable texture synthesis. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 777–786. ACM, 2005.
- [39] T. Leung and J. Malik. Detecting, localizing and grouping repeated scene elements from an image. In *Computer Vision ECCV'96*, pages 546–555. Springer, 1996.
- [40] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(3):418–433, 2005.
- [41] H.-C. Lin, L.-L. Wang, and S.-N. Yang. Extracting periodicity of a regular texture based on autocorrelation functions. *Pattern Recognition Letters*, 18(5):433–443, 1997.
- [42] Y. Liu, R. T. Collins, and Y. Tsin. A computational model for periodic pattern perception based on frieze and wallpaper groups. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(3):354–371, 2004.
- [43] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [44] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. In *Computer Vision–ECCV 2006*, pages 508–521. Springer, 2006.
- [45] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *Image Processing, IEEE Transactions on*, 17(1):53–69, 2008.
- [46] F. Meyer. Topographic distance and watershed lines. *Signal processing*, 38(1):113–125, 1994.
- [47] J. Michels, A. Saxena, and A. Y. Ng. High speed obstacle avoidance using monocular vision and reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 593–600. ACM, 2005.

- [48] T. Monks, J. Carter, and C. Shadle. Colour-encoded structured light for digitisation of real-time 3d data. In *Image Processing and its Applications, 1992., International Conference on*, pages 327–330. IET, 1992.
- [49] P. Müller, G. Zeng, P. Wonka, and L. Van Gool. Image-based procedural modeling of facades. *ACM Trans. Graph.*, 26(3):85, 2007.
- [50] S. M. Omohundro. *Five balltree construction algorithms*. 1989.
- [51] G. Oxholm and K. Nishino. Shape and reflectance from natural illumination. In *Computer Vision–ECCV 2012*, pages 528–541. Springer, 2012.
- [52] A. Panagopoulos, S. Hadap, and D. Samaras. Reconstructing shape from dictionaries of shading primitives. In *Computer Vision–ACCV 2012*, pages 80–94. Springer, 2013.
- [53] M. Park, K. Brocklehurst, R. T. Collins, and Y. Liu. Translation-symmetry-based perceptual grouping with applications to urban scenes. In *Computer Vision–ACCV 2010*, pages 329–342. Springer, 2011.
- [54] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. In *ACM transactions on graphics (TOG)*, volume 23, pages 664–672. ACM, 2004.
- [55] T. Ping-Sing and M. Shah. Shape from shading using linear approximation. *Image and Vision Computing*, 12(8):487–498, 1994.
- [56] J.-P. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *International Journal of Computer Vision*, 72(2):179–193, 2007.
- [57] B. Potetz and T. S. Lee. Statistical correlations between two-dimensional images and three-dimensional structures in natural scenes. *JOSA A*, 20(7):1292–1303, 2003.
- [58] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado. A state of the art in structured light patterns for surface profilometry. *Pattern recognition*, 43(8):2666–2680, 2010.
- [59] J. Salvi, J. Pages, and J. Battle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004.
- [60] G. Sansoni, M. Carocci, and R. Rodella. Calibration and performance evaluation of a 3-d imaging sensor based on the projection of structured light. *Instrumentation and Measurement, IEEE Transactions on*, 49(3):628–636, 2000.
- [61] F. Schaffalitzky and A. Zisserman. Geometric grouping of repeated elements within images. In D. A. Forsyth, J. L. Mundy, V. Di Gesu, and R. Cipolla, editors, *Shape, Contour and Grouping in Computer Vision*, LNCS 1681, pages 165–181. Springer-Verlag, 1999.
- [62] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 519–528. IEEE, 2006.
- [63] T. S. H. Shao and L. V. Gool. Zubud-zurich buildings database for image based recognition. Technical report, Swiss Federal Institute of Technology, 2004.

- [64] S. N. Sinha, P. Mordohai, and M. Pollefeys. Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.
- [65] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1470–1477. IEEE, 2003.
- [66] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM transactions on graphics (TOG)*, volume 25, pages 835–846. ACM, 2006.
- [67] R. F. Sproull. Refinements to nearest-neighbor searching ink-dimensional trees. *Algorithmica*, 6(1-6):579–589, 1991.
- [68] C. Strecha, R. Fransens, and L. Van Gool. Combined depth and outlier estimation in multi-view stereo. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2394–2401. IEEE, 2006.
- [69] J. Sun, G. Zhang, Z. Wei, and F. Zhou. Large 3d free surface measurement using a mobile coded light-based stereo vision system. *Sensors and Actuators A: Physical*, 132(2):460–471, 2006.
- [70] M. A. Tehrani, A. Saghaeian, and O. R. Mohajerani. A new approach to 3d modeling using structured light pattern. In *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on*, pages 1–5. IEEE, 2008.
- [71] S. Tran and L. Davis. 3d surface reconstruction using graph cuts with surface constraints. In *Computer Vision–ECCV 2006*, pages 219–231. Springer, 2006.
- [72] G. Vogiatzis, P. H. Torr, and R. Cipolla. Multi-view stereo via volumetric graph-cuts. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 391–398. IEEE, 2005.
- [73] S. Wenzel, M. Drauschke, and W. Förstner. Detection and description of repeated structures in rectified facade images. *Photogrammetrie, Fernerkundung, Geoinformation (PFG)*, 2007.
- [74] B. Wu, T. L. Ooi, and Z. J. He. Perceiving distance accurately by a directional process of integrating ground information. *Nature*, 428(6978):73–77, 2004.
- [75] C. Wu, J.-M. Frahm, and M. Pollefeys. Detecting large repetitive structures with salient boundaries. In *Computer Vision–ECCV 2010*, pages 142–155. Springer, 2010.
- [76] C. Wu, J.-M. Frahm, and M. Pollefeys. Repetition-based dense single-view reconstruction. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3113–3120. IEEE, 2011.
- [77] P. N. Yianilos. Data structures and algorithms for nearest neighbor search in general metric spaces. In *Proceedings of the fourth annual ACM-SIAM Symposium on Discrete algorithms*, pages 311–321. Society for Industrial and Applied Mathematics, 1993.
- [78] A. Zaharescu, E. Boyer, and R. Horaud. Transformesh: a topology-adaptive mesh-based approach to surface evolution. In *Computer Vision–ACCV 2007*, pages 166–175. Springer, 2007.

- [79] L. Zhang, B. Curless, and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *3D Data Processing Visualization and Transmission, 2002. Proceedings. First International Symposium on*, pages 24–36. IEEE, 2002.
- [80] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape-from-shading: a survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(8):690–706, 1999.
- [81] P. Zhao and L. Quan. Translation symmetry detection in a fronto-parallel view. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1009–1016. IEEE, 2011.
- [82] P. Zhao, L. Yang, H. Zhang, and L. Quan. Per-pixel translational symmetry detection, optimization, and segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 526–533. IEEE, 2012.