2022

Development of Correspondence Field and Its Application to Effective Depth Estimation in Stereo Camera Systems

Shichao Fu

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

UNIVERSITY
OF WOLLONGONG
AUSTRALIA

Development of Correspondence Field and Its Application to Effective Depth Estimation in Stereo Camera Systems

Shichao Fu

This thesis is presented as part of the requirements for the conferral of the degree:

Doctor of Philosophy

Supervisor:
Farzad Safaei

Co-supervisor:
Wanqing Li

The University of Wollongong
School of Electrical, Computer and Telecommunications Engineering

2, 2022

This research has been conducted with the support of an Australian Government Research Training Program Scholarship.

Declaration

I, *Shichao Fu*, declare that this thesis is submitted in partial fulfilment of the requirements for the conferral of the degree *Doctor of Philosophy*, from the University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualifications at any other academic institution.

Shichao Fu

June 8, 2022

Abstract

Stereo camera systems are still the most widely used apparatus for estimating 3D or depth information of a scene due to their low-cost. Estimation of depth using a stereo camera requires first estimating the disparity map using stereo matching algorithms and calculating depth via triangulation based on the camera arrangement (their locations and orientations with respect to the scene). In almost all cases, the arrangement is determined based on human experience since there lacks an effective theoretical tool to guide the design of the camera arrangement. This thesis presents the development of a novel tool, called correspondence field (CF), and its application to optimize the stereo camera arrangement for depth estimation.

CF is a mathematical model of the topology of correspondences in front of cameras and it quantifies the interaction between the camera and the scene being observed. In the thesis, rigorous analysis of this topology is provided and the closed form analytic expressions of the constant disparity surfaces as a function of camera arrangement parameters are derived. CF is then defined formally as the gradient of the disparity field. In addition, the relationship between the CF and the depth estimation accuracy is shown.

Based on the theoretical analysis of the CF, a novel iterative approach to optimizing the arrangement of a stereo camera with respect to the scene and an Expectation-Maximization algorithm are developed for effective acquisition of depth. Experimental results show that the accuracy of depth estimation can be improved by as much as 30% compared to the conventional camera arrangements.

Further, inspired by the involuntary movements of human eyes during fixation, a novel method is proposed to further improve depth estimation. The method is based on random perturbation of camera orientations guided by the CF theory. By perturbation, it is meant that the orientations of two cameras are changed within a small angular range. The experimental results show that the proposed method can further improve the accuracy of depth by close to 30% on average compared to a single optimal camera arrangement.

Acknowledgments

First and foremost, I am very grateful to my supervisors, Prof. Farzad Safaei and Associate Prof. Wanqing Li, for their valuable advice, continuous support and patience during my PhD studies. Their profound knowledge and vast experience have encouraged me throughout my academic research and daily life. Also, I would like to thank all the members of SECTE and my colleagues. It is their help and support that has made my studies and life in Australia a wonderful time. Special thanks to Sen Zhang, Yuanliang Li, and Wenyang Li for their generous help in my life. Finally, I would like to thank my parents, Qiuling Zhang and Tongshun Fu, and my girlfriend, Liu Liu. Without their great understanding and encouragement over the past eight years, I would not have been able to complete my studies.

Contents

Abstract **iv**

1 Introduction **1**

1.1 Research Background 1

1.2 Contribution 4

1.3 Organization 5

1.4 Publications 6

2 Related Work **7**

2.1 Depth Estimation in Computer Vision 7

2.2 Correspondence Matching Algorithms 8

2.2.1 Local-based stereo 10

2.2.2 Global-based stereo 16

2.2.3 Spsstereo 18

2.3 Deep learning based Stereo Matching 19

2.3.1 Module based learning approach 20

2.3.2 Pipeline based learning approach 21

2.3.3 End-to-end learning based approach 22

2.4 Camera Arrangement Optimization 23

2.4.1 Viewpoint Sampling in Light field 23

2.4.2 Camera Pose Selection 24

2.4.3 Iso Disparity and Correspondence Field 25

2.5 Camera Arrangement Perturbation 26

3 Correspondence Field 29

3.1 Disparity field in front of two cameras 29

3.2 2-surfaces 32

3.3 Extension of 2-surfaces to 3D 37

3.4 Disparity and Depth Estimation 40

3.4.1 Density of the disparity field 43

3.4.2 Direction of disparity field surfaces 45

3.4.3 Correspondence Field: the Gradient of the disparity field 47

4 Depth Estimation with CF based Optimization 49

4.1 CF-based camera optimization 49

4.1.1 Expectation maximization based algorithm 51

4.1.2 Initialization 52

4.1.3 E-Step: Estimate S given Θ 52

4.1.4 M-Step: Optimizing Θ given S 54

4.1.5 Termination Condition 55

4.1.6 Complexity 55

4.2 Experiments 56

4.2.1 Simulation dataset and parameters 57

4.2.2 Camera pair optimization 57

4.2.3 Evaluation on the TUM-SLAM dataset 63

4.2.4 Evaluation on the Middlebury multi-view dataset 66

5 Depth Enhancement by CF Perturbation 68

5.1 Camera orientation perturbation 69

5.1.1 Overview 70

5.1.2 Generation of a perturbation set P 71

5.1.3 Finding the optimal subset P^* from P for the scene 72

5.1.4 Selection of P^* and acquisition of associated images 75

5.1.5 Region based correspondence matching 77

5.1.6 Depth estimation and construction of a 3D point cloud 78
5.2 Experiments 79
5.2.1 Implementation 79
5.2.2 Evaluation on the Simulated Data 80
5.2.3 Evaluation on Simulated Data from Real Scene Models 84
5.2.4 Evaluation on Real Scenes 87
6 Conclusion 89
Bibliography 91

Chapter 1

Introduction

1.1 Research Background

Depth estimation is a key technology in computer vision and plays a fundamental role in various image processing tasks and systems. From single depth acquisitions via stereo matching like Kinect to large-scale geometric reconstructions based on multi-view stereo such as simultaneous localization and mapping (SLAM), structural motion (Mfs), and depth-based image rendering (DIBR) systems, retrieving depth information could directly affect the final output accuracy of these systems. In light field rendering, for example, the depth information is used to calculate the weight of the interpolation when synthesizing the virtual light, which significantly affects the final rendering quality. Meanwhile, in SLAM systems, depth estimation influences not only the resolution of the reconstructed geometry, but also the estimation of the camera pose, which is another important task of SLAM.

It is possible to obtain accurate depth information using specialized hardware, such as time-of-flight (TOF) cameras and event cameras, but the current generation of devices has limited range, accuracy, and resolution, and also imposes significant costs if multiple depth maps need to be drawn. Therefore, depth estimation by image processing techniques is still indispensable. The core algorithm for depth estimation is correspondence matching. It matches pixels at the same point/feature in the physical scene of the acquired image and calculates their coordinate differences, called disparity. The disparity is proportional to the depth value, resulting in a depth estimate for each pixel.

There is a wealth of literature related to the correspondence matching algorithms. This approach has been developed for decades and has grown into a huge and deep branch. Local-based correspondence matching algorithms commonly used filtering techniques to cope with smoothness in the early days, and then introduced global optimization algorithms to cope with edges, texture-free regions, and occlusions by specifying a cost function. Modern correspondence matching algorithms utilize machine learning algorithms, such as Convolutional Neural Network (CNN), ResNet, etc., to produce state-of-the-art performance.

The depth estimation algorithms themselves, such as those implementing high performance depth estimation, will improve the final results, however, an interesting area lies in the relationship between the image acquisition phase and depth estimation for stereo and multi-view systems. A large number of stereo systems use camera arrangements with parallel settings. For multi-view systems, such as free-view systems, a regular or spherical grid is typically used [1] . Although most systems use these convenient setups, this does not mean that they are optimal for depth estimation. The camera can be located in any position and orientation. Any combination of these positions will give a camera set which can generate a scene depth map. The upper limit of the accuracy of the depth map may be related to the camera position and orientation.

The hypothesis presented in this thesis is that the *arrangement of cameras*, i.e., the location and orientation of each camera with respect to the scene, also impacts the performance of the depth estimation. Assume that the spatial locations and the orientations of two cameras are denoted by a vector $\Theta = (\theta_1, \theta_2, l, d)$, where θ_1 and θ_2 represent the rotation angles of cameras, and l and d determine the distance between the cameras and the displacement of the midpoint from the origin respectively. When altering this camera arrangement, the mapping between disparity space and the 3D space can drastically change. This unlocks the potential for densifying triangulated spatial samples over the entire or selected sub-regions of the scene, to maximize the resolution of correspondence search space and thus improve the depth estimation. At the same time, different camera arrangements lead to different occluded and unoccluded regions. We believe that this improvement in camera arrangement benefits depth estimation accuracy, and importantly this benefit is independent of the correspondence matching algorithm. Its application can be used in multi-view based systems, such as structure-from-motion, SLAM and robot localization.

Few studies could be found in the camera arrangement optimization area. One early research is presented in iso-disparity by [2] that first represents the discretisation of stereo sampling in 3D space for a general camera arrangement. A primary study on the relationship between the camera arrangement and the pixel correspondences that are formed in front of the cameras is presented in [3] using a mathematical representation called the correspondence field (CF) of cameras. It is a mathematical representation of the relationship between cameras and scenes (shown in Fig.1.1). An intersection point of two rays from cameras is called a 2-point. The direction of the vector at each 2-point is normal to the 2-surface created by the set of 2-points associated with a given disparity. In the plane of a single row of pixels, each 2-surface is shown as a curve. The 2-surfaces represent constant disparity surfaces from the perspective of cameras with increasing distance from the camera plane. The topology of CF changes if the camera arrangement (position and orientation) is altered.

This thesis explores this correspondence field and shows that the correspondence field can be considered as a gradient field of the disparity. The mathematical theory of CF is improved by refining the definition of CF, deriving a closed form expression for the 2-surface, and discovering useful parameters for depth estimation. A CF based camera arrangement optimisation method is proposed that results in significant improvement on depth estimation when the camera is located at the optimal position and orientation. An EM based pair-wise camera arrangement optimization is produced to iteratively relocate pair of cameras to different parts of scenes to achieve a scene-adaptive optimization.

We then propose to extend this optimized position by a new camera alignment method, called camera perturbation, which is motivated by involuntary (and apparently random) movements of the human eye during fixation, called saccade [4]. In simple terms, the

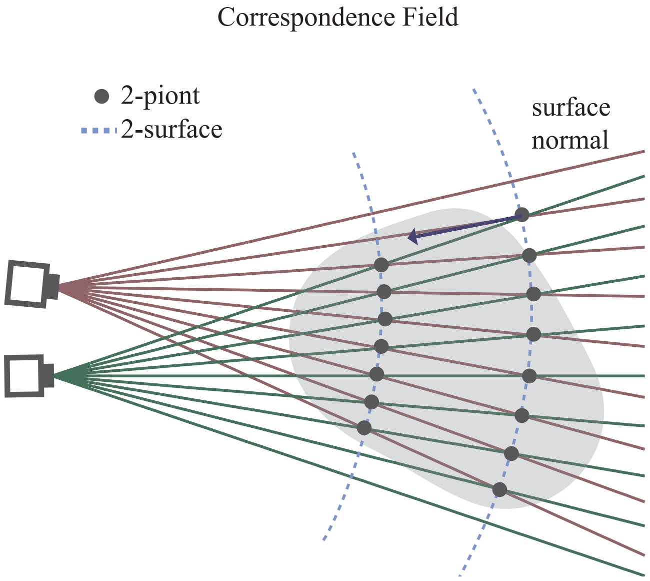


Figure 1.1: Correspondence field of stereo cameras and the illustration of 2-points and 2-surfaces in the field.

optimized camera arrangement moves slightly several times and the depth can be further refined. We validate the concept of perturbation and show that camera perturbation effects are on top of any improvements due to correspondence matching algorithms or arrangement optimization. In practice, based on these results, a robot may benefit from both larger scale re-positioning and small-scale movements of its cameras when scanning objects.

1.2 Contribution

This thesis provides the following contributions:

- *Analytical expression of the CF of two cameras*: The correspondence fields of the two cameras are mathematically analyzed to derive the closed-form equations for the CF surface.

- *the CF density and direction*: Evaluation of some critical parameters for the CF: the extended definition of disparity, the 2-surface density, 2-surface direction and 2-surface gradient field, which affect the accuracy of depth estimation.

- *CF based camera arrangement optimization*: A novel camera arrangement optimization method is proposed to improve the accuracy of stereo depth estimation. It is shown that the optimized arrangement of two cameras reduces the depth error and the extent of occluded areas.

- *CF based camera perturbation optimization*: The concept of camera arrangement perturbation is introduced and the impact of perturbation on the properties of CF in terms of depth accuracy is investigated. The perturbation process is designed into the iterative optimisation model of camera arrangement.

- *Experiments on perturbations*: Extensive experimental results using both synthetic and real scenes are provided to assess the improvements obtained by CF based camera arrangement optimization and perturbation optimization.

1.3 Organization

The thesis is organized as follows:

- Chapter 2 reviews the literature closely related to our study in terms of correspondence matching, sampling analysis, multi-view stereo camera pose selection, and correspondence field. The conventions and notations for the rest of this thesis are set in this chapter.

- Chapter 3 introduces the methodology of correspondence matching, deriving the CF equation, and finally defines the disparity gradient as the correspondence field. It formally introduces the concept of correspondence and its mathematical representation, and discusses how correspondence affects depth estimation.

- Chapter 4 An EM-based optimization method for camera arrangement optimization is provided, which provides a significant improvement to the depth estimation process.

- An extended camera perturbation strategy is presented in Chapter 5. The refined camera perturbation method builds on the method mentioned in Chapter 4, and it can further improve the depth accuracy without introducing occlusion. This chapter also provides a large number of comparisons with existing datasets.

- Chapter 6 summarizes the core contributions of the paper and opens up some avenues for future research.

1.4 Publications

This thesis is based on the following peer-reviewed publications:

S. Fu, F. Safaei and W. Li, ”Optimization of Camera Arrangement Using Correspondence Field to Improve Depth Estimation,” in IEEE Transactions on Image Processing, vol. 26, no. 6, pp. 3038-3050, 2017,

S. Fu, F. Safaei and W. Li, ”Improving Stereo Depth Estimation by Perturbation of Angular Orientation of Cameras,” to be submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence.

Chapter 2

Related Work

2.1 Depth Estimation in Computer Vision

Depth information is widely used in a large number of vision tasks and applications such as scene understanding, scene reconstruction, virtual and augmented reality, and obstacle avoidance. The depth estimation optimization provided in this thesis can be applied to two broad domains. One is the field of 3D reconstruction, such as Structure from Motion (SfM), Simultaneous Localization and Mapping (SLAM) and Multi-View Stereo (MVS). The other is in the area of computational imaging, such as image-based rendering and light field rendering.

Depth plays a vital role in the 3D reconstruction area, as it is one of their main purposes. For multi-view stereo, a 3D point cloud of the target scene can be constructed by a computationally intensive image depth map. Its applications can be real scene modelling, artifact restoration, and photorealistic 3D rendering. In SLAM systems [5], the depth information records the sparse and dense features of the environment on the one hand, and serves as an important parameter for solving and tracking the camera pose on the other. Such systems are heavily used for environment perception tasks in robotics, autonomous driving and augmented reality.

In the field of light field rendering, at an early stage, the final virtual image is synthesized from adjacent rays using a filter-based technique [6]. The filtering results are always blurry and flickering. Although various researchers tried to design sophisticated interpolation algorithms, purely statistical-based filtering does not provide satisfactory results. It is not until depth information is considered as a guide for ray interpolation that the light field rendering shows greater progress. Many asymmetric filters like Gaussian filters [7] [8] [9] are employed. Also, approaches based on geometry information become popular, like DIBR methods: 3D warping [10], layer depth imaging (LDI)[11].

Therefore, obtaining high-quality depth estimates, pre-processing and post-processing of depth maps becomes extremely important in these fields. Depth estimation has become a long-standing problem. Techniques in this category include correspondence matching, depth from focus, photometric stereo, time-of-flight (TOF) cameras, and event cameras. The use of specialized hardware is beyond the scope of this thesis, and the main focus of this thesis is on depth estimation using stereo methods, i.e. correspondence matching.

2.2 Correspondence Matching Algorithms

In computer vision, the correspondence matching process is an essential part of depth estimation. Correspondence matching refers to the procedure of searching and grouping rays belonging to the same part of the scene, from which depth information can be derived based on their geometric relationships. There are two main variants of correspondence matching, namely the local correlation-based approach and the global optimization-based approach.

Local-based methods are quite efficient as they decompose the depth estimation problem into specific procedures. Alternatively, global optimization-based methods define simultaneously the similarity cost, smoothness, and visibility of ray pairs to establish a global function.

The local correlation-based methods match each pair of rays by computing the similarity of the pair with the neighboring pairs within a local window. The basic similarity metrics exploited are the sum of squared differences (SAD), optic flow (OF), and normalized cross-correlation (NCC) [12]. An important concept in the local method is the cost aggregation proposed by Yoon and Kweon [13], where after the similarities built, the cost of neighours are weighted and aggregated within a window to obtain more reliable measurements. Additionally, segmentation and filtering techniques [14–16] are frequently applied during the matching procedure to improve the smoothness in the texture-less areas and highlight the edge sharpness. Local window-based methods have also been widely used in light field-based methods, i.e., robust patch-based block matching in the light field [14].

In terms of computation cost and suitability for real-time applications, local-based methods are highly efficient. However, the inherent locality makes them less efficient for occluded and texture-less areas. To overcome this difficulty and achieve high overall coherence, global optimization methods try to minimize the matching cost of the pixels over the entire image by rigorously modelling photon-consistency, smoothness, and visibility [17–19].

Accordingly, various methods regard the depth estimation as a global optimization cost function, which carefully manipulates the smoothness and visibility issue [20] [21] [22]. Diverse global optimization schemes have also been proposed to achieve higher overall coherence and accuracy such as belief propagation [23] and graph cut [24]. Graph cuts [24] achieve high-quality results at the expense of costly computation time.

In the light field area, many global optimization schemes have been proposed [23], to efficiently evaluate depth information in the epi-polar space. Notably, Wanner and Goldluecke [17] provide a consistent depth labelling technique on the 4D light field space with highly accurate results. Nevertheless, like any global optimization method, the high quality of the depth map is achieved at their very high computational cost.

Estimating the image depth for the free-viewpoint or light field-based systems can also be regarded as a multi-view correspondence matching problem, which inherits both local and global stereo methods [22] [23]. For example, Collins [25] presented a space sweeping approach by using a plane sweeping scheme through the volume of the 3D scene, which is also employed in rendering algorithms, e.g., Layer Depth Images(LDI) [11], and Surface camera light field rendering [26]. Plane sweep methods are suitable for arbitrary camera locations presenting an $O(n)$ algorithmic complexity, where n is the number of cameras. However, the planes established cannot be reused by different target

Stereo Matching Methods	Type
Guided Filter[27]	Local
Efficient Large-Scale Stereo Matching[28]	Local
Semi-Global Block Matching[29]	Semi-Global
Openrwr[30]	Global
Spstereo[31]	Global

Table 2.1: The five stereo algorithms used for experiments

viewpoints, and the quality generated is relatively inferior compared to elaborating local and global stereo matching methods.

In this thesis, we demonstrate the effectiveness of the CF-based camera arrangement optimization algorithm by comparing five top stereo matching algorithms in the KITTI stereo benchmark that can cope with wide-baseline stereo arrangement [27–31]. As shown in Table 2.1, these algorithms vary with respect to their local and global approaches, and in their performance and applications. Efficient Large-Scale Stereo Matching [28] and Guided filter [27] employ local based strategies. Openrwr [30] and Spstereo[31] use global based methods to fine-tune the depth improvement while requiring relatively large computational time and resources. The details of these algorithms, from local-based stereo to global-based stereo, are reviewed in the following sections.

2.2.1 Local-based stereo

Guided Filter based Stereo Matching

The method proposed by [27] is a typical local-based stereo algorithm that uses filtering and cost aggregation techniques. This type of method usually consists of three parts.

1. Creation of a cost volume.
2. Applying filtering techniques.
3. Post-processing of the depth map.

The cost volume C is a three dimensional array that stores the costs for choosing disparity k at pixel (u,v) of image I. At a given (u,v,k), the lower the cost, the greater the likelihood that this disparity is selected. The cost is usually derived from the matching cost functions that measure the photometric similarity of pixels. These metrics include the sum of absolute intensity differences (SAD), the sum of squared intensity differences (SSD), and the normalized cross-correlation (NCC). (shown in Eq.2.1)

$$\begin{aligned} SAD: \quad C(i,j) &= ||I_i - I_j|| \\ SSD: \quad C(i,j) &= ||I_i - I_j||^2 \\ NCC: \quad C(i,j) &= \frac{\sum_{(i,j)\in w}(I_i - \bar{I}_i)(I_j - \bar{I}_j)}{\sqrt{\sum_{i\in w}(I_i - \bar{I}_i)^2 \sum_{j\in w}(I_j - \bar{I}_j))^2}} \end{aligned} \tag{2.1}$$

where w is the size of the neighborhood window, and i and j represent the pixels of the left and right images, respectively.

The cost in [27] employs a truncation version of the combination of the SAD measurement and the second order SAD (SAD of the gradient of image I). The truncation operator is more resilient to noise and the second order SAD helps the overall smoothness and precision. This configuration is widely used in optical flow estimation.

The constructed cost volume contains noise from edges, occluded areas, untextured surfaces, and is not regularized. The second step aggregates (smooths) the cost over a support window by applying a specific filter. Since the window restricts the pixels contributing to the cost to be located in a small local area, this process is also known as the local method of stereo.

Under a certain disparity, the cost value will form a cost image with the same size as the input image, so the traditional convolution filters such as median filter and Gaussian filter, can serve to improve the reliability of the cost image, but these simple structure filters cannot handle the edges well. In order to preserve the edge sharpness, a bilateral filter is proposed. The weights $W_{ij}^{bilaterl}$ of the bilateral filter in Eq.2.2 takes into account both the distance weight $|\mathbf{x}_i - \mathbf{x}_j|$ of neighbouring pixels and the intensity similarity $|I_i - I_j|$ of color images, taking advantage of color consistency.

Although the bilateral filter successfully deals with the problem of over-smooth edges, the problem of backward gradient arises in the edge region. The currently popular guided image filter solves such a problem. The weights W_{ij}^{guided} of the guided image filter in

Eq.2.3 treats the output cost image as a linear transformation of the color input image, which can be solved by the local least squares method. The least-square approximation ensures the smoothness of the function and the linear features give appropriate neighborhood pixel weights to the edge regions. The final solution of the least-square method can be written in the format of a convolution filter, so that the guided filter is formulated. Due to the careful processing of the edges, the guided filter produces a better state of the art performance. All these filtering techniques can also be applied to the final depth map to improve the quality of the results.

$$W_{ij}^{bilaterl} = \frac{1}{K_i} exp(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{\sigma_s^2}) exp(-\frac{|I_i - I_j|^2}{\sigma_r^2}) \tag{2.2}$$

$$W_{ij}^{guided} = \frac{1}{\omega^2} \sum_{k:(i,j)\in w} (1 + \frac{(I_i - \bar{I}_k)(I_j - \bar{I}_k)}{\sigma_k^2 + \varepsilon}) \tag{2.3}$$

The final step is to refine the generated disparity map. This disparity map is always examined by performing a left-right consistency check. That is, the generated depth map for the left camera should be consistent with that of the right camera. And those badly estimated pixels will leave blank holes with other blank areas, such as occlusion areas. Usually, these blank regions are filled using inpainting algorithms. Luo et al. investigated depth-assisted inpainting algorithms based on Criminisi's inpainting method. Daribo [32] proposed an advanced inpainting algorithm that relies on texture and structure propagation, while Oh et al.[33] proposed that using background pixels instead of foreground for the fading region is more reasonable in terms of the definition of occlusion. In addition, both left and right views have two reference depth maps instead of only one, which is usually applied to reduce the hole region [8, 33–35]. For a summary of hole-filling methods, the reader is referred to [36].

In this thesis, besides this typical local based stereo matching algorithms [27], there are two other local based algorithms used in the experiment. They are introduced as follows.

Efficient Large-Scale Stereo Matching

The efficient large-scale stereo matching (ELAS) can be categorized as a patch-based stereo matching method. It contains three steps.

First, the disparity of a set of sparse support points is estimated by searching over the full range of disparity. The support points refer to points with disparity of relatively high confidence. These pixels are expected to be robustly matched due to their unique texture. A variety of methods are provided to compute stable correspondences by using sparse interest point descriptors, such as NCC, DSI [37, 38]. For the sake of both efficiency and effectiveness, the method generates support points by combining the horizontal and vertical Sobel filters.

In the second step, the method connects the support points and the remaining area into piecewise planes, where a Delaunay triangulation is applied on these points to form triangles. The disparity of these non-stable triangle areas can be searched with an initial value based on the linear interpolation value within triangle planes. By assuming that the disparity is piece-wise smooth, given the support points S, the interpolated disparity k_t can be derived as

$$k_t(S,u,v) = a_i * u + b_i * v + c_i \tag{2.4}$$

where i is the index of the triangle the pixel belongs to and the coefficient (a_i, b_i, c_i) can be obtained by solving a linear system. This interpolation gives a prior of the prediction of the disparity to estimate. And the method considers the prior to a sampled Gaussian. So the probability of the disparity given this prior is given as

$$p(k|S) = k_t + exp(\frac{k - k_t(S,u,v)}{2\sigma^2}) \tag{2.5}$$

and if the $|k - k_t|$ is over $3 * \sigma$ the probability is assigned to zero. In such a way, the disparity search range for each pixel is initialized with a reasonable range for efficiency.

Given a k candidate, the cost C of k can be formed by pixel $(I_l, I_r(k, I_l))$. The cost used in this method is a likelihood function based on the Laplace distribution as

$$C = p(I_r|I_l,d) = exp(-\beta||f(I_r),f(I_l)||_1) \tag{2.6}$$

where f is the photo-consistency metrics or so-called feature vector. That is, in the method, the f is calculated by the gradient in a 5x5 pixel neighbourhood computed from the Sobel filter.

The cost designed is under a probability model and the disparity estimation can be regarded as a maximum a-posteriori (MAP) estimation. So, the total energy function is defined as

$$E(I_r|I_l,d) = exp(-\beta||f(I_r),f(I_l)||_1) - log(k_t + exp(\frac{k - k_t(S,u,v)}{2\sigma^2}) \tag{2.7}$$

Overall, the approach forms a reasonable prior for the disparity searching range by triangulating reliable support points. This prior provides local smoothness for each triangle and solves the problem of low confidence of disparity, typically on the low-texture regions and tilted surfaces. This provides an efficient algorithm that reduces the search space and can be easily parallelized because of its local nature.

Semi-Global Block Matching

Another stereo matching algorithm used in this thesis is Semi-Global Matching (SGM) [29] . This method performs pixel-wise matching based on mutual information (MI) and the construction of a smoothness constraint in the global cost function.

For the local cost, unlike the common photon-consistent metrics, a probability-based metric is proposed on the basis of Mutual Information (MI). It consists of the entropy H of both left and right images and their joint entropy.

Mutual information [39] is widely used in computer vision areas such as object pose estimation, object alignment and shape from shading. However, its primary use is for non-rigid registration, MI performs the task similarly to evaluation of the disparities for sparsely sampled pixels. In this method, the MI could be used densely for disparities for every pixel.

$$C_{MI} = H_1(I_1) + H_2(I_2, k) - H_{12}(I_1, I_2, k) \tag{2.8}$$

H_1 and H_2 represent the entropy for the image left and right for a given disparity k and H_{12} is the joint entropy. H_1 does not rely on the disparity and should be a constant. If the disparity map is a one-to-one mapping, H_2 is almost a constant. Thus H_2 always indicates the occlusions. As a result, the joint entropy H_{12} often indicate the cost of intensity similarity.

The cost established from entropy is noisy and contains incorrect matches. Therefore, an additional constraint is needed, e.g., smoothness terms that penalize changes of neighbouring disparities. The pixel-wise cost and the smoothness constraints are expressed by defining the energy $E(k)$ as

$$E_k = C_{MI}(k) + T_1(k, w_1) + T_2(k, w) \tag{2.9}$$

where T_1 and T_2 are the truncated image gradient for smoothness regulation within a neighbouring window. T_1 adds a constant penalty for all pixels, for which the disparity changes slightly less than w_1 (i.e. 1 pixel). This lower penalty for small changes permits an adaptation to slanted or curved surfaces. T_2 adds a larger constant penalty for all larger disparity changes over w_2 which preserves discontinuities.

The calculation of the C_{MI} is performed iteratively. A random disparity map is firstly fed into the energy function and being improved in each turn. In practice, even low quality disparity maps could lead to a good probability estimation. So, the number of iterations could be few (e.g. 3). Also, a hierarchical calculation is proposed in this method, which recursively uses the sub-sampled disparity image, that has been calculated at half resolution, as initial disparity.

Unlike other local based methods that aggregate matching costs within a local window, the paper proposed a cost aggregation of the 1D row of the image in all different directions equally. This approach provides a robust, light-insensitive stereo matching algorithm in a wide range of applications. Furthermore, it is shown that the global cost function has a high computation efficiency as $O(whd)$.

Like other local based algorithms, the final estimation of disparity involves finding the disparity image k that minimizes the energy $E(k)$.

2.2.2 Global-based stereo

The local-based stereo solution deals with photo-consistency (similarity), smoothness and visibility separately. The global-based stereo solution combines all the different items into a single cost function with a similarity component, a smoothness component $E(i,j)$ (which measures neighbourhood intensity consistency) and a visibility cost $E(i,\infty)$. The upgraded cost function is not easily solved by simple traversal and numerical methods, because it is NP-hard. So the problem becomes a global optimization of this cost function.

In the global-based approach, various optimization algorithms have been developed to achieve better results, ranging from the basic Markov chain Monte Carlo (MCMC) to belief propagation (BP) and the emerging graph cut (GC) optimization. The belief propagation method helps to optimize the marginal distribution. The graph cut method is applied by representing image data as a neighborhood graph. In the graph, each vertex represents a pixel and the edge weights between two neighboring vertices represent the similarity between two neighboring pixels. The final quality depends on the design of the energy function, while the performance depends on the efficiency of the global optimization algorithm.

Openrwr Stereo

Openrwr algorithm [30] used in this thesis belongs to a typical global-based approach. Its cost function consists of two parts.

The first part comes from local matching costs such as the census transform [40] and gradient SAD. The census transform manipulates the pixels of the image pair into binary vectors and compares among patches of neighbouring pixels by using the Hamming distance. Gradient SAD metrics are extremely popular in filtering techniques such as guided filter [27]. These two metrics are combined with the weighting parameters to balance their performance. This kind of combination is resilient to illumination changes and reliable for outliers.

To make the cost function more robust, the authors propose a superpixel partitioning method with local cost aggregation within each superpixel in place of window based cost aggregation. At the same time, the computation time is reduced due to the reduced size of the graph. In the proposed method, the Simple Linear Iterative Clustering (SLIC) algorithm [41] is utilized for superpixel segmentation.

The second part is the cost smoothing term. The smoothing term determines that the system cannot be a single-pixel cost selection, but a global optimization algorithm. It evaluates the association between each cost by a custom weighting within adjacent distances. The smoothness constraint assumes that the changes of cost between neighboring pixels are small. However, in the regions of occlusion or depth discontinuity, the smoothness assumption usually fails.

The entire cost function requires a global optimization algorithm to trade-off the two components. Random walk with restart (RWR) is used as a global optimization algorithm. Compared with BP and GC, RWR is computationally efficient and theoretically optimal. The cost function is designed to have a minimum energy in closed form by calculating the derivation of the energy function E. Similar to the graph-cut algorithm, the cost connects the neighbours by 'edge' that contains a probability weight when the cost is propagated through. The weights are affected by the intensity similarity between neighboring superpixels. These factors allows the proposed method to achieve high-quality matching results at relatively lower computational cost. The matching cost is updated iteratively until the convergence is reached.

The key point in [30] is the use of the adaptive RWR algorithm (ARW), which focuses on the fact that global methods often produce unsatisfactory matches in the presence of occlusions or depth discontinuities. Pixels located in the occluded region cannot observe the ground truth match points. That is, the occluded pixels have no matches on the reference image, while the other pixels have at least one.

A visibility constraint was developed to take the occlusion into account. Two additional procedures are required. First, the occluded region is detected by a left-right consistency check. For each superpixel, values with left-right disparity exceeding 1 are occluded and are marked as binary $o_i = \{0, 1\}$.

Secondly, by using the occlusion labelling o, an additional fidelity term is computed to maintain depth discontinuities. The superpixels adjust their disparity value to k^* based on their current matching cost as

$$k^* = \frac{\sum w_{ij} * o_i * k}{\sum w_{ij} * o_i} \tag{2.10}$$

where w_{ij} is the similarity between two neighboring superpixels, j denotes the index of the neighbouring superpixel of the i-th superpixel, o_i is the occlusion label from the previous step. This results in a new disparity k^* for this super-pixel, due to the occlusion label and the aggregation of the neighbouring disparity, the updated disparity is most representative of the disparity in the foreground, with fewer artifacts in the over-smoothed data items in the depth discontinuity region. By using this k^*, additional fidelity term is given as

$$C(k^*, k) = min(\frac{(k^* - k)^2}{\sigma^2}, \tau) \tag{2.11}$$

This function has two benefits for optimization. Firstly, the depth boundaries are well preserved by conserving the illuminant changes of adjacent super-pixels. In practice, it helps to restore the details of the blurred backgrounds caused by the overuse of the smoothness constraints. Secondly, in relation to the first purpose, the truncation parameter τ denotes the maximum cost of disparities could be among neighbours. In this way, the depth boundaries with large discontinuity will be less penalized and thus have a higher probability of being retained. Finally, the aggregated matching cost is iteratively refined by the adaptive RWR algorithm.

2.2.3 Spsstereo

Spsstereo is developed based on an extension of semi-global block matching (SGM) [29]. The total energy function is $E(s, \pi, f, o, I, d)$ and contains up to 6 data items for global optimization. s is a segmentation label, π assigns a plane to each segment, f assigns an outlier label to each pixel, o assigns a line label to each pair of neighboring segments giving the occlusion state of the boundary between these segments, I is the reference image (for segment color similarity), and k is the smoothed semi-dense disparity provided in the previous SGM algorithm, to infer segmentation, plane, and boundary labels.

For optimization strategy, each slanted plane is optimized by closed least squares approximation and keeping the segments, outlier marks and line labels fixed. Occlusion labels are optimized with fixed segmentation, planes, and outlier marks. Segments and outlier marks are jointly optimized. Segmentation is an extension of the SLIC superpixel strategy in SGM.

The algorithm shows significant improvements, especially in terms of occluded pixels, showing the benefit of having joint energy that accounts for outliers on the occluded boundary between each superpixel.

2.3 Deep learning based Stereo Matching

Conventional stereo matching, whether based on local or global approaches, relies on a systematic pipeline solution; it starts with feature extraction, cost-volume building, cost-volume filtering and optimization. Different processes foster complex optimizations or models to deal with occlusion, featureless regions or highly textured areas with duplicated patterns. However, humans are adept at using a priori knowledge to solve such ill-posed problems. It is believed that the human mind can abstract the shapes and objects of a 3D scene and build strong prior knowledge to overcome these problems; for example, regions with less texture are often seen as familiar shapes and accurately match edge and interior regions. The depth of similar scenes can be perceived easily and efficiently by the binocular system.

Learning-based approaches [42] have become very popular as they attempt to exploit this prior knowledge by formulating the problem as a learning task. In particular, with the emergence of convolutional neural network (CNN) in computer vision and the usage of large scale training datasets resources, learning-based stereo matching has become the mainstream for stereo matching/multiview stereo matching tasks.

Learning-based depth estimation is advanced, as the name implies, however, it does not abandon the traditional stereo matching pipeline. In fact, learning-based approaches have developed this cost-volume pipeline to varying degrees. In general, most learning based approaches can be divided into three main categories, i.e., how it prefers traditional stereo matching architectures or purely learning-based tasks.

2.3.1 Module based learning approach

The first class of approaches mimics conventional correspondence matching techniques [43] by explicitly learning how to extract feature, or build cost volume to correspond pixels from the image pair. Machine learning-based algorithms can be applied to one or several modules of the pipeline, such as the feature extraction module, the similarity matching and cost aggregation modules, and the disparity estimation module. Each learning algorithm is trained independently of the other algorithms. In short, the learning-based approach tries to replace some of the complex algorithms of traditional correspondence matching algorithms in order to improve the final results.

Some deep learning based correspondence matching methods replace closed-form features with learned features [44–46]. Given the input includes two image patches, instead of building cost by a conventional descriptor such as NCC features, the convolution neural network (CNN) or full connected network is employed to compute their corresponding feature vectors. It maps intensity patches to descriptors by a supervised learning method with image patches of known disparity as the source.

Commonly, the learned features are fed into the next module, which computes the similarity cost. As with other stereo matching methods, the cost C measures the inverse likelihood that the source pixels on the reference image have a difference k. Recent work [45] has used decision networks consisting of fully connected (FC) layers instead of using similarity measures such as Sobel filtering, NCC, etc. Using decision networks instead of traditional similarity measures allows learning the appropriate similarity measure from the data instead of imposing one fixed solution from experience. Compared with classic correlation based features, they show higher accuracy but the speed becomes significantly slower.

The cost C is estimated for all possible pixel pairs (i, j) at different disparities k, so a 3-dimensional cost quantity $C(i, j, k)$ is created, and the final disparity is obtained by refining the cost quantity data and selecting from it. The original cost volume computed from the image similarities is noisy due to the presence of non-textured regions, occlusions, or duplicate textures. Traditional correspondence matching algorithms attempt to tackle this problem by using cost volume regularization [47, 48]. In these methods, the raw cost volume C will be refined by a global based or semi-global based algorithms to find the optimized disparity map. Typically, these cost aggregation and optimizer routines, such as the SGM algorithm mentioned earlier, are used.

Semi global matching (SGM) provides high accuracy and also keep a low computation cost. This is due to its sophisticated design of the energy function and fine-tuned weight among each data term, which control the smoothness and discontinuity of the disparity map. The learning based method tries to improve it in many ways. The SGM-Net incorporate the deep neural network into this global based architecture to create an adaptive energy function with improved weights. Also, Poggi [49] provides a neural network that refines the weighted aggregation process in SGM algorithm. The weights for every directional 1D scanline of the optimized cost volume are calculated by using a confidence map computed from this deep neural network.

2.3.2 Pipeline based learning approach

The above approach aims at equipping the traditional stereo matching pipeline with modern deep neural network modules. Recent work preserves the pipeline structure but directly replaces the components with fully differentiated models, thus enabling joint optimization and end-to-end learning of the network.

Besides the pre-mentioned feature extraction and similarity cost phase by using learned metrics, the cost volume regulation can also be done by a pure CNN-based network.

Some methods [50, 51] divide the 3D cost volume into a sequence of 2D convolutional layers, thus efficiently applying the 2D convolutions neural network for regularizing this 3D cost volume. However, they only consider aggregated costs along the spatial dimension and neglect aggregated costs along the disparity dimension. Yao et. al [52] normalizes these two-dimensional cost maps sequentially along the disparity direction by a gated recursive unit. This greatly reduces memory consumption and makes possible high-resolution reconstruction while capturing costs along both the spatial and disparity dimensions.

Given the regularized cost volume, the final disparity is commonly obtained by a winner-takes-all method. This winner-takes-all strategy is discrete and could not be back-propagated in the network training. To cope with the non-differentiability problem, a soft argmin operator is used in the last step, which also yields sub-pixel accuracy for the disparity estimation.

When the distribution of disparity is unimodal and symmetric, the soft argmin approximates the subpixel as a maximum a posteriori (MAP) solution [53]. When this assumption is not satisfied, the soft argmin possibly produces a solution leading to over smoothing. Chen et al [54] observed this phenomenon at the depth boundary, where the discontinued disparity follows a multimodal distribution. To address these issues, they performed a weighted averaging operation only for a window centred on the modalities with maximum probability, rather than using full-band weighted averaging for the entire range of disparities.

2.3.3 End-to-end learning based approach

In addition to the above two categories, pure end-to-end methods have become popular due to the advent of computational speed and data capacity, such as FlowNet [55] and DispNetS [56], which use an auto-encoder framework to superimpose the stereo images as input into a latent volume space to directly predict the final disparity map. These algorithms abandon the traditional modelling of cost-volume modules and show high computational speed in a wide range of applications. The final quality depends on the size of the training data set. They usually require a large amount of training data, which is difficult to obtain.

2.4 Camera Arrangement Optimization

The main objective presented in this thesis is to manipulate the camera arrangement for the sake of depth estimation improvement. Camera view selection or planning has already been studied in the computer vision and robotics communities. However, these proposed methods are tightly coupled to a specific reconstruction scheme and do not generalize well [57], or do not necessarily focus on the specific nature of the existing light field acquisition setups [57][58]. Little work has been done in this particular area for depth estimation. In this section, camera-related techniques for each field are discussed and a unique model for this field, named correspondence field, is introduced.

2.4.1 Viewpoint Sampling in Light field

A related idea of camera arrangement optimization lies in the light field area. Each sample in the light field can be considered as a camera viewport, therefore, the analysis of light field sampling indeed is a kind of camera arrangement optimization for the sake of light field acquisition and rendering. It would be desirable to optimize the configuration of cameras, such as their location or orientation, to improve the rendering quality.

Zhang [59] and Shidanshidi[60] studied the light field sampling theory that attempts to find an optimal sub-sampling set for rendering a certain view of the scene. The theoretical analysis [59] demonstrates that the depth variation of the scene determines the minimum sampling rate. Accordingly, Shidanshidi [60] proposed an objective evaluation approach for light field rendering by using different interpolation algorithms. Specifically, based on the 4D light field model, this work evaluates several interpolation techniques to identify the effective sampling density.

Bagnato [61] analyzed the sphere 4D light field model by applying a sphere harmonics function to deduct the sample rate formula with approximation, to prevent the aliasing effect of the sphere light field model. This strategy can be regarded as an extension of [59].

Fourier analysis of the light field is also examined by Lumsdaine [62] who presented a solution for image-based rendering in the frequency domain via the forward and inverse Fourier transform. Also, Gilliam [63] investigated the sampling rate problem in more detail by deducting the exact close form expression for the plenoptic spectrum, and presented a formula determining the minimum sample frequency for a slanted plane under a Lambertian scene. However, this sampling analysis emphasizes the interpolation of light ray, not setting targets for the depth estimation.

2.4.2 Camera Pose Selection

On the other hand, ongoing research on selecting the most suitable subset of cameras for 3D reconstruction is a popular topic especially for large multi-camera settings utilizing multi-view imaging [64–69]. The visual Simultaneous Localization and Mapping (SLAM) calculates simultaneously the camera poses, i.e., camera arrangement by regarding the camera as a rigid body, and the environment map, e.g., a point cloud of the scene. In current stereo or monocular-based SLAM methods, the environment map is constructed from a stereo matching process. However, to find a suitable image pair for stereo matching the system has to select the key-frames and the reference-frames [5, 70, 71]. This is because the image points in the key-frame will be triangulated and stored into the global environment map by exploiting the reference frame to complete the disparity search and 3D reconstruction. This process involves selecting a pair of optimal cameras, where one camera offers the key-frame and the other the reference-frame. In SLAM, the key-frame is mainly used to expand the scale of the current map. For example, a frame is selected as a key-frame when 40% of the frame is not overlapping with the previous key-frame [5, 72]. As for the reference frame, it is often heuristically selected based on the baseline formed by the key- and reference-frames [5]. Therefore, selecting both reference frames is not related to the accuracy of the depth estimation.

Other camera selection method in the SLAM area such as [73, 74] focus on camera placement optimization for visibility maximization and discovery of occlusion regions, which is not the main goal or scope of discussion in the thesis. In addition, Chen [75] employs modern reinforcement learning technique to facilitates the alignment and merging of point clouds obtained from different camera viewpoints. The camera optimization in this thesis focuses on improving the accuracy of correspondence matching by changing the camera arrangement.

2.4.3 Iso Disparity and Correspondence Field

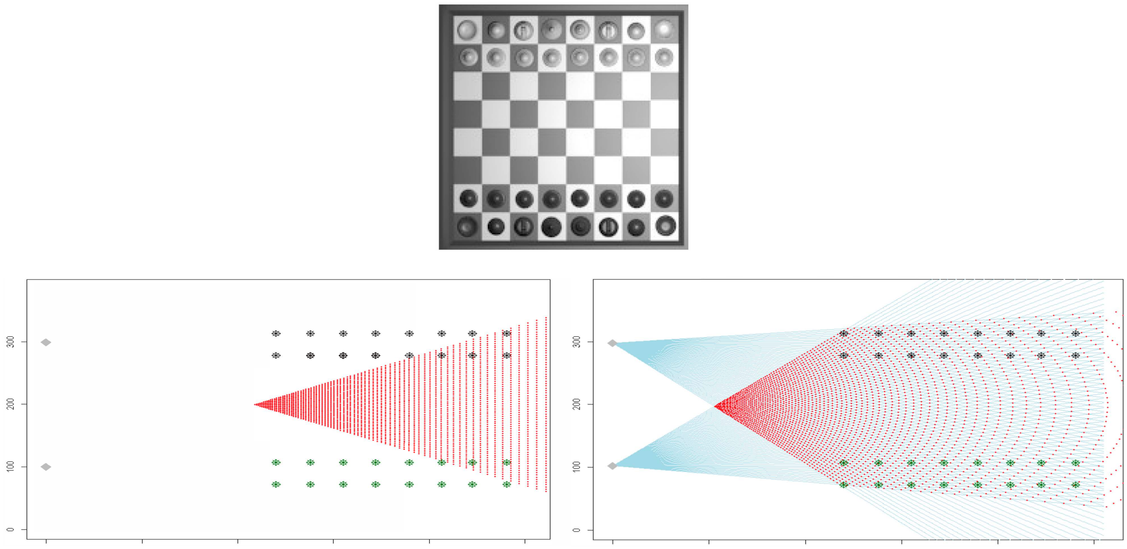
Camera arrangements such as converging or diverging are only briefly studied [76–80]. These settings are highly relevant to depth and correspondence matching. The first concept that tries to describe this geometry relationship is the iso-disparity presented in [2]. It firstly shows how one can configure a stereo camera pair head to align iso-disparity surfaces to scene structures of interest such as a vertical wall with qualitative visuals, allowing better and faster stereo results.

The mathematical properties of iso-disparity have not been further investigated until a mathematical expression, which is of particular relevance to this thesis, named the correspondence field, was introduced in [3] to quantify the relationship between the camera arrangement and the objects in a given scene.

The correspondence field describes the spatial topology of the intersecting rays of cameras, arranged as a number of layers or surfaces in the field of view of cameras. This field is first introduced into the light field to improve the light field rendering results. As mentioned in [3], the intersection point of camera rays is called 2-point, with its normal direction as 2-normal and the 2-points with the same disparity creating a 2-surface. The greedy strategy is proposed to numerically maximize 2-points in the target region.

Figures 2.1 shows two camera arrangement (initial and optimal) for one scene. The red points represent 2-points of the CF. It is clear that different arrangement of objects in the scene would require substantial changes to the camera configuration. It is also evident that the optimum camera configurations result in correspondence points near the objects (the red and black points from the top view of a chess scene), thus resulting in a significant improvement in the final rendering quality of light field.

Therefore, calculating the corresponding fields for various camera configurations in



(a)

Initial Configuration

$a_1 = 300$, $\theta_1 = 0°$; $a_2 = 100$, $\theta_2 = 0°$

(b)

Optimal Configuration

$a_1 = 290$, $\theta_1 = 20°$; $a_2 = 120$, $\theta_2 = -15°$

Figure 2.1: Initial and optimum CF for one scene.[3]

the pre-processing stage would provide an important practical advantage. As the scene changes in time, the acquisition system can change its camera arrangement, perhaps by evaluating the applicability of some precomputed configurations. In addition to the benefits for light field rendering, we believe that correspondence fields have the potential for depth estimation tasks.

In this thesis, we have systematically studied the CF model and extended its framework. In the next chapter, we present a detailed theoretical analysis of the CF model developed in this study. We present a closed-form relationship between the CF model and the depth estimation accuracy, and finally develop a CF-based optimization system that provides a novel and effective method to optimize the camera arrangement or the selection of key and reference frames in SLAM to manage a more accurate depth estimation to construct environmental maps.

2.5 Camera Arrangement Perturbation

As the theory of CF is closely associated with the fundamental notion of disparity, it is possible to relate the CF topology and binocular vision model for human depth perception.

For human vision [81–83], the correspondence field surface of zero disparity is termed as binocular horopter and the intersection of the center point of the two eyes form a fixation point. The disparity of near and far from fixation point is called cross and uncross disparity, respectively.

One of the main purposes of obtaining disparity is the ability to estimate the depth of objects in a scene. Unlike the correspondence matching process in computer vision, the human binocular disparity is usually associated with different visual attributes, such as luminance, chromaticity, object texture orientation, and surface complicity in the environment. Taking into account the various degrees of freedom of movement (orbital movement of the eyes as well as head movements), and the curvature of the human retina, the human visual geometry is in fact quite complex.

Unlike computer vision commonly using parallel setting of camera arrangement, the behaviour of the binocular disparity [84, 85] often drives converged viewpoint arrangement (vergence eye movements and accommodation). This behavior is directly related to many important human functions, such as touching objects, grasping targets, and even guiding route planning in 3D environments. This is one of the inspirations of using different camera arrangement for computer vision task.

Another interesting behaviour related to this thesis is the binocular saccades movement. The human eye requires coordinated movements of eye to make frequent saccades. A healthy eyes is able to converge accurately when targeting on the new fixation point, to prevent the emergence of double images. Gibaldi [86] also suggested that rapid binocular eye movements may reflect the distribution of binocular disparities. According to their findings, the eyes are more divergent than convergent in the lower visual area, reflecting the crossed and uncrossed disparity between the two hemispheres. Chau [84] also pointed out that rapid binocular movements are well adapted to the 3D environment, with the result that the need of larger visual corrections is minimized at the end of the saccades.

The relationship between the saccadic eye movements and the binocular disparity is controversial and need further investigation. Nonetheless, inspired by this phenomena, we propose a novel camera perturbation method like the behaviour of human eye saccadic movement, by rotating the camera within small angles for more robust depth estimation optimization. We theoretically demonstrate its benefits by building a fused CF, and the experiments show its advantages for better depth (see chapter 5 for more detail).

Chapter 3

Correspondence Field

In this chapter, we investigate a model closely related to correspondence, disparity and depth estimation, namely the correspondence field (CF). First, the basic topology of correspondence in physical space is visualized. Second, the theory of the disparity field is established by deriving its mathematical formulation. The derivation starts from the two-dimensional case and is extended to the three-dimensional space. Third, the key properties of the disparity field in terms of depth estimation are presented. The quantitative relationship between these properties and depth accuracy is derived. Finally, the concept of the correspondence field is introduced in order to optimize the camera arrangement.

3.1 Disparity field in front of two cameras

Figure 3.1 shows a top view of the rays associated with a single row of pixels in a stereo system. Each intersection of two rays is a correspondence, which is the basic element of the correspondence matching algorithm. These correspondence points are distributed in a specific pattern in physical space, which is considered to be related to the arrangement of cameras.

By connecting these correspondence points with the same disparity, these points can create a specific 2-D curve (or 3-D surface), which is called a 2-surfaces. 2-surfaces represent constant disparity surfaces at increasing distances from the camera plane. In the limiting case, when the pixel resolution is infinite and the disparity becomes a continuous

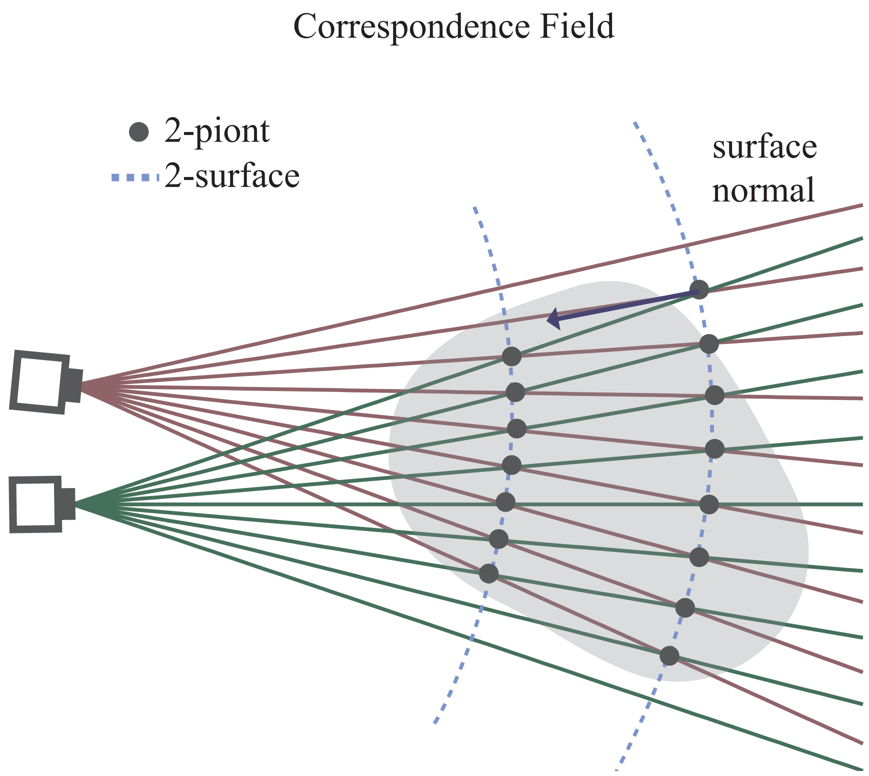


Figure 3.1: Correspondence of stereo cameras in physical space and the illustration of 2-points and 2-surfaces in the field.

scalar, the variations of disparity can be modelled as a scalar field.

The shape of the 2-surfaces determines how disparity value is translated into ’depth’ or distance in physical space. Such a relationship can be simple for conventional stereo systems, where the 2-surfaces form parallel planes. However, the shape changes and becomes more complex if the camera arrangement (position and orientation) is altered.

The topology of 2-surfaces is demonstrated by simulation in [3]. In this section, we develop an analytical expression for the 2-surface curve in a closed form so that the relationship between camera arrangement, disparity and depth can be quantified. To keep the derivation concise. The table 3.1 shows the notation used consistently in all formulas in every chapter.

Table 3.1: Table of notations.

Symbol	**Definition**
1.	**Variables**
x, y, z	3D Cartesian coordinate in 3D scene space
r, θ, ϕ	3D spherical coordinate
u, v	2D Cartesian coordinate in image space
u, v, s, t	4D Cartesian coordinate in light field space
f	focal length
c	camera location point
Θ	camera arrangement in CF representation (location and orientation)
l	baseline of the stereo camera
d	middle point of the stereo camera
k	disparity
$\mathbf{r}$	radius vector in 3D scene space
A	coefficient of CF equation
J	Jacobian matrix
T	perspective transformation function
$\varepsilon_r, \varepsilon_k$	uncertainty of point location **r** and disparity k
ε_p	perturbation bounds
Φ	energy function
Ψ	gradient field of disparity (correspondence field)
L, U	camera arrangment lower and upper bounds
S	scene
Ω	subsection of the scene
β	parameters of generalized normal distribution
$\mathbf{n}$	normal vector
N	number of perturbations
m	mid-point of the camera pair
M	number of regions
a	perturbation element
P	camera perturbation set

3.2 2-surfaces

The derivation of the 2-surface equation starts from the case of one epi-polar plane associated with the middle row of image pixels (2D) and then extends to all epi-polar planes to complete the deduction for the 3D case.

In 2D case, a Cartesian coordinate x-z is used to describe the scene and the arrangement of cameras from the top view (See Fig. 3.2). Consider two cameras mounted on a frame or rail along the x direction and denoted as c_1 and c_2, respectively. The cameras are able to move along this rail and also change their orientation within some reasonable bounds. Let the middle point between c_1 and c_2 be m. In this coordinate system, four parameters are sufficient to represent a given camera arrangement: θ_1, θ_2 representing the rotation angles of c_1 and c_2 respectively; l representing the distance between each camera and the mid-point m; and d being the displacement of m along the x axis. The arrangement parameters of this camera system are represented by $\Theta = (\theta_1, \theta_2, l, d)$. Also, the focal length of the camera is denoted by f in pixel units. In this thesis, it is assumed that all cameras are identical and f is a constant.

Any arrangement Θ will result in a specific 2-surfaces topology in front of the cameras. Let the position of a spatial point p be represented by the vector $\mathbf{r}$. Assume that for a camera arrangement Θ, p is located on the kth 2-surface curve $K(\mathbf{r}, \Theta) = k$ with disparity k. Its corresponding projection point coordinates on the image space of the two cameras are denoted by u_1 and u_2 shown in Fig. 3.2. Here u_1 and u_2 can be obtained by performing the perspective transformation of point p. The perspective transformation function is denoted by function T so that the image coordinate u for $i$the camera is obtained as

$$u_i = T_i(\mathbf{r}, \Theta) \tag{3.1}$$

where T_i represents perspective transformation of the ith camera. In this thesis, the pinhole camera model [87] is utilized, so the perspective transformation can be written as

$$u_i = T_i(\mathbf{r}, \Theta) = \frac{\mathbf{r}_x^i}{\mathbf{r}_z^i} f \tag{3.2}$$

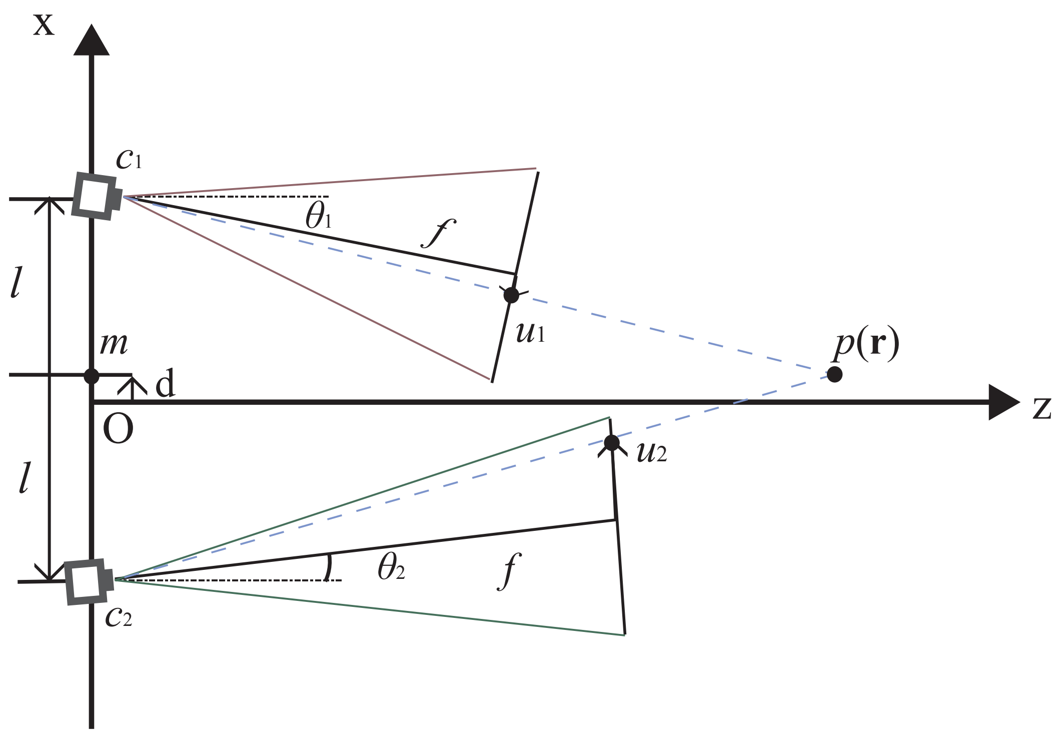


Figure 3.2: Arrangement parameters of a two camera system.

where $\mathbf{r}^i$ is the vector $\mathbf{r}$ seen in the coordinate of camera i's view. Commonly, camera i's view uses a coordinate system with origin at camera's centre with z axis parallel to camera's orientation and x axis parallel to camera's focal plane (See coordinates in red and blue in Fig. 3.3). $\mathbf{r}^i$ can be obtained through rotation and displacement from world coordinate to i's view coordinate system as

$$\mathbf{r}^i = \begin{pmatrix} \mathbf{r}^i_x \\ \mathbf{r}^i_z \end{pmatrix} = \mathbf{T}[\mathbf{r} + (l_i, 0)^\intercal] \tag{3.3}$$

$$= \begin{pmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{pmatrix} \begin{pmatrix} x + l_i \\ z \end{pmatrix} \tag{3.4}$$

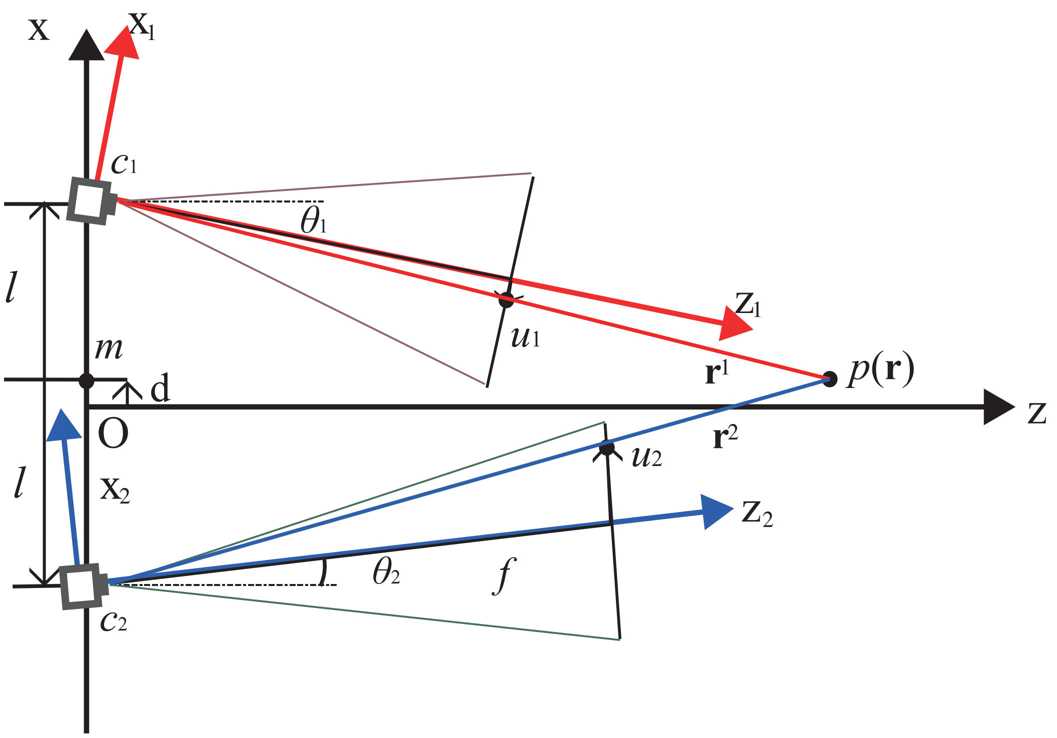


Figure 3.3: Coordinate system from two camera's view. The coordinate system for individual cameras $x_1 - z_1$ and $x_2 - z_2$ is marked in red and blue, respectively. The global coordinate is shown in black.

combine the Eq.3.2 and Eq.3.3, u_i is given as

$$u_i = \frac{(l_i + x)\cos\theta_i + z\sin\theta_i}{z\cos\theta_i - (l_i + x)\sin\theta_i} f,\ l_i = \begin{cases} -l, i = 1 \\ l,\ \ i = 2 \end{cases} \tag{3.5}$$

The denominator of u_i represents $\mathbf{r}_z^i$ value seen from the camera viewpoint. As our target scene should be in front of the camera focal plane, the $\mathbf{r}_z^i$ should be above zero. Given the coordinate u, every $\mathbf{r}$ on the kth 2-surface has a constant disparity k:

$$K(\mathbf{r}, \Theta) = u_1 - u_2 = k \tag{3.6}$$

From Eq.3.5 and Eq.3.6, the 2-surface equation K is derived as

$$\frac{(l+x)\cos\theta_i + z\sin\theta_i}{z\cos\theta_i - (l+x)\sin\theta_i} f - \frac{(-l+x)\cos\theta_i + z\sin\theta_i}{z\cos\theta_i - (-l+x)\sin\theta_i} f = k \tag{3.7}$$

as the denominator should not be zero, the equation could be expanded as

$$\begin{aligned} d^2\sin(\theta_1)\cos(\theta_2) - d^2\cos(\theta_1)\sin(\theta_2) - 2dz\sin(\theta_1)\sin(\theta_2) \\ -2dz\cos(\theta_1)\cos(\theta_2) - x^2\sin(\theta_1)\cos(\theta_2) \\ +x^2\cos(\theta_1)\sin(\theta_2) - z^2\sin(\theta_1)\cos(\theta_2) + z^2\cos(\theta_1)\sin(\theta_2) \\ = kf(-d\sin(\theta_1) - x\sin(\theta_1) + z\cos(\theta_1))(d\sin(\theta_2) - x\sin(\theta_2) + z\cos(\theta_2)) \end{aligned} \tag{3.8}$$

by re-organizing the Eq.3.8 in terms of x and z, it yields

$$\begin{aligned} \sin(\theta_1 - \theta_2)(-l^2 + x^2 + z^2) + 2Lz\cos(\theta_1 - \theta_2) &= \\ k(z\cos\theta_1 - \sin\theta_1(l+x))(z\cos\theta_2 + \sin\theta_2(l-x)) \end{aligned} \tag{3.9}$$

or

$$\begin{aligned} x^2(\sin(\theta_1 - \theta_2) - \mathrm{k}\sin\theta_1\sin\theta_2) + xz(\mathrm{k}\sin\theta_1\cos\theta_2 + \mathrm{k}\sin\theta_2\cos\theta_1) + \\ z^2(\sin(\theta_1 - \theta_2) - \mathrm{k}\cos\theta_1\cos\theta_2) \\ +z(\mathrm{k}l\sin\theta_1\cos\theta_2 - \mathrm{k}l\sin\theta_2\cos\theta_1 \\ +2L\cos(\theta_1 - \theta_2)) + \mathrm{k}l^2\sin\theta_1\sin\theta_2 - l^2\sin(\theta_1 - \theta_2) = 0 \end{aligned} \tag{3.10}$$

which can be written into a *quadratic* equation as Eq. 3.11.

$$A_1(x-d)^2 + 2A_2(x-d)z + A_3z^2 + 2A_4(x-d) + 2A_5z + A_6 = 0 \tag{3.11}$$

where

$$A_1 = \sin(\theta_1 - \theta_2) - kf\sin\theta_1\sin\theta_2 \tag{3.12}$$

$$\begin{aligned}
A_2 &= \frac{kf}{2}\sin(\theta_1+\theta_2) && (3.13)\\
A_3 &= \sin(\theta_1-\theta_2)-kf\cos\theta_1\cos\theta_2 && (3.14)\\
A_4 &= 0 && (3.15)\\
A_5 &= \frac{l}{2}(kf\sin(\theta_1-\theta_2)+2\cos(\theta_1-\theta_2)) && (3.16)\\
A_6 &= l^2(kf\sin\theta_1\sin\theta_2-\sin(\theta_1-\theta_2)) && (3.17)
\end{aligned}$$

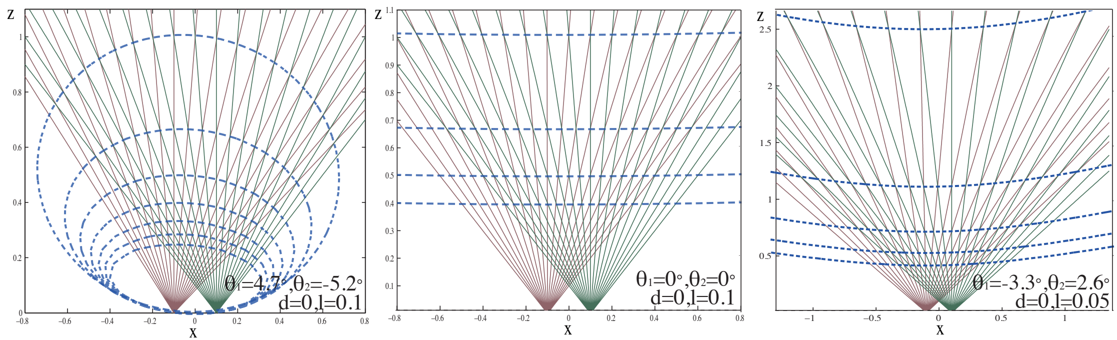
Given their quadratic nature, the 2-surfaces are, in general, conic curves. It is always possible to eliminate the $(x-d)z$ cross term in the equation by a suitable rotation of the axes by an angle θ_{12}, which describes the general direction of conic curve compared to the standard one. This direction is calculated as

$$\theta_{12} = \frac{1}{2}\arctan\left(\frac{A_3-A_1}{2A_2}\right) = -\frac{\theta_1+\theta_2}{2} \tag{3.18}$$

The shape of 2-surface is only dependent on the orientation of the two cameras through this simple relationship, which means we can easily control the ellipse/parabola surface direction. In addition, the eccentricity ε and the position (X,Y) of the origin of quadratic curve are

$$\begin{aligned}
\varepsilon &= \frac{\mathrm{k}\tan\left(\frac{1}{2}(\theta_1-\theta_2)\right)+2}{2-\mathrm{k}\cot\left(\frac{1}{2}(\theta_1-\theta_2)\right)} && (3.19)\\
X &= \frac{L\cos\left(\frac{1}{2}(\theta_1+\theta_2)\right)\sec^2\left(\frac{1}{2}(\theta_1-\theta_2)\right)(\mathrm{k}\sin(\theta_1-\theta_2)+2\cos(\theta_1-\theta_2))}{2\left(\mathrm{k}-2\tan\left(\frac{1}{2}(\theta_1-\theta_2)\right)\right)} && (3.20)\\
Y &= -\frac{L\sin\left(\frac{1}{2}(\theta_1+\theta_2)\right)\csc^2\left(\frac{1}{2}(\theta_1-\theta_2)\right)(\mathrm{k}\sin(\theta_1-\theta_2)+2\cos(\theta_1-\theta_2))}{2\left(\mathrm{k}+2\cot\left(\frac{1}{2}(\theta_1-\theta_2)\right)\right)} && (3.21)
\end{aligned}$$

To verify the above result, Fig. 3.4 shows the perfect alignment of 2-surfaces obtained by Eq. 3.11 and simulation. This figure also illustrates different arrangements of 2-surfaces. It can be seen that when the two cameras rotate towards each other (the *converged* orientation), the 2-surfaces form a set of ellipses, while when rotating away from each other (the *diverged* orientation), the curves will be hyperbola. In the special case where the cameras are facing forward in parallel, the 2-surfaces will be planar.



(a) (b) (c)

Figure 3.4: Exact matching of analytical and simulated 2-surfaces for different camera arrangements. From top to bottom: converged, parallel and diverged orientations

3.3 Extension of 2-surfaces to 3D

The derivation of the 2-surfaces on the horizontal plane is a special case of 2-surfaces in 3D space. In order to extend this idea into 3D, the 3D space is divided into infinite 2D epi-polar planes shown in Fig.3.5. Each epi-polar plane has their own 2D coordinate system as $\tilde{x}-\tilde{z}$. Since the derivation of one epi-plane is successfully developed, it is possible to perform the same derivation by converting the previous $x-z$ coordinates to $\tilde{x}-\tilde{z}$ coordinates of other epi-planes.

As shown in Fig.3.6, for a point p in Cartesian 3D space (x,y,z), the epipolar plane space to which it belongs is formed by connecting the point p and the camera positions c_1 and c_2. We can define a $(\tilde{x},\tilde{z})$ coordinate on this plane and the plane is parametrized by ϕ which is the angle between this epi-polar plane and the horizontal plane. In this case, the point (x,y,z) has a one-to-one mapping to the epipolar plane coordinates $(\tilde{x},\tilde{z},\phi)$. The transformation between Cartesian and epipolar coordinates is given by the relation:

$$\begin{cases} \tilde{x} = x \\ \tilde{z} = \sqrt{z^2+y^2} \\ \tan\phi = \frac{y}{z} \end{cases} \tag{3.22}$$

Given the coordinate transformation in Eq.3.22, the disparity k could be obtained in terms of $(\tilde{x},\tilde{z},\phi)$. The disparity in general is the coordinate difference on each epipolar line in the Fig.3.6. The coordinate on the epi-polar line is set the same as the [2, 87] for

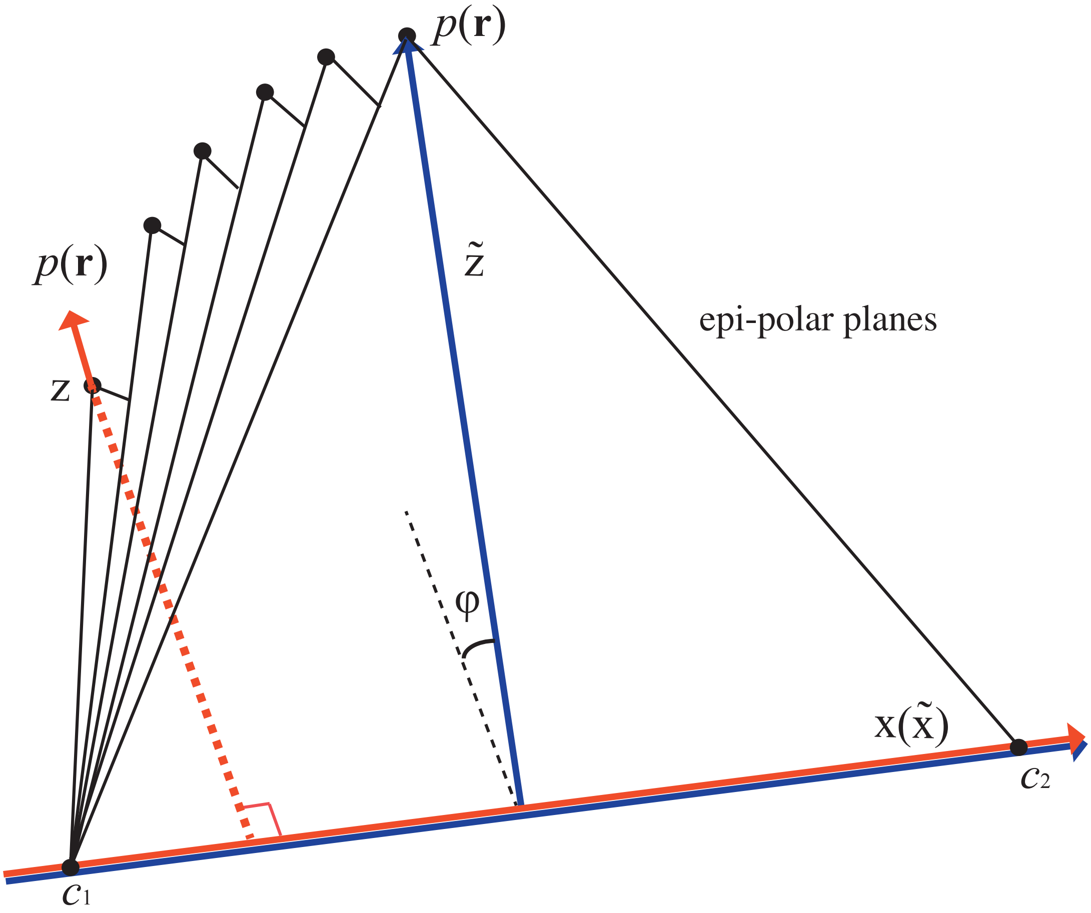


Figure 3.5: Extension of the 2-surfaces to 3D space can be achieved by decomposed the 3D space into infinite 2D epi-polar plane space. The previous $x-z$ plane is marked in red and the new coordinate on other epi-polar plane is labelled in blue.

convenience. By substituting the (x,z) to $(\tilde{x},\tilde{z})$, the previous perspective transformation in Eq.3.5 can be updated to calculate the image coordinate $\tilde{u}$ on the epi-polar plane as

$$\tilde{u}_i = T_{\tilde{u}_i}(\mathbf{r},\Theta,f) = csc\omega \cdot u_i \tag{3.23}$$

$$= csc\omega \cdot \frac{(l_i+\tilde{x})\cos\theta_i + \tilde{z}\sin\theta_i}{\tilde{z}\cos\theta_i - (l_i+\tilde{x})\sin\theta_i} f \tag{3.24}$$

where $T_{\tilde{u}}$ is a projection function that calculates the epi-polar coordinate $\tilde{u}$ of point p in the image plane. As shown in Fig.3.6, ω is an auxiliary variable that represents the epi-polar line slope angle on image plane, which is a constant for each epi-polar plane.

After the transformation of getting the coordinate $\tilde{u}$ for every epi-polar plane, The con-

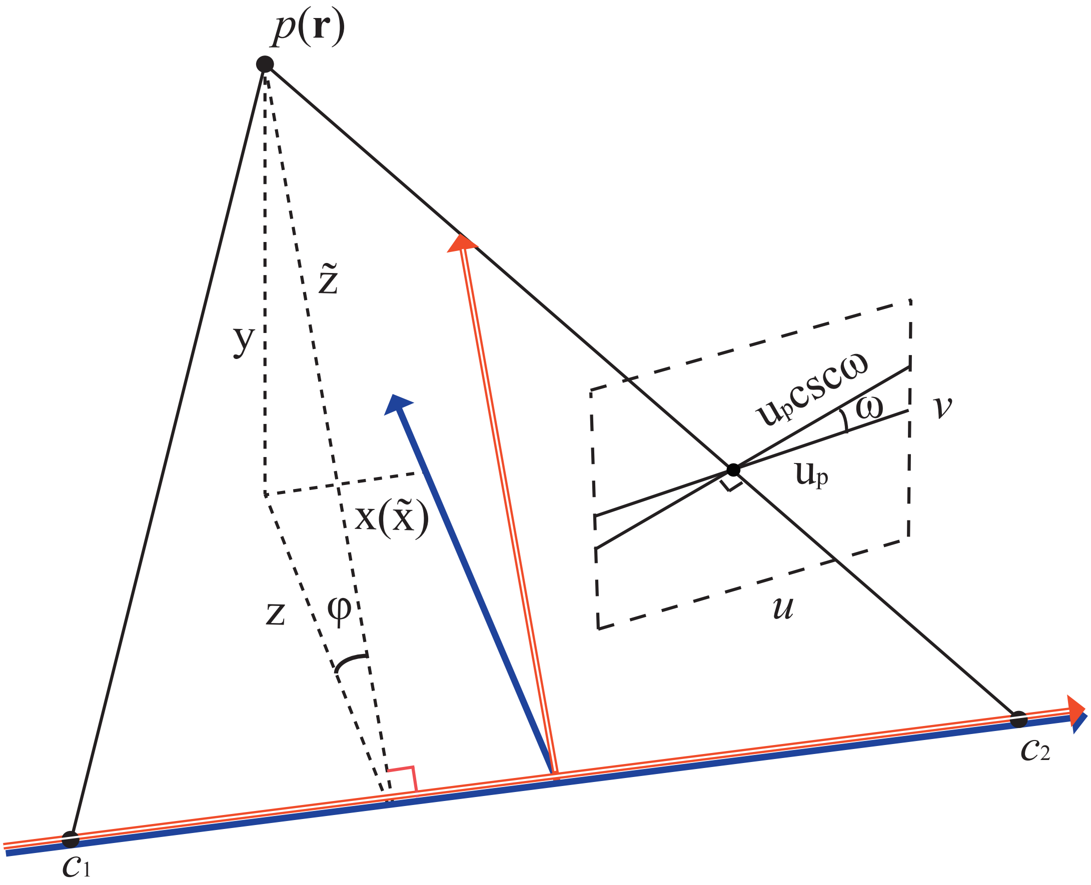


Figure 3.6: Extension of the 2-surfaces to 3D space can be achieved by replacing the 2D coordinates marked in blue solid line to the epi-polar coordinates system marked in red double solid line.

stant disparity surface at $K(\mathbf{r},\Theta,f)=k$ is computed as

$$K(\mathbf{r},\Theta,f)=T_{\tilde{u}_1}(\mathbf{r},\Theta,f)-T_{\tilde{u}_2}(\mathbf{r},\Theta,f)=k \tag{3.25}$$

And the 2-surfaces equation for each epi-polar plane can be derived as

$$A_1(\tilde{x}-d)^2+2A_2(\tilde{x}-d)\tilde{z}+A_3\tilde{z}^2+2A_4(\tilde{z}-d)+2A_5\tilde{z}+A_6=0 \tag{3.26}$$

where

$$\begin{aligned} A_1 &= \beta_2\sin\theta_1\cos\theta_2-\beta_1\sin\theta_2\cos\theta_1 \\ &\quad -kf\sin\theta_1\sin\theta_2 \end{aligned} \tag{3.27}$$

$$A_2 = \cot\phi[(\mu_1 - \mu_2)\cos\theta_1\cos\theta_2 + \frac{kf}{2}\sin(\theta_1+\theta_2)] \tag{3.28}$$

$$\begin{aligned} A_3 &= \cot\phi^2[\beta_1\sin\theta_1\cos\theta_2 - \beta_2\sin\theta_2\cos\theta_1 \\ &\quad -kf\cos\theta_1\sin\theta_2] \end{aligned} \tag{3.29}$$

$$A_4 = 0 \tag{3.30}$$

$$A_5 = \frac{l}{2}[kf\sin(\theta_1-\theta_2) + (\beta_1+\beta_2)\cos(\theta_1-\theta_2)] \tag{3.31}$$

$$\begin{aligned} A_6 &= l^2(kf\sin\theta_1\sin\theta_2 + \beta_1\sin\theta_2\cos\theta_1 \\ &\quad -\beta_2\sin\theta_1\cos\theta_2) \end{aligned} \tag{3.32}$$

where β is calculated as

$$\beta_i^2 = \csc\phi^2\csc\theta_i^2 + 1 \tag{3.33}$$

The definition of camera arrangement $\Theta = (l, d, \theta_1, \theta_2)$ remains unchanged. Figure 3.7 demonstrates the 3D 2-surface which is like a bullet and multiple 2-surfaces constitute a semi-ellipsoid onion.

3.4 Disparity and Depth Estimation

Disparity k can be described as a *scalar field* by utilizing the 2-surfaces equation $K(\mathbf{r}, \Theta)$. The field accurately depicts the distribution of k over the physical space. In this way, it is possible to know the uncertainty of the disparity in different regions before performing the correspondence matching algorithm. This allows us to optimize the camera arrangement for depth estimation.

For general camera arrangements, this uncertainty is associated with the relationship between the disparity k, the pixel index coordinates u, and the perceived ”depth” (spatial

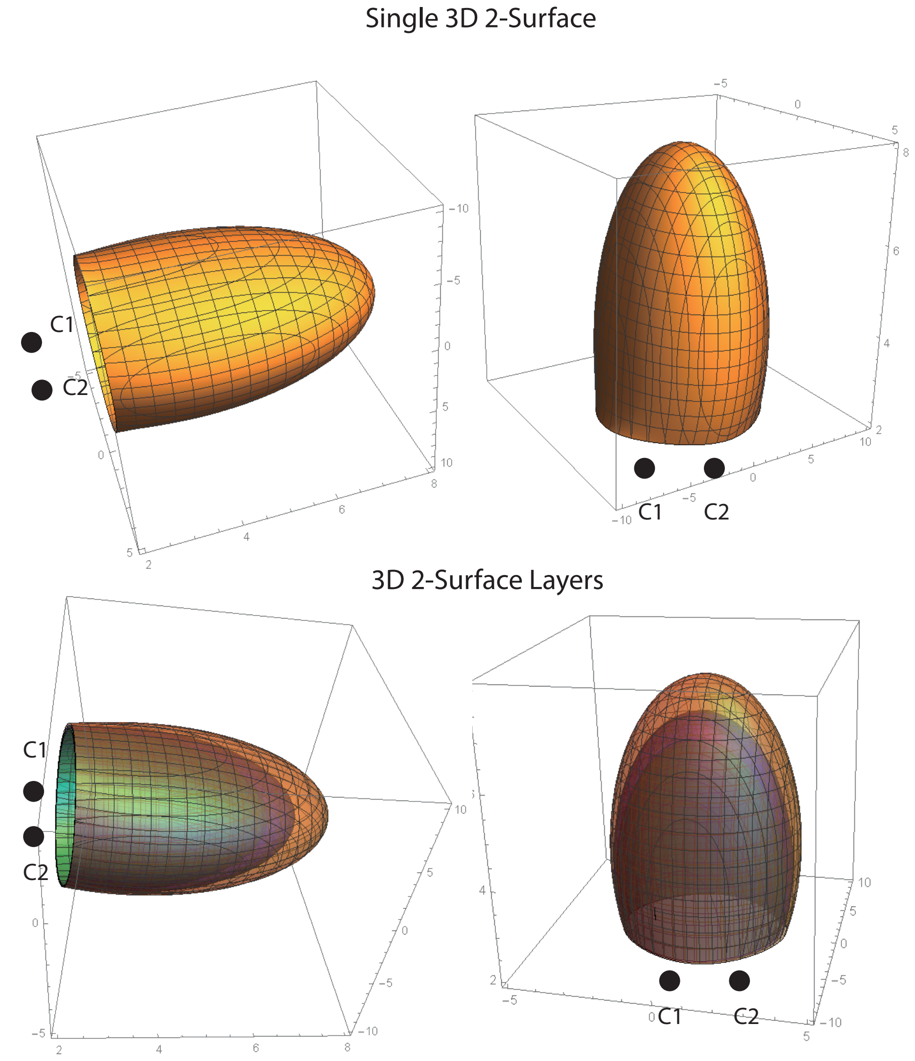


Figure 3.7: Demonstration of 3D 2-surfaces with a given camera arrangement.

lcoation **r**), written as

$$\mathbf{r} \;=\; H(k,u,\Theta) \tag{3.34}$$

The function H is the triangulation function, which maps the pixel of coordinate u with disparity k to its spatial location **r**. This equation describes how the uncertainty of the disparity is propagated to depth. Figure 3.8 visualizes this triangulation process. The entire uncertainty region for every coordinate u is a ring area highlighted in grey. The error in k, that comes from the correspondence matching algorithm, would lead to an uncertainty region $(k \pm \delta k, u)$, which is shown as a bold red line. The uncertainty is

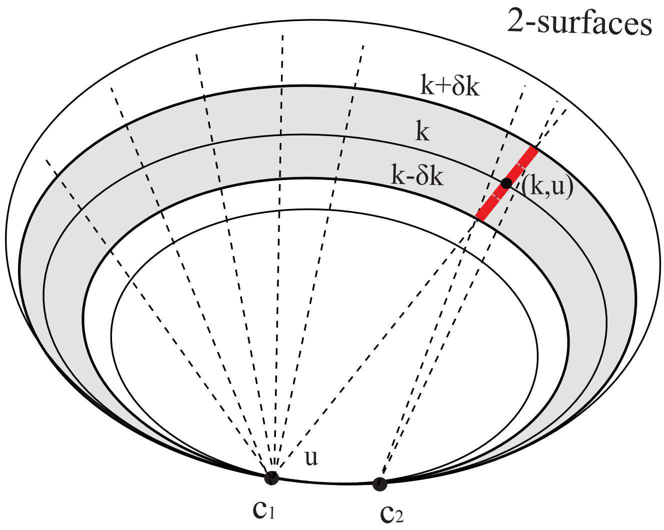


Figure 3.8: The error region caused by the triangulation process for a general camera arrangement.

symmetric with respect to the disparity measure k, however, the error in disparity can cause an error in the geometrical extent of depth which is asymmetrical. This error region is generalized to the spatial interval between the 2-surfaces.

Briefly, the distance of the interval between successive 2-surfaces, which could be measured by the density of 2-surfaces in a unit area, is important for the uncertainty of the disparity. This value is largely determined by the 2-surface topology and varies with the camera arrangement. To quantify this property, we propose a ”density” measure of the disparity field.

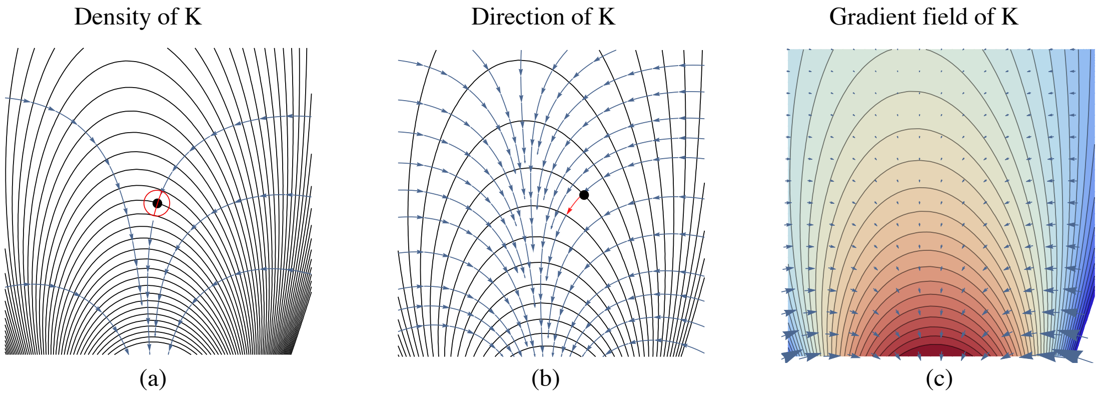


Figure 3.9: Examples of disparity field properties. (a) Density of the disparity field. (b) Direction of the disparity field. (C) Gradient field of disparity.

3.4.1 Density of the disparity field

The *density* is defined as the norm value of the gradient of the disparity field $||\nabla K||$. Shown in figure 3.9 (a), the *density* represents the maximum changes of disparity k at a point $\mathbf{r}$ whose direction is along the 2-surfaces normal. Higher density means that the value of k is changing rapidly around this point and reduces the spatial distance between successive 2-surfaces, therefore improving the accuracy.

The explicit relationship between the density of the disparity and the uncertainty caused by triangulation process could be derived as follows. According to multivariate error analysis [88], error matrix $\vec{\varepsilon_r}$ including both errors in $z-axis$ and $x-axis$ can be derived from Eq.3.34.

$$\vec{\varepsilon_r} = \begin{pmatrix} \varepsilon_x \\ \varepsilon_z \end{pmatrix} = J_H \varepsilon_k \tag{3.35}$$

$$J_H = \begin{pmatrix} J_x \\ J_z \end{pmatrix} = \begin{pmatrix} \frac{\partial H_x}{\partial k} \\ \frac{\partial H_z}{\partial k} \end{pmatrix} \tag{3.36}$$

where J_H is the Jacobian matrix of H. Eq. 3.35 indicates that the error in depth estimation stems from two processes, one is ε_k, which is dependent on the accuracy of stereo matching algorithm, the other is the Jacobian matrix J_H of triangulation process which depends on the camera arrangement . It should be noted that the multiplicative relationship between the two factors indicates that both are important for depth estimation accuracy. When the matching algorithm is fixed, J_H will determine the error.

Based on our definition above, the density of the disparity field can be derived as

$$||\nabla K(\mathbf{r})|| = \sqrt{(\frac{\partial k}{\partial z})^2 + (\frac{\partial k}{\partial x})^2} \tag{3.37}$$

$$= \sqrt{(\frac{\partial k}{\partial H_z})^2 + (\frac{\partial k}{\partial H_x})^2} \tag{3.38}$$

$$= \sqrt{\frac{1}{J_z^2} + \frac{1}{J_x^2}} = \left(\sqrt{\frac{J_z^2 J_x^2}{J_z^2 + J_x^2}}\right)^{-1} \tag{3.39}$$

The expression under the square-root in the right side of Eq. 3.39 is in the form of Harmonic mean of two terms J_z^2 and J_x^2. This shows that the norm of the gradient of the disparity field that represents surface density is inversely proportional to the square root of Harmonic mean of the triangulation error squared J_x^2 and J_z^2.

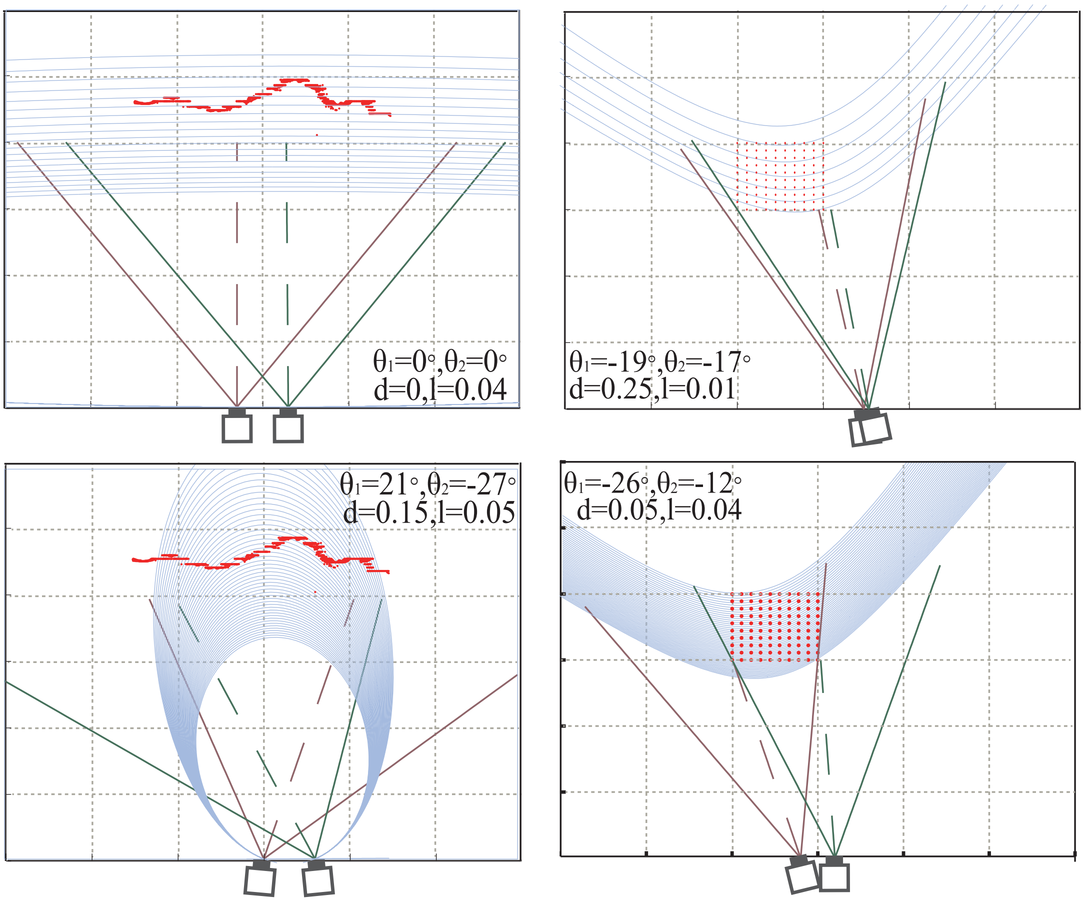

$\theta_1=0^\circ,\theta_2=0^\circ$
$d=0,l=0.04$
$\theta_1=-19^\circ,\theta_2=-17^\circ$
$d=0.25,l=0.01$
$\theta_1=21^\circ,\theta_2=-27^\circ$
$d=0.15,l=0.05$
$\theta_1=-26^\circ,\theta_2=-12^\circ$
$d=0.05,l=0.04$

Figure 3.10: Examples of density optimization for an area and a point cloud.

Therefore, the density is an indicator that can objectively determine the bound on the accuracy of depth estimation for a given camera arrangement.

The density of the disparity field is not constant across the scene. The depth estimation accuracy will be better for areas with higher density. The first column of Fig. 3.10 shows an object surface and the 2-surfaces from the top view of the scene before and after optimization of the density of k. It is clear that after optimization in the bottom figure, the density of 2-surfaces in the vicinity of the object has increased. The second column of Fig. 3.10 demonstrates a density optimization for a rectangular region, marked as a set of red dots, and gives a comparison between the low density arrangement and the optimized density arrangement. The camera arrangement Θ for each case is also shown.

The closed form of the density value can be obtained by combining the 2-surface equation K and the norm of the gradient equation as

$$\begin{aligned}
&\|\,\nabla \boldsymbol{K}\,\| = \\
&\sqrt{\Big(\Big(\Big(2\, l \left(l^2 - x^2 + z^2\right) - \cos(2\,\theta_1)\,(l + x)\,(l - x - z)\,(l - x + z) + \\
&\qquad (l - x)\,(2\,z\,(\sin(2\,\theta_2) - \sin(2\,\theta_1))\,(l + x) - \cos(2\,\theta_2)\,(l + x - z)\,(l + x + z))\Big)^2 + \\
&\qquad z^2 (2\,z \sin(2\,\theta_1)\,(l - x) + 2\,z \sin(2\,\theta_2)\,(l + x) - \cos(2\,\theta_2)\,(l + x - z)\,(l + x + z) + \cos(2\,\theta_1)\,(l - x - z)\,(l - x + z) + 4\,l\,x)^2\Big) \Big/ \\
&\quad \Big(2\,(\sin(\theta_1)\,(x - l) + z\cos(\theta_1))^4\,(\sin(\theta_2)\,(l + x) + z\cos(\theta_2))^4\Big)\Big)}
\end{aligned}$$

3.4.2 Direction of disparity field surfaces

Besides the density of the disparity field, there is another important factor for the error of disparity. Consider the following scenario in figure 3.11. The scene contains a slope wall object marked in red with two different camera arrangements. In figure 3.11(a), the 2-surfaces (shown in blue lines) are in a default planar arrangement associated with parallel cameras, while in figure 3.11(b) the direction of 2-surfaces are better aligned with the object surface. Both scenes have the same density over the object area and the estimated disparity is marked in blue crosses. Shown in 3.11(a), the correspondence matching algorithm has a high probability of generating levelling error due to rapid variation of k, which causes the zigzag effect. This phenomena is largely alleviated by adjusting the 2-surface orientation to align object surface in figure 3.11(b). In this way, we define another useful measure to quantify the direction of the disparity field to tackle this problem.

The *direction* of the disparity field at a point is measured by the normal of the 2-surface at that point (Shown in figure 3.9 (b)). The surface direction can be calculated as

$$\mathbf{n}_s = \frac{\nabla K(\mathbf{r})}{||\nabla K(\mathbf{r})||} \tag{3.40}$$

In stereo matching, if the normal vector of 2-surface is parallel to the normal of the object surface, the disparity variation along the 2-surface would be small, resulting in

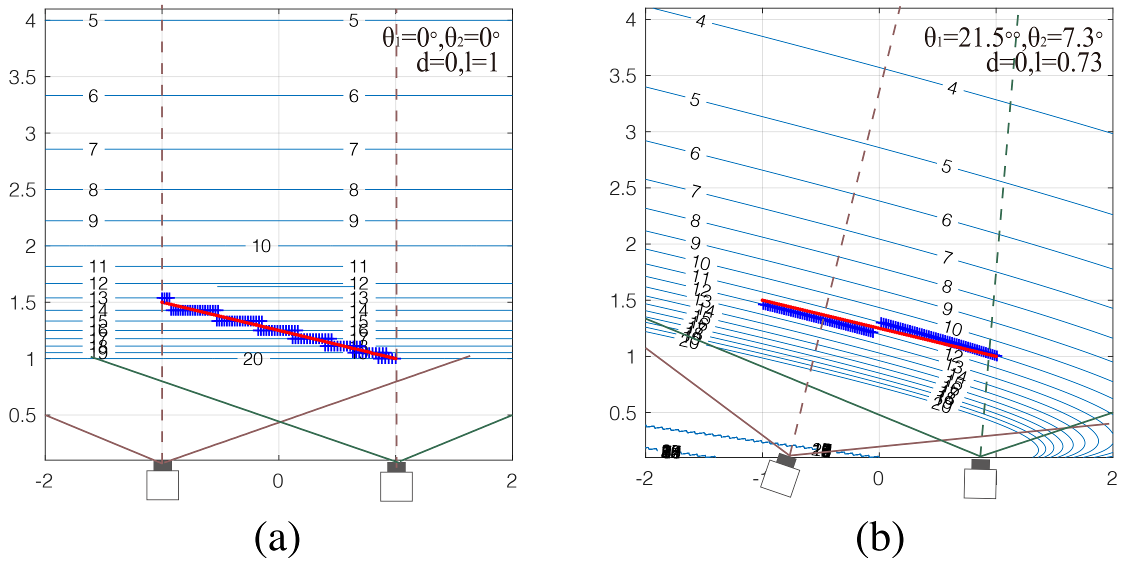


Figure 3.11: Examples of changing direction of 2-surface. (a) 2-surface of conventional camera arrangement. (b) Adjusted camera arrangement with 2-surface aligned to object surface.

reduced depth levelling errors. In contrast, when the 2-surface normal vector is perpendicular to the normal of the object's surface, there will be significant depth variations and high likelihood of occlusions.

This direction property can be used to control the alignment between 2-surface and the directions of object surfaces in the scene. This would also be helpful to minimize occlusions. Fig. 3.12 demonstrates a camera arrangement optimization based on the above objective. The first row shows that cameras can adaptively select the best view with respect to a rotating bunny scene according to the orientation of bunny's surface. The second row shows that cameras can align themselves according to the direction of object surface. This may also help with occluded areas, for example in this case certain parts of the scene in the middle are occluded when using the parallel arrangement. The direction optimization rotates the cameras into an arrangement which can cover and see more details in these areas.

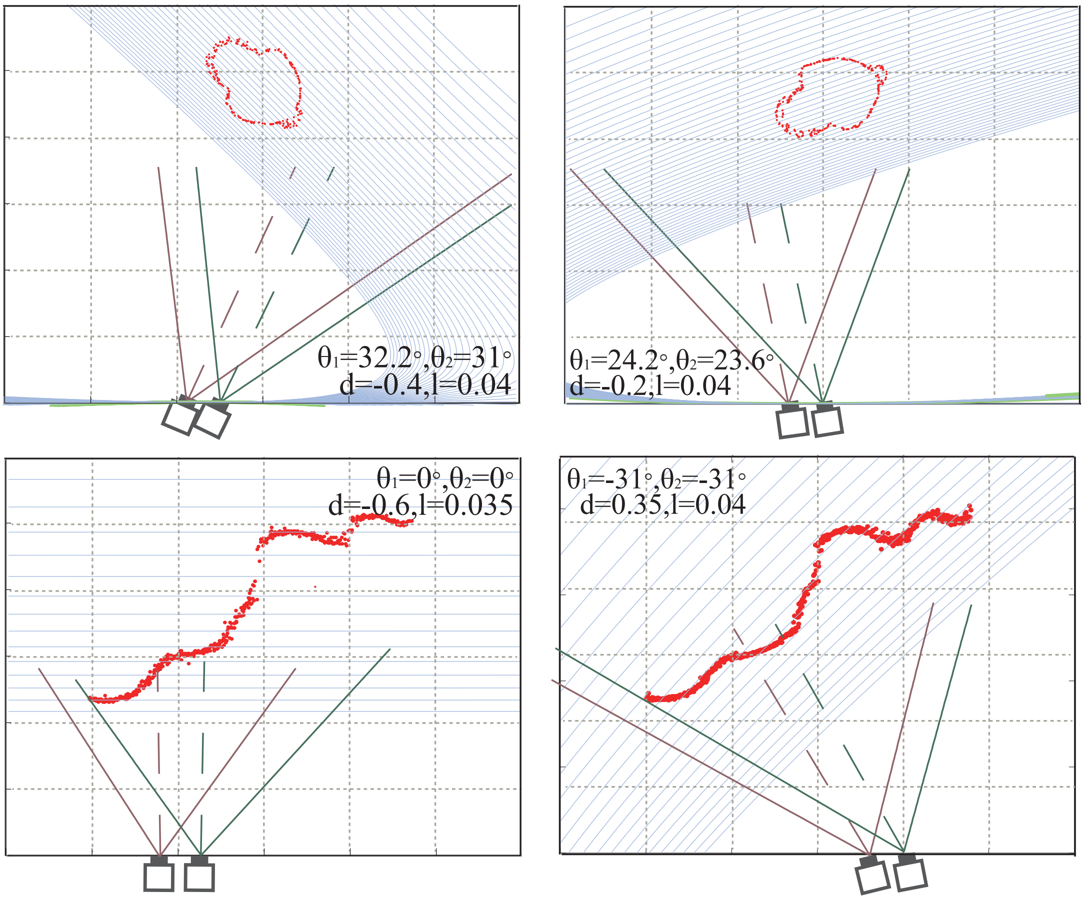

θ_1=32.2°,θ_2=31°
d=-0.4,l=0.04
θ_1=24.2°,θ_2=23.6°
d=-0.2,l=0.04
θ_1=0°,θ_2=0°
d=-0.6,l=0.035
θ_1=-31°,θ_2=-31°
d=0.35,l=0.04

Figure 3.12: An example of direction optimization for an object and a scene with occlusion.

3.4.3 Correspondence Field: the Gradient of the disparity field

When evaluating the goodness of the camera arrangement for depth estimation, both properties, density and direction of the disparity field, need to be considered. As described above, the gradient of disparity field can quantify both of these parameters.

The gradient $\nabla K(\mathbf{r})$ for point $\mathbf{r}$ could be used to maximize a directional density based on the object surface normal. Given an object surface element in Fig. 3.13, the gradient field ∇K should be decomposed into two components ∇K_t and ∇K_n, one is along the surface, the other is along the surface normal, which measure the density of the disparity along the two direction, respectively. As discussed above, when the 2-surface direction is aligned with the object surfaces, it is likely to minimize the error of depth estimation and the area of occlusion. Accordingly, the system should maximize the gradient of disparity

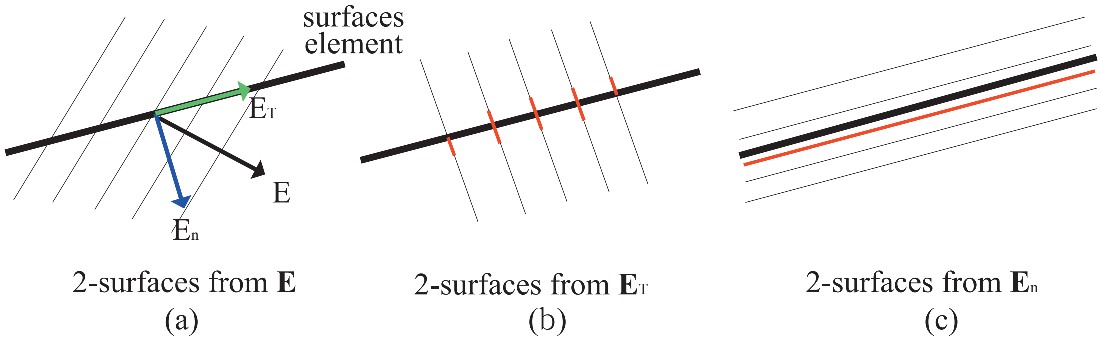


Figure 3.13: Illustration of disparity gradient field and its components.

along the surface normal direction as

$$K_r(\mathbf{r},\Theta) = \nabla K(\mathbf{r},\Theta)\cdot\mathbf{n}_r \tag{3.41}$$

The ∇K is considered as a key for the accuracy of the depth estimation. Therefore, the correspondence field is formally defined as follows.

Definition

The *correspondence field of two cameras (CF), denoted by* Ψ*, is the gradient of disparity within the common field of view of the cameras.*

If we assume that the camera properties, such as focal length or resolution, are fixed, then CF can be represented as $\Psi(\mathbf{r},\Theta) : \mathbb{R}^3 \to \mathbb{R}^3$, where $\mathbf{r}$ is the position vector for the location in the field of view, and Θ is the camera arrangement as defined before. Hence

$$\Psi(\mathbf{r},\Theta) \equiv \nabla K(\mathbf{r},\Theta) \tag{3.42}$$

With the above definition, $\Psi(\mathbf{r},\Theta)$ is a conservative field; its magnitude $|\Psi(\mathbf{r},\Theta)|$ is a measure of density and its direction determines the direction. In the next chapter, we will build an objective function based on Ψ to find the optimal camera arrangement for a given scene.

Chapter 4

Depth Estimation with CF based Optimization

In this chapter, the CF-based camera arrangement optimization (CFC) method is developed, investigated and experimented. First, the objective function, which is the core of optimization, is described. Second, an expectation maximization (EM) framework is introduced for iterative optimization based on the objective function. The procedure and performance of the EM-based approach are presented. Finally, this CF-based camera arrangement optimization (CFC) method is experimented on both synthetic dataset and real scene dataset. The results show that the proposed CFC improves the accuracy of depth estimation by up to 30% for a variety of scenes.

4.1 CF-based camera optimization

By using the correspondence field Ψ, an objective function can be defined to optimize the arrangement of cameras for the depth estimation. This function takes the point $\mathbf{r}$ on the object surface and its normal $\mathbf{n}_r$ as input variables. Their values can be either an initial guess or an estimate from the previous iteration of the algorithm. As described in the previous chapter, for each $\mathbf{r}$, the dot product between the Ψ and the surface normal $\mathbf{n}_r$ represents the good alignment as well as density for depth estimation and the objective function, Φ, can be defined as

$$\Phi(\Theta, \mathbf{r}) = \int_{\Omega} |\psi(\mathbf{r}, \Theta) \cdot \mathbf{n_r}\, d\mathbf{r} = \int_{\Omega} |\nabla K \cdot \mathbf{n_r}|\, d\mathbf{r} \tag{4.1}$$

where Ω is the region of interest in the scene. If Ω is a set of descrete points, for example representing a point cloud, then the above function will be in the form of a summation instead of integral. The discretized form of Φ can be written as

$$\Phi(\Theta, \mathbf{r}) = \sum_{\mathbf{r} \in \Omega} |\psi(\mathbf{r}, \Theta) \cdot \mathbf{n_r}| \tag{4.2}$$

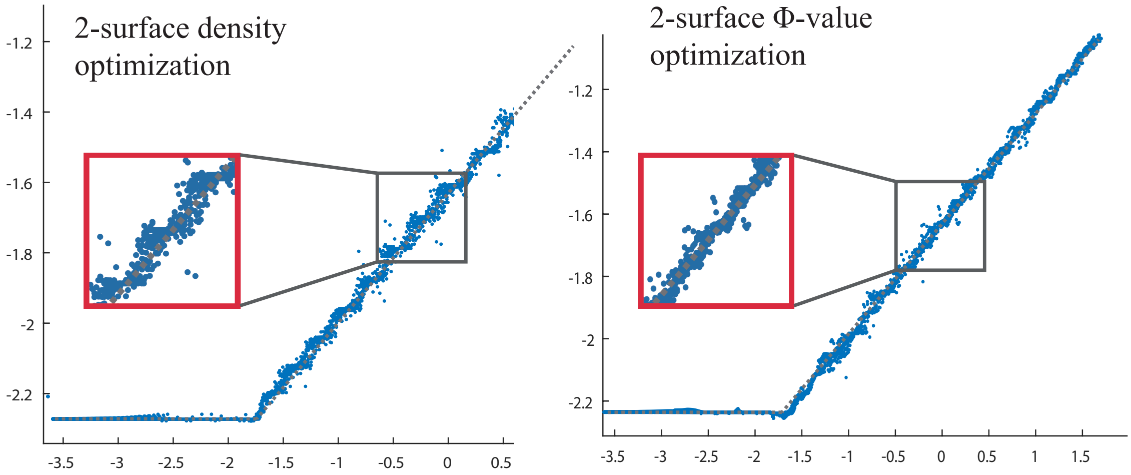


Figure 4.1: An example of optimization using density of the disparity field and Φ value of the correspondence field.

Demonstration of Φ Optimization

Fig. 4.1 compares the estimated points of a sloping object surface by optimizing density of disparity field alone and optimizing based on the gradient field Φ value above. It can be seen that optimization based on the density (the left plot) results in some sort of stair-like levelling error by neglecting direction, while Φ-value based optimization in the right plot gives a more accurate estimation.

4.1.1 Expectation maximization based algorithm

Based on the above analysis and discussion, the arrangement of a pair of cameras can be optimized for depth estimation by maximizing Φ, i.e.

$$\arg\max_{\Theta} \quad \Phi(\Theta, S) \quad \text{s.t.} \quad L \leq \Theta \leq U \tag{4.3}$$

where $\Theta \in \mathbb{R}^4$ represents the camera pair arrangement $(\theta_1, \theta_2, d, l)$; $[L, U]$ are the upper and the lower bounds for variations of Θ respectively; and S represents the scene surface including surface points $\mathbf{r}$ and their normals $\mathbf{n}_r$.

Since the scene S is unknown, it has to be estimated while the camera arrangement Θ is being optimized. An expectation maximization (EM) approach that iteratively estimates S and adjusts parameters of Θ to maximize the objective function is adopted. The expectation (E) step first estimates S and creates an expectation function with respect to the S based on the current Θ. Φ is chosen as the expectation function in this thesis as it pertains to the accuracy of depth estimation. The maximization (M) step computes parameters Θ maximizing the Φ found in the E-step. The estimated Θ is then used to determine the distribution of S in the next E step. The algorithm is detailed in Algorithm 1.

Algorithm 1 CF-based camera optimization

```
Input: $I \leftarrow$ Images from cameras
Output: $\Theta \leftarrow$ Camera arrangement parameters
Set the initial arrangement $\Theta^{(t)}|_{t=0}$
while $||\Theta^{(t+1)} - \Theta^{(t)}|| > T_{stop}$ do
    E-step : estimate $S$ under the given arrangement $\Theta^{(t)}$ by using stereo and normal estimation algorithm $G$:
        $S(\Theta^{(t)}) = G(\Theta^{(t)}, I(\Theta^{(t)}))$    (4.4)
    formulate the expected value function:
        $\Phi(\Theta|S(\Theta^{(t)}))$    (4.5)
    M-step : adjust arrangement to $\Theta^{(t+1)}$ given the estimated S as
        $\Theta^{(t+1)} = \arg\max_{\Theta} \; \Phi(\Theta|S(\Theta^{(t)})) \quad \text{s.t.} \quad L \leq \Theta \leq U$    (4.6)
end while
```

4.1.2 Initialization

At the initial stage, there is no geometry information S of the scene. The initial camera arrangement Θ can be obtained by maximizing density of the disparity field for the whole bounding area of a scene. The objective function Φ can be expressed as

$$\Phi(\mathbf{r},\Theta) = \int_{\Omega} ||\psi(\mathbf{r};\Theta)|| d\mathbf{r} \tag{4.7}$$

In our implementation, the cameras start from a conventional parallel setting, with angles $\theta_i = 0$, centroid location d_0 and distance to centroid l_0.

4.1.3 E-Step: Estimate S given Θ

This stage attempts to estimate S given the current arrangement, Θ, of the cameras. Namely, all the location $\mathbf{r}$ of object surfaces in the scene with the corresponding normal $\mathbf{n}_r$ to the surface at each point. Estimation of S consists of three stages.

The first stage is depth estimation shown in Fig. 4.2 (a). Given the current camera arrangement, the system will estimate a depth map by using a suitable stereo matching algorithm. Five top stereo matching algorithms are employed in this thesis to assess the robustness of camera arrangement optimization. These are Slanted plane smoothing stereo matching (*spsstereo*) [31], Efficient large-scale stereo matching (*libelas*) [28], Robust stereo matching using adaptive random walk with restart algorithm (*openrwr*) [30], Stereo matching based on the fast cost-volume filtering (*costfilter*) [27] and Accurate and efficient stereo by semi-global matching (*blockmatching*) [29].

The second stage is noise reduction. All the valid pixels are triangulated to create a point cloud. A statistical density-based outlier detection [89] is performed to remove the outliers where the number of neighbour points is set to N_m. All points that have a distance larger than T_s standard deviation of the mean distance to neighbouring points will be marked as outliers and removed. Then the filtered point cloud is sub-sampled uniformly into N_s points to efficiently analyse its structure, as shown in Fig. 4.2 (b).

In the third stage, shown in Fig. 4.2 (c), the refined point cloud will be decomposed into

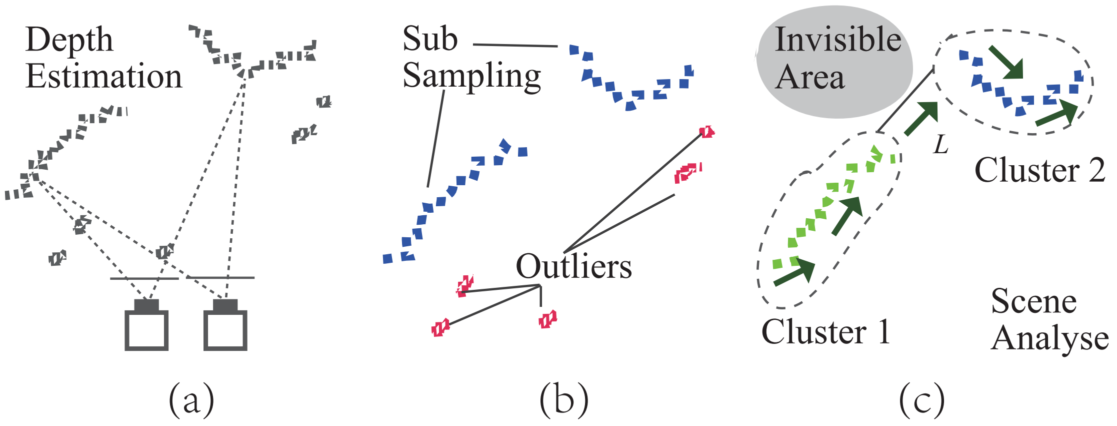


Figure 4.2: Different steps during the scene analysis: (a) depth estimation; (b) noise reduction; and (c) clustering.

a number of clusters. The decomposition considers point cloud density and a DBSCAN [89] clustering technique is utilized to identify point groups with sufficient density and appreciable separation. The DBSCAN algorithm can be broken down into the following steps: First, for every point, define its neighbours by a specified range, and identify the core points with more than a minimum required neighbours. Then, find the connected components of core points on the neighbor graph, ignoring all non-core points. Each connected graph of the core points is a cluster. Finally, assign each non-core point to a nearby cluster if it is at least one neighbour of any point in this cluster, otherwise assign it to noise. The generated clusters are then used in the optimization stage.

When two neighbouring clusters are connected by plane L, illustrated in Fig. 4.2 (c), the area behind L is invisible. This area is regarded as an occluded area. The direction of the plane L could be regarded as a guide for assessing the 2-surface direction to probe an occluded area. To include this information in the system, points are sampled uniformly from the estimated plane and merged with the observed point cloud.

The estimation of normal $\mathbf{n}_r$ at a point $\mathbf{r}$ is done by a principal component analysis (PCA) on a local point patch. The PCA algorithm approximates a tangent plane at a given point by regression on its neighboring points. The normal of the point is defined as the eigenvector corresponding to the smallest eigenvalue of the covariance matrix of its neighbours. Consider $\mathbf{W}(\mathbf{r})$ to be the neighbouring N_w number of points around a point

$\mathbf{r}$, then $\mathbf{n}_r$ is the eigenvector of the covariance matrix $\mathbf{W}\mathbf{W}^{-1}$.

Given the estimated S, the expected value function Φ could be obtained by Eq.4.2. It is noted that by treating $\mathbf{r}$ and $\mathbf{n}_r$ as constants, the only parameter in Eq.4.2 is now camera arrangement Θ. Therefore, it is possible to perform numerical optimization algorithm to find the optimal Θ in the next M-step.

4.1.4 M-Step: Optimizing Θ given S

In this algorithm, the constraints of the problem (limitation of the movements and rotation angle) are defined as box constraints within bound $[L,U]$. Therefore, we use a trust-region-reflective optimization algorithm [90] to solve this box constrained non-linear optimization problem shown in Eq.4.6. In each iteration, the parameter Θ is moved by a trial step $\Delta\Theta$, which is computed by solving a quadratic approximation of the target function

$$\Delta\Theta = \min_{\Delta\Theta} \quad \frac{1}{2}\Delta\Theta^T H \Delta\Theta + \Delta\Theta^T g \text{ s.t. } \quad ||Ds|| \leq \Delta \tag{4.8}$$

where g is the gradient of Φ at the current iteration and H is the symmetric matrix which approximates the Hessian of Φ and $\Delta > 0$ is a trust region radius. Eq. 4.8 can be solved as a two dimensional subspace minimization and the step $\Delta\Theta$ is calculated using a preconditioned conjugate gradient process [91] where $\Delta\Theta$ satisfy

$$(D^{-1}HD^{-1} + diag(g)J)D\Delta\Theta = -D^{-1} \tag{4.9}$$

In Eq. 4.9, D is the Dirac function and J is the Jacobian of Φ. When the components of $\Delta\Theta$ vector $(\Delta\Theta_1, \Delta\Theta_2, ..\Delta\Theta_i., \Delta\Theta_n)$ crossed boundary constraints $[L,U]$ by $\mathbf{o} = (o_1, o_2, ..o_i.., o_n)$ (o_i is zero if it does not cross the boundary). A reflection process is made and the step $\Delta\Theta$ is changed to $(\Delta\Theta_1 - 2o_1, \Delta\Theta_2 - 2o_2, ..\Delta\Theta_i - 2o_i., \Delta\Theta_n - 2o_n)$ for this iteration to realize constraint optimization.

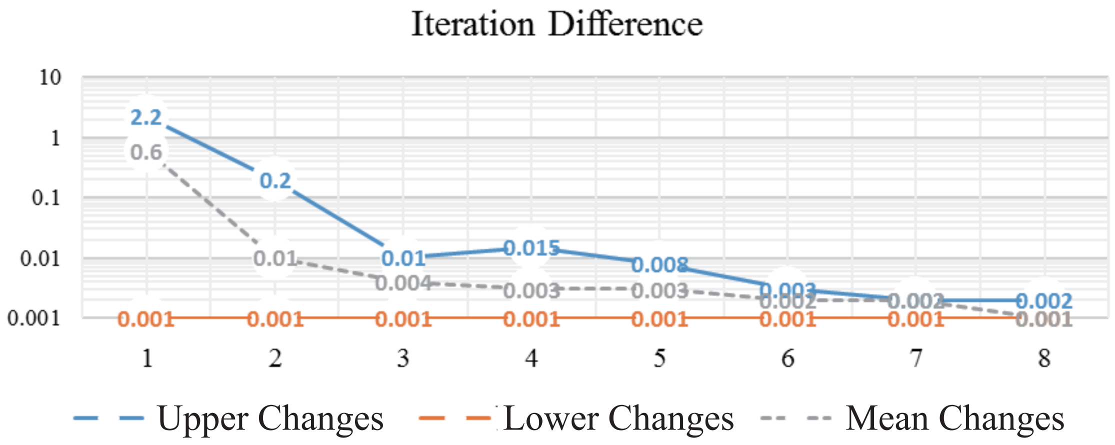


Figure 4.3: The change of Θ between two consecutive iterations of the optimization. The upper and lower bounds are the maximum and minimum change over all scenes.

4.1.5 Termination Condition

The optimization of Θ is terminated if the absolute change of Θ in two consective iterations is less than a threshold T_{stop}. The change is measured by the distance between the Θ in current and previous iteration as $||\Theta^{(t+1)} - \Theta^{(t)}||$. Fig. 4.3 shows the average change of four camera arrangement parameters Θ of all experiments. The camera arrangement usually converges to a stable result in a small number of iterations, though an explicit proof for convergence of camera arrangement is yet to be established.

4.1.6 Complexity

The algorithm consists of two steps, the E-step and M-step. The E-step involves the process of triangulation and normal estimation. The computational complexity of triangulation depends on the number of triangulated pixels and is in the order of $O(wh)$, where w and h are respectively the height and width (in pixels) of the primary image. The computation of normal estimation is a function of the number of estimating points N_p and the number of neighbours B. Its complexity is in the order of $O(B^2N_p)$. Generally, N_p is equal to the pixel number wh so that the overall complexity of the E-step is $O(B^2wh)$. The M-step mainly involves the correspondence field computation. In each EM iteration, the CF is calculated for each 2-point, hence, the complexity is $O(N_p)$. In addition, a typical stereo matching algorithm's complexity is $O(Rwh)$, where R is the number of disparity scanning layers, which is usually around 100 and can be much more if fractional-disparity is employed.

It is noted that stereo matching, point cloud triangulation and normal estimation are necessary for the proposed method, and their complexity increases linearly with the number of pixels, i.e. $O(wh)$. While CF computation depends on the number N of points in the point cloud, which is equal to wh if every pixel is used for optimizing camera arrangement. Consequently, the complexity of the proposed method would grow at the same rate as the stereo matching algorithm. However, in practice, N_p can be far smaller than wh, because in a dense point cloud neighbouring points have very similar values. In our implementation, we down-sampled the pixels by a factor of 8, that is $1/64$ of wh, for the calculation of Ψ.

In the experiments, the CPU time spent on the EM algorithm was almost negligible compared to that on the stereo matching, because B are commonly smaller than R. Also, the computation time of Ψ (requiring 8 additions and 10 multiplications) is much smaller than the computing time spent on the calculation of photo-consistency, e.g. NCC metric, in stereo matching and on the calculation of filtering operation, e.g. smoothing per 8×8 pixel block. On a PC with an Intel i7 4770k CPU and single core enabled, it takes about 4-6 seconds on average for stereo matching whereas the optimization process could be finished within 500ms each cycle.

4.2 Experiments

This section presents an experimental validation of the proposed CF-based Camera Arrangement Optimization (CFC). The accuracy of depth estimation after CFC is compared with the conventional parallel camera arrangement for simulated data as well as real scenes. The improvement of the depth estimation are mostly evaluated and described by the error reduction in percentage, that is $(\frac{InitialDepthError}{CFCDepthError} - 1)\%$ in the following experiments.

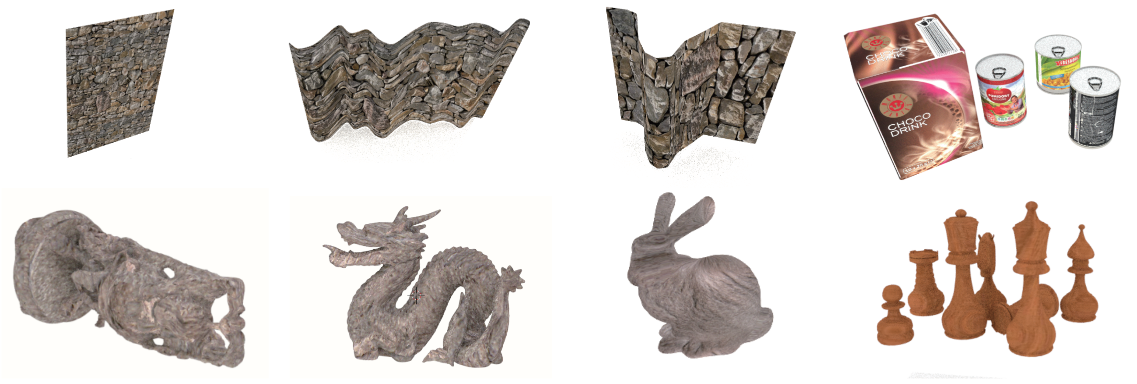


Figure 4.4: Various scenes used in the evaluation dataset.

4.2.1 Simulation dataset and parameters

Currently available public datasets focus on quality improvement in stereo matching algorithm [92], and provide acquired images based on fix arrangements. These datasets, therefore, are not suitable for evaluating camera arrangement optimization. Consequently, eight different scenes are created in Blender 3D by using the widely known 3D models, including the Stanford Scanning Models [93]. For each scene, the input images for stereo matching can be obtained from arbitrary arrangements. In order to make robust evaluation of the optimization algorithm, scenes are built with various sizes, shapes, and distribution of objects. As illustrated in figure 4.4, the first three scenes have rock walls with various shapes. These structures make it easy to understand and show the effects of optimization algorithms. The cans and the chess scenes include different degrees of occlusions. In addition, three sculptures, the buddha, dragon and rabbit in Stanford dataset are tested, which show surface complexity and require high depth accuracy to resolve.

4.2.2 Camera pair optimization

The experiments started from the parallel camera arrangement and ended with an optimized camera arrangement. These two arrangements are compared in terms of the accuracy of the depth estimation over the eight scenes. Meanwhile, the optimization on each scene is repeated five times with five different stereo matching algorithms. These are Slanted plane smoothing stereo matching (*spsstereo*) [31], Efficient large-scale stereo matching (*libelas*) [28], Robust stereo matching using adaptive random walk with restart algorithm (*openrwr*) [30], Stereo matching based on the fast cost-volume filtering (*cost-filter*) [27] and Accurate and efficient stereo by semi-global matching (*blockmatching*) [29].

In the experiments, the scene range Ω was set to $[-2m,2m] \times [2m,4m]$ representing left, right, front and back, respectively. The cameras were assumed to be on a rail at $Z_{rail} = -4m$ with distance to center $l_0 = 0.04m$ and centroid location $d_0 = 0$. The cameras intrinsic parameters were set as follows: $f = 600, resx = 720, resy = 320$ pixels. The labelled depth was firstly de-noised with $T_r = 0.0078$, followed by outlier filtering [89], in which the point group size N_m was set to 10 and the standard deviation threshold T_d was set to 2.00. The sub-sampling points N_s was set to 3000 and the number of neighbours N_w in normal estimation was set to 15. Four parameters of camera arrangement $(\theta_1, \theta_2, l, d)$ were estimated as described in section 3.2. The rotation angles θ_1, θ_2 were constrained within $\pi/6$ for both directions, the distance to centroid l has a range $[0.02, 0.05]$ and the centroid location can be moved from $-2m$ to $2m$. The accuracy of the depth estimation is commonly measured by the mean of Hausdorff distance between the final estimated point cloud and the ground truth model of scenes [94]. In our system, the ground truth depth of each ray is known.

Fig. 4.5 demonstrates some interesting effects when controlling different parameters for CF optimization. The two images in the first row show a chess scene and its depth error map estimated by using initial parallel stereo arrangement. Then different factors of CF optimization are examined and shown in a 2×2 image matrix in Fig. 4.5. The rows of this matrix represent possible options associated with the changeable parameters of the camera arrangement, i.e., only allowing rotation of cameras or rotation plus relocation. The columns of this matrix represent the optimization objective, i.e., choosing density optimization alone (Eq. 4.7) or CF-based optimization. The results are shown as four error maps. From the first row, it can be seen that the total error can be reduced by 10% to 20% when only allowing rotation. Full parameter optimization can achieve 30% error reduction (shown in the second row). From another perspective, CF-based optimization provides 31.7% error reduction when allowing rotation and movement and 17.4% when only rotation is allowed, compared with 9.7% and 29.1% respectively when density optimization alone is employed.

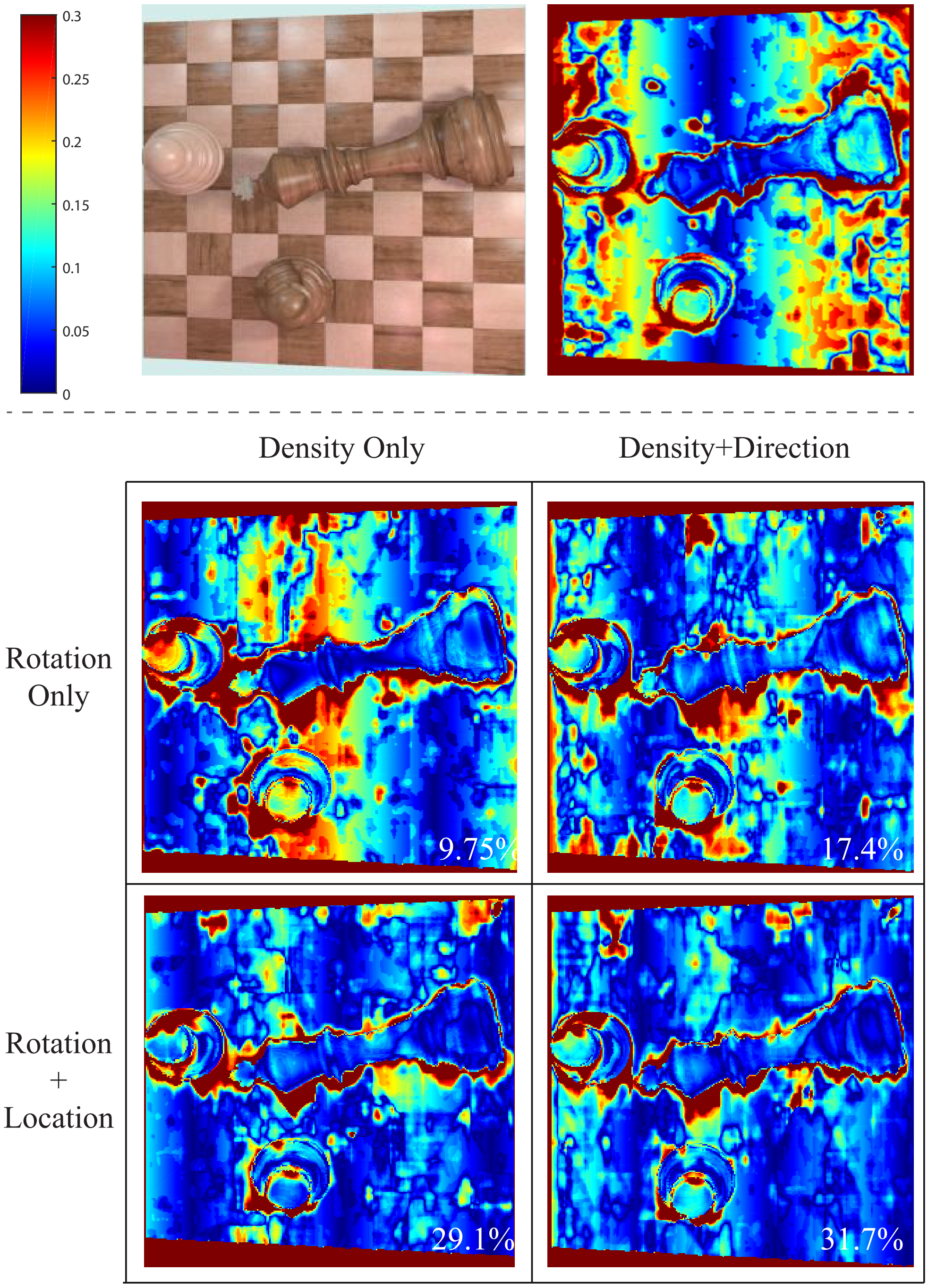


Figure 4.5: Example depth maps under different optimization: The first row shows the scene and its disparity map with conventional parallel setting. The second and third rows are the resulting depth error maps by controlling camera arrangement based on the stated optimization strategy.

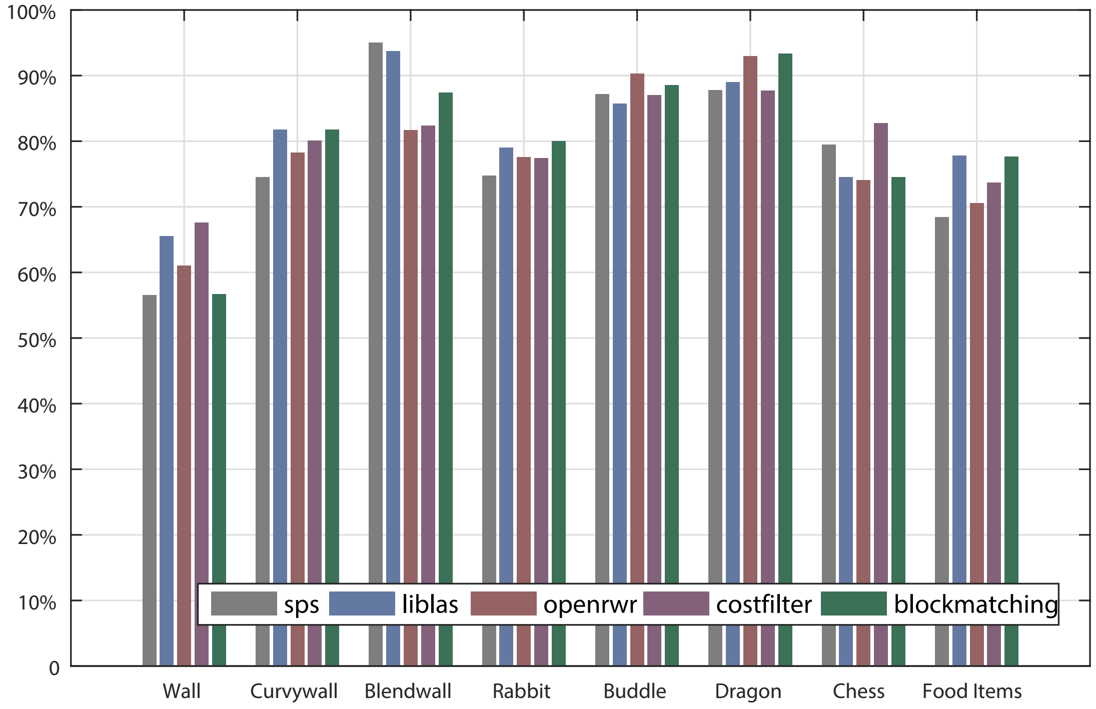


Figure 4.6: Depth error reduction in percentage with respect to parallel stereo arrangement by using five stereo algorithms over the eight simulated scenes

Fig. 4.6 shows overall statistics of the depth accuracy improvement by using CFC. The x-axis represents eight different scenes and y-axis represents the depth error in percentage compared to initial depth errors. It can be seen that the proposed CFC optimization has largely improved the accuracy of depth estimation for all stereo algorithms. CFC produces stable accuracy and reduces the error by 10% to 30% on average and can reach 45% for the curvy wall scene.

Fig. 4.7 shows the initial and final camera arrangements for scenes shown in sub-figures (a) and (b), respectively. The two camera arrangements turn out to be significantly different, i.e., $(0,0,0.03,0)$ and $(-4.3°,-6.6°,0.05,-0.22)$. In the top view plot, the red circle points show the estimated point cloud while the grey cross points are the ground truth. It can be seen that CFC method reduces the depth error when estimating curved shapes, and the estimated points closely align with the ground truth. The colour coded depth points of the 3D objects are displayed next to the main plot for regions labelled one and two, where the darker color means the closer locations of the objects. The optimized camera arrangement presents an improvement in smoothness and details, e.g. in the sub-figure one, the

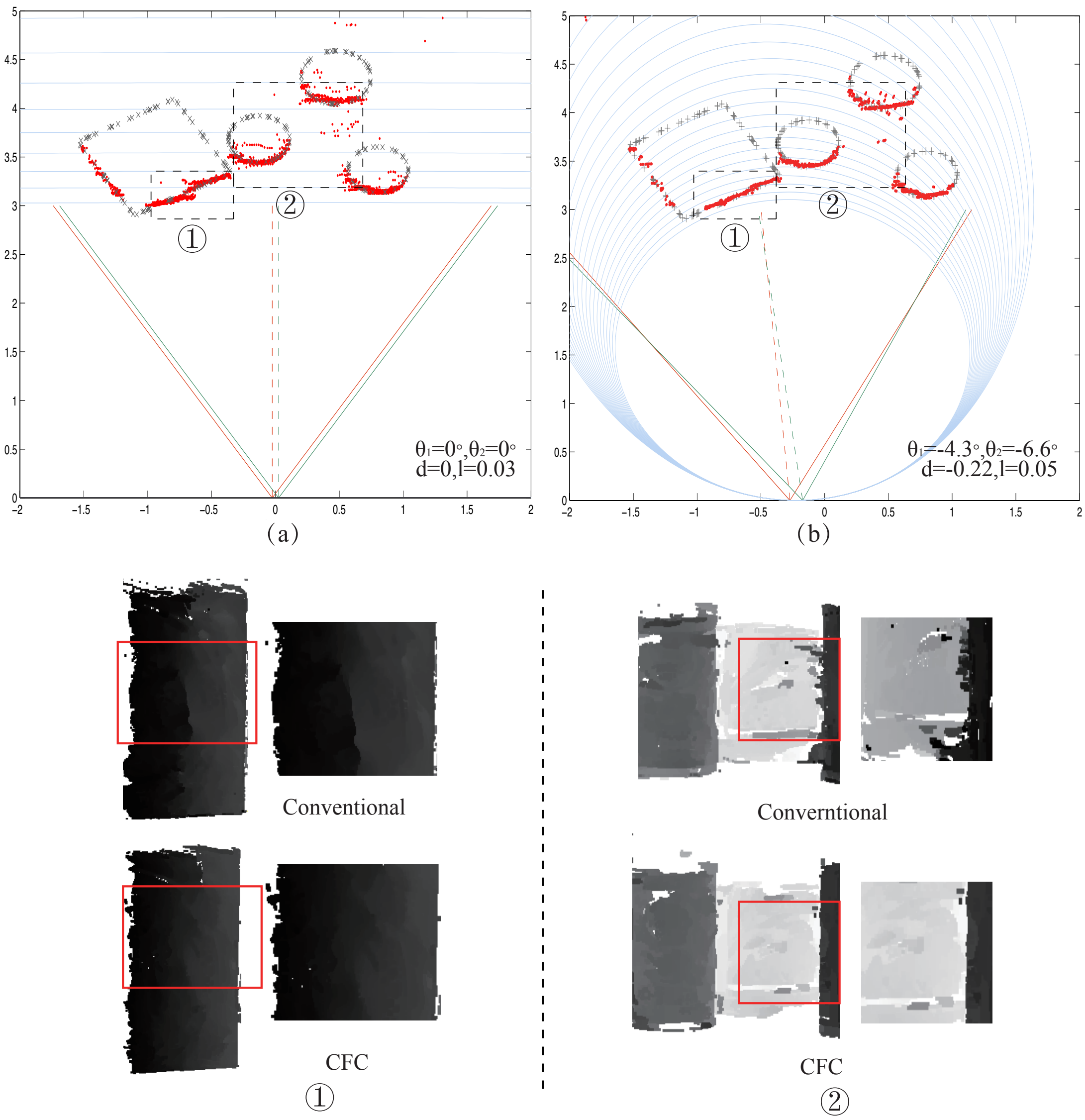


Figure 4.7: The top view of the scene and the estimated 3D point cloud based on: (a) conventional, and (b) CFC arrangements. The arrangement of cameras is shown besides the camera pairs.The depth maps of marked regions in (a) and (b) are enlarged and shown on the right.

parallel arrangement shows incorrect depth discontinuity in the middle of the plane due to the error of sub-pixel interpolation (it is noted that the density of the disparity field is low) while the CFC method shows smooth and consistent surface estimation for this object as sufficient correspondences are arranged around the object location. As for region two, the quadratic shape of 2-surface by converged camera setting meets the direction of all three cylindric objects which help increase smoothness in the point cloud and reduce occlusion at the edges of objects.

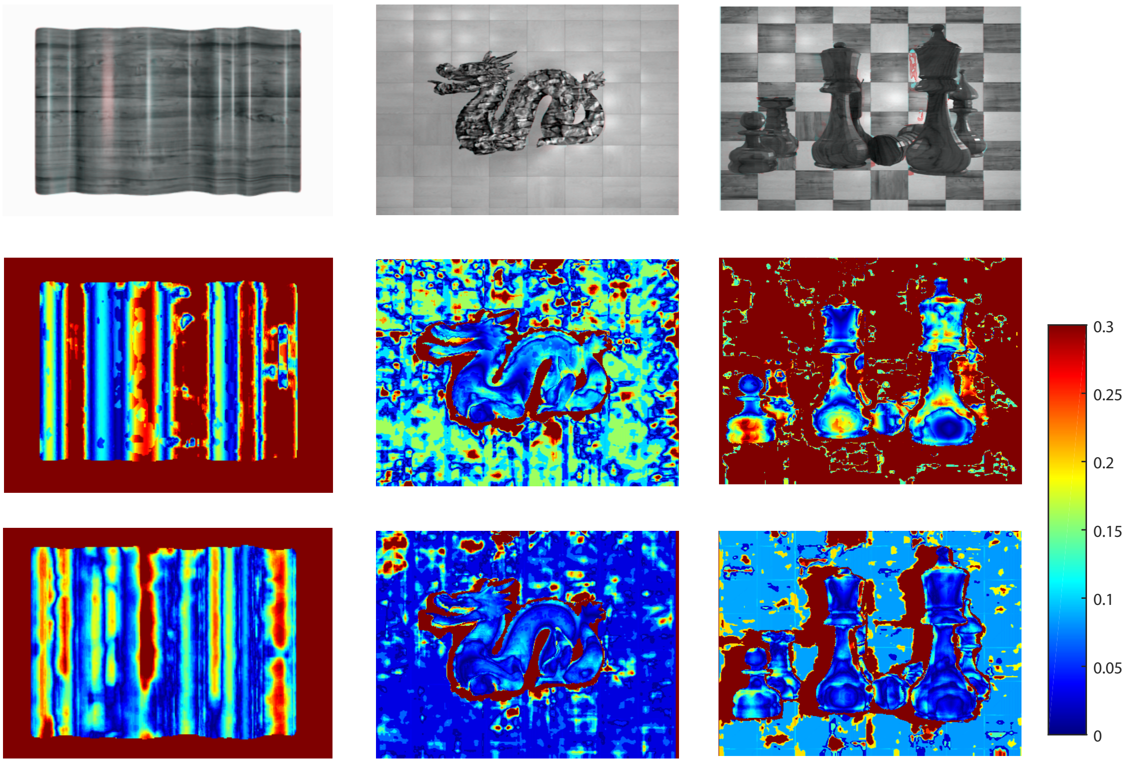


Figure 4.8: Example depth error maps before and after optimization in simulation results. The error from large to small is coded from red to blue color.

Scene	Spstereo			Liblas			Openrwr			Cost Filter			Block Matching		
office_1	1.5911	1.2064	**20.94%**	1.3414	1.1891	14.71%	1.2814	1.0419	16.25%	1.5062	1.3390	7.84%	1.2120	1.1069	15.59%
office_2	1.4505	1.1271	**13.82%**	1.2348	1.0693	14.40%	1.1320	0.9455	16.92%	1.4018	1.1275	20.62%	1.3377	1.0595	22.98%
desk_ 1	1.4039	0.8333	42.04%	1.1690	0.7810	26.88%	1.3063	0.7340	**41.00%**	1.2290	0.8032	29.85%	0.9653	0.6832	27.20%
desk_ 2	1.6868	1.4283	10.95%	1.2417	1.0810	13.62%	1.4427	1.1340	**21.15%**	1.1320	1.0255	15.27%	1.1802	1.0122	14.06%
teddy_1	0.1168	0.0795	31.17%	0.3130	0.2888	4.59%	0.0975	0.0697	28.00%	0.1270	0.0857	**33.77%**	0.1273	0.0885	27.52%
teddy_2	0.0910	0.0712	23.61%	0.3110	0.2983	7.07%	0.0955	0.0722	22.70%	0.1163	0.0834	**20.21%**	0.1107	0.0821	27.48%
xyz_1	2.2809	1.5918	30.21%	2.4317	1.3603	**44.06%**	1.8520	1.5205	17.90%	2.3526	1.4398	38.80%	1.9013	1.4637	23.02%
xyz_2	1.5006	1.1218	**29.54%**	1.2116	0.9984	17.60%	1.7302	1.2482	27.86%	1.8628	1.5244	18.19%	1.5141	1.3101	13.47%

Table 4.1: Depth estimation accuracy with five stereo algorithms on eight scenes

Fig. 4.8 illustrates some improved depth maps of different scenes. The first row shows the scene views, the second row shows the depth error maps using the initial arrangement, and the optimized depth error maps are presented in the last row. The error is coded into a color map with fix absolute range $[0\ 0.03m]$. The comparison between the initial and optimized depth error maps shows significant depth error reduction for the foreground part of the scene. Also, the background with low disparity range is improved after CF is densified.

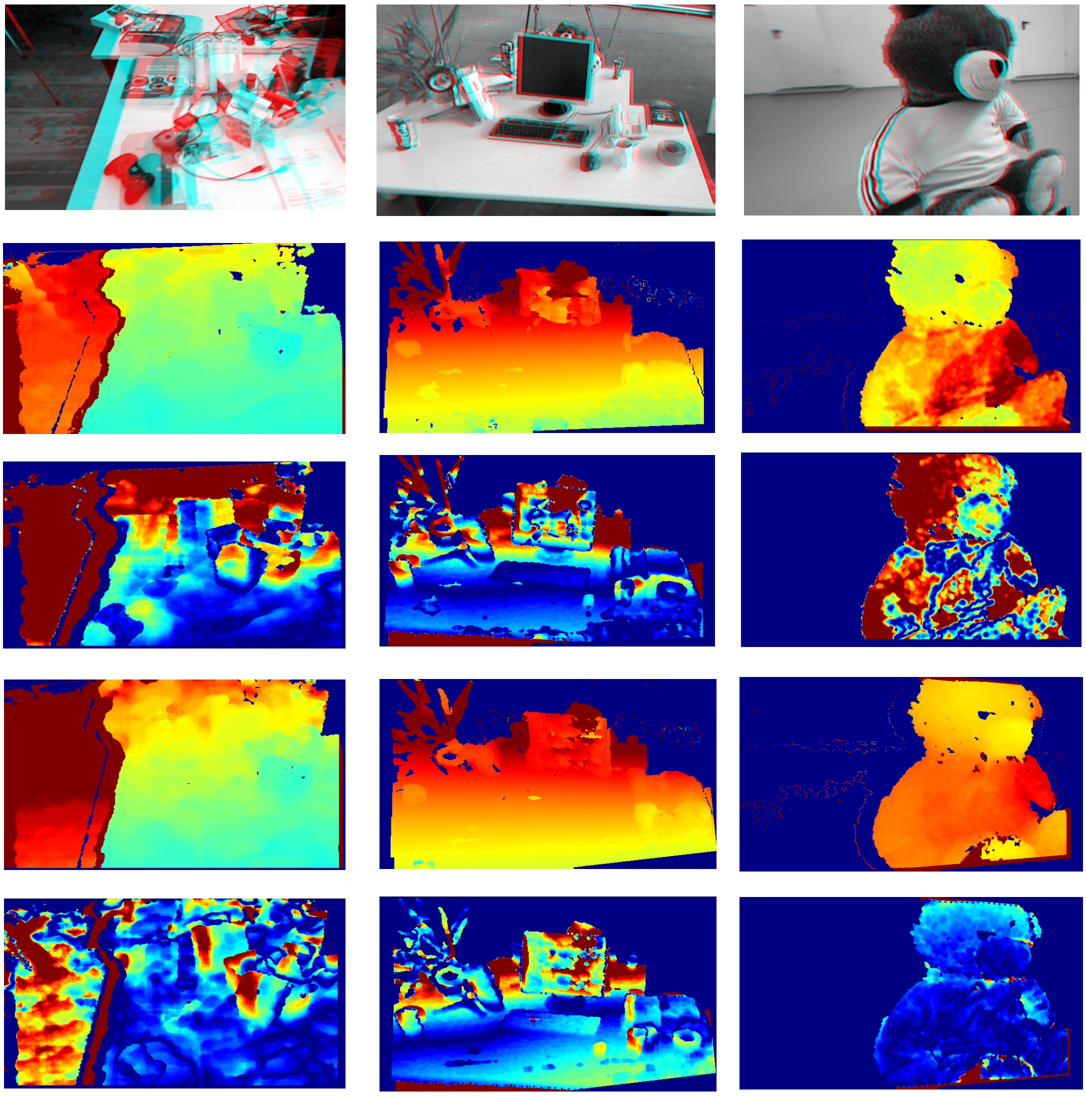


Figure 4.9: Example depth error maps before and after optimization in simulation. First row: scene image, second and third row: initial depth map and depth error map, fourth and fifth row: optimized depth map and depth error map. The error increases from blue to red

4.2.3 Evaluation on the TUM-SLAM dataset

Camera arrangement optimization for real scenes requires different camera poses with ground truth depth data for images captured from each pose. There is no specific benchmark dataset for this purpose. Nonetheless, visual SLAM datasets include a variety of scenes that have large range of complexity, ground truth camera data as well as RGB-D images. Their frame selection process can be cast as a problem of camera arrangement optimization, in which each possible camera arrangement is given a score (e.g. Φ value) and the best will be selected. Therefore, the TUM RGB-D benchmark [70] dataset is used in this thesis.

In our experiments, the camera pose generated from an external device is considered as a known arrangement for optimization and the depth information obtained from the Kinect device is considered as the ground truth. The scenes were selected from the 3D reconstruction subset of the TUM RGB-D dataset.

As TUM RGB-D dataset is designed for SLAM, we compare the depth map error estimated by a SLAM algorithm and that estimated by a pair of frames selected with the highest Φ value by CFC. That is, key-frames and reference frames were firstly selected by the well-known LSD-SLAM algorithm [5], denoted as SLAM selection. Then the key-frame is kept for fair comparison and the reference frame is then selected by using CFC optimization algorithm. As the dataset only includes a discrete set of camera poses and the EM algorithm assumes that the camera arrangement can be changed continuously, the strategy used for SLAM is the winner-takes-all (selecting the best among all these available poses). Lastly, the depth error generated from the two selections are compared. The experiments were carried out with five stereo algorithms mentioned before on eight different scenes.

The results are shown in Table 4.1. Each row shows the depth error for one scene with five different stereo matching algorithms. For each algorithm, the first column shows the depth error using SLAM selection. The middle column shows the error by using the CFC optimized arrangement. The third column shows error reduction in percentage. The error reduction varies for different scenes, but for most scenes, the proposed method reduces the error by 30%, where the highest reduction is 45%.

Figure 4.9 demonstrates the depth and depth error images from optimized camera poses and from SLAM camera pair. Bad estimation areas in red are largely reduced for both the foreground and the background, e.g. the floor part of the desk scene, the left plant in the computer scene and the skin of teddy in the teddy scene. This experiment provides one possible way to embed CFC in a SLAM algorithm for better frame selection, leading to more accurate depth estimation. Also the CFC algorithm can be applied at the level

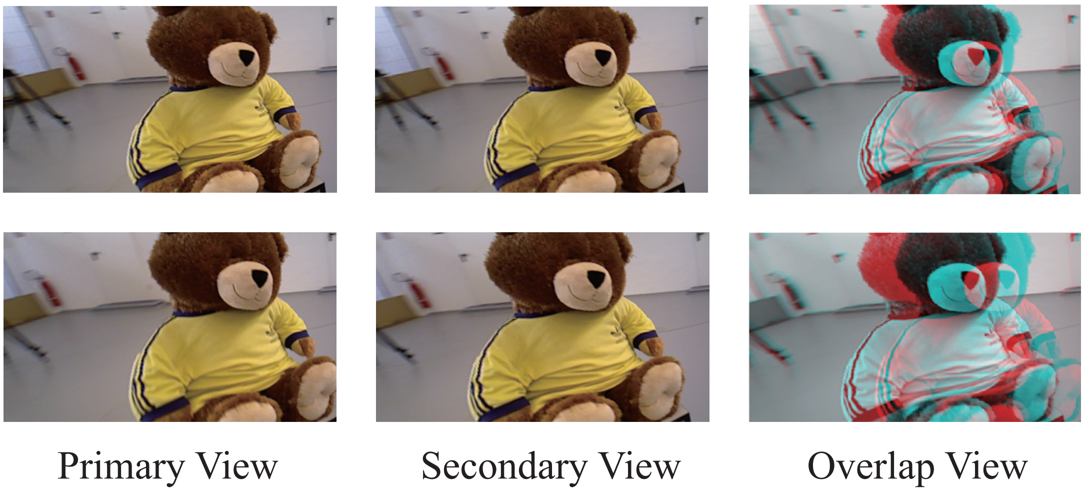


Figure 4.10: Example views from the cameras for the unoptimized camera arrangement (top row) and the optimized arrangement (bottom row).

of specific spatial points or sub-regions to enable multi-reference frame selection, i.e., each point or subregion of the scene selects the most suitable camera by comparing their objective value. With improved depth estimation, the pose estimation in SLAM would be improved as well.

Fig. 4.10 shows the captured images for a real scene by cameras before and after optimization. The first row represents unoptimized camera arrangement and the second row represents the optimized one. From the left to right, the figure shows the rectified images from the two cameras and their overlap, demonstrating the disparity range. It is noted that in the images from the optimized arrangement, the teddy model is larger, which results in more correspondence pixels on the teddy bear to perform more accurate depth estimation.

The Φ values of the optimization function for all possible camera poses are plotted in the figure 4.11 with four stereo algorithms. In each sub-figure the covariance and average depth error are decreasing when the Φ value increases. In the areas of lower score, there are still some camera poses which provide good results, however, the range of the depth error shrinks significantly and consistently as the optimization score increases, thus showing the effectiveness of our optimization strategy and the independence of the optimization algorithm from the stereo matching algorithm used.

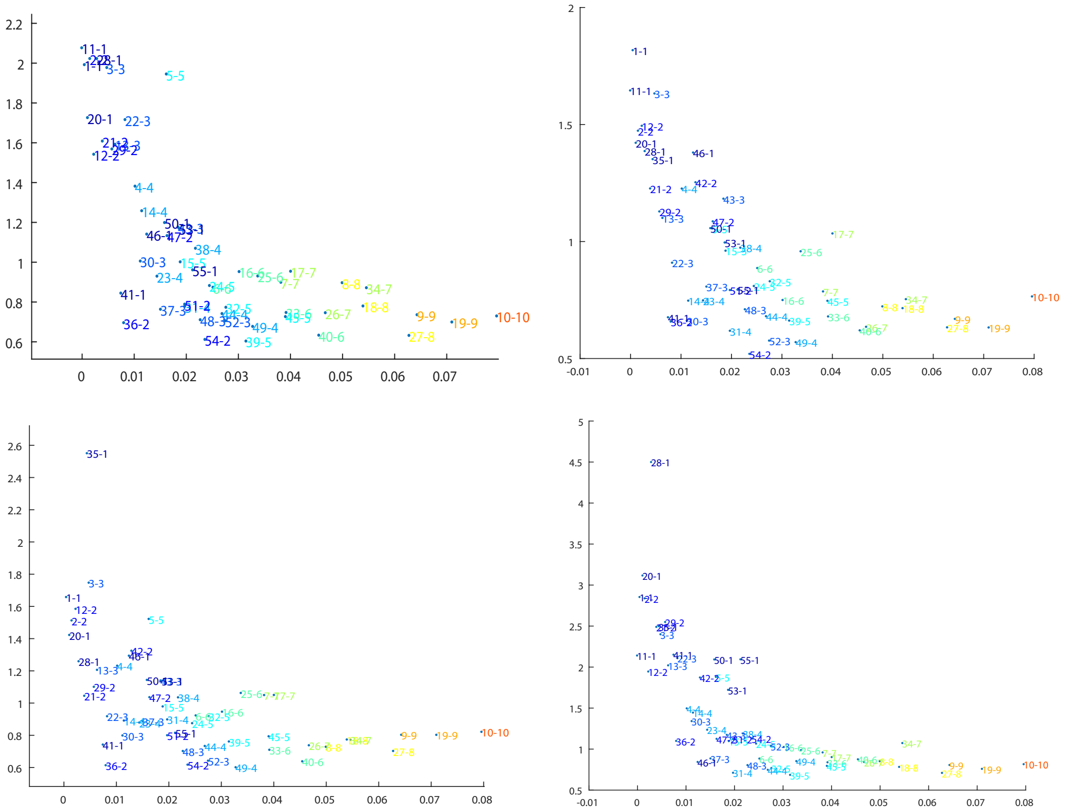


Figure 4.11: Robust evaluation of CF optimization for a scene, x-axis represents the CF optimization score, y-axis shows the depth error against ground-truth value. Each figure shows an optimization by using a stereo matching algorithm

4.2.4 Evaluation on the Middlebury multi-view dataset

The Middlebury benchmark [92] consists of models captured by a semi-sphere arrangement of camera arrays. For each camera's image, we firstly use the neighbour reference view to obtain an initial scene geometry S, followed by the camera arrangement optimization algorithm to find the best secondary camera to generate a depth map. The generated depth map from each camera is transformed into the point cloud and merged by using robust surface reconstruction. Fig. 4.12 shows the resulting mesh model of Dino using the proposed method (the left model) compared with the ground-truth mesh (the right model).

Fig. 4.13 provides an overview of the camera pair selected by the camera arrangement optimization, the x-axis represents the primary camera index while the y-axis represents the secondary camera index. Each point in the figure gives a selected camera pair used for the multi-view stereo. In most cases, the first to the third nearest camera of the primary

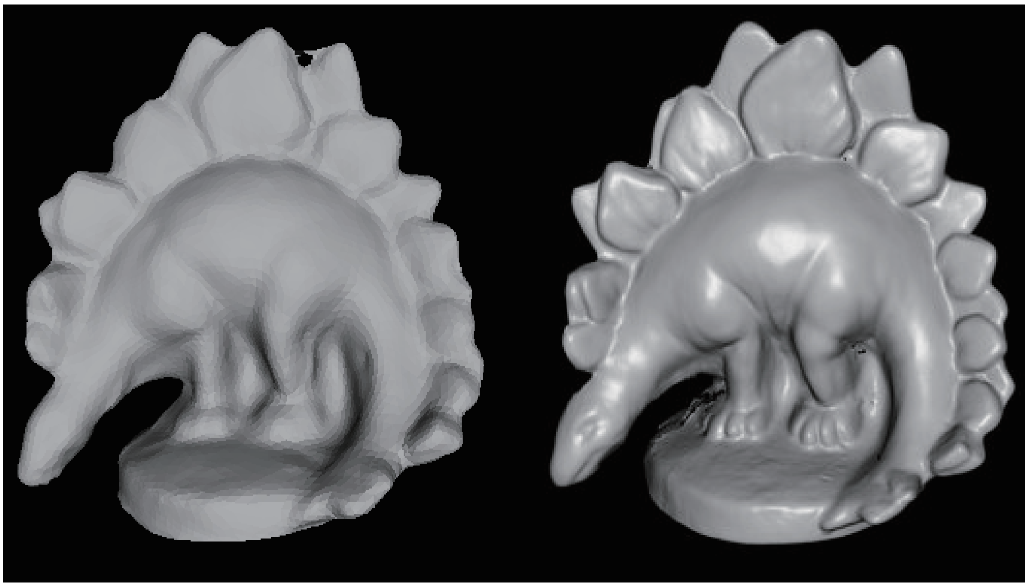


Figure 4.12: Demonstration of the resulting mesh model of Dino in the Middlebury dataset. Left: the Dino model generated by using the proposed optimization method. Right: the ground-truth Dino model

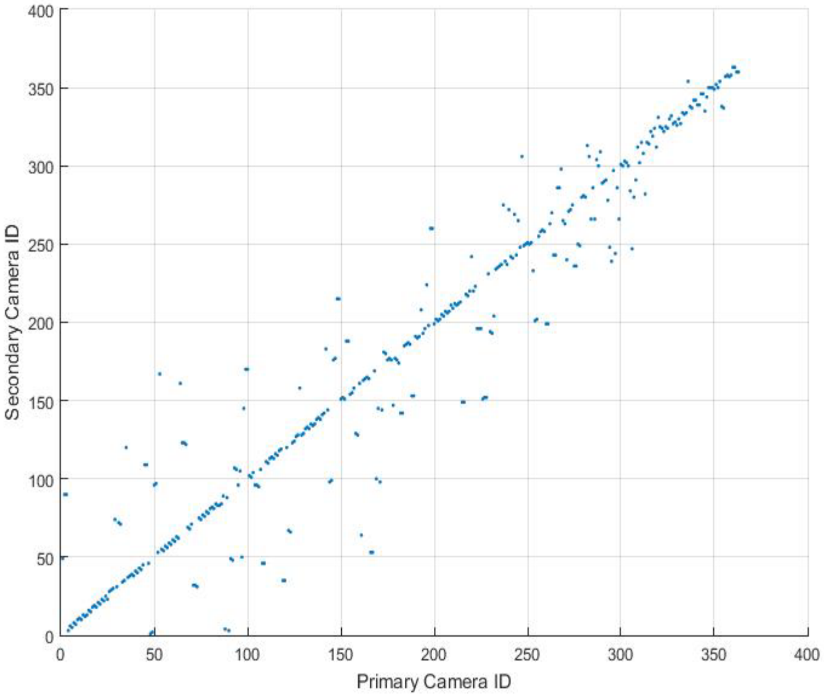


Figure 4.13: The optimized camera pair ID selected based on the CF algorithm

camera is selected as the best secondary (reference) camera. Some ID pairs show large differences because they are at the end and at the beginning of a ring.

Chapter 5

Depth Enhancement by CF Perturbation

In our previous chapter, we stipulated that, for a given scene, there is an optimal camera arrangement for depth estimation. Using an iterative optimization, we progressively moved cameras from an arbitrary initial position towards an optimal point to refine estimation of the surfaces and their depth in the scene. In this chapter, we propose to extend the previous work by *perturbing* the above-mentioned optimal camera arrangement to further improve the accuracy of depth estimation.

Assume that the spatial locations and the orientations of two cameras are denoted by a vector $\Theta = (\theta_1, \theta_2, l, d)$, where θ_1 and θ_2 represent the orientation of cameras, and l and d determine the distance between the cameras and the displacement of the midpoint from the origin respectively. A possible perturbation of this camera arrangement would involve small rotations of cameras, i.e., $(\theta_1 + \Delta\theta_1, \theta_2 + \Delta\theta_2, l, d)$, where $|\Delta\theta_1|, |\Delta\theta_2| \leq \varepsilon_p$. In other words, the locations of cameras are not changed but their orientations are altered by (small) angular shifts $\Delta\theta_1, \Delta\theta_2$ (positive or negative) and the magnitude of change is bounded by ε_p. A finite or infinite set of perturbed camera arrangements is defined as a perturbation set P, which will be used in our algorithm as described later.

Our approach is motivated by the involuntary movements of human eyes during fixation, referred to as saccade [95]. Saccades are rapid and ballistic movements of eyes around the fixation point, which means that humans see the world by a series of saccades interspersed with fixations [96]. A careful investigation of CF shows that camera perturbation alters those CF parameters that determine accuracy of depth (e.g. sampling density, error bound, levelling errors). It seems plausible, therefore, that by combining information from perturbed states, more accurate estimation may be obtained.

This chapter is organized as follows. First, we introduce the concept of camera arrangement perturbation and investigate the impact of the perturbation on the properties of CF. Second, an optimization method is developed to find an optimal set P^* from a perturbation set P for a scene. Then, an inverse sampling method is proposed to select the most relevant elements of P^* given practical constraints on computation and camera control. Last, a new perturbation-enhanced camera optimization (PEC) method is proposed for combining the information contained in P^* into an optimal and single *fused CF* for depth estimation, and empirical verification on simulated and real data that shows substantial improvement of depth accuracy.

5.1 Camera orientation perturbation

The starting point for the perturbation algorithm is a relatively optimum camera arrangement Θ^* obtained by either the CFC algorithm as presented in [97] or other available algorithms. We retain the world coordinates used in the previous chapter to calculate the CF and its perturbations. Using this initial camera arrangement, an estimate of the depth map, including object surfaces and their orientations in the scene, is obtained. The aim is to further improve the depth estimate by perturbing this initially optimized camera arrangement, which involves changing the camera *orientations* within a prescribed range while maintaining the location and the distance between cameras. In other words, l and d remain the same.

5.1.1 Overview

Let the ith perturbed camera arrangement be $\mathbf{a}_i = (\theta_{1i}, \theta_{2i})$, where θ_{1i}, θ_{2i} denote the orientation angles of first and second cameras respectively. The change in orientation with respect to the initial arrangement can be denoted by $\Delta\theta_{1i}, \Delta\theta_{2i}$ respectively. We assume that the change in orientation is limited to a small range ε_p, in other words $|\Delta\theta_{1i}|, |\Delta\theta_{2i}| \leq \varepsilon_p \; \forall i$.

A *perturbation set* of camera arrangement is denoted by P. An element of this set is a particular camera arrangement $\mathbf{a}$. A perturbation set may contain a finite number of elements, for example $P = \{\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_n\}$. Alternatively, the set P may include a continuous set of perturbed camera arrangements within a range of $\pm\varepsilon_p$. In other words, $P = \{\mathbf{a} : |\Delta\theta_1|, |\Delta\theta_2| \leq \varepsilon_p\}$.

Each element $\mathbf{a}_i$ in the perturbation set is a camera arrangement and therefore has its CF, which is denoted by $\Psi(\mathbf{a}_i)$. Note that the CF of each element of P can be generated analytically using the CF equations obtained in Chapter 3 without obtaining images from the cameras or physically rotating the cameras. Let $\mathbf{I}_i = (I_{1i}, I_{2i})$ be the images I_{1i}, I_{2i} associated with camera 1 and 2 from that viewpoint defined by $\mathbf{a}_i$, respectively.

The proposed algorithm to improve depth estimation by utilizing camera perturbation consists of the following steps:

1. Generation of a perturbation set P.

2. Finding the optimal elements of P for each region of the scene. The set of these optimal elements is denoted by P^* where $P^* \subseteq P$.

3. Selection of a subset of suitable elements of P^* based on practical constraints, and acquisition of image pairs associated with these camera positions.

4. For each region, performing correspondence matching on the images associated with its camera arrangement in P^* for the region to generate a disparity map.

5. Generation and merge of 3D point clouds of all regions to form the 3D point cloud of the scene.

5.1.2 Generation of a perturbation set P

The elements of a perturbation set may be generated randomly, for example by randomly assigning values to each θ_{1i}, θ_{2i} according to a probability distribution function. On the other hand, the pair of angles may be generated so that it corresponds to a practical *trajectory* path of cameras as they rotate around their original position as shown in Figure 5.1.

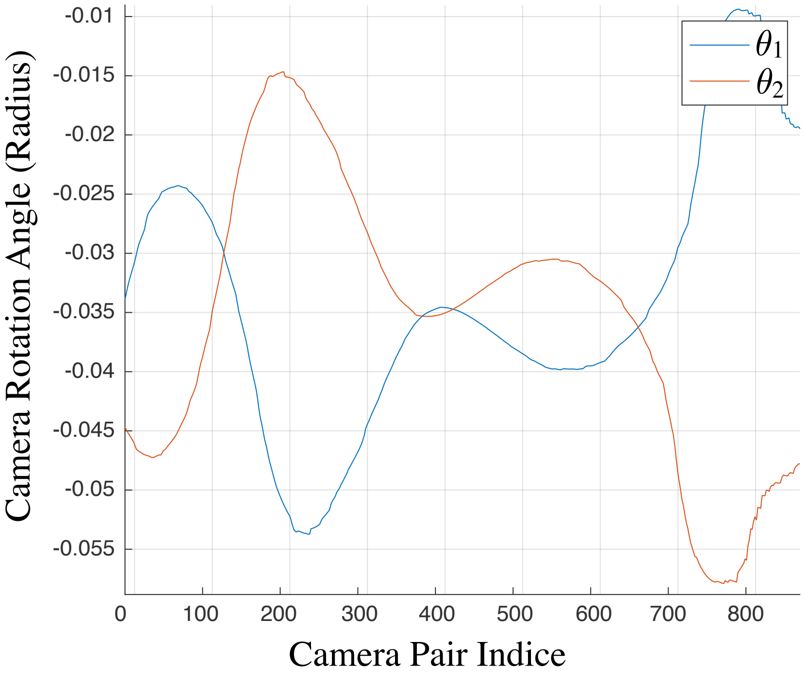


Figure 5.1: Illustration of the changes in camera orientations for a possible trajectory of camera movements. The angular orientation of two cameras θ_{1i}, θ_{2i}. The X-axis represents a sequence of camera pairs whose indices start at the left region of the scene and end at the right area, thus forming a path.

It is important to note that the generation of P does not require any actual movement of cameras or capturing of images from the scene and is purely a theoretical step. Each element of this set is simply a *possible* camera orientation within the prescribed range. Whether this element is *useful* in improving the depth estimate will be determined in the next step. The pair of images for a particular camera arrangement will only be required if the selection criteria in step 3 are satisfied. For this reason, P may be considered as a continuous set, having all possible camera arrangements generated by the trajectory.

The composition of a set P is affected by two issues. First, the upper bound of camera orientation changes ε has to be decided. A compromise has to be made between very small values of ε_p (insufficient variation in CF) or too large (adding noise and affecting the field of view). The approach adopted in this thesis is to assess this range based on the CF variations required to match the object surfaces. This approach will be presented as part of the example of Figure 5.2 in the next section.

The second issue is to choose the trajectory of camera orientation changes. For the experimental results of this thesis, we have adopted a random and arbitrary trajectory. Further research on improving this step is left for the future.

5.1.3 Finding the optimal subset P^* from P for the scene

The scene is divided into a number of *regions*. Let the j^{th} region be Ω_j. We already have an initial estimate of the object surfaces in this region obtained by an algorithm such as the CFC algorithm [97]. This step aims to find the most suitable element of P for estimating the depth of objects in each region. The optimization uses the set of CF associated with the elements of P, i.e., $\Psi(\mathbf{a}_i)$ and chooses the best $\mathbf{a}_i$ for each region.

To clarify this step, consider a simple example shown in Figure 5.2. The optimum camera arrangement obtained by a CFC algorithm is shown as black arrows at the bottom of Figure 5.2(a), and its associated CF is shown as a set of curves covering the scene in blue. The estimated object surface is visible as a wavy curve. In Figure 5.2(b) the orientations of cameras are perturbed within a range of ± 10 degrees with the resolution of one degree as shown by the black arrows. Assume that the two cameras are perturbated in the same way, this results in a discrete perturbation set P with 21 elements, numbered from 1 to 21. The scene is partitioned into 9 regions along the x-axis marked on the bottom of the figure 5.2(b). In the figure, only 7 of 21 perturbations are selected. The 2-surfaces for each region associated with ith camera perturbation are shown with a set of black lines. It can be seen that for different parts of the object surface, the perturbation provides a suitable CF that aligns well with the object. The most suitable CF for each region is identified by its index number in black text. For example, in Figure 5.2(b), camera arrangement 20 is better aligned with the left-hand side of the object surface, while numbers 2 and 7 are more suitable for the middle part.

Figure 5.2 (c) shows the range of CF directions (the maximum and minimum angle is shown in the red and blue line corresponding to the perturbations) for each point along the object surface. Depending on the range of variations needed for this object surface, a suitable value for the max range of perturbation ε_p can be selected.

Figure 5.2 (d) shows a bar chart, each bar representing the percentage improvement of depth (compared to CFC) for each part of Figure 5.2 (b). It is clear that camera perturbation provides a better camera orientation for each region of the scene and hence improves the accuracy of depth estimation.

The objective function for finding the optimum matching between regions of the scene and elements of P is as follows:

$$z(\Omega_j) = \arg\max_{\mathbf{a}\in P} \int_{\mathbf{r}\in\Omega_j} |\Psi(\mathbf{a},\mathbf{r}) \cdot \mathbf{n}_{\mathbf{r}}| \, d\mathbf{r} \tag{5.1}$$

In words, the objective is to find an element of P that maximizes the aggregate dot product of CF and the object normal in this region. As described in [97], given that CF surfaces are quadratic surfaces, this algorithm is fast. Importantly, this step does not require any image processing, as it relies on matching the CF of the camera arrangement to the scene.

In the limit, the size of the region Ω_j may be reduced to a single point $\mathbf{r}$ and the object surface can be represented by a single normal vector at that location. The objective function to find the optimum element of P for this point is simplified to:

$$z(\mathbf{r}) = \arg\max_{\mathbf{a}\in P} |\Psi(\mathbf{a},\mathbf{r})| \cdot \mathbf{n}_r \tag{5.2}$$

Therefore, after completing this step, for each region Ω_j or point $\mathbf{r}$ in the scene, we obtain a camera arrangement $\mathbf{a}$ associated with the optimum CF for that region or point, i.e., $z(\mathbf{r})$ represents the mapping $\mathbf{r} \rightarrow \mathbf{a} \in P$. The collection of these arrangements can be

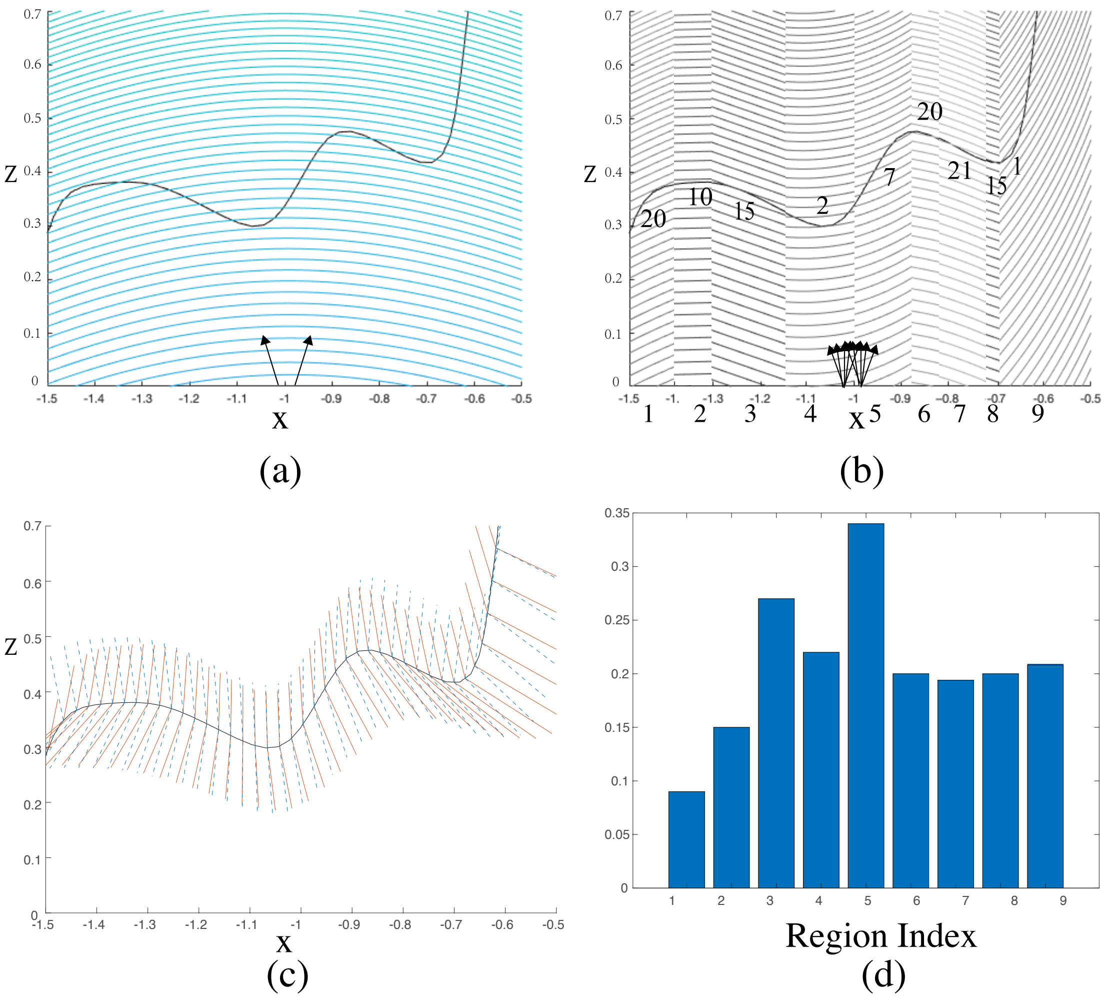


Figure 5.2: (a) Initial camera arrangement obtained by the CFC algorithm and its CF. (b) Optimum camera arrangements for each region of the scene. (c) The range of variations of the CF vectors for each point on the object surface. (d) Percentage improvement on accuracy by using the optimum camera arrangement for each region compared to the global optimization achieved by the CFC algorithm [97] without any perturbation.

considered to be the optimal set of the perturbation set for this scene and the combined CF is referred to as the *fused CF*. The fused CF is denoted as $\Psi(\mathbf{r}, z(\mathbf{r}))$ or $\Psi(\mathbf{r}, z(\Omega_j)))$ in a region-based approach. Figure 5.2(b) is an example of a region based fused CF. Figure 5.3 shows an example when the size of regions approaches a single point. For a single camera pair, the shape of the CF's 2-surfaces or iso-disparity surfaces is quadratic. However, it can be observed that for a reasonable number of perturbations within a range, the fused CF has 2-surfaces that can align well with arbitrarily object surfaces. This is a significant result and the underlying reason for the significant improvement that is obtained in depth estimation.

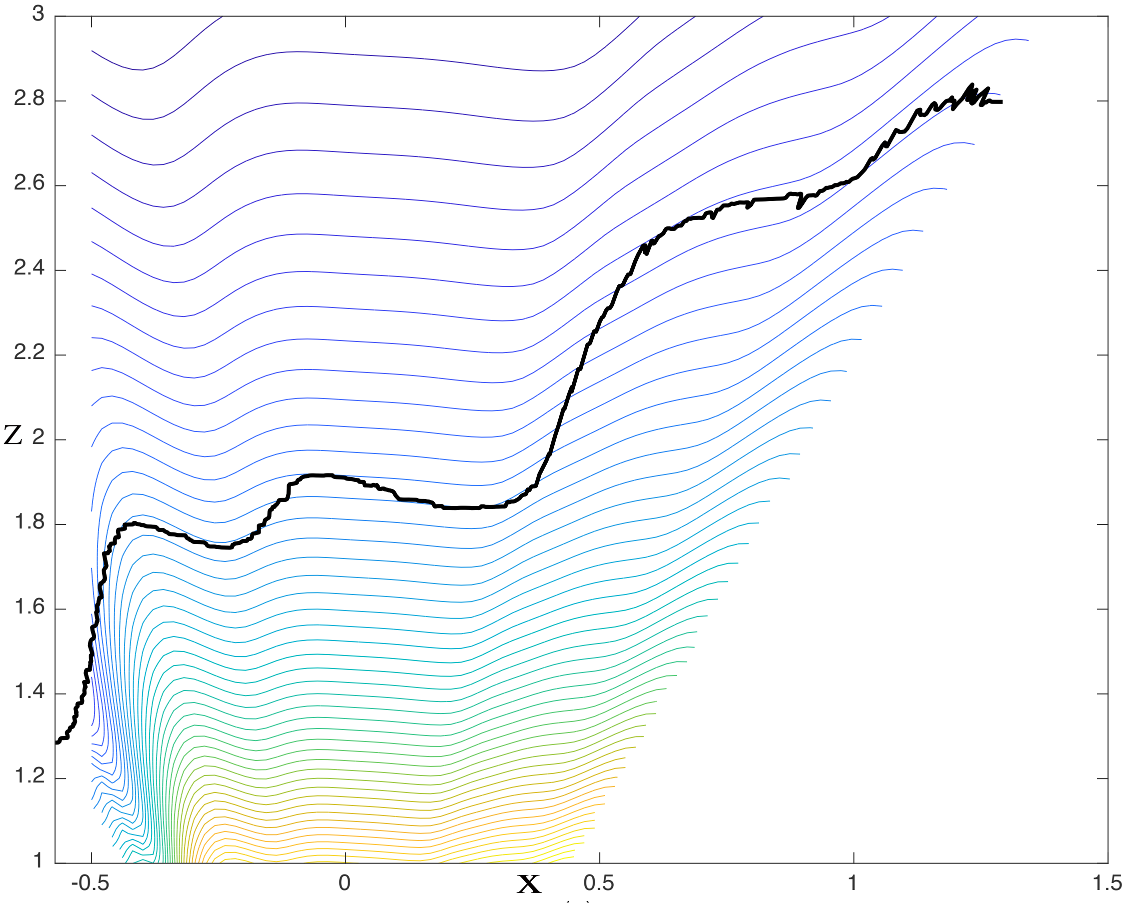


Figure 5.3: The fused CF associated can match well the scene surface with a perturbation set of a limited size.

Depending on the nature of object surfaces in the scene, some elements of P may be more useful than others. For example, in Figure 5.2(b), camera arrangements 20 and 15 are used twice but some of the other arrangements (such as 5 or 19) do not appear to be suitable for any part of the scene. Therefore, a subset of camera arrangements P is often required or can be afforded practically for the scene. This set of optimal camera arrangements is denoted by P^*.

5.1.4 Selection of P^* and acquisition of associated images

One strategy of selecting elements from P to form P^* is to measure the usefulness of each camera arrangement in P, for instance, a histogram (or distribution function) of how many times each arrangement has been selected as optimal for the regions in the scene as in the previous step. This is shown in the example of Figure 5.4(a). The more times the arrangement is chosen, the more usefulness it is.

Apart from the usefulness measure represented by this distribution, there is also the practical matter of being able to rotate the camera and capture the actual images of the scene from that particular viewpoint. Limits on the accuracy of camera control and the frame rate of the camera will inevitably introduce some error in the ability to obtain these images. It may also be impractical to obtain a large number of image pairs and some merging of nearby orientations may be necessary. Nevertheless, it is possible to choose a finite number of camera arrangements that best fit the histogram and obtain images (or perhaps selecting from video frames as the cameras pan). Since this step is critically dependent on practical features of camera control and frame rate, the selection process adopted in this thesis is based on the limits on computation, i.e. number of affordable perturbed arrangements and, hence, the histogram of camera arrangements being selected alone.

Let us assume that we intend to select the best N arrangements in P. The usefulness can be characterized by the histogram in Figure 5.4 (a). The vertical axis is the proportion of times that the camera arrangement (identified by the index on the horizontal axis) was selected during the optimal matching step. The corresponding cumulative distribution function is shown as the blue line in Figure 5.4 (b). It is possible to perform an inverse sampling of the cumulative distribution to choose the best N samples. The sampling points are shown as vertical lines with equal interval partitioning of the vertical axis. Figure 5.5 compares the fused CF shape before and after the above inverse sampling. It is noted that the sampled version, i.e. P^*, is less smooth compared to the original P, but the overall shape of the object surface is tracked relatively well.

The choice of the number of samples N may depend on many constraints as well as the complexity of the scene and further research is left for the future. However, our experiments have shown that in many cases a modest value of N is sufficient. Figure 5.6 demonstrates the reduction in depth estimation error versus N. The depth estimation error is reduced by as much as 40% compared to a single camera arrangement as N approaches

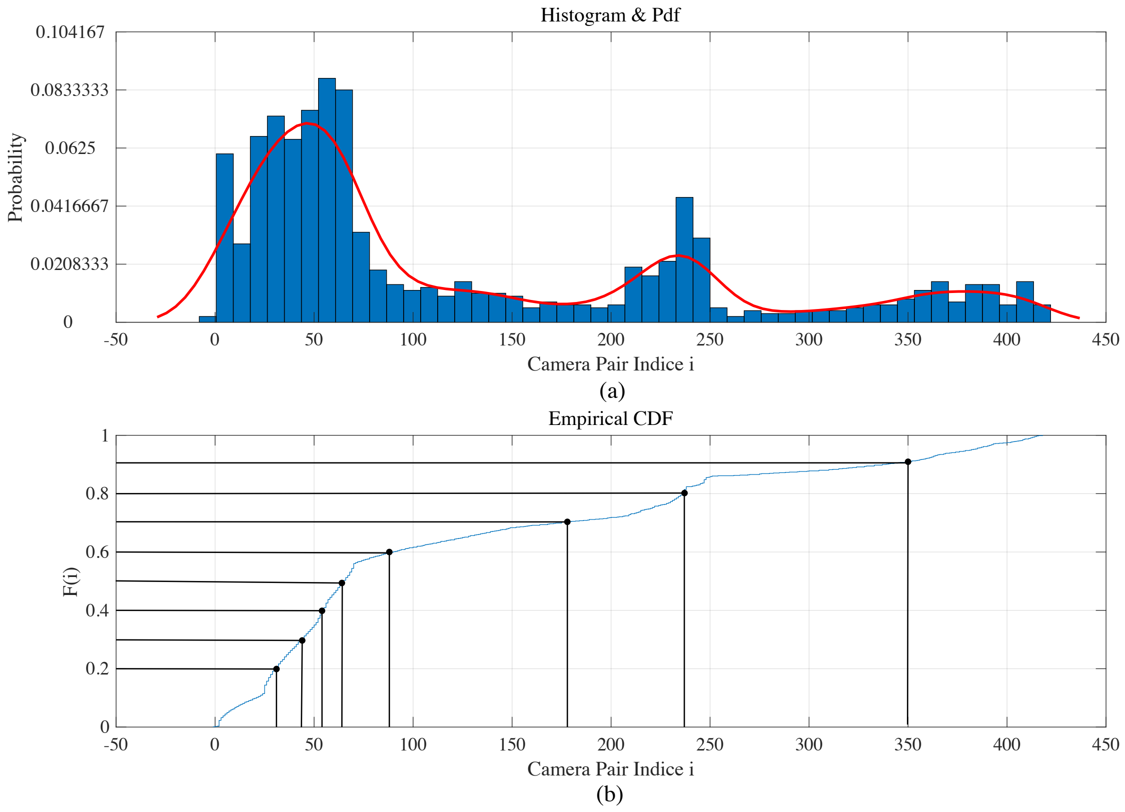


Figure 5.4: (a) The histogram and approximate probability density function (i.e. curve) of usefulness. The x-axis represents the camera indices i. (b) Cumulative distribution function (CDF) $F(i)$ and illustration of equal interval sampling of the CDF.

20 and seems to converge to a stable level after that.

5.1.5 Region based correspondence matching

After selecting the N best perturbed camera orientations to form P^* in the previous step, N pairs of images that correspond to the N arrangements are physically acquired. One possible approach is to pass this set of images to a multi-camera or multi-view depth estimation algorithm. However, it is likely that the optimal association between the camera arrangement, hence, the image pairs, and the regions of the scene is not utilized by such a generic multi-camera algorithm. In this thesis, each pair of images is utilized to obtain the depth of objects in the regions for which the arrangement is optimal.

Specifically, region Ω_j is associated with the camera arrangement $z(\Omega_j) \in P^*$ and image pair (I_1^j, I_2^j) is captured using the arrangement. Ω_j maps to the areas $R_{1,2}(\Omega_j)$ in (I_1^j, I_2^j), respectively. A conventional correspondence matching algorithm could be used

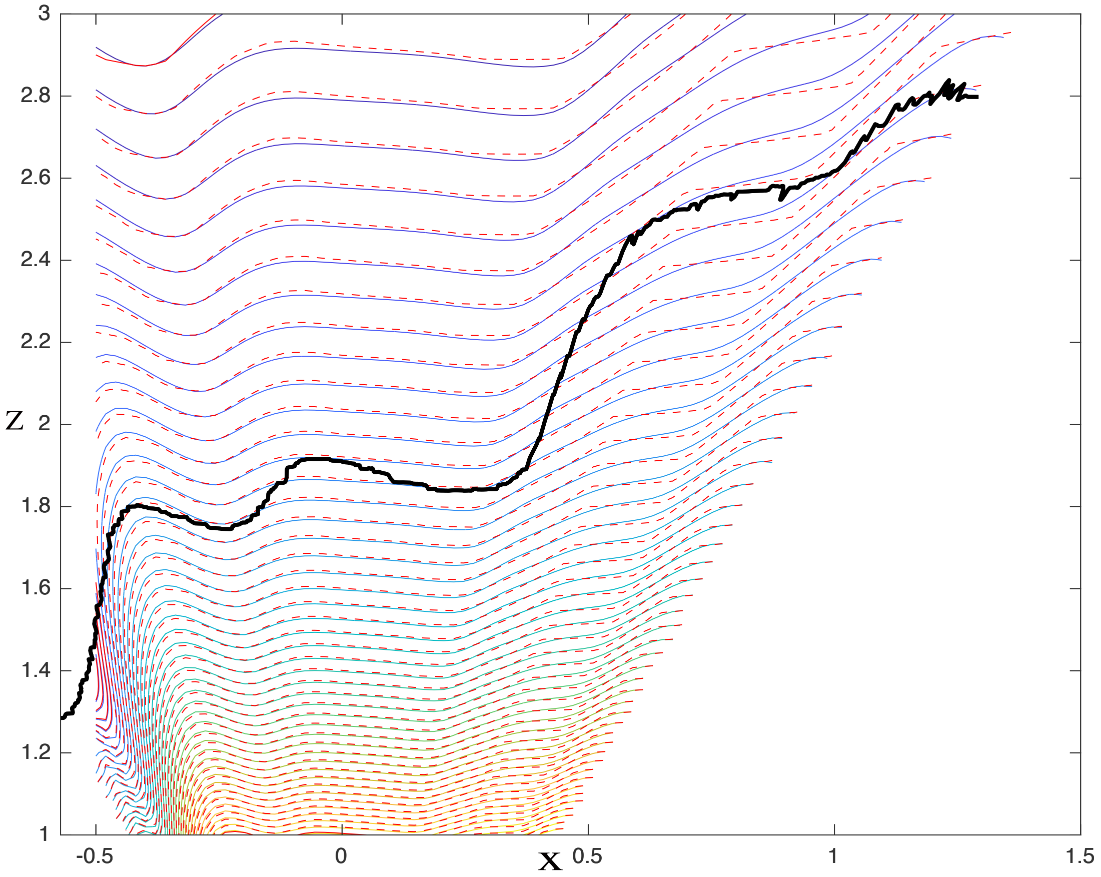


Figure 5.5: Fused CF generated from P (blue curves) and the selected best $N = 8$ elements P^* (red curves) for the surface (black curve).

to obtain the disparity map k^j for Ω_j. In this thesis, we use the same stereo algorithms employed in CFC algorithm presented in [97] for a fair comparison. Any stereo algorithm or post stereo matching refinement algorithm can be used.

5.1.6 Depth estimation and construction of a 3D point cloud

A depth map for each region Ω_j can be obtained through triangulation from its disparity map k^j and the corresponding camera arrangement. A 3D point cloud for Ω_j with respect to the camera arrangement is generated. This 3D point cloud is then transformed to a global coordinate as shown in Fig. 5.2. 3D point clouds for all regions are merged together in the global coordinate system to form the 3D point cloud of the scene.

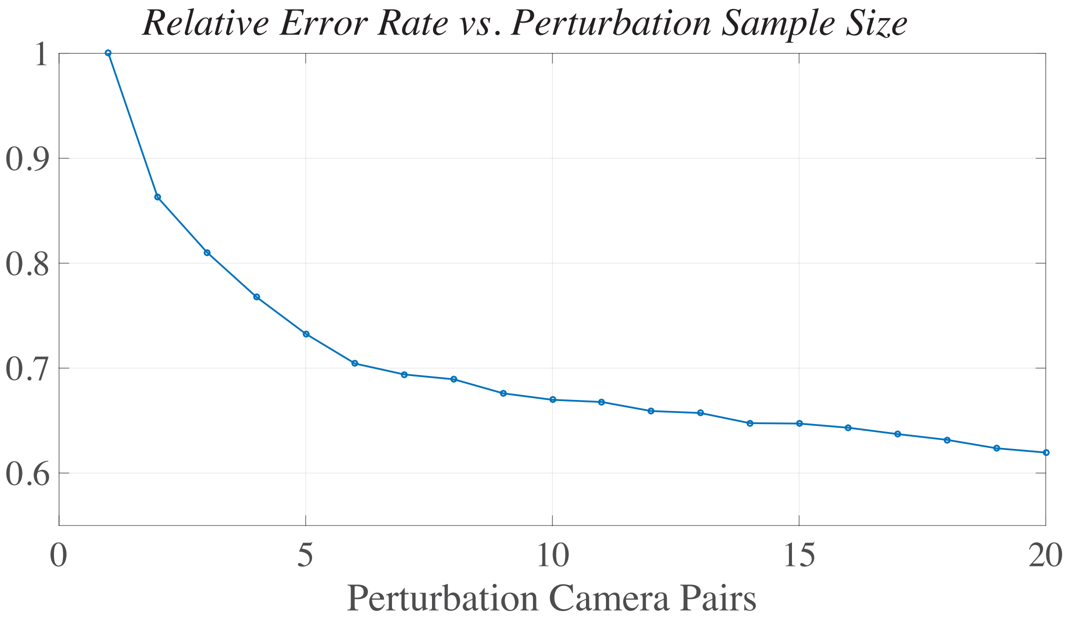


Figure 5.6: Depth estimation error with different number N of perturbations in P^*. The relative error is set to 100% for $N = 1$ and converges to 60% when $N > 20$.

5.2 Experiments

This section presents the implementation and experimental validation of the proposed perturbation-enhanced camera optimization (PEC) algorithm. A depth accuracy comparison with the previously proposed CF-based camera alignment optimization (CFC) is performed and tested on simulated data as well as on real scenarios. Readers are referred to Section 4.2 for comparison between the CFC algorithm and other typical depth estimation algorithms. The PEC algorithm with N perturbed arrangements is denoted as PEC-N.

5.2.1 Implementation

The proposed PEC algorithm can be implemented by first dividing the scene into regions according to the initial CF and estimated depth map, then selecting the optimal arrangement for every region to form P^*. Such implementation would be subject to how well the obtained regions could be potentially accommodated by a possible perturbation. In this thesis, we have implemented the algorithm in a bottom-up approach by integrating region segmentation of scenes with the selection of the N optimal camera arrangements P^*. Details are as follows.

1. Generate possible perturbations P from the initial camera parameter Θ, perturbation range ε_p and its interval $\Delta\varepsilon_p$. This would result in $(\frac{2\varepsilon_p}{\Delta\varepsilon_p}+1)^2$ possible perturbations considering the orientation of left and right cameras can be perturbed independently within $[-\varepsilon_p, \varepsilon_p]$.

2. For each pixel $i, i = 1, 2, \cdots, M$ of the scene, select the best camera arrangement $\mathbf{a_i}$ from P, where M is the number of pixels.

3. Generate the histogram of selected camera arrangements by setting the bin-size to $\Delta\theta \times \Delta\theta$, where $\Delta\theta$ is the bin-size of the perturbation angles for both left and right cameras. Select N best perturbations to form P^* according to the cumulative distribution as detailed in Section 5.1.3

4. Take N pairs of images of the scene and perform correspondence matching to generate the 3D point cloud of the scene. Note that for the sake of validating the effectiveness of perturbation, the correspondence matching in the current implementation is image based rather than region-based as described in Sections 5.1.5,

Specifically, in all the experiments, $\varepsilon_p = 15°$, $\Delta\varepsilon_p = 0.01°$, $\Delta\theta = 0.06°$ and $N = 10$.

The accuracy of depth estimation is usually measured by the mean of Hausdorff distances between the ground truth and estimated point clouds [94]. In the following evaluation, the depth error map shows the accuracy from the left camera view in order to align with the ground truth. This is consistent with the CFC evaluation [97]. The left view depth map of the PEC algorithm is obtained by perspective transformation from the generated point cloud.

5.2.2 Evaluation on the Simulated Data

This section provides a comprehensive evaluation of the PEC method for the various synthetic scenes.

Simulation Dataset and Parameters

Synthetic scenes were created using widely known 3D datasets such as Stanford Scanning Models, Sketchfab and Blend Swap. These synthetic datasets provide a variety of geometries of varying complexity and ground truth. Nine scenes are used to test the proposed PEC. As shown in Figure 5.7, the first four are individual objects with complex surfaces, the next four scenes have complex occlusion and multiple objects.

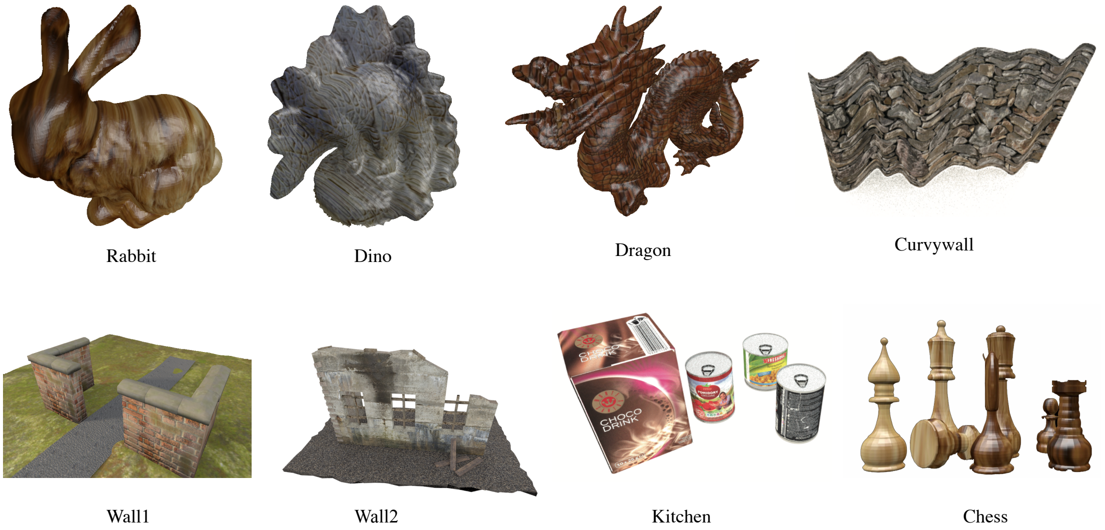


Figure 5.7: Illustration of the synthetic scenes used in the experiments.

The settings of the simulated scenes and experiments follow the same as those in [97]. The spatial range was set to $[-2m, 2m] \times [2m, 4m]$ representing left, right, front and back, respectively. The cameras were assumed to be on a rail at $Z_{rail} = -4m$ with distance to center $l_0 = 0.04m$ and centroid location $d_0 = 0$. The intrinsic parameters of the camera were set as follows:$f = 600$, $res_x = 720$, $res_y = 320$ pixels. The CFC parameters are identical to the configuration in [97].

Results and comparision

Figure 5.8 gives a qualitative illustration of the PEC-10 method compared with CFC algorithm described in section 4.1 with different matching algorithms. It is important to note that the PEC-10 algorithm has little improvement on the occluded regions compared with the CFC depth map. This is mainly because of small perturbation in camera orien-

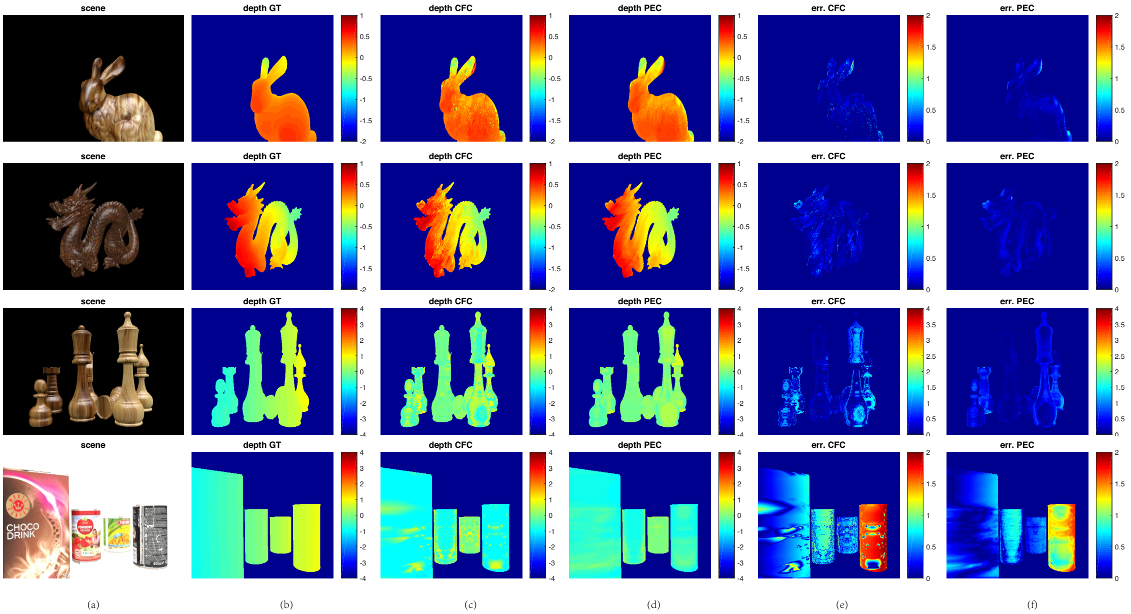


Figure 5.8: Illustration of the results of PEC and CFC. (a) scene view from the camera. (b) ground truth depth map (c) depth estimated by CFC. (d) depth estimated by PEC. (e) estimation error of CFC. (f) estimation error of PEC.

tation and mapping to the left view for comparison. Four different scenes are shown in Figure 5.8 (a). The depth estimated by the CFC and PEC-10 algorithms is shown in Figure 5.8 (c) and Figure 5.8 (d), respectively, followed with absolute error map (e,f) against ground truth map of Figure 5.8 (b). Zero error is shown in blue and the error increases when the color turns red. As can be seen from the overall depth error, the PEC-10 algorithm handles all parts of the object surface well. From the dragon and bunny objects, it can be seen that the depth errors for large continuous area can be reduced to a large extent. For the chess and kitchen objects, PEC-10 is quite advantageous in multiple curvy thin objects like cylinders. Overall, the PEC-10 algorithm further improves the smoothness of

Scene	Spstereo [31]			Optical Flow [28]			Openrwr [30]			Cost Filtering [27]		
	PEC	CFC	Imprv.	PEC	CFC	Imprv.	PEC	CFC	Imprv.	PEC	CFC	Imprv.
bunny	0.0395	0.0536	26.31%	0.0087	0.0104	16.35%	0.0311	0.0547	43.14%	0.0397	0.0550	27.82%
dragon	0.2161	0.2276	5.05%	0.0317	0.0401	20.95%	0.1529	0.2683	43.01%	0.1533	0.2780	44.86%
dino	0.0917	0.1458	37.11%	0.0139	0.0182	23.63%	0.0865	0.1076	19.61%	0.0869	0.1535	43.39%
curvysurface	0.0179	0.0305	41.31%	0.0765	0.0841	9.04%	0.0285	0.0514	44.55%	0.0841	0.1489	43.52%
wall1	0.0175	0.0307	42.92%	0.0126	0.0165	23.64%	0.0240	0.0413	41.89%	0.0210	0.0425	50.59%
kitchen	0.5170	0.5577	7.3%	0.3424	0.4727	27.57%	0.5218	1.0211	48.9%	0.4893	0.7372	33.63%
chess	0.0465	0.0616	24.51%	0.0094	0.0101	6.93%	0.0384	0.0667	42.43%	0.0655	0.0765	14.38%
wall2	0.0792	0.0975	18.77%	0.0581	0.0884	34.28%	0.1914	0.2450	21.88%	0.0984	0.1110	11.35%
average			**25.41%**			**20.95%**			**38.17%**			**33.69%**

Table 5.1: Depth estimation accuracy with four stereo matching algorithms on eight scenes

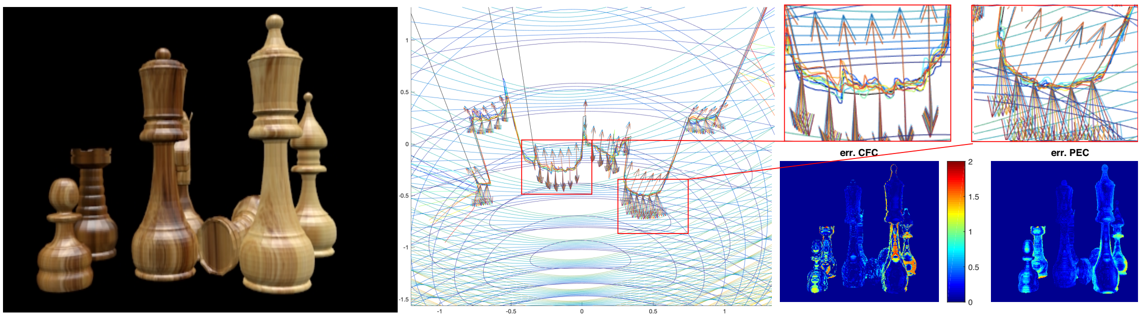


Figure 5.9: Illustration of the chess scene. **Left**: chess scene. **Middle**: the top view of the CF 2-surface (each color curve) and CF direction (color arrow) over the objects (gray curve) **Right**: illustration of zoomed-in surface areas and the error maps of the CFC and PEC-10 algorithms.

the depth map and significantly reduces the noise as expected.

Figure 5.9 gives an example of the PEC-10 algorithm on the chess scene. After reaching the optimized camera location through the CFC [97] algorithm, the camera is perturbed with the PEC-10 perturbation strategy. The generated 2-surfaces are represented by thin curves, and the estimated depths are represented by lines of different colors. The directions of the 2-surfaces for each perturbation are given by the arrows in the middle of Figure 5.9. In the upper right of Figure 5.9, the plots are a zoomed-in look of the 2-surfaces around the object surface, while the bottom right shows the error maps of depth estimated by CFC and PEC-10, respectively. The proposed algorithm performs well in estimating the curve shape compared to the CFC algorithm.

Table 5.1 and Figure 5.10 compare the accuracy of depth maps obtained by PEC-10 and CFC with different optical flow/stereoscopic matching algorithms [27, 29–31], and it can be seen that for all types of optical flow and stereo algorithms, the perturbation strategy helps to improve the accuracy of the depth estimation. The average improvement in accuracy is about 30%. The maximum improvement for each scene is shown in bold. For scenes such as ‘wall1’, the maximum improvement can even be up to 50%, which shows the effectiveness of the perturbation strategy.

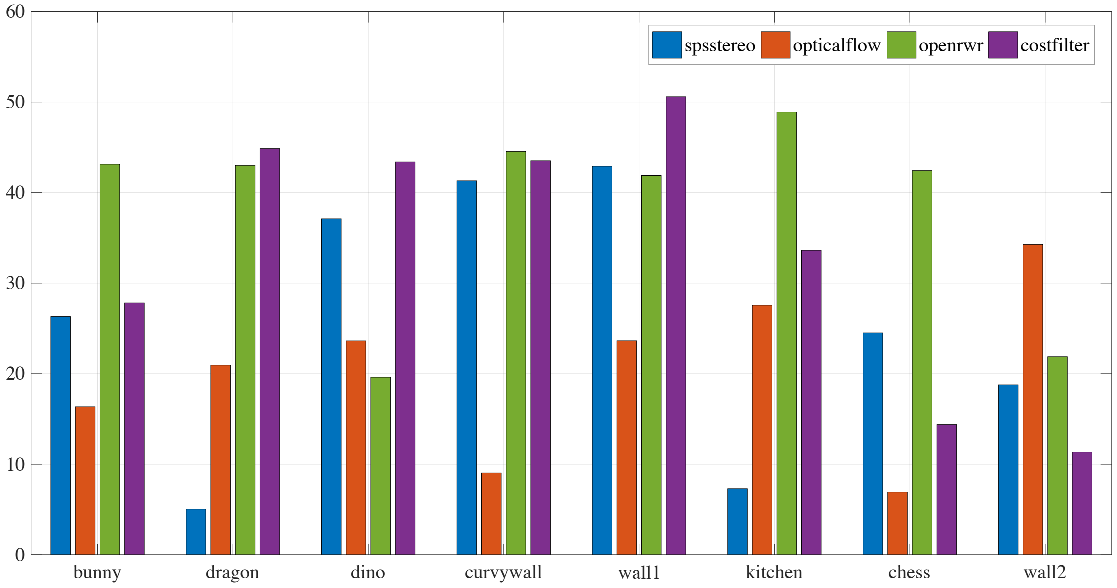


Figure 5.10: Improvement of depth accuracy in percentages comparing the estimated depth maps obtained by the PEC-10 to the CFC algorithms on the synthetic 3D scenes using the same matching algorithm. From left to right: bunny, dragon, dino, house, curvy surface, wall, kitchen, chess, wall2. The vertical axis is $(1-(PECerror/CFCerror))\%$

Impact of N

The choice of N will impact the performance. Fig. 5.11 shows the error reduction achieved by PEC at different N in comparison to CFC [97]. The error reduction is on ‘bunny’ using the matching algorithm [28] was used. As expected, when increasing N the error is reduced and when N reaches a certain value, i.e. 10, the error reduction becomes saturated.

In general, the more complex the scene is, the larger the N is required and more image pairs would be physically acquired. Empirical evidence shows that the value of N in the order to from a few up to twenty is sufficient for most scenes. This is also demonstrated by the results shown in Table. 5.1 where $N = 10$.

5.2.3 Evaluation on Simulated Data from Real Scene Models

Most real acquisition datasets, such as visual SLAM (Simultaneous Localization and Mapping) datasets and MVS (Multi-View Stereo) datasets, have a specific camera arrangement structure, where the spacing between cameras and the distance from the cam-

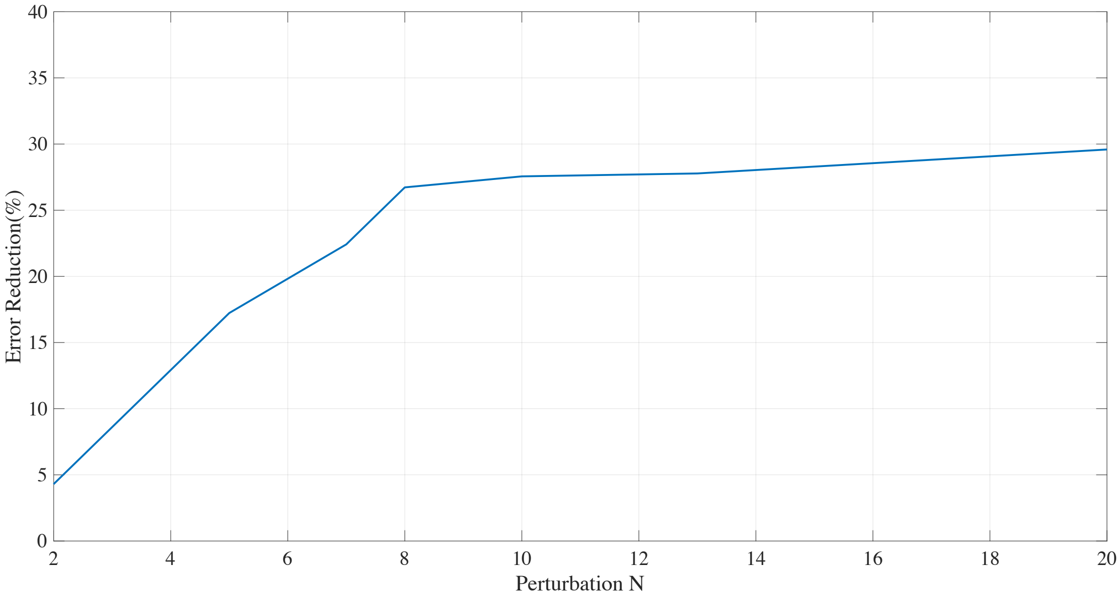


Figure 5.11: Performance of PEC at different N and number of regions for object 'bunny'.

era center point to the scene object are almost equal. This sparse camera arrangement is not conducive to optimization and evaluation, especially for the proposed perturbation strategy, which requires a dense arrangement over a small angular range. To overcome this problem, we propose an evaluation strategy for reconstructing data using real scenes. The MVS and SLAM datasets of real scenes are first reconstructed by COLMAP [98, 99] or other 3D reconstruction algorithms, and the resulting meshes are considered as ground truth data. These models are used for simulation. The differences between these experiments and the previous one lie in the sources of the 3D models.

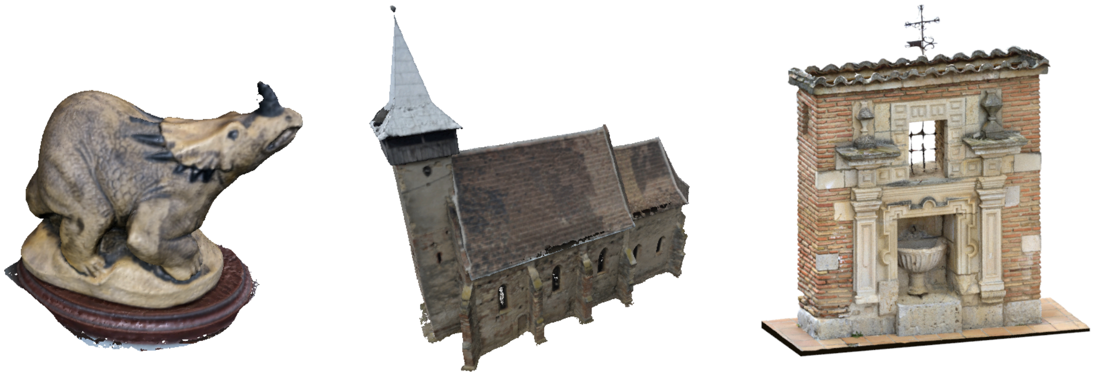


Figure 5.12: Illustration of the real scene models used in the experiments.

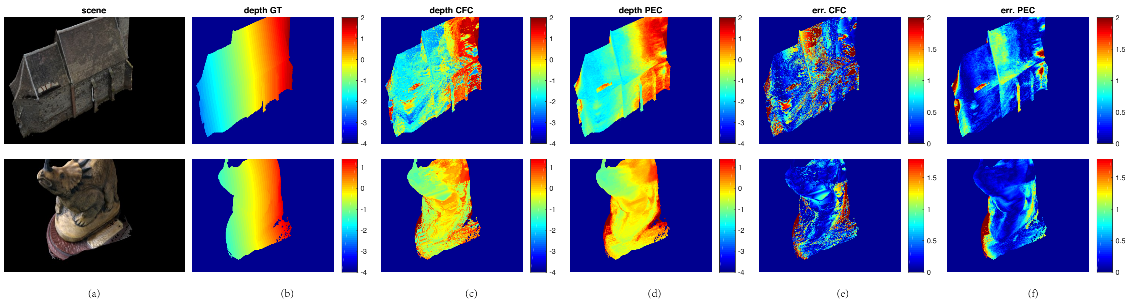


Figure 5.13: Illustration of the evaluation results of PEC and CFC on simulation data using real scenes. (a) scene view from the camera. (b) ground truth depth data (c) depth estimated by CFC. (d) depth estimated by PEC. (e) estimation error of CFC. (f) estimation error of PEC.

Three scenes are reconstructed from real visual and LIDAR data provided in Sketchfab[a], including buildings with complex occlusions (Fig. 5.12). Similar to the synthetic object benchmark, the camera is considered to move freely in the scene space, which is feasible for real scenes, as drones are now widely used for image acquisition of real scenes. Specifically, The spatial range was set to $[-2m, 2m] \times [2m, 4m]$ representing left, right, front and back, respectively. The cameras were assumed to be on a rail at $Z_{rail} = -3m$ with distance to center $l_0 = 0.04m$ and centroid location $d_0 = 0$. The intrinsic parameters of the camera were set as follows:$f = 1050$, $res_x = 1280$, $res_y = 720$ pixels. The experiments provide the qualitative visual evaluation of the scene, as well as a quantitative analysis of the dataset in terms of accuracy and completeness.

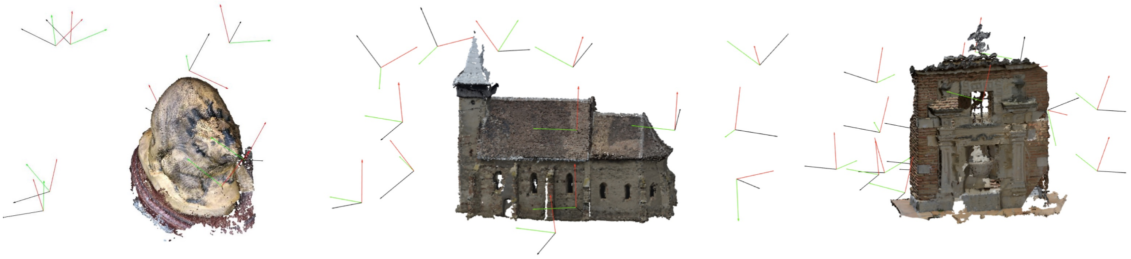


Figure 5.14: The reconstructed dense point cloud from the real scene data. From left to right: animal, church and arch.

Figure 5.13 illustrates the depth estimation error of the PEC-10 algorithm and CFC, and it can be seen that the proposed algorithm performs well in a real scene consisting of

[a] https://sketchfab.com/

textures and high geometric complexity. For the church scene, PEC alleviates the error in the roof area and reduces the upper error bound (red part). For the animal scene, the low error part (blue area) is enlarged and the error is even smaller. Figure 5.14 also illustrates the 3D model reconstructed from the 3D point cloud obtained by the PEC method. The overall model preserves the shape well.

Table 5.2 shows the accuracy of the CFC and PEC-10 algorithms, and the improvement in accuracy achieved by the PEC algorithm is significant. In the arch scene, the maximum improvement of 53% using the Openrwr algorithm is achieved.

5.2.4 Evaluation on Real Scenes

To simulate the camera perturbation in practice, a DSLR camera with gamble (DJI Robin-S) was used to create a real scene image set. The camera was mounted on gamble facing the objects, then the camera was slightly rotated using a mobile application. After that the gamble was translated by a base line distance on a stable track and performed perturbation for another view. The whole process was repeated several times.

Afterwards, all the images were processed using the COLMAP globally optimized 3D reconstruction algorithm [98, 99] to construct a ground truth 3D model (with high computational resources and time), and then we select only two key-views with and without perturbations to demonstrate the advantages of the algorithm.

In particular, the spatial range was set to $[-1m, 1m] \times [0m, 1m]$, $l_0 = 0.02m$ and centroid location $d_0 = 0$. The intrinsic parameters of the camera were set as follows:$f = 937$, $res_x = 1920$, $res_y = 1080$ pixels.

Figure 5.15 shows the four scenes to examine the perturbation technique. Figure 5.16 shows the depth estimation by the CFC and PEC-10, respectively. The image from the

Scene	Spstereo [31]			Openrwr [30]			Cost Filtering [27]		
	PEC	CFC	Imprv.	PEC	CFC	Imprv.	PEC	CFC	Imprv.
animal	0.1352	0.2024	26.67%	0.1143	0.2411	21.88%	0.1628	0.2549	36.15%
church	0.0274	0.0319	13.84%	0.0222	0.0475	53.33%	0.0347	0.0556	37.67%
arch	0.1101	0.1142	3.61%	0.1042	0.1850	43.69%	0.1264	0.1687	25.06%

Table 5.2: Depth estimation accuracy for real world scenes

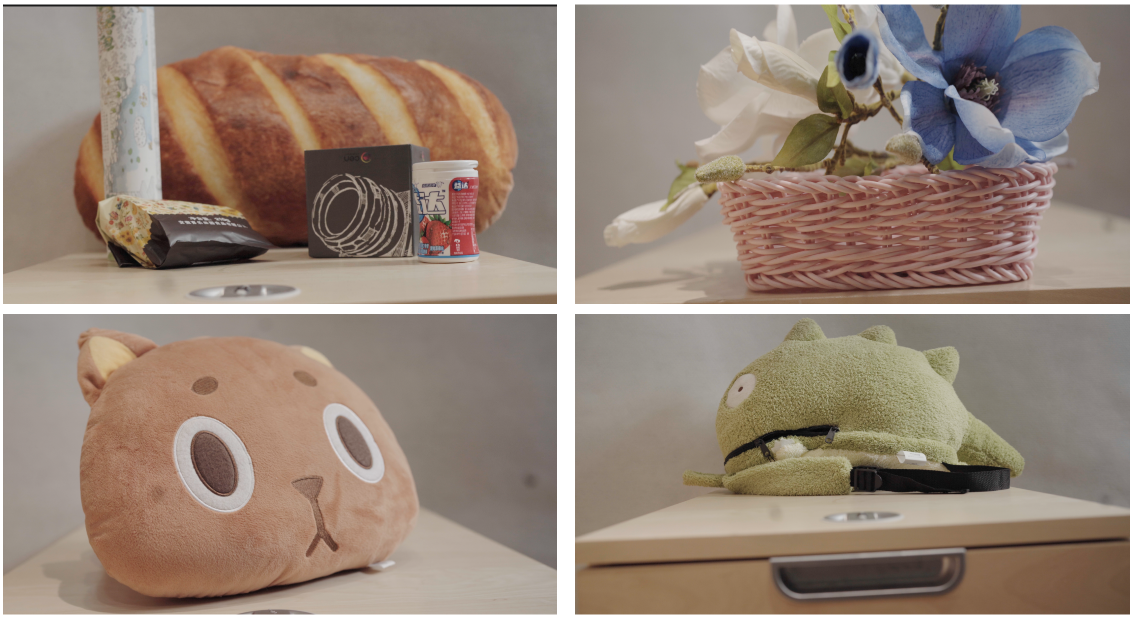


Figure 5.15: Illustration of the real scenes used in the experiments.

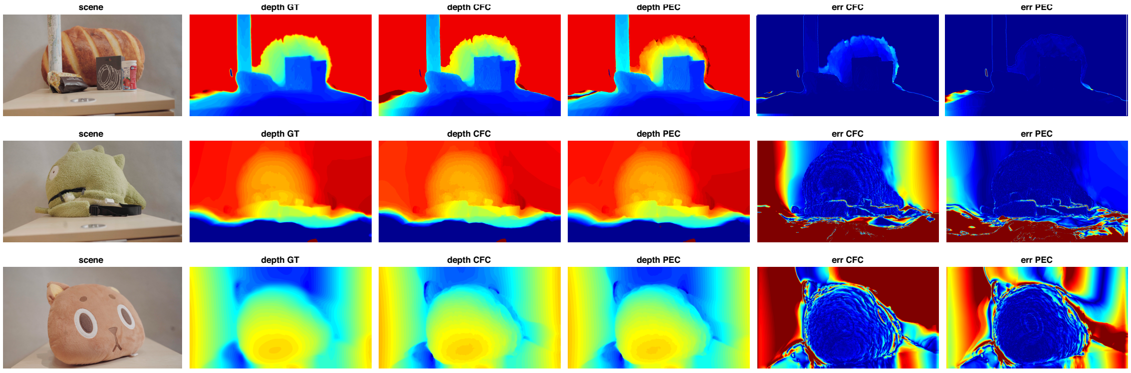


Figure 5.16: Illustration of evaluation of the CFC and PEC algorithms on the real scenes.

left to right is arranged as other experiments (image, ground truth, depth map of CFC, depth map of PEC, error map of CFC and error map of PEC). It can be seen that the perturbation technique improves the depth estimation well and mitigates the error parts. The total improvement in error could be 20-30% in various practical scenes.

Chapter 6

Conclusion

In this thesis, a closed-form formula of the 2-surface is developed, which proves the quadratic form of the constant disparity surface. The density and orientation of the disparity field are investigated to evaluate the suitability of the camera arrangement for the scene.

Importantly, we investigate the relationship between disparity, depth and camera alignment and introduce a vector field, called the camera's correspondence field (CF), which is the gradient field of the disparity in the space in front of the cameras. The proposed CF naturally combines the strengths of both the density and direction of the disparity field and provides a solution to find the optimal camera arrangement for depth estimation.

A novel camera arrangement optimization algorithm based on CF (CFC) is proposed to improve the accuracy of depth estimation. It is demonstrated that CF can be used to guide/select camera alignment for the efficient acquisition of 3D information. Extensive experimental results using five of the popular stereo matching algorithms show that an optimal camera arrangement can significantly improve the accuracy of depth estimation by about 30%. Furthermore, this improvement is independent of the matching algorithm, suggesting that it is additional or complementary to any correspondence matching method.

In addition, inspired by the involuntary eye movements during human fixation, a camera perturbation model is developed and a depth refinement optimization method is proposed without introducing additional occlusion regions for multi-view stereo images. This CF-based camera perturbation optimization method (PEC) is proposed for improving depth estimation. Experiments on synthetic and real models and simulated data of real scenes validate that the proposed PEC algorithm can improve the accuracy of depth estimation, including complex non-convex objects/scenes. The aim of this study is to further extend the analysis of multi-view cameras beyond pair-wise stereo. The PEC algorithm can be fully integrated with existing multi-view 3D algorithms.

In general, CF provides an approach to studying how well multiple cameras can sample or interact with scenes. This is a further step beyond the theory of epipolar geometry [87] which focuses on the geometric relationship among multiple cameras without considering the geometric complexity of the scenes. The studies in this thesis have demonstrated that CF can be used in guiding/selecting camera configurations for effective acquisition of 3D information. Also, the successful combination of CF and saccades indicates a promising path towards new mathematical modelling of 3D visual perception.

In future work, the relationship between CF and occlusion area and visibility coverage can be investigated for the application of next best view (NBV) in SLAM or robotic systems. Moreover, the CF theory and rigorous formulation of multiple cameras are still undeveloped, which leaves the potential for global camera alignment optimization for 3D geometric reconstruction.

Bibliography

[1] E. Camahort, F. Abad, and D. Fussell, "A line-space analysis of light-field representations," *Graphical Models*, vol. 71, no. 5, pp. 169–183, 2009.

[2] M. Pollefeys and S. Sinha, "Iso-disparity Surfaces for General Stereo Configurations," in *European Conference on Computer Vision (ECCV).* Springer Berlin Heidelberg, 2004.

[3] F. Safaei, P. Mokhtarian, H. Shidanshidi, W. Li, M. Namazi-Rad, and A. Mousavinia, "Scene-adaptive configuration of two cameras using the correspondence field function," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2013.

[4] S. P. Liversedge and J. M. Findlay, "Saccadic eye movements and cognition," *Trends in Cognitive Sciences*, vol. 4, no. 1, pp. 6–14, 2000.

[5] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-Scale Direct Monocular SLAM," in *European Conference on Computer Vision (ECCV)*, 2014.

[6] G. Lippmann, "Épreuves réversibles donnant la sensation du relief," *Journal de Physique Théorique et Appliquée*, 1908.

[7] I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Distance dependent depth filtering in 3D warping for 3DTV," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2007.

[8] Z.-W. Liu, P. An, S.-X. Liu, and Z.-Y. Zhang, "Arbitrary view generation

based on DIBR," in *International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS)*, 2008.

[9] I. Daribo and H. Saito, "Bilateral depth-discontinuity filter for novel view synthesis," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2010.

[10] W. R. Mark, L. McMillan, and G. Bishop, "Post-rendering 3D warping," in *Proceedings of the Symposium on Interactive 3D Graphics (I3D)*, 1997.

[11] J. Shade, S. Gortler, L.-w. He, and R. Szeliski, "Layered depth images," in *ACM SIGGRAPH Conference on Computer Graphics*, 1998.

[12] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[13] K.-J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.

[14] V. Vaish, R. Szeliski, C. L. Zitnick, S. B. Kang, and M. Levoy, "Reconstructing occluded surfaces using synthetic apertures: Stereo, focus and robust measures," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[15] X. Tan, C. Sun, D. Wang, Y. Guo, and T. Pham, "Soft Cost Aggregation with Multi-resolution Fusion," in *Lecture Notes in Computer Science*, ser. Euporean Conference on Computer Vision (ECCV). Springer, 2014, vol. 8693, pp. 17–32.

[16] K.-J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.

[17] S. Wanner and B. Goldluecke, "Globally consistent depth labeling of 4D light

fields," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[18] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *IEEE International Conference on Computer Vision (ICCV)*, 2001.

[19] M. G. Mozerov and J. van de Weijer, "Accurate Stereo Matching by Two-Step Energy Minimization," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 1153–1163, 2015.

[20] E. H. Adelson and J. Y. A. Wang, "Single lens stereo with a plenoptic camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 99–106, 1992.

[21] T. Stich, A. Tevs, and M. Magnor, "Global depth from epipolar volumes - A general framework for reconstructing non-lambertian surfaces," in *International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, 2007.

[22] T. E. Bishop and P. Favaro, "Full-resolution depth map estimation from an aliased plenoptic light field," *Lecture Notes in Computer Science*, vol. 6493, no. 2, pp. 186–200, 2011.

[23] T. Li, X. Ji, and Q. Dai, "Depth map recovery for multi-view using belief propagation," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV)*, 2009.

[24] B. Goldlcke and M. A. Magnor, "Joint 3d-reconstruction and background separation in multiple views using graph cuts," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[25] R. T. Collins, "Space-sweep approach to true multi-image matching," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1996.

[26] J. Yu, L. McMillan, and S. Gortler, "Scam light field rendering," in *Pacific Graphics*, 2002.

[27] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, "Fast Cost-Volume Filtering for Visual Correspondence and Beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 504–511, 2013.

[28] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision (ACCV)*, 2010.

[29] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[30] S. Lee, J. H. Lee, J. Lim, and I. H. Suh, "Robust stereo matching using adaptive random walk with restart algorithm," *Image and Vision Computing*, vol. 37, pp. 1–11, 2015.

[31] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *European Confernece on Computer Vision (ECCV)*, 2014.

[32] I. Daribo and B. Pesquet-Popescu, "Depth-aided image inpainting for novel view synthesis," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2010, pp. 167–170.

[33] K.-J. Oh, S. Yea, and Y.-S. Ho, "Hole filling method using depth based in-painting for view synthesis in free viewpoint television and 3-D video," in *Picture Coding Symposium (PCS)*, 2009.

[34] S. Zinger, L. Do, and P. H. N. de With, "Free-viewpoint depth image based rendering," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5, pp. 533–541, 2010.

[35] Y.-M. Feng, D.-X. Li, K. Luo, and M. Zhang, "Asymmetric bidirectional view synthesis for free viewpoint and three-dimensional video," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 4, pp. 2349–2355, 2009.

[36] L. Azzari, F. Battisti, and A. Gotchev, "Comparative analysis of occlusion-filling techniques in depth image-based rendering for 3D videos," in *ACM Workshop on Mobile Video Delivery (MoVid)*, 2010.

[37] A. Bobick and S. Intille, "Large occlusion stereo," *International Journal of Computer Vision (IJCV)*, vol. 33, pp. 181–200, 1999.

[38] C. Rabe, T. Müller, A. Wedel, and U. Franke, "Dense, robust, and accurate motion field estimation from stereo image sequences in real-time," in *European Confernece on Computer Vision (ECCV)*, 2010.

[39] P. Viola and W. Wells, "Alignment by maximization of mutual information," *International Journal of Computer Vision (IJCV)*, vol. 24, no. 2, pp. 137–154, 1997.

[40] R. Zabih and J. I. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *European Confernece on Computer Vision (ECCV)*, 1994.

[41] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[42] S. Khan, H. Rahmani, S. Shah, and M. Bennamoun, "A guide to convolutional neural networks for computer vision," *Synthesis Lectures on Computer Vision*, vol. 8, no. 1, pp. 1–207, 2018.

[43] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision (IJCV)*, vol. 47, no. 1-3, pp. 7–42, 2002.

[44] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[45] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. Berg, "Match- net: Unifying feature and metric learning for patch-based matchingj. zbontar and y. lecun, computing the stereo matching cost with a convolutional neural network, in ieee cvpr, 2015, pp. 15921599." *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[46] J. Zbontar and Y. Lecun, "Computing the stereo matching cost with a convolutional neural network," *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[47] W. Chen, X. Sun, L. Wang, Y. Yu, and C. Huang, "A deep visual correspondence embedding model for stereo matching costs," *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[48] W. Luo, A. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[49] M. Poggi and S. Mattoccia, "Learning a general-purpose confi- dence measure based on o(1) features and a smarter aggregation strategy for semi global matching," *International Conference on 3D Vision (3DV)*, 2016.

[50] J. Pang, W. Sun, J. Ren, C. Yang, and Q. Yan, "Cascade residual learning: A two-stage convolutional neural network for stereo matching," *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2017.

[51] Z. Liang, Y. Feng, Y. Chen, and L. Zhang, "Learning for disparity estimation through feature constancy," *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[52] Y. Yao, Z. Luo, S. Li, T. Shen, T. Fang, and L. Quan, “Recurrent mvsnet for high-resolution multi-view stereo depth infer- ence,” *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[53] S. Tulyakov, A. Ivanov, and F. Fleuret, “Practical deep stereo (pds): Toward applications-friendly deep stereo matching,” *International Conference on Neural Information Processing Systems (NIPS)*, 2018.

[54] C. Chen, X. Chen, and H. Cheng, “On the over-smoothing problem of cnn based disparity estimation,” *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[55] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der, D. Smagt, T. Cremers, Brox, and Flownet, “Learning optical flow with convolutional networks,” *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[56] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovit-Skiy, and T. Brox, “A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation,” *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[57] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, “Multi-view stereo for community photo collections,” in *EEE International Conference on Computer Vision (ICCV)*, 2007.

[58] Y. Furukawa and J. Ponce, “Accurate, dense, and robust multiview stereopsis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[59] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum, “Plenoptic Sampling,” in *Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, ser. SIGGRAPH ’00, New York, NY, USA, 2000, pp. 307–318.

[60] H. Shidanshidi, F. Safaei, and W. Li, “Objective evaluation of light field rendering

methods using effective sampling density," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, 2011.

[61] L. Bagnato, P. Frossard, and P. Vandergheynst, "Plenoptic spherical sampling," in *International Conference on Image Processing (ICIP)*, 2012.

[62] A. Lumsdaine, L. Lin, J. Willcock, and Y. Zhou, "Fourier analysis of the focused plenoptic camera," in *The International Society for Optical Engineering (SPIE)*, 2013.

[63] C. Gilliam, P. L. Dragotti, and M. Brookes, "A closed-form expression for the bandwidth of the plenoptic function under finite field of view constraints," in *International Conference on Image Processing (ICIP)*, 2010.

[64] A. Hornung, B. Zeng, and L. Kobbelt, "Image selection for improved multi-view stereo," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[65] I. Kostrikov, E. Horbert, and B. Leibe, "Probabilistic labeling cost for high-accuracy multi-view reconstruction," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[66] A. Ladikos, S. Ilic, and N. Navab, "Spectral camera clustering," in *International Conference on Computer Vision Workshops (ICCVW)*, 2009.

[67] M. Mauro, H. Riemenschneider, A. Signoroni, R. Leonardi, and L. Van Gool, "An Integer Linear Programming Model for View Selection on Overlapping Camera Clusters," in *International Conference on 3D Vision (3DV)*, vol. 1, 2014, pp. 464–471.

[68] S. Shen, "Accurate multiple view 3D reconstruction using patch-based stereo for large-scale scenes," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1901–1914, 2013.

[69] K. Wang, G. Zhang, and H. Bao, “Robust 3D reconstruction with an RGB-D camera,” *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4893–4906, 2014.

[70] G. Klein and D. Murray, *ECCV*, 2008, ch. Improving, pp. 802–815.

[71] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “ORB-SLAM: A Versatile and Accurate Monocular SLAM System,” *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[72] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, “DTAM: Dense Tracking and Mapping in Real-time,” in *International Conference on Computer Vision (ICCV)*, 2011.

[73] G. Olague and R. Mohr, “Optimal camera placement for accurate reconstruction,” *Pattern Recognition*, vol. 35, no. 4, pp. 927–944, 2002.

[74] A. Steinitz, “Optimal camera placement,” Master’s thesis, EECS Department, University of California, Berkeley, May 2012.

[75] Y. Chen, M. Tsukada, and H. Esaki, “Reinforcement learning based optimal camera placement for depth observation of indoor scenes,” *CoRR*, vol. abs/2110.11106, 2021.

[76] V. V. Petrov and K. A. Grebenyuk, “Improved stereoscopic imaging with converged camera configuration,” in *Saratov Fall Meeting 2006: Coherent Optics of Ordered and Random Media VII.* International Society for Optics and Photonics, 2007, p. 65360T.

[77] X. Song, Y. Wu, L. Yang, and Z. Liu, “Object position measuring based on adjustable dual-view camera,” in *IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 2013, pp. 1–6.

[78] T. Yoshida and T. Fukao, “Dense 3D reconstruction using a rotational stereo camera,” in *IEEE/SICE International Symposium on System Integration (SII)*, 2011, pp. 985–990.

[79] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, “Scene reconstruction from high spatio-angular resolution light fields,” *ACM Transactions on Graphics*, vol. 32, no. 4, 2013.

[80] D. Gallup, J. M. Frahm, P. Mordohai, and M. Pollefeys, “Variable baseline/resolution stereo,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[81] A. Harrold and P. Grove, “Binocular correspondence and the range of fusible horizontal disparities in the central visual field,” *Journal of vision*, vol. 15, no. 8, p. 12, 2015.

[82] B. Backus, M. Banks, R. Van Ee, and J. Crowell, “Horizontal and vertical disparity, eye position, and stereoscopic slant perception,” *Vision Research*, vol. 39, no. 6, pp. 1143–1170, 1999.

[83] K. Schreiber, D. Tweed, and C. Schor, “The extended horopter: Quantifying retinal correspondence across changes of 3d eye position,” *Journal of vision*, vol. 6, pp. 64–74, 2006.

[84] T. Chauhan, Y. Hjja-Brichard, and B. R. Cottereau, “Modelling binocular disparity processing from statistics in natural scenes,” *Vision Research*, vol. 176, pp. 27–39, 2020.

[85] E. Seemiller, B. Cumming, and T. Candy, “Human infants can generate vergence responses to retinal disparity by 5 to 10 weeks of age,” *Journal of Vision*, vol. 18, no. 6, pp. 17–17, 2018.

[86] A. Gibaldi and M. Banks, “Binocular eye movements are adapted to the natural environment,” *Journal of Neuroscience*, vol. 39, no. 15, pp. 2877–2888, 2018.

[87] R. Hartley and A. Zisserman, *Multiple View Geometry in computer vision*. Cambridge University Press, 2003.

[88] A. Clifford, *Multivariate Error Analysis*. John Wiley & Sons, 1973.

[89] M. Ester, H.-p. Kriegel, J. S, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise.” AAAI Press, 1996, pp. 226–231.

[90] J. J. Moré and D. C. Sorensen, “Computing a Trust Region Step,” *SIAM Journal on Scientific and Statistical Computing*, vol. 4, no. 3, pp. 553–572, 1983.

[91] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. Society for Industrial and Applied Mathematics, 1994.

[92] D. Bradley, T. Boubekeur, and W. Heidrich, “Accurate multi-view reconstruction using robust binocular stereo and surface meshing,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[93] S. C. G. Laboratory, *http://graphics.stanford.edu/data/3Dscanrep/*, Std.

[94] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, “A comparison and evaluation of multi-view stereo reconstruction algorithms,” in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[95] J. E. Hoffman and B. Subramaniam, “The role of visual attention in saccadic eye movements,” *Perception & Psychophysics*, 1995.

[96] K. Rayner, “Eye movements and attention in reading, scene perception, and visual search,” *Quarterly Journal of Experimental Psychology*, 2009.

[97] S. Fu, F. Safaei, and W. Li, “Optimization of camera arrangement using correspondence field to improve depth estimation,” *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 3038–3050, 2017.

[98] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, “Pixelwise view selection for unstructured multi-view stereo,” in *European Conference on Computer Vision (ECCV)*, 2016.

[99] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.