Sampling Models in Light Fields

THÈSE N^{O} 7003 (2016)

PRÉSENTÉE LE 8 JUIN 2016

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE DE COMMUNICATIONS AUDIOVISUELLES
PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Zhou XUE

acceptée sur proposition du jury:

Prof. R. Hersch, président du jury
Prof. M. Vetterli, Dr L. A. Baboulaz, directeurs de thèse
Prof. P. L. Dragotti, rapporteur
Dr C. Perwass, rapporteur
Prof. P. Frossard, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
Suisse
2016

Acknowledgments

I feel tremendously lucky to have Prof. Martin Vetterli and Dr. Loïc Baboulaz as my two advisors. “A journey of a thousand miles begins with a single step” is a Chinese quotation from Laozi, which Martin mentioned to me multiple times to guide my research. Martin, thank you so much for your support and enormous patience. Your methodology is not only a bright beacon for my research in this wonderful adventure of my Ph.D, but also a priceless asset for my work and life in the future. Loïc, thanks for supporting me, for spending hours reading my thesis and papers, for helping me with my English and for showing me the importance of communication skills. Furthermore, I sincerely thank Dr. Paolo Prandoni, whose semester project lead me to this fantastic lab. Thanks for all the inspiring discussions and your generous help with my thesis writing.

I would also take this opportunity to express my gratitude to my thesis committee members, Prof. Roger Hersch, Prof. Pascal Frossard, Prof. Pier Luigi Dragotti and Dr. Christian Perwass. Thanks a lot for agreeing to assess my work. Without their patience, insightful suggestions and friendly reminders, my thesis will not be in the current shape.

I am very thankful to my dear friend Niranjan, with whom I started to share the same office since day one in EPFL. I really appreciate all the research discussions and interesting conversations, which help me through this journey. I owe many thanks to our lab’s big boss, Jacqueline, for all the keen and patient helps. I would also like to thank all my LCAV colleagues: Runwei, Gilles, Marta, Mitra, Robin, Feng, Zichong and other present and previous members. A very special thanks to my colleague Hanjie. An interesting conversation with him directly lead to the final chapter of my thesis.

My friends, Xiaolu, Xinchao, Mingfu, Jian, Jingge, Ye, Hao, Xifan, Min, Nan, Wenqi and many more I cannot list here, thank you all for making my life in Lausanne colorful. I also want to thank my two best friends, Tianran and Jinfeng for their invaluable friendship and absolute trust in me.

Last but not the least, I would like to thank my parents, Shouying and Yujun for their unconditional love and support, as well as the curiosity and tenacity they instilled in me. Finally, I sincerely thank my beloved wife, Liting, for her love, endless support, continued optimism and patience. She gives me the strength to resurrect dreams after defeat, without which this thesis would not be possible. This thesis is dedicated with love, to my family.

Abstract

What is the actual information contained in light rays filling the 3-D world? Leonardo da Vinci saw the world as an infinite number of radiant pyramids caused by the objects located in it. Nowadays, the radiant pyramid is usually described as a set of light rays with various directions passing through a given point. By recording light rays at every point in space, all the information in a scene can be fully acquired.

This work focuses on the analysis of the sampling models of a light field camera, a device dedicated to recording the amount of light traveling through any point along any direction in the 3-D world. In contrast to the conventional photography which only records a 2-D projection of the scene, such camera captures both the geometry information and material properties of a scene by recording 2-D angular data for each point in a 2-D spatial domain. This 4-D data is referred to as the light field. The main goal of this thesis is to utilize this 4-D data from one or multiple light field cameras based on the proposed sampling models for recovering the given scene.

We first propose a novel algorithm to recover the depth information from the light field. Based on the analysis of the sampling model, we map the high dimensional light field data to a low dimensional texture signal in the continuous domain modulated by the geometric structure of the scene. We formulate the depth estimation problem as a signal recovery problem with samples at unknown locations. A practical framework is proposed to recover alternately the texture signal and the depth map. We thus acquire not only the depth map with high accuracy but also a compact representation of the light field in the continuous domain. The proposed algorithm performs especially well for scenes with fine geometric structure while also achieving state-of-the-art performance on public data-sets.

Secondly, we consider multiple light fields to increase the amount of information captured from the 3-D world. We derive a motion model of the light field camera from the proposed sampling model. Given this motion model, we can extend the field of view to create light field panoramas and perform light-field super-resolution. This can help overcome the shortcoming of limited sensor resolution in current light field cameras.

Finally, we propose a novel image based rendering framework to represent light rays in the 3-D space: the circular light field. The circular light field is acquired by taking photos from a circular camera array facing outwards from the center of the rig. We propose a practical framework to capture, register and stitch multiple circular light fields. The information presented in multiple circular light fields allows the creation of any virtual camera view at any chosen location with a 360° field of view. The new representation of the light rays can be used to generate high quality contents for virtual reality and augmented reality.

Keywords: Light field, sampling model, depth recovery, motion model, light field registra-

tion, light field stitching, circular light-field, virtual reality.

Zusammenfassung

Welche Informationen enthalten die Lichtstrahlen, die unsere 3D Welt ausfüllen? Leonardo da Vinci sah die Welt als eine unendliche Anzahl an Sehpyramiden, die durch die Objekte in dieser Welt definiert sind. Heutzutage beschreibt man eine solche Sehpyramide gewöhnlich als eine Menge von Lichtstrahlen mit verschiedenen Richtungen, die alle durch einen vorgegebenen Punkt gehen. Wenn man alle Lichtstrahlen an allen Punkten eines 3D Raums aufnimmt, dann hat man alle Informationen der Szene vollstndig erfasst.

Der Fokus dieser Arbeit liegt auf der Analyse der Abtastmodelle von Lichtfeldkameras. Eine Lichtfeldkamera ist ein Gerät, das die Menge an Licht messen kann, die in einer bestimmten Richtung durch einen bestimmten Punkt im 3D Raum transportiert wird. Im Gegensatz zu konventionellen Fotokameras, die nur 2D Projektionen einer Szene aufnehmen, erfassen Lichtfeldkameras Geometrie und Materialeigenschaften einer Szene indem sie die Lichtmenge pro Richtung (2 Winkel) fr jeden Punkt eines 2D Bildraumes abspeichern. Solche 4D Daten bezeichnet man als ein Lichtfeld. Das Hauptziel dieser Doktorarbeit ist es, solche 4D Daten von einer oder mehreren Lichtfeldkameras, basierend auf den vorgeschlagenen Abtastmodellen, zu nutzen.

Wir stellen zuerst einen Algorithmus vor, der aus einem gegebenen Lichtfeld die Tiefe rekonstruiert. Auf Grund der Analyse des Abtastmodells projizieren wir die hochdimensionalen Lichtfelddaten auf ein kontinuierliches Texturensignal mit tieferer Dimension, das durch die geometrische Struktur der Szene moduliert wird. Wir formulieren die Tiefenrekonstruktion als ein Signalwiederherstellungsproblem mit Abtastwerten an unbekannten Stellen. Unser System stellt alternierend das Texturensignal und die Tiefenkarte wieder her. So erhalten wir nicht nur die Tiefenkarte, sondern auch eine kompakte und kontinuierliche Darstellung des Lichtfeldes. Unser Algorithmus funktioniert besonders gut fü Szenen mit detaillierten geometrischen Strukturen. Wenn man den Algorithmus an ffentlichen Datenstzen testet, sind die Resultate vergleichbar mit dem aktuellen Stand der Technik.

In einem zweiten Schritt bercksichtigen wir mehrere Lichtfelder um die Menge der aufgenommen Informationen zu erhhen. Vom vorgeschlagenen Abtastmodell leiten wir ein Bewegungsmodell fr die Lichtfeldkamera ab. Mit diesem Bewegungsmodell knnen wir das Gesichtsfeld erweitern um Lichtfeldpanoramas zu erstellen und um Lichtfeldsuperauflösung zu erreichen. Auf diese Weise kann das Problem der limitierten Sensorauflsung herkmlicher Lichtfeldkameras berwunden werden.

Schliesslich präsentieren wir ein neuartiges, bildbasierendes Renderingframework, das Lichtstrahlen im 3D Raum darstellt: das zirkulre Lichtfeld. Ein zirkuläres Lichtfeld wird aufgenommen indem man mit einer kreisfrmigen Anordnung von Kameras, die alle nach aussen gerichtet sind, Fotos schiesst. Dann stellen wir ein geeignetes System vor, das mehrere zirklulre Lichtfelder aufzunehmen, registrieren und zusammensetzen kann. Die Informationen, die in mehreren zirkulren Lichtfeldern vorhanden sind, erlauben es eine virtuelle Kamera mit einem Sichtfeld von 360° an einem beliebigen Ort zu platzieren. Mit Hilfe dieser neuen Darstellung von Lichtstrahlen

knnen hochqualitative Inhalte fr die virtuelle oder erweiterte Realitt generiert werden.

Schlagwörter: Lichtfeld, Abtastmodell, Tiefenwiederherstellung, Bewegungsmodell, Lichtfeldregistrierung, Lichtfeldstitching, zirkuläres Lichtfeld, virtuelle Realität.

Contents

Chapter 1

Introduction

> If you cannot explain it simply, you don't know it well enough.
>
> *Albert Einstein*

The problem of faithfully capturing and representing the full three-dimensional information of a real-world scene is arguably as old as photography itself. Thomas Wedgwood started experimenting with light-sensitive chemicals around the year 1800 and in 1826 Nicphore Niépce produced the first permanent photograph with a camera obscura and a pewter plate coated with bitumen; just twelve years later, Charles Wheatstone proposed the first 3-D photography method by introducing the stereoscope [54]. Since then, 3D photography (and, obviously, film) has been an active research topic with countless innovations and rediscoveries. The importance of 3D information goes well beyond the world of art and entertainment: industrial application range from long established domains such as robotics and industrial inspection, to the recently emerging areas of human-computer interaction (with products like Kinect and Leap Motion), 3-D printing and virtual reality.

Recovering 3D information from 2D images is a challenging inverse problem, one that we tend to take for granted given the extreme sophistication of the human visual system. From the point of view of projective geometry, the stereoscope is the most straightforward 3D imaging system and one that is very close to the way human vision works. The principle behind the stereoscope is parallax: an object viewed along two different lines of sight exhibits an apparent displacement and this displacement is greater the closer the object is to the viewer. When a 3D scene is captured with a camera, this angular diversity is lost: in the case of a pinhole camera because only one light ray from the object can reach the sensor; in the case of a normal camera, because the bundle of rays that go through the aperture are integrated on the sensor. However, as soon as we have at least two viewpoints (as in stereoscopy) at least a portion of the angular information can be recovered and, in so doing, part of the geometry of the scene.

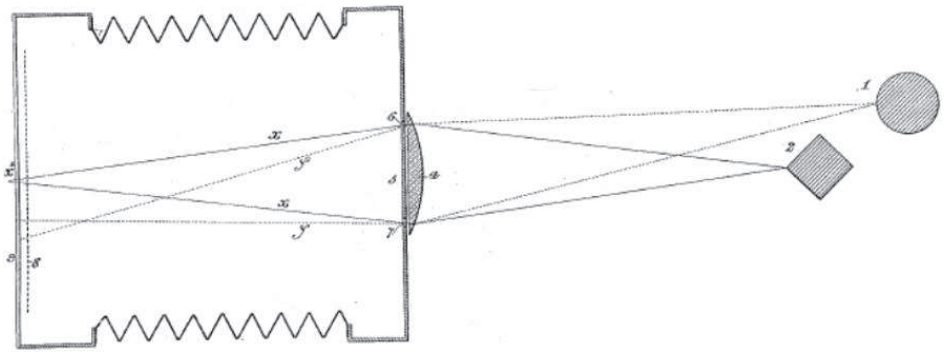

Figure 1.1: The camera design from Ive's patent in 1902[26].

It's worth mentioning in passing that the human visual system, in reconstructing a mental 3D image of a perceived scene, relies on not only binocular cues but also on the visual semantics of the real world; so much so, in fact, that even with a single functioning eye, humans still have the ability to perceive depth. This is at the basis of fascinating research topics such as shape from shading [55], shape from texture [3] and shape from silhouette [6][7]; although certainly interesting, these contributions are however beyond the scope of the present work.

Consider now the opposite point of view (no pun intended): instead of a monocular view of the world, assume we have access to the *plenoptic* representation of a scene. This means we have information about every single light ray passing through every possible point in space. Clearly, the plenoptic information contains the complete 3D information about the scene; just as clearly, the plenoptic function is a highly idealized entity whose high dimensionality needs to be brought back to manageable scale. In this thesis we focus on how to efficiently sample both the spatial and angular information of the light rays for a given scene: we investigate the models to represent the light rays and the devices for recording them, together with their spatial and angular information; these devices are commonly referred to as light-field or plenoptic cameras. We will build a very versatile light-field camera prototype; we will derive sampling methods for the light field and develop algorithmic methods to infer depth information for the scene.

1.1 Seeing the World through Light Field Cameras

To record the 3D information of a scene, we need to be able to capture individual light rays with their incidence angles; as we said before, standard camera cannot do that because each pixel [1] encodes the average of a bundle of rays crossing the aperture.

The first patented device to address this problem was proposed by Ives in 1903 [26] (see Figure 1.1). With a uniform array of pinholes positioned in front of a camera sensor, each pixel obtains a narrow bundle of light rays and, in the ideal case, only a single light ray will pass through each pinhole; this "barrier-based" design allows the camera to record the spatial and angular information of the light rays. Conceptually, Ives' camera is an extension of the stereoscope to a large array of pinhole cameras. .

1. We will use the term "pixel" independently of the analog or digital nature of the imaging device, since we can always "scale in" to a basic picture element in both cases.

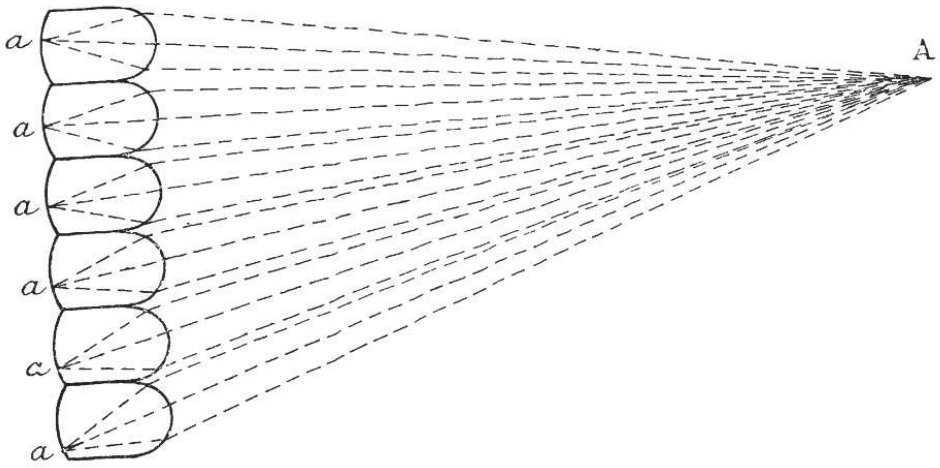

Figure 1.2: The camera design from Lippmann's work in 1908[34].

Unfortunately, the well-known drawback of pinhole cameras is the limited amount of light that reaches the photographic sensor, so that they can be used only when the subject allows for very long exposure times. To compensate for this, in 1908 Lippmann proposed to replace the pinhole array with an array of microlenses, as shown in Figure 1.2. Each microlens is focused at infinity, so that each pixel measures a bundle of parallel light rays from a particular direction[34]. This technique is usually referred to as *integral photography* and it significantly increases the efficiency of the imaging process.

Lippman's idea provided the basis for a design that persists in modern devices; while many scientists endeavored to reinvent and adapt the light-field camera over the past century [22, 27, 28], one of the most influential contributions was put forth by Adelson and Wang in 1992 [2]. As shown in Figure 1.3, they introduced a relay lens to bring the focal plane of the microlens array onto the sensor. The resulting camera captures a continuum of viewpoints since each macropixel (i.e. the ensemble of pixels subtended by a single microlens) records the distribution of light within the main lens. This design simplifies the assembly and calibration of the light-field camera, yet makes the whole device too long to be portable.

In 2005, Ng et al. implemented the modern incarnation of integral photography[40]. Compared to Adelson and Wang's design, the optical path of Ng's light-field camera is significantly shortened with the removal of the relay and field lens. A microlens array is cemented on the image sensor at a distance equal to the microlens focal length, as in Lippmann's camera. Modern technology, however, allows the manufacture of arrays where the single microlens is vanishingly small compared to the main lens, and yet sufficiently precise to produce a sharp image. The result is that each microlens measures the spatial location of the incoming light rays with a resolution equal to the elements in the array, while each pixel in the sensor captures the individual direction of each light ray.

Note that as the microlens array is focused at infinity, the pixels in each macropixel are used for the acquisition of the angular component of the light rays, and this results in a low spatial resolution of the final rendered images, which equal to the number of microlenses in the array. To address this issue a modified design was proposed by Lumsdaine and Georgiev[36] by using a

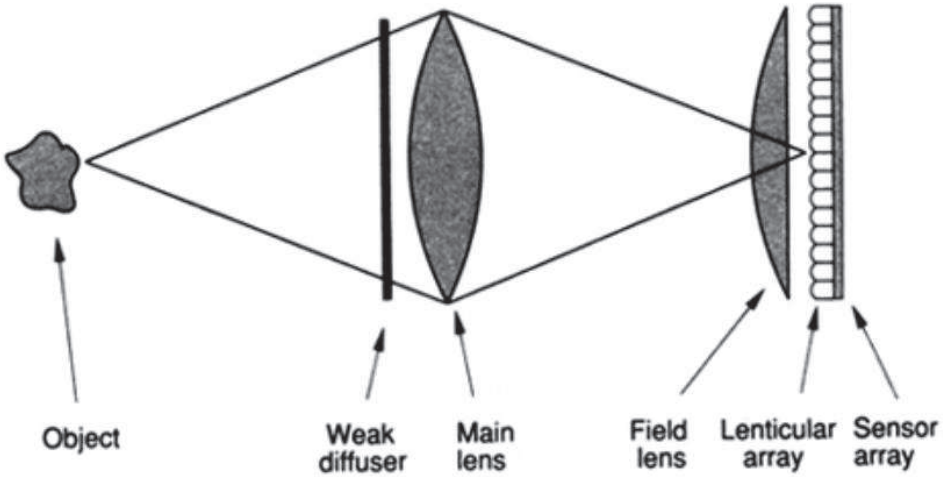

Figure 1.3: The camera design from Adelson's work in 1992[2].

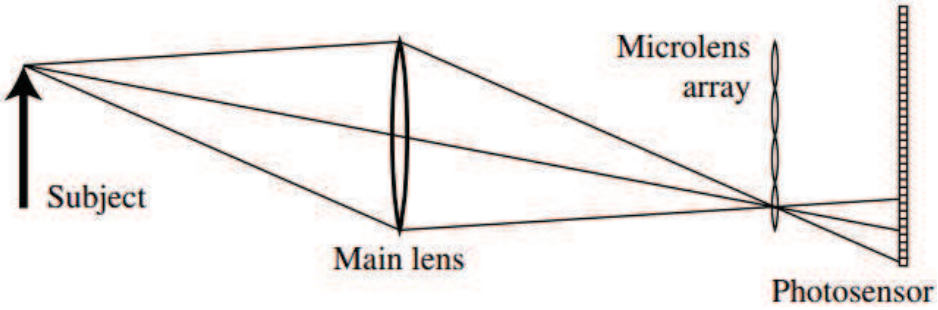

Figure 1.4: The camera design from the work of Ng et al.[40].

microlens array that is focused at a given distance instead of infinity.

In 2009, the first commecially available light-field camera to appear on the market was launched by Raytrix, using a design similar to Lumsdaine's. To increase the depth of field Raytrix employs three different types of microlens, such that the camera can cover a larger depth range; the design however also increases the complexity of the acquisition process. As a consequence, Raytrix cameras are individually designed for their specific task: in industrial inspection, for example, the camera is designed for a particular depth range, so that it can achieve a sufficient resolution for the high accuracy demanded by the quality control process. As the image sensor of a light-field camera is used for recording both the spatial and angular information, it is important to determine the proper trade-off based on the application.

In 2012, the first consumer-grade light-field camera was launched by Lytro, Inc., a company founded by Ng in 2006. They released the first generation Lytro camera and Lytro ILLUM in 2012 and 2014, respectively; both devices use the camera design proposed by Ng [40]. Instead of focusing on depth reconstruction, as in the case of Raytrix, the Lytro camera is capable of re-synthesizing images focused at arbitrary depths after the photo is taken; this type of a posteriori focusing is a long-standing problem in photography. In addition to the refocusing

Figure 1.5: The Lytro Immerge. This camera is for the capture of virtual-reality videos.

application, users can also change the perspective of the photos and even create depth maps with the acquired data. The major advantage of the consumer cameras is that they are simple to use and do not require any technical knowledge with respect to the light-field acquisition. In 2015 Lytro announced the release of a new light-field camera for capturing virtual-reality videos called Immerge (see Figure 1.5); the device produces videos in which the viewpoints can be changed freely.

As we shortly outlined, the many incarnations of the light field camera in history have tried to provide a mechanism to capture both the spatial and angular information of light rays. Given the unmanageable dimensionality of the problem, each design implements a sampling model for the light rays and this model is the crucial factor in the performance and potential applications for each device. In this thesis, we investigate these sampling models and apply them to the problem of depth recovery (in Chapter 3) and light-field registration (in Chapter 4). Finally, we also demonstrate a reinterpretation of the standard light field by introducing a circular light-field that is specifically designed for virtual reality in Chapter 5.

1.2 Thesis Outline

We hope that the subject matter of this thesis will be accessible to a broad range of readers, including researchers working in the fields of signal processing, computer vision, virtual reality, and photographers working with light-field cameras. Here is a broad chapter-by-chapter breakdown of the content.

Sampling Models for Light-Field Cameras (Chapter 2)

This chapter, which establishes the foundations of the thesis, introduces the basic concept of light field and presents a general framework for analyzing the sampling models of light field cameras. We describe an implementation consisting of a main lens in front of a moving pinhole camera. Building on this simple model, we demonstrate various sampling models under different camera settings and clarify the relationship between camera design and the sampling periods in the spatial and angular domains. The theoretical results are validated via a prototype, whose design and assembly is described in detail, which allows for flexible sampling periods in both spatial and angular domains and which will be used as a powerful experimental tool for the depth recovery in Chapter 3 and registration algorithms in Chapter 4.

Depth Recovery from Surface Light-Fields (Chapter 3)

In this chapter we propose the concept of a low-dimensional surface light-field: the high-dimensional light field is modeled as a low-dimensional surface light-field modulated by the geometry structure of the scene. By using the surface light-field, we can cast the traditional depth reconstruction problem as one of bandlimited signal recovery from unknown sampling locations, and we propose a novel and practical framework for exploiting the properties of the surface light-field for high-accuracy depth recovery. We perform experiments on synthetic public datasets and achieve state-of-the-art performance. The proposed algorithm best suits sophisticated scenes within a small depth range. We also use our light-field camera prototype to run the algorithm on datasets of a 3-D printed object with micrometer resolution and on oil paintings.

Light-Field Registration and its Applications (Chapter 4)

In this chapter, we derive the motion model of a light-field camera by using its sampling model. We propose the use of a light-field camera as an image scanner and perform light-field stitching to increase the size of the acquired data. More specifically, we describe how to stitch multiple light-fields under two different scenarios: by camera translations and by camera translations and rotations. We discuss the impact of extending the spatial and angular dimensions of the stitched light-field. We also address the issue of algorithmic light-field registration by describing a direct method in the Fourier domain and a feature-based method for large camera displacements.

Circular Light Fields for Virtual Reality (Chapter 5)

In this chapter we propose a novel, circular model for the light-field aimed at virtual reality applications; the model allows for the rendering of new views with a 360-degree field of view from any chosen location. A circular light-field is created with a set of cameras mounted on a circular rig; since the size of the rig limits the range of the rendering, we show how to extend the coverage by registering and stitching multiple circular light-fields. In addition, we can also create the circular light-field by pointing the camera towards the center of the circular rig. Then, instead of capturing the environment, we capture a object positioned at the center and use the acquired data to render this object for different viewing angles at different distances.

1.3 Thesis Contributions

The main contributions in this thesis are in the investigation of light field sampling models and their applications:

— we derive theoretical results on sampling models for the light field and build a working prototype of a light field camera with flexible sampling periods to validate the theory

— we introduce the concept of surface light field and design a novel algorithm around it to achieve high-accuracy results in depth estimation

— we describe a motion model for the light-field in order to create stitched light fields with a wider field of view from multiple acquisitions

— we introduce a new representation for the light rays called the circular light-field, allowing for the rendering of novel views with a 360-degree field of view at any chosen location.

Chapter 2

Sampling Models for Light-Field Cameras

A journey of a thousand miles begins with a single step.

Laozi

Einstein once said that if he had one hour to save the world he would spend fifty-five minutes defining the problem and only five minutes finding the solution. We face a similar situation when dealing with new computational imaging devices. Light-field cameras are efficient and promising imaging devices. Before diving right into the applications and algorithms to utilize them, we should take a step back and invest time and effort to improve our understanding of it.

In this chapter, we identify and analyze sampling models of light-field cameras. We first give a short review of the plenoptic function and demonstrate how it is simplified into the 4-D light field in Section 2.1. We then analyze the sampling process of light fields and derive the sampling models of the light field camera in Section 2.2. We discuss the variations of the sampling model under different camera settings and demonstrate the relations between the sampling periods in spatial and angular domains and the light field camera designs. Finally, based on the proposed sampling model, we construct a light field camera that has flexible sampling periods for experimental purposes in Section 2.3.

2.1 Introduction and Related Work

Identifying the sampling model of the camera is a fundamental step, before processing the acquired data. It entails the use of the properties of a given scene and the imaging system. We review the concept of the light field to demonstrate the challenges and advantages in capturing

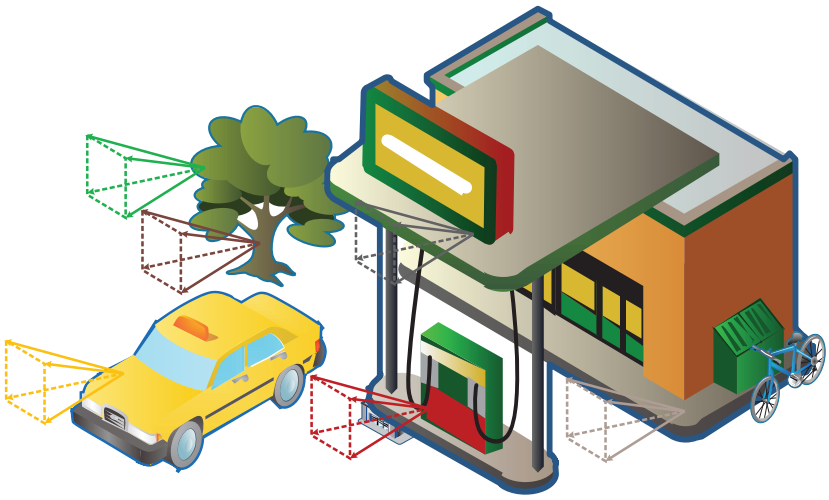

Figure 2.1: A world full of radiant pyramids.

the light rays that flow in a given space instead of a 2-D projection of the scene. We discuss the representations of the 4-D light field and afterwards present the analysis of the 2-D light field.

2.1.1 Plenoptic function

What is the actual information contained in the light rays filling the 3-D world? Leonardo da Vinci once wrote in his manuscript on painting: "The air is full of an infinite number of radiant pyramids caused by the objects located in it. These pyramids intersect and interweave without interfering with each other during the independent passage throughout the air in which they are infused."

Nowadays, radiant pyramids are usually referred to as pencils of light rays as shown in Figure 2.1: the pencil being a set of rays that pass through any given point in space. Building on this concept, Adelson and Bergen introduced the concept of light fields and developed a theory of the plenoptic function in [1].

To capture one pencil of rays, we can place a pinhole camera at the given point (C_x, C_y, C_z) in a given 3-D space and measure the intensity distribution of the light rays. Then each measurement can be parameterized with spherical coordinates (θ, ϕ) where θ and ϕ denote the azimuth and elevation of the light ray passing through the point (C_x, C_y, C_z), respectively. As for the light ray itself, its wavelength λ and measuring time t are also taken into consideration. Finally we end up with a seven dimensional function $L(C_x, C_y, C_z, \theta, \phi, \lambda, t)$ describing the light rays filling the 3-D world.

2.1.2 4-D light-field representation

Although the 7-D plenoptic function forms a complete representation of the light rays filling a given space, it is still problematic to capture due to its high dimensionality. To tackle this problem, we can remove the variables that are not crucial to extract the information of the scene. First by assuming a static scene, the variable t can be neglected. Then by using a monochrome

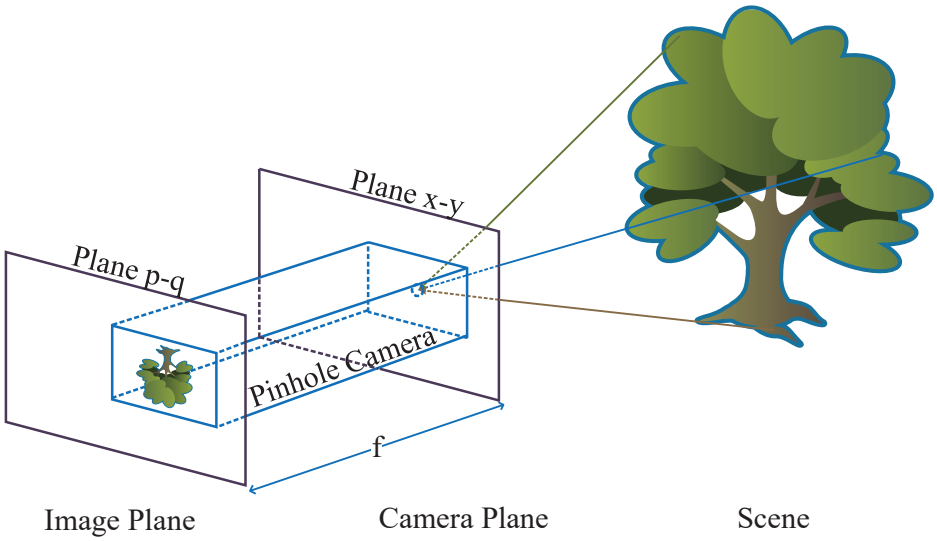

Figure 2.2: The representation of the 4-D light field. A pinhole camera is positioned with the pinhole on the plane $x - y$. The captured image is on the plane $p - q$. The distance f between the plane $x - y$ and $p - q$ is the focal length of the pinhole camera.

camera, the wavelength λ can also be neglected. Furthermore, by assuming the intensities of the light rays remain constant while propagating in space, the remaining 5 dimensions of the function can be further reduced to 4. The 4-D function is usually referred to as the 4-D light field.

In this thesis, to represent the light rays in a given 3-D space, we use a two-plane parameterization that is also used by Levoy and Hanrahan in [32]. In the 3-D space, each light ray can be uniquely identified by its intersections with two pre-defined planes $x - y$ and $p - q$ parallel to each other. Then radiance intensity of the light ray is assigned to the 4-D light-field function $L(x, y, p, q)$ by the 4-D index of the intersections on these two parallel planes.

We illustrate how 4-D light fields are defined and created in Figure 2.2. In the setup, a moving pinhole camera on plane $x - y$ is used to record light rays emitted from the scene. The image plane of the pinhole camera is defined as the second plane $p - q$. Both of these planes are parallel to each other and are perpendicular to the optical axis. The 4-D light field is the set of images captured as the camera moves on the camera plane. The $p - q$ plane records the direction of each light ray that passes through the corresponding pinhole located on the $x - y$ plane. By normalizing the pinhole camera's focal length f to be 1, the coordinate vector (p, q) becomes the direction vector for each corresponding light ray.

2.1.3 Analysis of 2-D light field

In any image pair taken at two different locations, a point in the 3-D space is projected to a stereo pair in the two images, with a disparity determined by its depth. For the sake of simplicity, we demonstrate the relation between the standard stereo system in 2-D and the acquisition setup of the 2-D light field in Figure 2.3.

In the 2-D case, the camera in the stereo system captures only 1-D images. By simply fixing

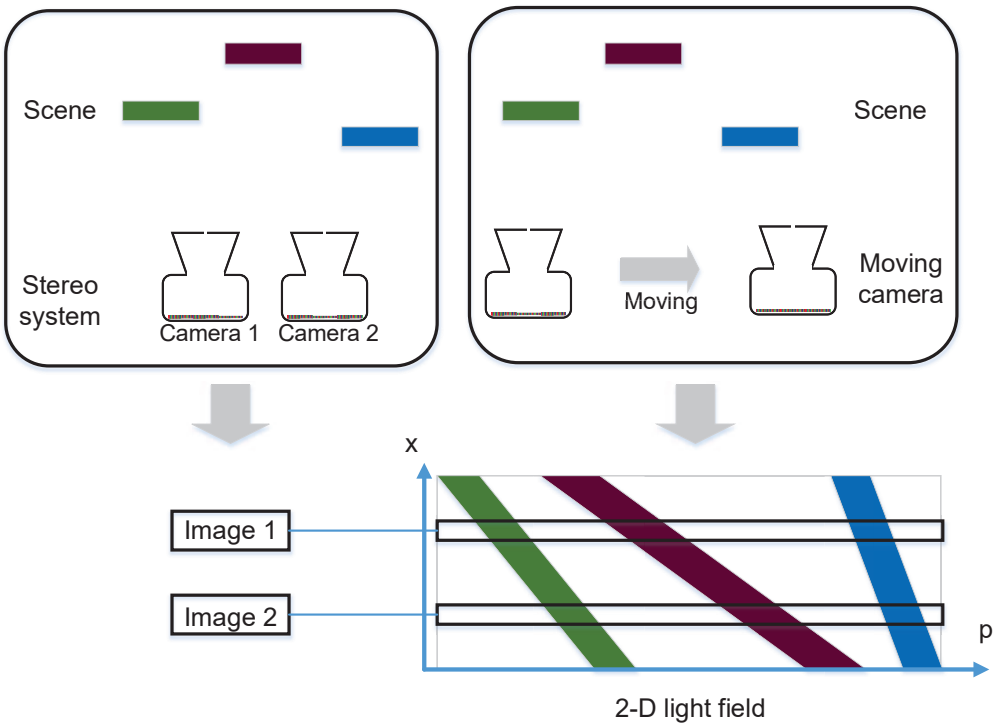

Figure 2.3: Illustration of the 2-D light field. We demonstrate the relation between an image pair captured by a stereo system and a 2-D light field captured by a moving camera.

the variable y and q, the 4-D light becomes a 2-D light field; this 2-D light field is also known as the epipolar plane images (EPIs) in the computer-vision community. This concept originally comes from Bolles et al. [12].

In Figure 2.3, we use 3 color bars to represent the tree in Figure 2.2. On the top left, we show a standard stereo system that captures two images of the scene. On the top right, we show a moving camera that captures the 2-D light field of the scene. The image pair are two rows in the 2-D light field. We can clearly see how the disparities become lines with different slopes.

When capturing a given point in the space with a stereo camera-system, the point corresponds to a unique disparity value because the baseline between the two cameras is fixed. As for the acquisition system of the light field, the baseline among cameras becomes a variable in our setup. For a given point, the observed disparity values are linearly proportional to the baseline. Therefore, we observe line structures in the 2-D light field $L(x, p)$, in which x denotes the camera position and p denotes the image. More specifically, a line in the 2-D light field corresponds to a set of light rays emitted from the same point. When the surface is Lambertian, the intensity of the line slice remains constant. The Lambertian property is widely used to estimate depth maps by computing the slope of these lines in the 2-D light fields.

More specifically, the relation between the variables x and p can be explained with the standard stereo system as shown in Figure 2.4. Two cameras are positioned by a translation of Δx that is perpendicular to the optical axis. Δx is also referred to as the baseline between these two cameras. We use the variable z to denote the depth value of the point being observed and

Δp to denote the disparity on the image sensor. Then the depth can be estimated by

$$z = \frac{\Delta x}{\Delta p}.$$

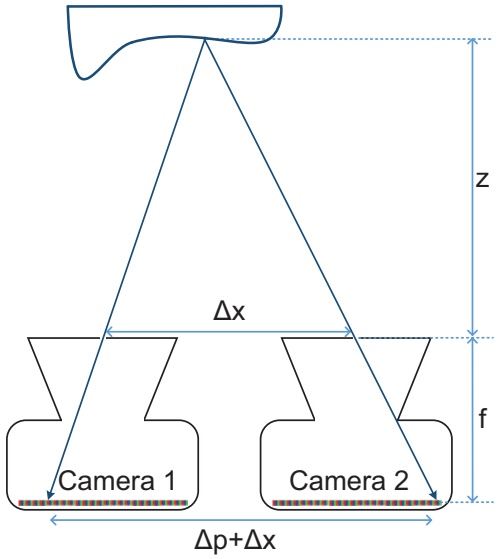

Figure 2.4: A standard stereo vision system. The baseline between two cameras is Δx, whereas the disparity is Δp.

As for the relation between x and p in the 2-D light field, we can formulate the depth as the line orientation as follows

$$z(x, p) = \frac{dx}{dp}. \tag{2.1}$$

Equation (2.1) shows an interesting phenomena: in the 2-D light field, a given point corresponds to a line of which the slope is determined by the distance from the point to the camera plane. All the entries on the line correspond to light rays emitted from this particular point. In other words, a line slice in the 2-D light field corresponds to a set of light rays passing a particular point that is determined by the slope of the line slice.

For a better understanding, in the 2-D light field, the slope of a horizontal line is zero, thus the horizontal line corresponds to a point on the camera plane, which is the pinhole. In a general case, any line in the 2-D light field can be seen as a 1-D image captured by putting a pinhole camera at the corresponding position that is determined by the slope of the line. By slicing the 2-D light field, we obtain a radiant pyramid mentioned in Section 2.1.1. We usually refer to these line slices as images captured by virtual pinhole cameras. So when we put this virtual pinhole camera on the surface of a given object, we capture a 1-D image that is a set of light rays emitting from the same spot on the surface. This particular 1-D image can be seen as a part of the reflectance function of the material of the surface as it is a set of radiance measurements at different directions.

2.1.4 Devices for light-field acquisition

As described by Levoy and Hanrahan in [32], the acquisition of the light fields can be very straightforward, as shown in Figure 2.5. They captured a set of photographs with an array of cameras uniformly distributed on a planar surface. Each camera records a set of light rays with varying directions, while the camera itself records the intersection of the light ray and the camera plane.

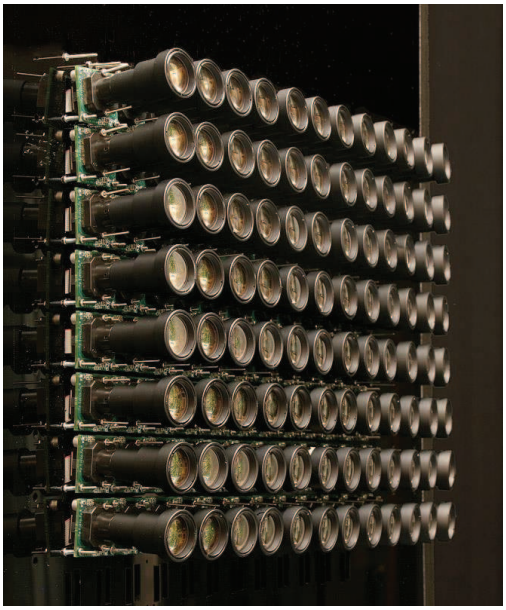

Figure 2.5: Camera array for light field aquisition. This photo is one example from the Stanford multi-camera arrays.

However, this type of acquisition has no flexibility in the imaging process. Once the camera array is set up, the lens of the camera and the baseline between the cameras are both fixed. Whenever the scene being photographed changes, the bulky setup has be adjusted accordingly. There are many alternative acquisition systems with various features. Here we introduce the two main commercial products in the market: Lytro and Raytrix as shown in Figure 2.6.

Lytro has an implementation much more compact than the bulky system shown in Figure 2.5. By placing a microlens array in front of the sensor, the spatial and angular information can be recorded separately. To effectively capture the scene, the sensor and microlens array are positioned behind a main lens. The main lens is used as a relay device that projects the world into the camera. Hence the microlens array and sensor capture the light field the same way as the camera array.

Inside the Lytro camera, the distance between the microlens array and the sensor is fixed to be the focal length of the microlens. Each pixel under the microlens corresponds to a beam with various directions, which means the microlens is focused at infinity. This results in blurry images under each microlens. But the blurring kernel does not change much for objects at different depths. This makes the modeling of the acquired light-field data quite straightforward.

Raytrix has a slightly different design compared to Lytro: the microlens of Raytrix is in

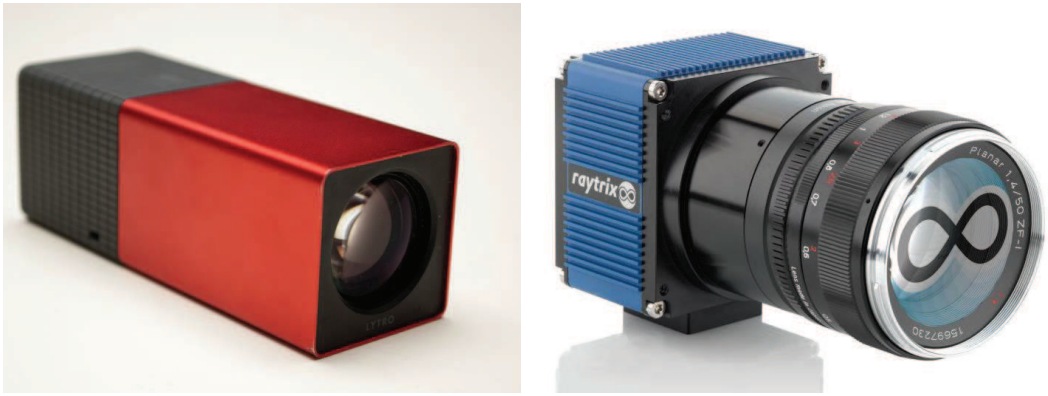

Figure 2.6: Two commerical light field cameras: Lytro on the left and Raytrix on the right.

focus thus the image under each microlens is sharper than the one from Lytro. This also means that only the objects in a certain depth range can be clearly captured by the microlens. The acquired data has different properties according to their depth values. To address this problem, Raytrix uses three different microlenses such that the whole microlens array covers a larger depth-range. This setting greatly increases the resolution of the acquired data, but it also increases the complexity of analyzing and processing the data.

In the following sections, we show how these settings affect the formulation of the sampling model of the light-field cameras.

2.2 Sampling Models under Various Configurations

We demonstrate several sampling models under different camera configurations. We begin with the simplest case where the acquired data are direct measurements from the continuous light field with a moving pinhole camera. Then we investigate the influence of pixel size and the microlens array on the sampling model. By applying these models, we demonstrate some example sampling models of the light field cameras currently on the market.

2.2.1 Sampling model with pinhole camera

In our analysis, we use a moving pinhole camera behind a main lens to capture the light field, as shown in Figure 2.7. The moving pinhole-camera captures the spatial and angular information, whereas the main lens is the optical relay that adjusts the imaging process and the sampling kernel of the system. We assume the pinhole camera to be ideal and each pixel on its sensor to be infinitely small. Therefore, each pixel only captures one light ray for each point of the scene passing through the pinhole. Without the loss of generality, we carry out the analysis in the 2-D light field for the sake of clarity and simplicity.

The imaging process shown in Figure 2.7 is as follows: By moving a pinhole camera behind a main lens with a step size T_x between each acquisition, multiple 1-D images from the same scan line are stacked to form the 2-D light field. The step size T_x is the spatial sampling-period on the x dimension. As for the pinhole camera, its pixel size is defined as T_p, whereas its focal length is normalized to one. Thus T_p directly represents the angular sampling-period on the dimension p.

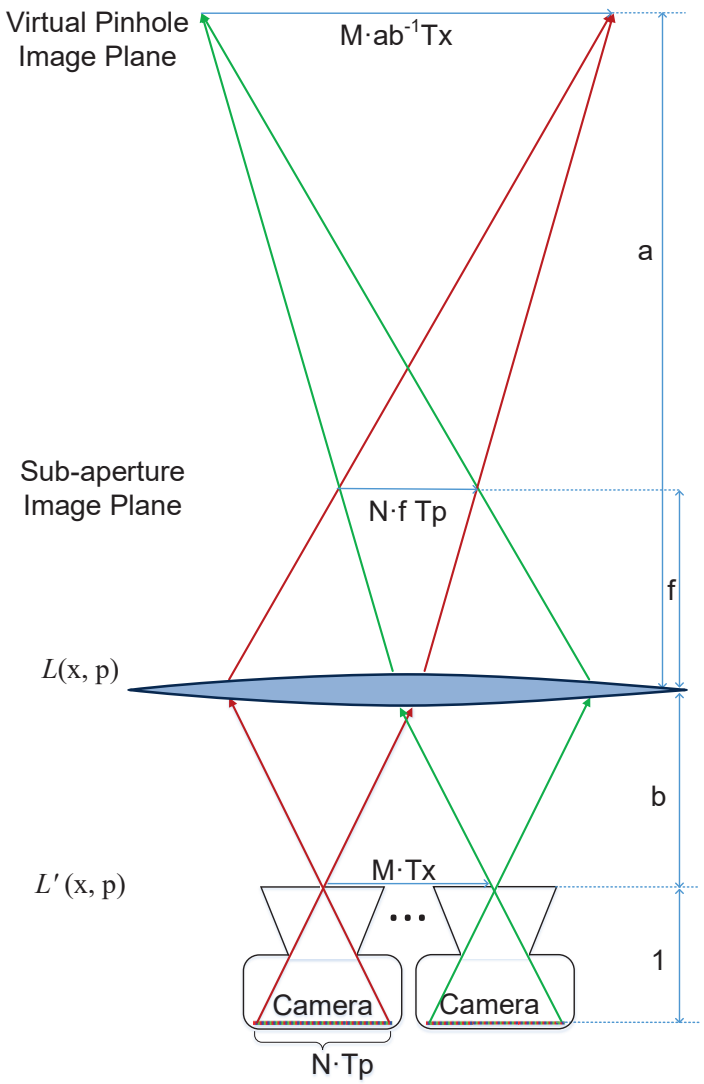

Figure 2.7: The setup of the constructed light field camera. A camera is moving behind the main lens with a step-size at T_x. The focal length of the moving camera is normalized to 1 and its sampling period is T_p. The moving camera is b meters behind the main lens.

The variable M and N represent the total number of discrete samples in the x and p dimension, respectively.

More specifically, each pinhole camera image is a set of light rays that converges at each pinhole inside the imaging system. Each set of light rays also converges outside the imaging system as if there was a virtual pinhole camera taking the picture. In addition, we can also view the light-field data in an alternative way as follows: we form a set of light rays by choosing those with the same directions from each pinhole camera. These parallel light rays converge at the focal plane of the main lens, which is usually referred to as the sub-aperture image plane.

Clearly, the baseline between the virtual pinhole images corresponds to the sampling period T_x, whereas the baseline between the sub-aperture images is proportional to the sampling period T_p. Usually the sub-aperture image is referred to as a set of samples in the spatial domain and the virtual pinhole camera image is referred to as a set of samples in the angular domain.

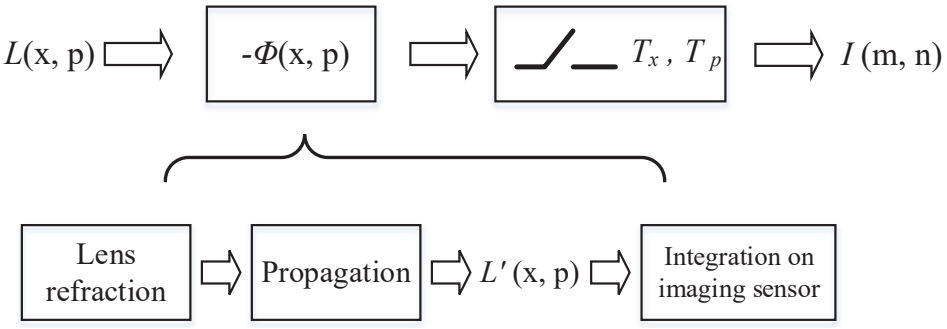

Figure 2.8: Sampling operator of plenoptic camera.

The overview of the sampling model is generalized in Figure 2.8, where the function $\phi(x,p)$ represents the sampling kernel and the vector (T_x, T_p) represents the sampling period in the x and p dimension, respectively. The light field $L(x,p)$ is defined on the plane of the main lens that is outside the camera, whereas $L'(x,p)$ is an intermediate light field defined on the plane of the moving pinhole-camera. Without considering the lens refraction and propagation, the acquisition of the discrete light-field data $I(m,n)$ is a direct measurement of the intermediate light field $L'(x,p)$. We further simplify the imaging process by assuming that each pixel captures only one light ray passing through the vanishingly small pinhole and derive the sampling process as follows:

$$I(m,n) = \langle L'(x,p), \delta(x/T_x - m, p/T_p - n) \rangle$$

where $x, p \in \mathrm{R}$ and $m, n \in \mathrm{Z}$.

The acquired discrete data $I(m,n)$ is a set of measurements of the light field $L'(x,p)$ defined on the pinhole-camera plane, which means $L'(x,p)$ is defined on a plane inside the light-field camera. To fully understand the property of the imaging system, we need to establish the relation between the discrete data and the continuous light field defined on the plane of the light field camera outside the imaging system as shown in Figure 2.7. We use the sampling kernel $\phi(x,p)$ to denote this process as shown in Figure 2.8.

There are two operating blocks between $L'(x,p)$ and $L(x,p)$: the refraction of the main lens and the propagation. By using the tools from ray transfer analysis, we formulate the refraction and propagation as two ray transfer matrices $\mathbf{A}_f$ and $\mathbf{A}_b$, respectively, as follows:

$$\mathbf{A}_b = \begin{bmatrix} 1 & b \\ 0 & 1 \end{bmatrix}, \quad \mathbf{A}_f = \begin{bmatrix} 1 & 0 \\ -f^{-1} & 1 \end{bmatrix},$$

where b denotes the propagation distance and f denotes the focal lens of the main lens. These matrices are applied directly in the ray space (x,p), which is defined by the ray location x and direction p that is the same definition as used in the light field.

In addition to these factors, the sampling kernel $\phi(x,p)$ is also affected by the property of the main lens. Here we assume the main lens to be perfect and the point-spread function to be a Dirac. Finally, the relation between the light field $L(x,p)$ defined on the main lens and the intermediate data $L'(x,p)$ is formulated as follows:

$$L'(\begin{bmatrix} 1 & b \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -f^{-1} & 1 \end{bmatrix} \begin{bmatrix} x \\ p \end{bmatrix}) = L([x\ p]^T),$$

where we use the vector form $[x\ p]^T$ to denote the two coordinates x and p for the sake of simplification. We use $\mathbf{A} = \mathbf{A}_f\mathbf{A}_b$ to represent the consecutive operations $\mathbf{A}_f$ and $\mathbf{A}_b$. Then the sampling model of the whole imaging system is formulated as follows:

$$\begin{aligned} I(m,n) &= \langle L'([x\ p]^T), \delta_{\mathbf{T}\cdot[m\ n]^T}([x\ p]^T))\rangle \\ &= \langle L(\mathbf{A}^{-1}\cdot[x\ p]^T), \delta_{\mathbf{T}\cdot[m\ n]^T}(\mathbf{x})\rangle \\ &= \langle L([x\ p]^T), \delta_{\mathbf{T}\cdot[m\ n]^T}(\mathbf{A}\,[x\ p]^T)\rangle \\ &= \langle L([x\ p]^T), \delta_{\mathbf{A}^{-1}\mathbf{T}\cdot[m\ n]^T}([x\ p]^T)\rangle. \end{aligned} \tag{2.2}$$

With the sampling model, we specify the sampling periods in both the spatial and angular dimension in Figure 2.9. We demonstrate the sampling pattern of the light field camera in both the intermediate light field $L'(x,p)$ and the actual light field $L(x,p)$ outside the camera. The actual sampling pattern in the world coordinates is a parallelogram grid that is determined by the operation matrix $\mathbf{A}$. The sampling grid is determined by

$$\boldsymbol{A}^{-1}\boldsymbol{T} = \begin{bmatrix} 1 & -b \\ \frac{1}{f} & 1-\frac{b}{f} \end{bmatrix} \begin{bmatrix} T_x & 0 \\ 0 & T_p \end{bmatrix} = \begin{bmatrix} T_x & -bT_p \\ \frac{1}{f}T_x & -\frac{b}{a}T_p \end{bmatrix},$$

where a is the plane of focus of the main lens and it can be calculated with the thin lens equation as

$$\frac{1}{a} + \frac{1}{b} = \frac{1}{f}. \tag{2.3}$$

The columns of the matrix $\boldsymbol{A}^{-1}\boldsymbol{T}$ correspond to the sampling period in terms of microlens and pixel, individually. The first column $[T_x\ \frac{1}{f}T_x]^T$ determines the distance between pixels under the same pinhole camera and the second column $[bT_p\ -\frac{b}{a}T_p]^T$ determines the distance between the same pixel under two consecutive pinhole cameras as shown in Figure 2.9.

The sampling pattern can be adjusted by changing the propagation distance b and the focal length f of the main lens. Compared to the traditional camera array, the light-field camera is a compact imaging system with a flexible sampling grid. Intuitively, the matrix $\mathbf{A}$ projects the pinhole-camera plane outside the imaging system, and the position of the virtual camera plane can be adjusted by the propagation distance b and the focal length f.

There are some alternative perspectives on the 2-D light-field data. The 1-D image $L(x)$ created by fixing the variable p is usually referred to as a sub-aperture image. It is a beam formed by selecting a single light ray in the same direction from each pinhole camera. Therefore the beam converges at the focal plane of the main lens. The baseline between neighboring sub-aperture images is determined by $T_p \cdot f$, which is also the angular resolution. Intuitively, a smaller baseline corresponds to a more refined acquisition for different view angles.

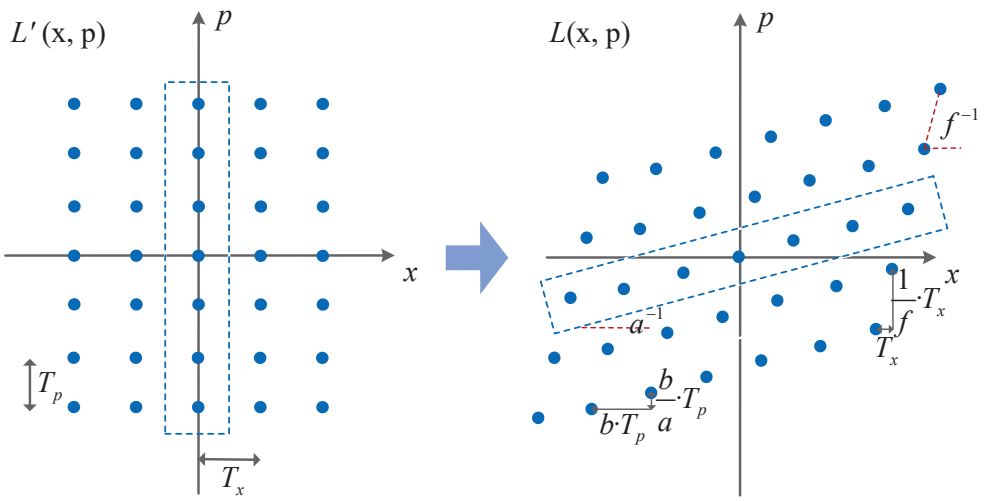

Figure 2.9: Sampling grid in the intermediate light field $L'(x, p)$ and the light field $L(x, p)$ outside the camera. The measurements within one rectangluar box represent one microlens image.

The 1-D image $L(p)$ created by fixing the variable x is referred to as a pinhole-camera image. The virtual pinhole-camera plane is determined by the propagating distance b and can be calculated with $a = (f^{-1} - b^{-1})^{-1}$. For an object located at a meters in front of the main lens, each virtual pinhole camera records a single spatial location. The baseline between the virtual pinhole cameras is $T_x \cdot ab^{-1}$ that also describes the spatial resolution.

2.2.2 Sampling model with microlens array

In practice, a microlens array is positioned in front of the sensor to replace the moving pinhole camera. Instead of capturing a single light ray, a bundle of light rays pass through the microlens and they are integrated by each pixel on the imaging sensor.

There are two factors to consider when analyzing the practical sampling model with a microlens array. First, the diameter of the microlens affects the amount of light rays integrated on the imaging sensor. Second, the pixel size is not infinitely small, and we use T_p to denote the pixel size.

Microlens array

As shown in Figure 2.10, we demonstrate the imaging process of the pinhole camera, the microlens focused at infinity and the microlens focused at a finite distance, respectively. One sample on the sensor behind the pinhole corresponds to one light ray in the space. One sampling point behind the microlens focused at infinity corresponds to a set of parallel light rays. In the last case, one sample corresponds to a set of light rays that are converging on the plane of focus at a distance a' and can be calculated with the thin lens equation directly. Here we use a' to represent the plane of focus of the microlens whereas we use a to represent the plane of focus of the main lens. The variable a' is used for modeling the blurring kernel whereas the variable a is

used in modeling the sampling grid of the light-field camera.

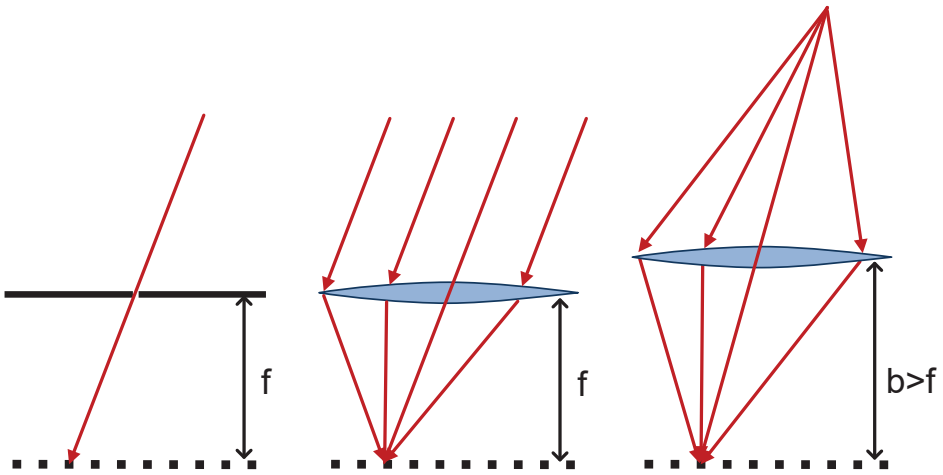

Figure 2.10: The difference among a pinhole, a microlens focused at infinity and a microlens in focus.

For each sampling device in Figure 2.10, we also show the corresponding sampling patterns of these imaging systems in the light field in Figure 2.11. For each pixel under the microlens, it integrates over a line slice in the light field. As the diameter of the microlens is also the sampling period in the x dimension, the coverage of the line slice is T_x in the x dimension.

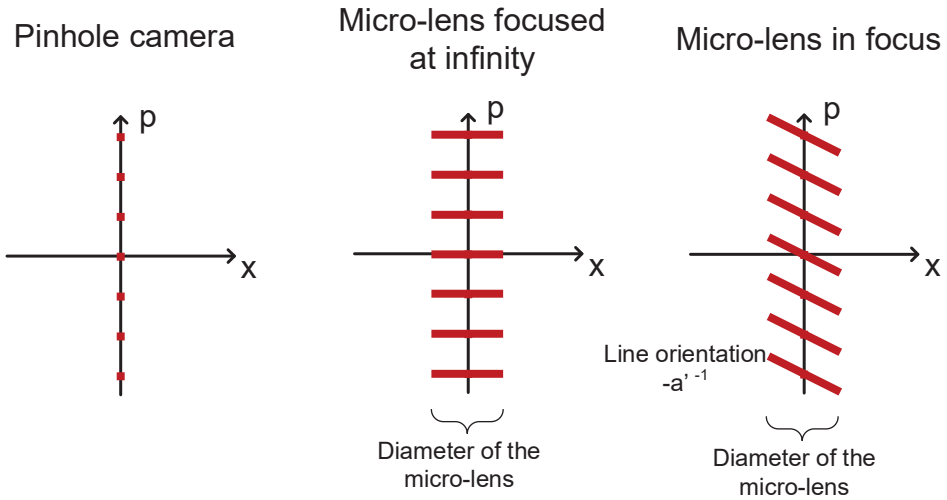

Figure 2.11: The sampling pattern in the 2-D light field for the pinhole camera, the microlens focused at infinity and the microlens in focus.

Furthermore, the slope of the line slice is determined by where the microlens is focused at. As shown in Figure 2.11, when the microlens is focused at infinity, each pixel integrates a bundle of parallel light rays. The set of light rays corresponds a horizontal line slice in the 2-D light

field. When the microlens is focused at a' meters away, each pixel integrates a bundle of light rays converging at a meters away. Then the set of light rays corresponds to a line slice with the slope at ${a'}^{-1}$.

We use an indicator function $\mathbb{1}_{\mathcal{D}}(x,p)$ to specify the integration domain. The integration domain for the microlens focused at infinity is defined as

$$\mathcal{D} = \{(x,p)|x \in [-\frac{T_x}{2}, \frac{T_x}{2}), p = 0\}, \tag{2.4}$$

whereas the integration domain for the microlens in focus is defined as

$$\mathcal{D} = \{(x,p)|x \in [-\frac{T_x}{2}, \frac{T_x}{2}), p = -\frac{1}{a'}x\}. \tag{2.5}$$

Therefore, the sampling kernel of the light field camera changes from the Dirac function to an integration over a line slice. The orientation of the line slice depends on the microlens' plane of focus.

Pixel size

We analyze the effect of the pixel size by following a similar methodology to the one used in the analysis of the microlens. As shown in Figure 2.12, we demonstrate the imaging process of the pinhole camera with a vanishingly small pixel-size and a finite pixel-size, respectively.

In the ideal case, as shown on the left in Figure 2.12, each pixel only captures one light ray through the pinhole, and they are direct measurements of the light field. In practice, as shown on the right in Figure 2.12, a set of light rays with similar angular information is integrated on one pixel. The measurement of each pixel corresponds to an integration at the angular domain p, over a vertical line segment in the 2-D light field.

We use the indicator function $\mathbb{1}_{\mathcal{D}}(x,p)$ to specify the integration domain as

$$\mathcal{D} = \{(x,p)|x = 0, p \in [-\frac{T_p}{2}, \frac{T_p}{2})\},$$

which is very similar to the formulation in Equation (2.4) and Equation (2.5). The integration areas are all extended from a single point to a line slice in the 2-D light field.

The analysis with a pinhole can be easily extended to a microlens. The line slices shown in Figure 2.11 become rectangles and parallelograms shown in Figure 2.13.

Examples of sampling models

Our analysis on the sampling models focuses on the sampling periods and sampling patterns. As introduced in Section 2.3, there are two types of light field camera products on the market: Lytro and Raytrix. As they adopt two different focusing mechanisms for the microlens array, their sampling models are quite different. Lytro is usually referred to as the plenoptic camera or plenoptic 1.0, where the microlens array is focused at infinity. Raytrix is usually referred to as the focused plenoptic camera or plenoptic 2.0, as their microlens array is not focused at infinity. We show how the focusing mechanisms affect the sampling models in the following sections.

Inside the Lytro camera, the distance between the microlens array and the sensor is the focal length of these microlenses. Therefore, each pixel captures a bundle of parallel light rays. As shown in Figure 2.11, each pixel value is the integration on a horizontal slice in the light field.

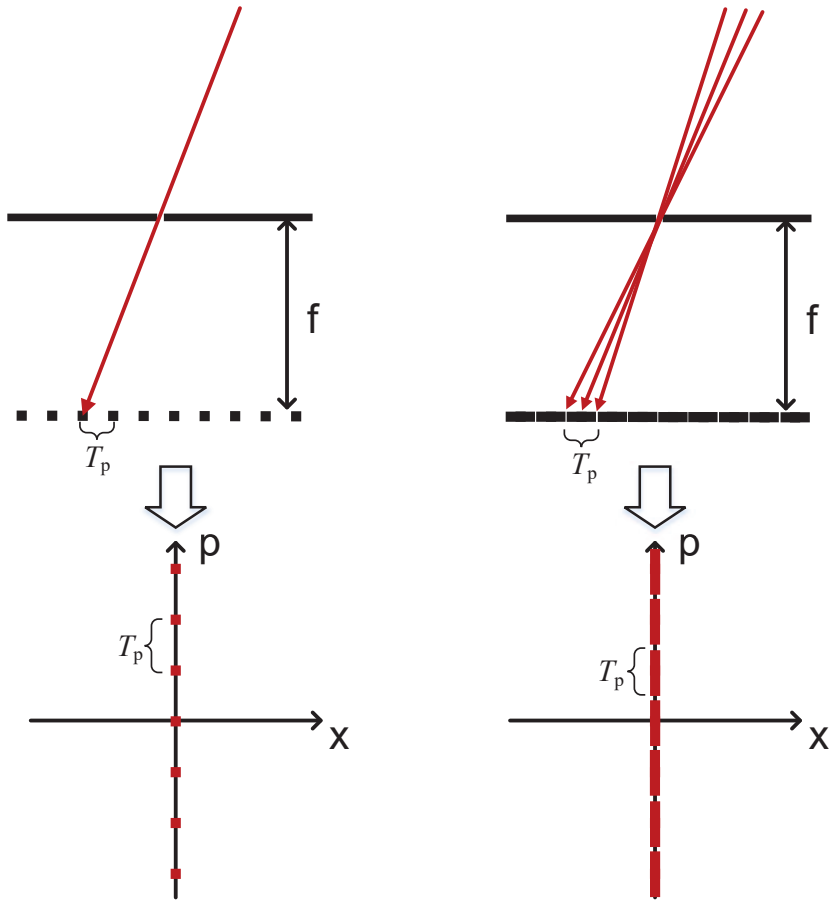

Figure 2.12: The sampling pattern of a pixel in practice. On the left we show the sampling setup of a pinhole camera with an vanishingly small pixel-size and its sampling pattern in the 2-D light field. On the right we show the sampling setup with a pixel size that equals to the sampling period T_p and its sampling pattern in the 2-D light field.

By considering the pixel size T_p, the integration area becomes a rectangle as shown in Figure 2.13. The integration area represented with the indication function $\mathbb{1}_{\mathcal{D}}(x,p)$ where

$$\mathcal{D} = \{(x,p)|x \in [-\frac{T_x}{2}, \frac{T_x}{2}),\ p \in [-\frac{T_p}{2}, \frac{T_p}{2})\}$$

By using Equation (2.2), the imaging process for the Lytro camera can be formulated as follows:

$$\begin{aligned} I(m,n) &= \langle L([x\ p]^T), \delta_{\mathbf{A}^{-1}\mathbf{T}\cdot[m\ n]^T}([x\ p]^T) * \mathbb{1}_{\mathcal{D}}([x\ p]^T \rangle, \\ \textit{where} & \\ \mathcal{D} &= \{(x,p)|x \in [-\frac{T_x}{2}, \frac{T_x}{2}), p \in [-\frac{T_p}{2}, \frac{T_p}{2})\}. \end{aligned}$$

The sampling kernel is determined by the sampling period in the x and p dimension, without

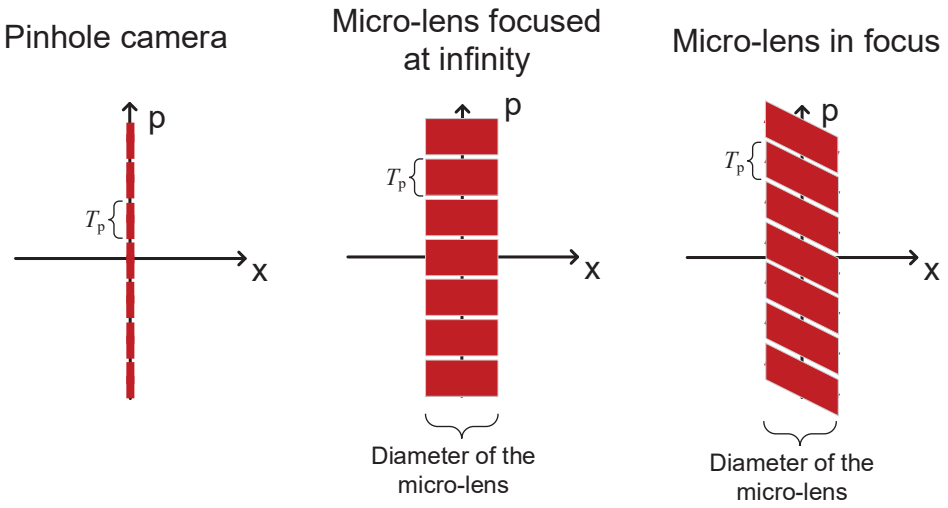

Figure 2.13: The sampling pattern in the 2-D light field for the pinhole camera, the microlens focused at infinity and the microlens in focus with a normal pixel size.

considering the point-spread function of the main lens and microlens. This sampling kernel however is defined on the intermediate light field $L'(x,p)$. To deduce the sampling kernel in $L(x,p)$, the ray transfer matrix $\mathbf{A}$ should be applied to the original blurring functions: the rectangular-shaped and parallelogram-shaped function for Lytro and Raytrix, respectively.

The drawbacks of the Lytro camera is that the image behind each microlens is blurry because the microlens is focused at infinity. The plenoptic camera with a focused microlens is proposed and implemented in the Raytrix camera.

Inside the Raytrix, the distance between the microlens array and the sensor is larger than the focal length of these microlenses. Each pixel focuses at a plane inside the camera as shown in Figure 2.10. As each microlens is in focus, they can only capture the scene within the focused range. The objects behind or in front of the plane of focus are still blurry, similar to those captured by the microlens inside a Lytro. To increase the focus range, three different types of microlenses are used to capture objects at different depths. Here we demonstrate only the sampling model for one of the microlenses as

$$\begin{aligned} I(m,n) &= \langle L([x\ p]^T), \delta_{\mathbf{A}^{-1}\mathbf{T}\cdot[m\ n]^T}([x\ p]^T) * \mathbb{1}_{\mathcal{D}}([x\ p]^T \rangle \\ \textit{where} & \\ \mathcal{D} &= \{(x,p)|x \in [-\frac{T_x}{2}, \frac{T_x}{2}), p - \frac{1}{a'}x \in [-\frac{T_p}{2}, \frac{T_p}{2})\}. \end{aligned}$$

Clearly, the sampling grids of the Lytro and Raytrix cameras are almost the same, but their sampling kernels are quite different. The sampling grid is determined by the sampling period T_x in the x dimension and T_p in the p dimension. These two variables are determined by the diameter of the microlens and by the pixel size that is normalized by the focal length of the microlens, respectively.

As for the sampling kernels, the major difference is in their shapes. The sampling kernel of Lytro is a rectangle in the 2-D light field. When the sampling kernel is applied to the light field,

the blurring effect is almost uniform across the scene.

The sampling kernel of Raytrix is a parallelogram. When the slope of the parallelogram matches the slope of the lines in the light field, the acquired microlens image is sharp and in focus. These lines correspond to objects at the focused plane of the microlens. There are three types of microlenses and they are focused at different distances in order to extend the range of focus. The data from Raytrix with three sampling kernels requires post-processing to be fully used. The parallelogram shape of the sampling kernel increases both the resolution of the microlens image and the complexity of processing the data.

2.3 Experimental Light-Field Camera

Using the sampling model, we demonstrate the relation between camera designs and the sampling periods in the light field. To verify the sampling model, we construct a light-field camera with flexible sampling periods in both angular and spatial domains. This way, we have a better understanding of the sampling model and we can acquire real-life data to work with. In the following chapter, we will use the constructed camera to acquire light fields and recover depth maps of the scene from the data to test the proposed light field registration algorithms.

2.3.1 Basics of light field cameras

As mentioned in Section 2.2, a standard light-field camera is built by adding a microlens array in front of the imaging sensor. The integrated light rays are then separated and recorded by the sensor under the microlens array. Once the microlens array is mounted, the sampling period in the spatial dimension can no longer be modified. And as the distance between the imaging sensor distance and the microlens array cannot be adjusted once mounted, the angular resolution of the acquired data cannot be modified either, as our analysis of the sampling model explains. Instead, we construct a prototype of a light field camera that has a more flexible configuration in various sampling periods.

The two key elements in a standard light-field camera are the main lens and the microlens array. Instead of the microlens array, we use a simpler realization in which a pinhole camera moves behind the main lens. In this setup, we can change the sampling period in the angular domain by changing the pinhole camera's focal length and the sampling period in the spatial domain by changing its moving step size. More specifically, we use a linear stage to move a digital single-lens reflex (DSLR) camera of which the aperture is set to be small enough to be emulated as a pinhole camera. Therefore, our sampling pattern is similar to Lytro, instead of being similar to Raytrix.

Capturing similar light fields as Lytro

The light field data acquired with Lytro has a 300×300 microlens array under which there are 11×11 images. Therefore, the 4-D light field data has the following dimensions: $11 \times 11 \times 300 \times 300$. In the proposed setup, the DSLR camera has to take 300×300 photos for one light-field data. After the photo is taken with DSLR, an 11×11 area is cropped from it. Clearly, the required amount of photos is too large to be practical, whereas many samples in each photo are wasted by cropping. Therefore, we look into the duality of the light field and propose an efficient acquisition strategy to construct an experimental light-field camera.

2.3.2 Duality of the sampling model

To increase the efficiency of the prototype, we first revisit the sampling model of the light-field camera. The light field can be seen as a plane of sub-aperture images by fixing the variable (p, q). As shown in Figure 2.14, the 4-D data is presented as a (p, q) grid of images that are referred to as the sub-aperture images. We show two sub-aperture images on the sides, where we can observe the horizontal disparity.

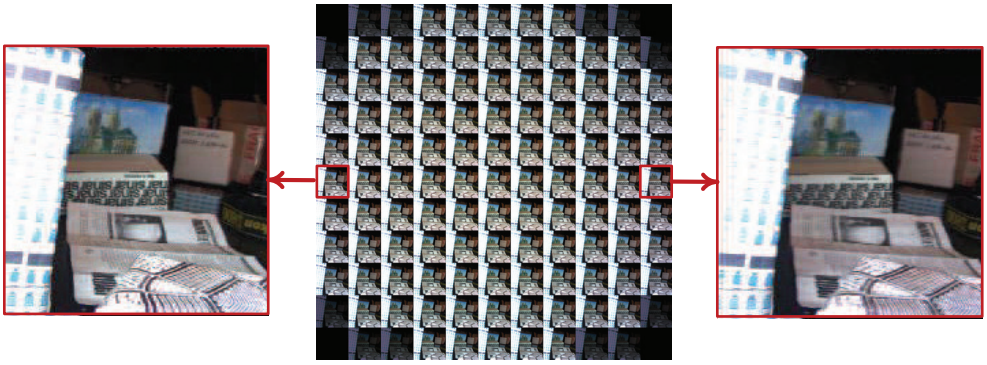

Figure 2.14: An example of the 4-D light field arraged as a (p, q) grid of the sup-aperture images. We show two sub-aperture images on both sides.

The 4-D can be seen as a plane of virtual pinhole-camera images by fixing the variable (x, y). As shown in Figure 2.15, the 4-D data is presented as an (x, y) grid of pinhole-camera images, on the left. We show a magnified area in the 4-D data, where we can observe the pattern of the pinhole-camera images.

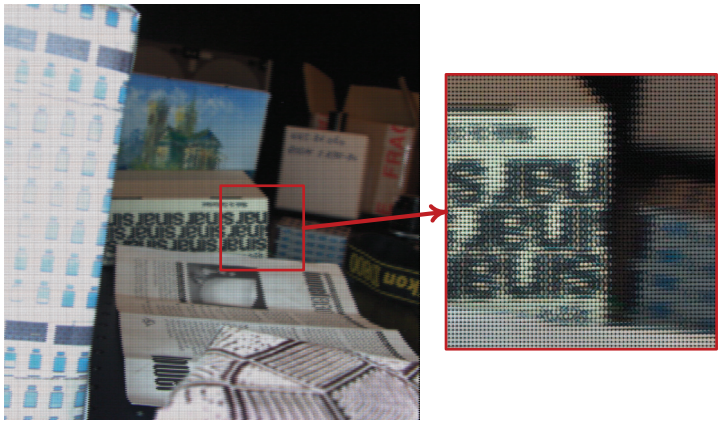

Figure 2.15: An example of the 4-D light field arranged as a (x, y) grid of the virtual pinhole-camera images. We show a magnified area such that the pattern of the virtual pinhole-camera images can be observed.

Let us take a closer look at the setup of a standard light field camera in Figure 2.16 on the left. A moving pinhole camera is positioned b meters behind the main lens. The acquired images

are equivalent to the images taken at virtual pinhole camera plane that is a meters in front of the main lens where $a = (f^{-1} - b^{-1})^{-1}$.

We use the term virtual pinhole-camera images to refer to this data, as each image corresponds to a set of light rays that converge at the virtual pinhole camera plane a meters in front of the main lens. The virtual pinhole-camera plane is usually very close to the object. Each image captures the angular variation of the limited spatial range. Therefore the pinhole-camera image is referred to as a set of the samples in the angular domain.

In another arrangement of the light-field data, we choose light rays with the same angular values from each pinhole camera. All the light rays have the same direction and converge at f meters in front of the main lens. These images are referred to as sub-aperture images. A sub-aperture image is usually referred to as a set of samples in the spatial domain as the light rays from each sub-aperture image covers a large field of view.

Although pinhole-camera images and sub-aperture images are referred to as angular samples and spatial samples, they are fundamentally two sets of light rays with converging planes at a meters and f meters in front of the main lens. Whether they are samples in the spatial or the angular dimension largely depends on the positions of the converging planes relative to the scene.

In our setup, we propose to reverse the purpose of the two sets of images by using the virtual pinhole-camera images to record spatial samples and by using the sub-aperture images to record angular samples. For example, to capture the data with the same dimensions as that of a Lytro, we use the pinhole camera to capture 11×11 photos with a resolution of 300×300 pixels. The proposed setup is shown on the right in Figure 2.16.

To ensure that the two imaging systems have the same resolution, the parameters of the proposed system have to satisfy the following constraints:

$$\begin{aligned} M_1 &= N, \\ N_1 &= M, \\ f_1 \cdot T_{p1} &= \frac{a}{b} T_x, \\ \frac{a_1}{b_1} T_{x1} &= f \cdot T_p. \end{aligned}$$

To clearly establish the relation of the proposed setup to the standard light-field camera, we align the sub-aperture image plane and virtual pinhole-camera plane in these two imaging systems. The alignment has to satisfy the following equation:

$$a - f = a_1 + f_1.$$

Note that the distance between the main lens and the pinhole camera is shorter than the focal length of the main lens. Therefore the virtual pinhole-camera is actually behind the plane of the pinhole camera. By carefully choosing the focal length of the main lens and adjusting its distance to the pinhole camera, these two imaging systems can be aligned perfectly.

2.3.3 Light field camera prototype

In the proposed setup, we use a DSLR camera with a 105 mm macrolens and a linear stage as shown in Figure 2.17. There are three rails in total and two of them are used to move the DSLR camera in horizontal and vertical directions. The third one is used to move the object such that we can capture multiple light fields by translation and create data-sets for light field

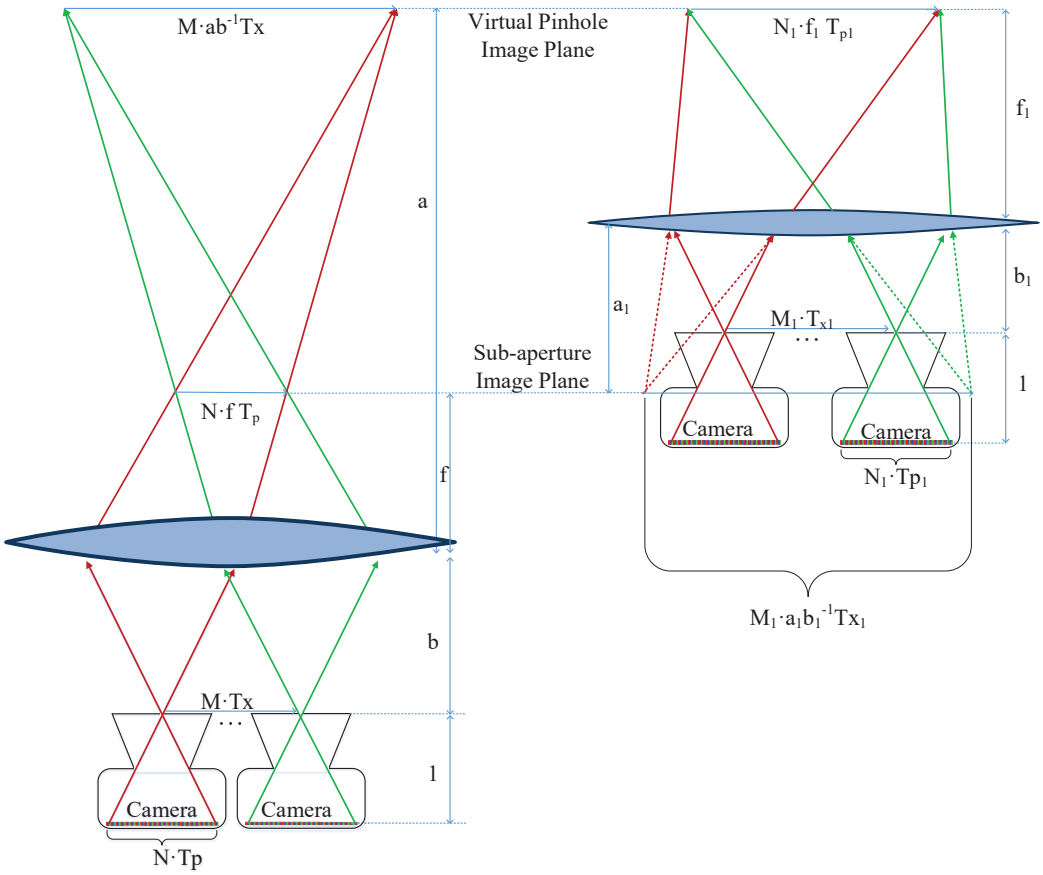

Figure 2.16: The comparison between a standard light field imaging system on the left and the proposed light field imaging system on the right. We align the image planes which sample the angular and spatial dimensions, respectively from the two imaging system. On the left, the sub-aperture image is a set of light rays which are from each pinhole camera. The virtual pinhole camera image is the image taken by each pinhole camera. On the right, the sub-aperture image is the image taken by the pinhole camera whereas the virtual pinhole camera image is a set of light rays from each pinhole camera.

registration. We position a main lens between the linear stage and the object. The diameter of the main lens is 13.3 cm and its focal length is 500 mm. The position of the main lens can be adjusted in all directions directly by the knobs on the main lens.

Here we still use T_x and T_p to denote the moving step size of the DLSR camera and the pixel size normalized by the camera's focal length respectively. As discussed in Section 2.2, the baseline between the sub-aperture images corresponds to the sampling period in the angular domain. For the proposed setup, the baseline is calculated as $T_x \cdot ab^{-1}$, which depends on the moving step size and the distance between the main lens and the DSLR camera. The baseline between the virtual pinhole camera images is calculated as $T_p \cdot f$ where f denotes the focal length

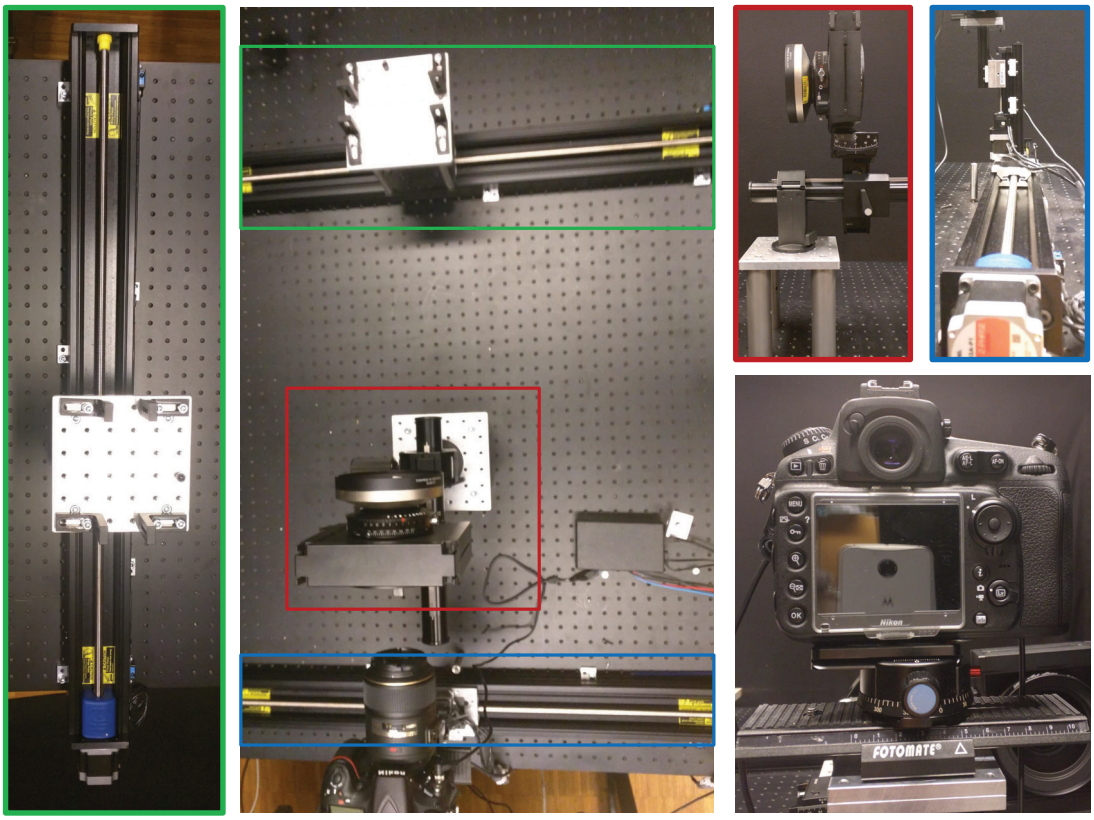

Figure 2.17: Assembly of the light-field camera. The middle figure shows the overview of the system. There are three parts: the camera and the x-y linear stage in the blue rectangle; the main lens in the red rectangle; the third linear stage to move the target to capture in the green rectangle.

of the main lens.

Clearly, the sampling periods in the angular and spatial dimensions are reversed for the proposed light field camera setup. We adjust the angular resolution of the light field by changing the moving step size T_x on the linear stage and positioning the main lens. As for the spatial resolution of the light field, we adjust the focal length of the DSLR camera to change T_p. The specific parameters of the constructed light field camera are shown in Table.2.1.

Parameters of the constructed light field camera.	
T_x	$5\mu m$
T_p	$5\mu m$
DSLR focal length	105 mm
Main lens focal length	500 mm

Table 2.1: Parameters of the imaging system

We show some example light fields in Figure 2.18. We apply different sampling periods for different scenes. For scenes with high frequency textures, we increase the spatial resolution. For scenes with fine geometric structures, we increase the resolution in the angular domain. We also show an example of the light field from metal with a complex reflectance function. The high angular resolution well preserves the high frequency components in its reflectance function.

2.4 Conclusions

In this chapter, the foundation has been established for the models and methods used in this thesis. Under the framework of geometric optics, the imaging process of the light field camera can be well analyzed by applying a ray-transfer-matrix analysis. We have proposed a simple sampling model with a moving pinhole camera and then extended the model with the analysis of microlenses and pixel sizes. The sampling model is a useful tool for understanding and utilizing the light-field camera and the acquired data. By using the proposed sampling model, we constructed a light-field camera prototype with flexible sampling periods in both spatial and angular domains. In the following chapters, it serves as a powerful experimental tool for depth recovery and registration algorithms.

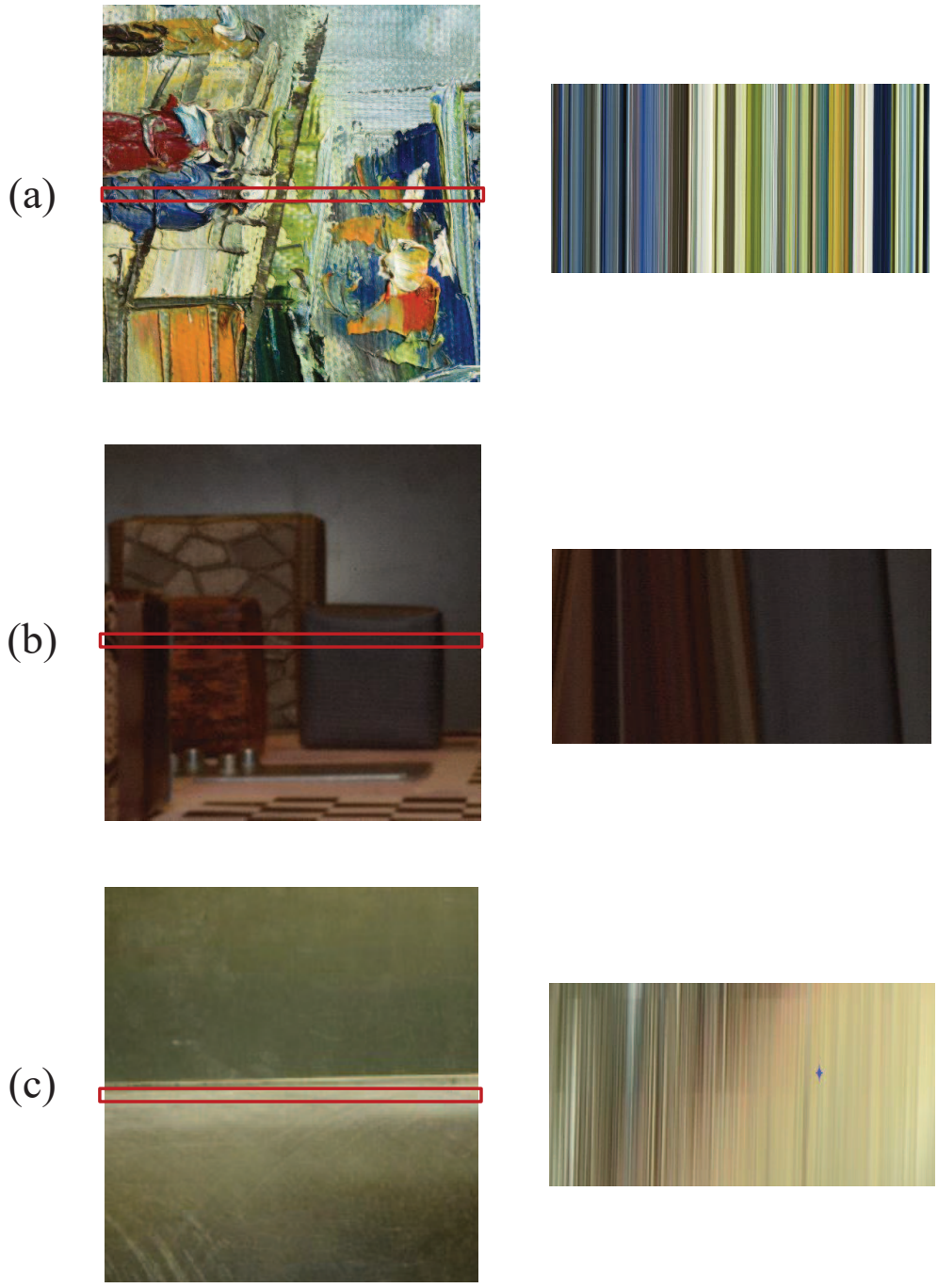

Figure 2.18: Some examples of the acquried light fields. The images on the left have a resolution of 512×512 pixels and the 2-D light fields of the central row of the left images have a resolution of 256×512 pixels, which is a very high angular resolution. More specifically, row (a) shows a light field of an oil painting. Row (b) shows a scene with multiple objects postioned in a range of 50 centimeters. Row (c) shows a small metal plate where we can observe a dramatic intensity-change of the 2-D light field in the angular dimension.

Chapter 3

Depth Recovery from Surface Light-Fields

From the side, a whole range; from the end, a single peak:
Far, near, high, low, no two parts alike.
Why cant I tell the true shape of Lu-shan?
Because I myself am in the mountain.

Su Shi, translated by Burton Watson

The primary goal of the work presented in this chapter is to achieve high-accuracy depth recovery results from light-field data. We first give a short review of the depth-recovery problem and the state-of-the-art algorithms in Section 3.1. In Section 3.2, we propose a concept of a low-dimensional surface light-field that is the underlying model of the high-dimensional light field. The surface light-field model is the result of our discussion about the sampling model of the light-field camera and the analysis of the depth resolution of the light-field camera. We exploit the properties of surface light-fields to increase the accuracy of the depth-recovery algorithm in Section 3.3. We formulate the depth-estimation problem as a problem of signal recovery with samples at unknown locations. Finally, we demonstrate the experimental results in Section 3.4. The reconstruction results of our algorithm achieve state-of-the-art performance on public data-sets. We also test the proposed algorithm with our custom light-field camera on a 3-D printed target with fine geometric structure, as well as on oil paintings.

3.1 Introduction and Related Work

Although light-field cameras originate from a concept of integral photography introduced by Ives and Lippmann over 100 years ago, they have regained popularity recently after the development of consumer light-field cameras. The first commercial product was released by Raytrix in 2011, and in 2012 the first consumer product was released by Lytro. Light-field cameras capture each light ray with its radiance, direction and 3-D position, respectively. Compared to photos taken with standard cameras, these acquired data contain much more information to describe and reconstruct the scene, which is usually referred to as the light field [32] or lumigraph [32].

Depth reconstruction is a challenging computer-vision problem that has been studied for more than three decades. Light-field cameras offer new solutions to this problem because they can perceive the depth information of the scene directly. Despite the compact form of a light-field camera, the acquired data can be seen as a set of photos captured from densely and regularly placed cameras. Take Lytro as an example, its acquired light field is equivalent to a set of photos captured with a 11×11 camera array. When these cameras are sufficiently dense, any point in a given 3-D space is projected in the light field as a line with its orientation determined directly by its depth. By analyzing the line structure in the light field, the depth information of the scene can be fully recovered.

In this Chapter, we see the depth recovery problem as an inverse problem, in which the forward operator is the imaging process of the light field. We propose to reverse the forward operator and recover both the geometry structure and the scene appearance. Simply put, the traditional photos and light fields are both observations of the scene appearance, modulated by its geometry structure. It is almost impossible to reverse the imaging process with a single photo. However, with the additional sampling dimension in the light field, we see the possibility of fully separating depth information and scene appearance from each other.

3.1.1 Related work

The concept of light field was originally introduced by physicists who described the flow of light rays as a field. It is a seven-dimensional function that describes each light ray in the scene with its wavelength, 3-D location, direction and time. The dimensionality of the light field is reduced to 4-D in [32] by Levoy and Hanrahan with the name 4-D light field, and in [25] by Gortler et al. with the so called name Lumigraphs, respectively. They both parameterize the light rays by their intersections with two pre-defined planes parallel to each other. The 4-D light field is actually the radiance function that records all the light rays' intensities and directions, therefore their trajectories in the 3-D world.

The 4-D light fields can be seen as a set of photos captured by a camera that moves on a camera plane. In any two given photos, one point in the 3-D space corresponds to a stereo pair with a disparity determined by its depth and the baseline between these two photos. In the light field, the baseline becomes a variable and it linearly determines the disparity for a given point. This becomes apparent when considering the 2-D light field, which is also known as the epipolar plane images (EPIs) in the computer vision community [12]. The points in the 3-D space are projected onto the 2-D light field as lines with different slopes that are determined by their depth values. When these points are on a Lambertian surface, the corresponding lines have constant colors. The Lambertian assumption makes the lines straightforward to detect, thus this assumption is widely used in the work related to EPIs.

There has been much work on scene reconstruction by exploiting this kind of line structure. One of the first approaches by Bolles et al. [12] uses subsequent line fitting in EPIs to reconstruct 3-D structure. Later Baker et al. use zero crossings of the Laplacian to achieve a similar goal [4][5]. The main idea behind these methods is to use the local derivative in the 2-D light field or the structure tensor in the 4-D light field to estimate the corresponding depth value. Line fitting as mentioned in [12] or minimizing a weighted path in [38] are both ways for extracting lines of constant intensities through the 2-D light field.

To further emphasizing the line structure, many algorithms try out a set of pre-defined orientations in a discrete fashion, in order to achieve more stable depth-recovery results. Criminisi et al. use an iterative method to extract groups of EPI-lines that they call EPI-tubes and these EPI-lines have the same slope that is a discrete set defined by the boundaries of the depth [18]. Berent et al. use a region competition method with active contours to segment the EPI-tubes and enforce correct occlusion ordering [9]. Kim et al. measure the color variance with a modified Parzen window estimation using an Epanechenikov kernel [30]. Tosic et al. design a special kernel with assumed depth in the scale-depth space and apply it to light fields to find locations at the assumed depth [48].

Wanner and Goldluecke extend the line structure in the 2-D light field to a tensor structure in the 4-D light field, and their algorithm obtains robust local-depth estimations and their reliability [51]. They perform depth labeling to the full entry of the light field and then enforce a consistent visibility by restricting the spatial layout. They further extend their work in a variational framework and apply it to light field super-resolution [52].

However, most of these methods only recover the depth maps with the discrete values defined by a finite set of orientations, which largely limits the resolutions of the depth values. Meanwhile, the assumption of the constant color intensity along the lines also constrains the size of the depth map to be equivalent to the spatial resolution of the light field. The spatial resolution of a light-field camera is determined by a microlens array positioned in front of the sensor. The resolution of the microlens array is quite limited compared to the imaging sensor of a standard camera. Therefore, the resolution of the recovered depth map is also quite limited.

Unlike the modeling in the previous methods, we model the acquired 4-D light fields as 2-D continuous texture signals that are "painted" on the scene surfaces, thus are modulated by the depth map of the scene. Furthermore, we see the depth recovery problem as an inverse problem, in which the forward operator is the imaging process of the light field. We show that the inverse problem is fundamentally a signal recovery problem in which a band-limited signal is sampled at unknown locations. This type of problem is first addressed in [49] in the case of discrete-time data and then discussed in the case of signals in continuous time domain in [13]. We propose a practical algorithm that can recover the depth map and the texture signal at the same time. By using a Fourier series to represent the light field in a lower dimension, we are able to recover the depth map with more refined details for the full entries of the light field. This kind of representation also shows great potential in synthesizing and compressing the light field.

3.2 Surface Light-Fields

In this section we investigate how to improve the precision of the depth-recovery algorithm from light fields and propose to use a surface light-field for depth recovery. The surface light-field is a 2-D (or 1-D) signal mapped from the 4-D (or 2-D) light-field data and it has great potential

to improve the performance of depth-recovery algorithms.

We first discuss the reasons for using the surface light-field from the perspective of the depth resolution of a stereo system and a light-field camera. Then we present a model that describes the scene as texture signals "painted" on the surfaces. We use this model to analyze the properties of surface light-fields and finally establish the relation between surface light-fields and the standard light-fields.

3.2.1 Motivation

How to increase the precision of the depth-recovery algorithms? To answer this question, we first look into the depth resolution of a standard stereo system. Then we extend the analysis of the stereo system to a light-field camera and the light-field data.

The simplest way to increase the depth resolution is to move the observing camera closer to the object. There are indeed other factors that have to be taken into consideration. But the light-field can be seen as a set of photos taken by virtual cameras whose positions can be flexibly adjusted by the camera parameters of the light-field camera. Thus we exploit this property to increase the depth resolution and further propose a surface light-field for depth recovery.

Depth resolution analysis of a stereo vision system

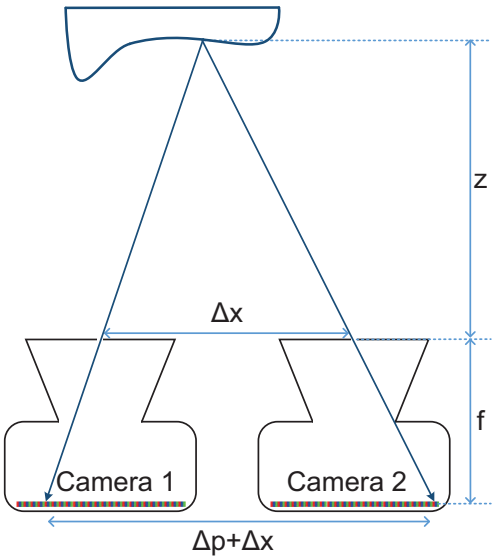

Figure 3.1: A standard stereo vision system. The baseline between the two cameras is Δx, whereas the disparity is Δp.

A standard stereo vision system is shown in Figure 3.1. Two cameras with the focal lengths f are positioned by a translation of Δx that is perpendicular to the optical axis. The translating distance Δx is also referred to as the baseline between these two cameras. We use the variable z to denote the depth value of the point being observed in the Figure. Then the depth can be

estimated by

$$z = f\frac{\Delta x}{\Delta p},$$

where Δp represents the disparity on the imaging sensor.

Given the camera setup, the depth resolution T_z is determined by the minimum disparity T_p that is also the pixel size normalised by the focal length f as follows:

$$T_z = \frac{z^2}{\Delta x} T_p. \tag{3.1}$$

Intuitively, a higher depth resolution requires a larger baseline Δx, a smaller pixel size T_p, and a smaller depth z. As the pixel size T_p and baseline Δx are usually fixed once the system is set up, we turn to the variable z for improving the depth resolution.

The depth z is defined by the distance between the object and the imaging system, thus can be adjusted by moving the object or the imaging system. Two factors prevent us from positioning the cameras too close to the object. First, cameras usually have a minimum focusing distance. Second, even when we use a macro lens that is specifically designed for taking photos of objects at a close range, we also need to take care of the illuminations of scene. Especially for an indoor environment, additional devices are required for close-range illumination as the camera may block the ambient light and even cast shadows on the object.

However, when we use the concept of virtual pinhole-cameras, which is a way to describe data captured by a light-field camera, the constraints on the minimum value of the variable z is no longer a problem. For a set of virtual pinhole-cameras, the variable z is defined as the distance from the object in the scene to the virtual pinhole-camera plane that is also the focused plane of the light-field camera. Thus, by changing the plane of focus of the light-field camera, the variable z can easily approach 0 without any lighting and focusing problems.

Depth Resolution analysis of light-field camera

We deduce the depth resolution of the light-field camera by using the virtual pinhole cameras for depth estimation. The light-field camera setup and related parameters are illustrated in Figure 3.2. It is implemented by moving a camera behind the main lens. The step size of the moving pinhole-camera is defined as T_x, whereas the sampling period of each pinhole camera is defined as T_p. The total number of samples is $M \cdot N$ where M denotes the number of samples in x and N denotes the number of samples in p dimension, respectively.

Each image captured by the camera is a sub-aperture image in the light field and corresponds to one position on the sub-aperture image plane as shown in Figure 3.2. The virtual pinhole-camera plane is at the focal plane of the main lens, whereas the sub-aperture image plane is a meters behind the main lens. The position of the sub-aperture image plane is determined by the thin lens Equation (2.3) mentioned in the Chapter 2. Our experimental setup of the light-field camera is shown in Figure 3.3.

Here are the related variables in the light-field camera:

— The baseline Δx between the neighboring virtual pinhole-cameras is fT_p.

— The minimum disparity is actually the sampling period of the virtual pinhole-camera, which can be calculated as $T_x f^{-1}$.

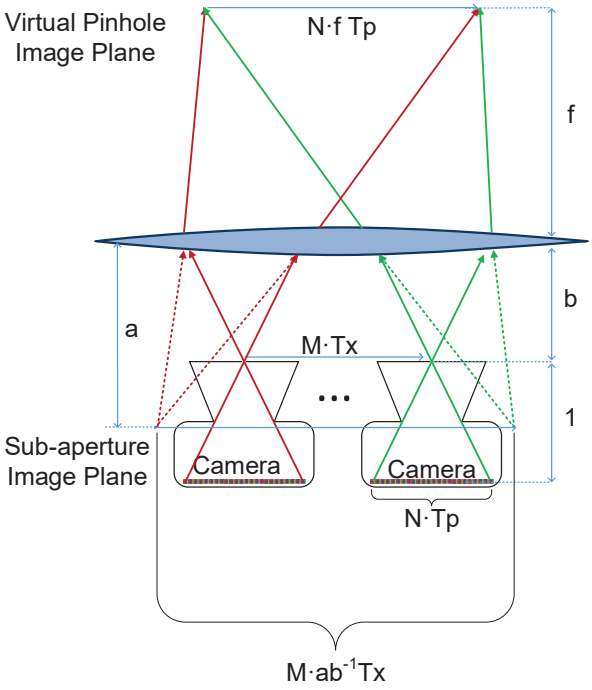

Figure 3.2: The setup of the constructed light-field camera. A camera moves behind the main lens with a step-size at T_x. The focal length of the moving camera is normalized to 1 and its sampling period is T_p. The moving camera is b meters behind the main lens. We use the green and red solid-lines to illustrate the paths of the light rays captured by different cameras, respectively. As the main lens changes the light paths, the acquired dataset is equivalent to a set of photos taken at a meters behind the main lens, by a set of virtual cameras. We use the dotted lines to represent light rays captured by the virtual cameras.

Then we put these variable into the general depth resolution model (3.1) and derive the depth resolution of the constructed light-field camera is as follows:

$$\Delta z = \frac{z^2}{fT_p} T_x f^{-1} = \frac{z^2}{f^2} \frac{T_x}{T_p}. \tag{3.2}$$

From the formulation (3.2) of the depth resolution, we can conclude that the three key factors that increase the resolution are: (1) the distance between the object and the virtual pinhole-camera plane, (2) the main lens' focal length and (3) the step size of the camera motion.

However, each neighbouring virtual pinhole-cameras can only observe the scene within a depth range that is determined by the sensor size. In practice, the sensor size is described by the number of discrete pixels. To illustrate the depth range, we represent the depth value in a discrete manner as follows:

$$Z_n[i] = f \frac{n \cdot b}{i \cdot \Delta d} = f^2 \frac{n \cdot T_p f_D^{-1}}{i \cdot T_x},$$

where i represents the disparity index in the pinhole-camera images and n represents the baseline

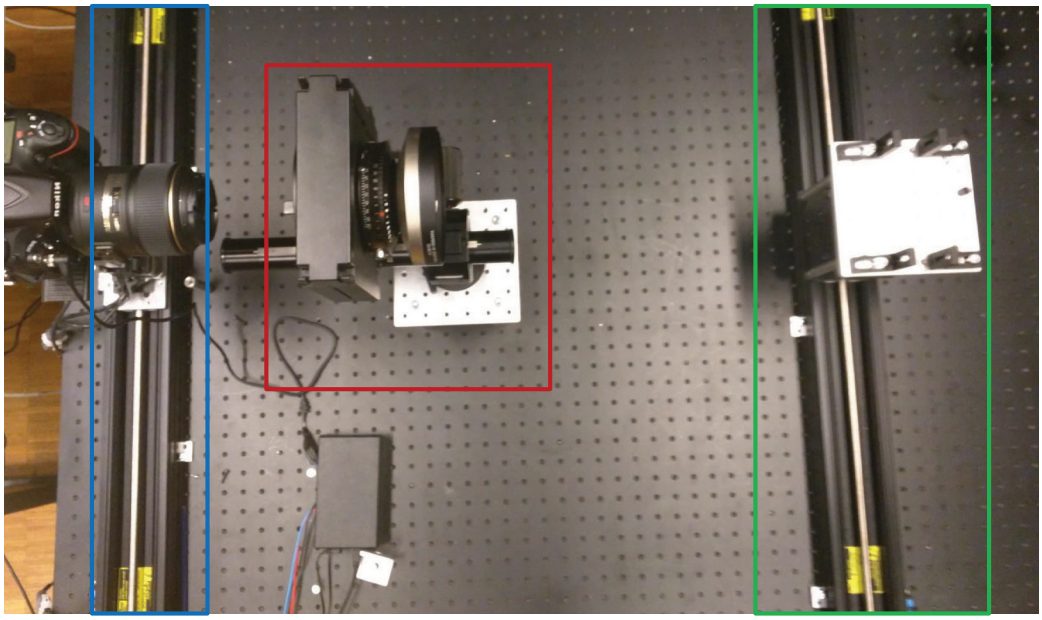

Figure 3.3: A custom light-field camera. It is constructed with a DSLR camera controlled by a linear stage (in the blue rectangle) mounted behind a main lens (in the red rectangle). The object then is positioned on anther linear stage (in the green rectangle).

between the virtual pinhole cameras. As z is defined as the distance between the object and the virtual camera-plane, it can be a negative value. More specifically, z is a positive value when the object is in front of the virtual pinhole-camera plane and a negative value when the object is behind the virtual pinhole-camera plane. These two cases are symmetric, thus we only address the scenario when z is positive.

The disparity index i begins at 1 and its maximum value depends on the sensor size of the camera. Take the constructed light-field camera as an example, the maximum translation is 6.4 mm and the maximum index is 1280 when the step size is set to be 5 μm. As for the index n, the default value is 1 when we only use the neighboring virtual pinhole-cameras. Here to increase the depth range of the acquisition, we set $n = 1, 2, 3$. Then we show the depth resolution and the depth range for this discrete setup in Figure 3.4. By using a maximum of 4 neighboring virtual pinhole cameras, we can achieve a 15 μm resolution within a 1 cm range.

From the depth-resolution analysis of a light-field camera, we conclude that fine depth-resolution can be achieved by projecting the virtual pinhole-camera plane close to the object in the scene. The tradeoff of the resolution improvement is the sacrifice of the depth range.

This observation is the motivation for further exploration of the usage of this kind of data. In the following section, we propose a surface light-field which is defined by sets of light rays sampled by a virtual pinhole camera plane close to the surfaces in the scene. Different from standard light-field, the surface light-field has a lower dimensionality and is thus simpler to model.

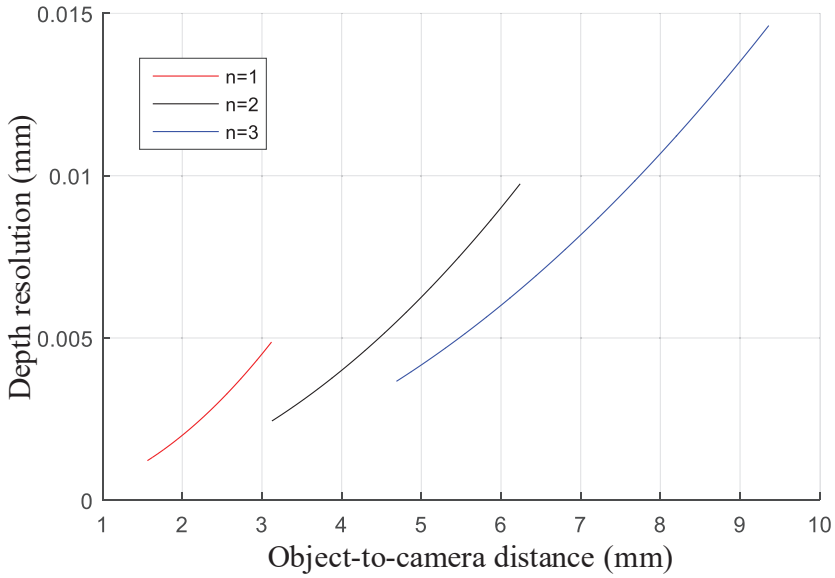

Figure 3.4: Depth resolution of the constructed light-field camera. n represents the baseline between the virtual pinhole cameras used for depth estimation.

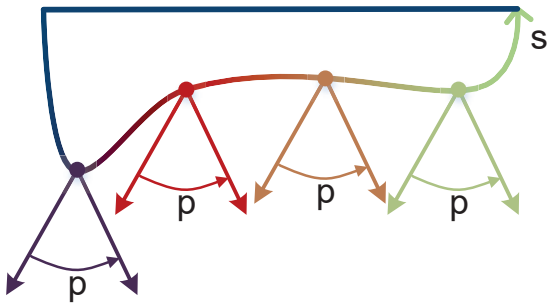

Figure 3.5: The texture signal $f(s, p)$ on the surface. The 2D radiance function is determined by its curvilinear coordinate s on the surface and the emitting direction p.

3.2.2 Definitions and notations

In the computer graphic community, the surface light-field was originally referred to as a function that assigns a color value to each light ray that originates on a surface [39].

In this thesis, we denote by 'texture'and the function $f(s, p)$ to refer to these direct measurements from the surface as shown in Figure 3.5. Note that s is the curvilinear coordinate on the surface, whereas p is the emitting direction of the light ray from the surface.

As for the light-field acquisition, it is almost impossible to capture the surface light-field directly. Thus, we relax the constraints of the initial definition of surface light-field. We change the reflectance function on the surface to a set of light rays captured by a pinhole camera that is positioned closely to the surface. For a surface within a small depth range, we simply put a

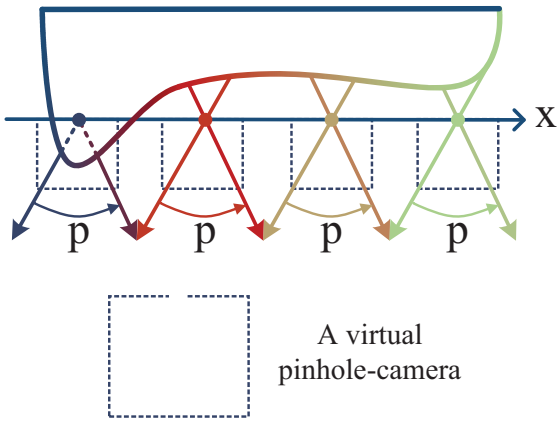

Figure 3.6: A virtual pinhole camera plane close to the surface. The light-field data $L(x, p)$ is defined by the camera position x and the direction p of the recorded light ray.

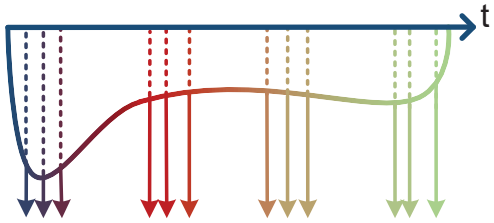

Figure 3.7: The 1-D surface light-field $g(t)$. Given the Lambertian assumption of the surface, the 2-D light-field data $L(x, p)$ becomes the 1-D surface light-field $g(t)$ by identifying the origin of each light ray.

pinhole-camera plane to capture the light rays emitted from the surface as shown in Figure 3.6. There are two advantages in using this framework. Firstly, the pinhole cameras that are close to the surface have a high depth resolution, as discussed in Section 3.2.1. Secondly, these virtual pinhole-cameras can reduce the possibility of occlusions for scenes with small depth range. In practice, the scene can first be segmented by the depth range and then be captured and processed respectively.

The surface light-field shown in Figure 3.6 is just the standard light field that is captured on the virtual pinhole-camera plane close to the surface. We further simplify the definition of the surface light-field and reduce its dimensionality as $g(t)$ by restricting the scene to be Lambertian surfaces as shown in Figure 3.7.

We establish the relations among the texture signal $f(s, p)$, surface light-field $g(t)$ and the 2-D light-field data $L(x, p)$ as follows. Given the Lambertian assumption, the 2-D texture function $f(s, p)$ becomes $f(s)$ as the point on the surface emits the same light rays for all directions. Then we assume that the surface light-field $g(t)$ is a result of the texture signal $f(s)$ "painted" on the surface. This framework is also used in the work of Do et al. [21]. The light-field data $L(x, p)$ is a set of measurements from the surface light-field $g(t)$. The sampling locations on $g(t)$ are determined by the corresponding depth values of each entry in the light field $L(x, p)$.

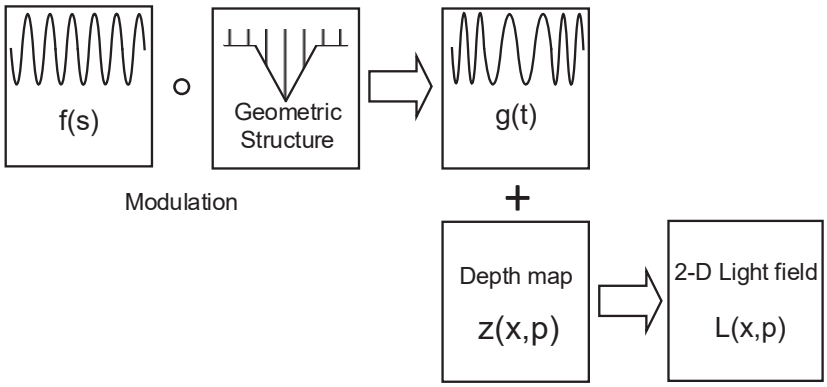

Figure 3.8: An illustration of the relations among the texture signal $f(s)$, the 1-D surface light-field $g(t)$ and the 2-D light field $L(x, p)$.

The formulation of the surface light-field requires the surfaces in the scene to be Lambertian. But in practice, when the reflectance function of the surfaces is smooth, the surface light-field can still be used to model the painted signal.

3.2.3 Surface model: textures painted on surfaces

In Figure 3.8, we demonstrate how the geometry structure of a given scene modulates the 1-D texture signal $f(s)$ to the 1-D signal $g(t)$ and then how $g(t)$ is mapped to the 2-D signal $L(x, p)$ with the information of the depth map $z(x, p)$.

The relation between $f(s)$ and $g(t)$ is well studied in [21] and the following work [23][24]. In their work, the texture signal $f(s)$ is assumed to be a band-limited signal. They conclude that the surface light-field $g(t)$ is band-limited only when the surface on which $f(s)$ is painted is flat. To address general scenes, they use the essential bandwidth that is defined as the bandwidth where most of the signal energy resides. The essential bandwidth of the surface light-field $g(t)$ is estimated as the product of the bandwidth of the painted signal $f(s)$ times the maximum absolute derivative of the surface curvilinear coordinate along a certain direction. In conclusion, the bandwidth of the surface light-field $g(t)$ depends on the maximum and minimum surface depths, the maximum frequency of the painted signal $f(s)$, and the maximum surface slope.

In our setup, we consider only the essential bandwidth of the signal $g(t)$ and use the Fourier series to approximate the signal $g(t)$. As long as the bandwidth of the Fourier series covers the essential bandwidth, we can achieve a good approximation of the underlying 1-D texture signal $g(t)$ and the acquired 2-D light field $L(x, p)$.

3.2.4 Mapping light fields to surface light-fields

We use the following assumptions in our analysis. First, we pose Lambertian constraints on the surfaces in the scene. Therefore light rays from the same origins on the surface have the same radiance intensities. Second, we assume the texture $f(s)$ that is "painted" on the surfaces to be band-limited. Last but not least, we assume the surfaces in the scene to be continuous. Thus the surface light-field $g(t)$ can be approximated with a Fourier series as discussed in the

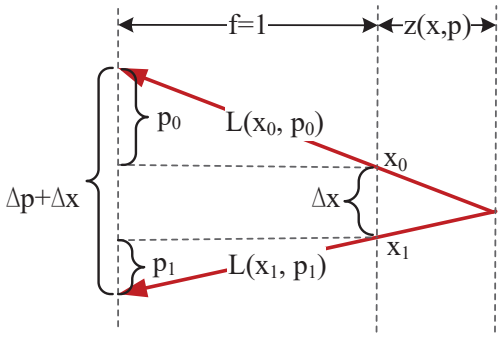

Figure 3.9: Illustration of the geometry relation between the light ray $L(x_0, p_0)$ and $L(x_1, p_1)$ which are both emitted from one origin. The two light rays are recorded by their intersections with the plane x and p.

previous section.

Under these assumptions, the 2-D light field is a set of measurements of the 1-D surface light-field and the measuring locations are determined by the geometric structure of the scene. To map the 2-D light field $L(x, p)$ to the 1-D texture $g(t)$, we trace each light ray back to its originating position on the surface.

As shown in Figure 3.9, we choose two arbitrary light rays $L(x_0, p_0)$ and $L(x_1, p_1)$ from the same origin. The variable $z(x, p)$ stands for the perpendicular distance between the origin of the light ray $L(x, p)$ and the camera plane. Under the Lambertian assumption of the surface, the two light rays satisfy the following equation

$$L(x_0, p_0) = L(x_1, p_1).$$

As they are from the same origin, we can also claim

$$z(x_0, p_0) = z(x_1, p_1).$$

Then we formulate the correspondence between these light rays as

$$x_0 - x_1 = z(x_0, p_0)(p_0 - p_1). \tag{3.3}$$

Equation (3.3) describes the general correspondence for two arbitrary light rays. Here we use an extreme case to determine the location of the origin. We set p_1 to be 0 to trace the sampling location on the surface. The origin of the light rays can be calculated as follows:

$$x_1 = x_0 + p_0 \cdot z(x_0, p_0). \tag{3.4}$$

The Equation (3.4) describes the ray tracing process and identifies the sampling location of $L(x_0, p_0)$ on the surface. In a light field, the sampling location for each light ray can be uniquely determined by the depth $z(x, p)$ as shown in Equation (3.4). Therefore, the 2-D light field can be mapped to the 1-D texture signal painted on the surface as follows:

$$L(x, p) = g(t(x, p)) = g(t(x + p \cdot z(x, p), 0)) = g(x + p \cdot z(x, p)).$$

In conclusion, we map a 2-D light field to a 1-D surface light-field and the operation can also be extended to map 4-D light fields to the 2-D surface light-fields.

3.3 Depth-Recovery Algorithm

Light fields are fundamentally the samples on the surface light-fields and the corresponding sampling locations are determined by the geometric structure of the surfaces in the scene. By using the surface light-field, we propose a practical framework to fully exploit the potential of the acquired light-field data for depth reconstruction.

3.3.1 Problem formulation

In this section, we first show how the surface light-field is represented with a finite Fourier series and then formulate the problem of depth recovery from the high dimensional light field as a problem of recovering the low dimensional surface light-field from unknown sampling locations.

Fourier series as signal model

As described in Section 3.2, the surface light-field is usually not band-limited, unless the band-limited texture is painted on a flat surface. Fortunately, most energy of the surface light-field is contained in a finite band-width that is referred to as the essential band-width in this thesis. In practice, to represent a 1-D surface light-field $g(t)$, we approximate the surface light-field with the Fourier series with a total number of $2L+1$ complex exponentials as follows:

$$g(t) = \sum_{k=-L}^{k=L} a_k \exp(2\pi j k t), \qquad t \in [0, 1)$$

where the fundamental frequency of the signal is 1, given that t is normalized into the range of $[0, 1)$.

Let us review the problem. We are given a set of measurements in the form of the 2-D light field $L(x, p)$, which is a set of samples on the 1-D signal $g(t)$. We assume the essential bandwidth of the signal $g(t)$ to be $2L+1$ and use the Fourier series to approximate the signal. From the 2-D data $L(x, p)$, we are given a total number of $N = N_x \cdot N_p$ samples where N_x and N_p are the number of samples in the x and p dimension, respectively. Finally, our goal is to find the amplitudes $\{a_k\}_{k=-L}^{L}$ of the Fourier series and the sampling locations $\{t_i\}_{i=0}^{N-1}$.

For the sake of simplicity, we represent the Fourier series in the matrix form are as follows:

$$\begin{bmatrix} g(t_0) \\ \vdots \\ g(t_{N-1}) \end{bmatrix} = \begin{bmatrix} W_0^{-L} & \cdots & W_0^{0} & \cdots & W_0^{L} \\ \vdots & & \vdots & & \vdots \\ W_{N-1}^{-L} & \cdots & W_{N-1}^{0} & \cdots & W_{N-1}^{L} \end{bmatrix} \begin{bmatrix} a_{-L} \\ \vdots \\ a_0 \\ \vdots \\ a_L \end{bmatrix} \tag{3.5}$$

where $W_i = \exp(2\pi j t_i)$.

Signal recovery from unknown sampling locations

We model the forward operator of the light field as a sampling process of the 1-D surface light-field where the sampling locations are determined by the depth map. We want to solve the

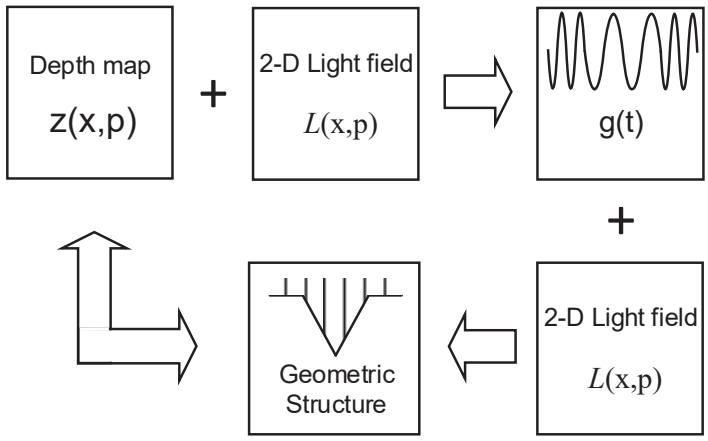

Figure 3.10: An illustration of the relations among the texture signal $f(s)$, the 1-D surface light-field $g(t)$ and the 2-D light field $L(x, p)$.

inverse problem by identifying the sampling locations and recover the signal. In this problem, both the sampling locations $\{t_i\}_{i=0}^{N-1}$ and amplitudes $\{a_k\}_{k=-L}^{L}$ of the Fourier series are unknown and there is no unique solution to Equation (3.5).

In practice, we approach this inverse problem by minimizing the difference between the acquired light-field data and the recovered surface light-field. This back-projection can enhance the stability and robustness of the algorithm. More specifically, to minimize the cost function $\|\mathbf{g}-W\mathbf{a}\|_2$, we use a method that is similar to the one used in [13]. Then, we update the sampling locations $\{t_i\}_{i=0}^{N-1}$ and amplitudes $\{a_k\}_{k=-L}^{L}$, alternately.

In conclusion, with the alternating least squares algorithm, we minimize the cost function $\|\mathbf{g} - W\mathbf{a}\|_2$ to recover the sampling locations and amplitudes of the Fourier series. The objective function of the goal can be formulated as

$$W, \mathbf{a} = \underset{W,\mathbf{a}}{\operatorname{argmin}} \|\mathbf{g} - W\mathbf{a}\|_2 \tag{3.6}$$

3.3.2 Algorithm overview

In our algorithm, we approach the signal recovery problem by updating the sampling locations $\{t_i\}_{i=0}^{N-1}$ and amplitudes $\{a_k\}_{k=-L}^{L}$ of the Fourier series, alternately, to minimize the difference between the surface light-field $g(t)$ and the 2-D light field $L(x, p)$. As the 2-D light field $L(x, p)$ is a set of discrete measurements of the 1-D surface light-field, we can directly render an estimation $\hat{L}(x, p)$ of the 2-D light field $L(x, p)$ from the sampling locations $\{t_i\}_{i=0}^{N-1}$ and amplitudes $\{a_k\}_{k=-L}^{L}$. Then we can compare the estimated light-field $\hat{L}(x, p)$ to the original measurements to verify the accuracy of the sampling locations and amplitudes. This back-projection operation enhances the stability and robustness of the depth estimation algorithm.

We present an illustration of the algorithm in Figure 3.10. On the one hand, given the depth map $z(x, p)$, we recover the 1-D surface light-field $g(t)$ in the continuous domain from the discrete 2-D light-field data $L(x, p)$. On the other hand, we can reconstruct the geometry structure by tracing the 2-D light-field data back to the 1-D surface light-field $g(t)$.

Algorithm 3.1 Depth-Recovery Algorithm

Initialize the depth map $z(x, p^0)$ and the converging factor $c^0 = -\infty$
Construct the matrix $\mathbf{W}^0$ with the depth map $z(x, p)^0$
for $i = 0$ to C **do**
  $\mathbf{a}^{i+1} = \mathbf{W}^{\dagger i+1}\mathbf{g}$
  $\hat{\mathbf{g}} = \mathbf{W}^{\dagger i+1}\mathbf{a}^{i+1}$
  $c^{i+1} = \|\mathbf{g} - \hat{\mathbf{g}}\|^2$
  if $c^i - c^{i+1} < \epsilon$ **then**
    Break
  end if
  $z = \underset{z}{\operatorname{argmin}} E(z)$
  where $E(z) = \int \psi(|\hat{g}(x + zp) - L(x, p)|^2) + \beta\psi(|z + \frac{z_p}{z_x}|^2) + \gamma\psi(|\nabla z|^2)dxdp$
  Construct the matrix $\mathbf{W}^{i+1}$ with the depth map $z(x, p)^{i+1}$
end for
return $z(x, p), \hat{g}(t)$

We present an overview of the proposed algorithm in Algorithm 3.1. Given the initial depth-map $z(x, p)$ of the scene, we map the 2-D light field $L(x, p)$ to the 1-D surface light-field $g(t)$ by applying

$$g(x + p \cdot z(x, p)) = L(x, p).$$

We use the Fourier series to represent the surface light-field $g(t)$. To recover the amplitudes $\{a_k\}_{k=-L}^{L}$ of the Fourier series, we first construct the matrix W where $W_i = \exp(2\pi j t_i)$. Then the amplitudes $\{a_k\}_{k=-L}^{L}$ can be recovered by solving

$$\mathbf{a} = \underset{\mathbf{a}}{\operatorname{argmin}} \|\mathbf{g} - W\mathbf{a}\|_2, \tag{3.7}$$

which is a variation of the minimization problem in Equation (3.6) by fixing the variable W. Equation (3.7) can be solved directly by

$$\mathbf{a} = W^{\dagger}\mathbf{g},$$

where $W^{\dagger}$ is the pseudoinverse of W.

Given the solution of $\mathbf{a}$, we then estimate the sampling locations $\{t_i\}_{i=0}^{N-1}$ by minimizing

$$W = \underset{W}{\operatorname{argmin}} \|\mathbf{g} - WW^{\dagger}\mathbf{g}\|_2, \tag{3.8}$$

which is also a variation of the minimization problem in Equation (3.6) by fixing $\mathbf{a}$.

In practice, it is intractable to optimize (3.8) over the sampling locations $\{t_i\}_{i=0}^{N-1}$ directly. We minimize the difference between the discrete measurements and the continuous function $\hat{g}(t)$, which is the Fourier series with the amplitudes $W^{\dagger}\mathbf{g}$. The difference is modeled as the objective function as follows:

$$\begin{aligned} E_d(z) &= \int \sqrt{|\hat{g}(x + zp) - L(x, p)|^2 + \epsilon^2}dxdp \\ &= \int \psi(|\hat{g}(x + zp) - L(x, p)|^2)dxdp. \end{aligned} \tag{3.9}$$

Here ψ is defined as

$$\psi(s^2) = \sqrt{s^2 + \epsilon^2},$$

which is an increasing concave function and ϵ is a small positive constant. The introduction of ψ increases the robustness of the energy function with only introducing the constant ϵ.

Note that the energy model in Equation (3.9) only considers the data difference between the discrete data and continuous signal locally without taking any interactions between neighbouring pixels into account. Hence, it runs into problems for constant signals or even low frequency signals. Furthermore, a direct minimization of Equation (3.9) is also vulnerable to outliers. To address these issues, we introduce a further assumption on the smoothness of the depth map as follows:

$$E_s(z) = \int \gamma\psi(|\nabla z|^2)dxdp$$

with γ being a weight between two terms.

The smooth term $\psi(|\nabla z|^2)$ poses only the smoothness in a general way without considering the properties of light fields. The most used property of a 2-D light field is its line structure. The line structure also exists in the depth map $z(x, p)$, which can be formulated as follows

$$z(x, p) = z(x + z(x, p) \cdot \Delta p, p + \Delta p),$$

which can be developed to get

$$z_x \cdot z(x, p)\Delta p + z_p \Delta p = 0,$$

where z_x and z_p are used to represent the derivative of $z(x, p)$ in the x and p dimensions respectively.

Finally we add the new structure term and rewrite the objective energy function as

$$E(z) = \int \psi(|\hat{g}(x + zp) - L(x, p)|^2) + \beta\psi(|z + \frac{z_p}{z_x}|^2) + \gamma\psi(|\nabla z|^2)dxdp. \tag{3.10}$$

Here minimizing the energy function in Equation (3.10) over z is equivalent to minimizing Equation (3.6) over $\mathbf{W}$. With the depth map estimated from Equation (3.10), we can derive the sampling locations $\{t_i\}_{i=0}^{N-1}$ on the continuous signal $g(t)$ and update the matrix $\mathbf{W}$. A complete loop of the proposed algorithm is then achieved and both the depth map $z(x, p)$ and the continuous signal $\hat{g}(t)$ are recovered.

We evaluate the recovered results by calculating the difference c between the recovered signal $\hat{g}(x + p \cdot z(x, p)$ and the observed light field $L(x, p)$ as

$$c = \|L(x, p) - \hat{g}(x + x \cdot z(x, p))\|_2.$$

We repeat the whole algorithm as long as the difference c continues to decrease significantly. The whole algorithm is illustrated in Algorithm 3.1.

The proposed algorithm requires an initialization of the sampling locations. We propose a local disparity-estimation algorithm applied to the discrete light field directly.

Initializing the sampling locations

To initialize the sampling locations, we have to estimate the depth map for each entry of the light-field data $L(x, p)$. Here we apply a simple extension of the standard stereo method to

estimate the initial depth map from light fields. Within the light-field data, any sub-aperture image pair or pinhole camera image pair can be used to estimate the depth map of the scene. By utilizing the image pairs in the data, we can reconstruct the depth value for each entry.

The disparity between the neighboring sub-aperture image pairs can be formulated as

$$L(x,p) = L(x + \Delta x, p + 1 \cdot T_p),$$

whereas the disparity between the neighboring pinhole-camera image pairs is formulated as

$$L(x,p) = L(x + 1 \cdot T_x, p + \Delta p).$$

In the standard stereo system, the depth value is determined by the disparity, baseline and focal length of the camera. For the sub-aperture image, the baseline is defined by $T_p \cdot f$, whereas the focal length is f. However, we cannot directly use the disparity to estimate the depth, because a different sub-aperture image would have a different range in the angular domain.

By choosing the neighboring light rays from the same pinhole camera, we can estimate their disparity on the main lens as $T_p \cdot b$. As the baseline between the neighboring pinhole-cameras is $T_p \cdot f$, the pixels with the same index from two neighboring sub-aperture images have an offset in the angular domain as $T_p \cdot (b - f)$. Furthermore, the depth value estimated from the sub-aperture images is the relative depth from the focal plane of the main lens. We derive the depth as follows:

$$\begin{aligned} z(x,p) &= f\frac{T_p \cdot f}{\Delta x + T_p \cdot b - T_p \cdot f} + f \\ &= f\frac{\Delta x + T_p \cdot b}{\Delta x + T_p \cdot b - T_p \cdot f}. \end{aligned}$$

For the pinhole-camera image, the baseline is defined by T_x, whereas the focal length is 1. Then the virtual object captured by the pinhole camera locates at $\frac{T_x}{\Delta p}$ in front of the pinhole camera plane. Then the actual depth $z(x,p)$ can be deduced with the thin lens model

$$\frac{1}{f} = \frac{1}{z(x,p)} + \frac{1}{b - \frac{T_x}{\Delta p}}.$$

As $T_p/\Delta x$ and $\Delta p/T_x$ are equivalent in the light field, we can observe that the depth values derived from the neighboring sub-aperture images and pinhole cameras are exactly the same.

Here we propose to use the local disparity between the neighboring pinhole cameras to estimate the initial depth map. The local disparity Δp between the neighboring virtual pinhole cameras is denoted with u to avoid confusion of symbols, and we use 1 to denote the incremental value $1 \cdot T_x$ in the discrete domain as follows

$$L(x,p) = L(x + 1, p + u).$$

To increase the robustness and derive an energy function $E(z)$, we pose the smoothness constraints of the disparity across the light field as follows:

$$E(u) = \int \psi(L_z^2) + \gamma\psi(|\nabla u|^2) dx dp,$$

where L_z is used to represent $L(x,p) - L(x+1, p+u)$ for the sake of brevity, ψ is the increasing concave function for reducing the outliers' influence, and γ is a weight factor between the data constraint and the smoothness constraint.

As $E(u)$ is highly nonlinear, it can only be optimized locally. We compute the Euler-Lagrange equations that must be fulfilled for a local optimum:

$$\Psi'(L_z^2) \cdot L_p L_z - \gamma \Psi'(|\nabla u|^2) \Delta u = 0,$$

where we use L_p to denote $\partial_p L(x+1, p+u)$. We use the a fixed point iteration method to find the local minimum that is widely used in optical flow problems [14].

Although we have only discussed the 2-D case, it is straightforward to extend the algorithm to the 3-D light field (the image sequence captured by 1-D translations) and 4-D light field.

More specifically, to estimate the local disparity we choose to use the neighboring virtual pinhole-camera images instead of the sub-aperture image. Intuitively, one point in the 3-D world is usually observed by all the sub-aperture images. The optimal depth estimation requires a global optimization that considers all the involved images. But the same point is usually captured by a limited number of virtual pinhole-cameras because the virtual pinhole-camera is usually very close to the point. Thus, when using the local images to estimate the depth, it is more efficient to use the neighboring pinhole-camera images instead of the sub-aperture images.

Once we recover the disparity map $u(x, p)$ of the light field, we can map the 2-D light data to the 1-D surface light-field with

$$t(x, p) = x + \frac{1}{u(x, p)} \cdot p,$$

and reconstruct the matrix $\mathbf{W}$ for the proposed algorithm to recover the depth map from surface light-fields.

After we estimate the sampling locations by minimizing Equation (3.10), we directly update the amplitudes of the Fourier series. Furthermore, to avoid noise and outliers we also perform a median filtering of the intermediate results; more specifically, the median filter is applied along the lines to maintain the line structure in 2-D light field.

3.4 Experimental Results

In this section, we test the proposed algorithm on various datasets. We begin with the simulations on 2-D light fields. By using parametric models of the geometry structure and texture, we generate a 2-D scene painted with band-limited texture. We test the proposed algorithm and discuss its performance under varying conditions such as noise level and the quality of the initial depth map.

We then run the algorithm on a synthetic public dataset from [50] and compare our algorithm with the state-of-the-art algorithms.

Finally, we test our algorithm on real-life datasets captured with the constructed light-field camera from Chapter 2. We test our algorithm on two different types of objects: a 3-D printed object and an oil painting.

3.4.1 2-D light-field simulations

To illustrate and validate the proposed algorithm, we consider a synthetic scene as shown in Figure 3.11. In the scene, there is a piece-wise linear wall with varying slopes between neighboring knots. The painted texture signal $g(t)$ has a bandwidth of L. The dimension of the discrete data

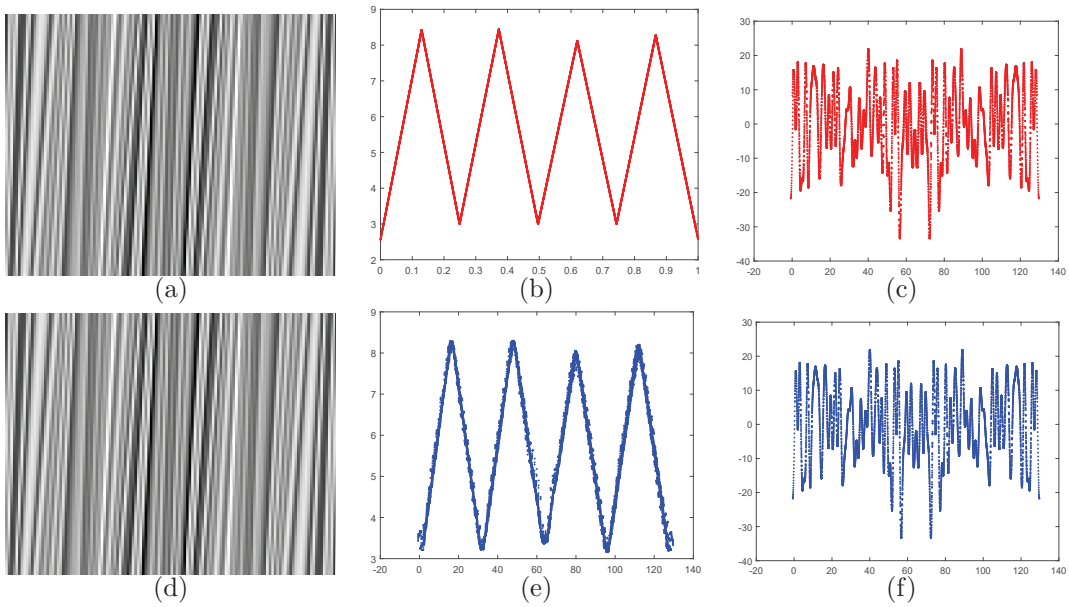

Figure 3.11: A toy model of the 2-D light field. The top row shows the ground truth: (a) the 2-D light field, (b) the original 1-D depth map and (c) the 1-D texture signal. The bottom row shows the results from the proposed algorithm: (d) the rendered 2-D light field, (e) the reconstructed depth map and (f) the recovered 1-D texture signal.

is $M \times N$ where M and N specify the number of samples in the spatial and angular dimension, respectively. We create a 2-D light field with the dimension $M = 101$ and $N = 128$ and the bandwidth of the texture signal is set as $L = 64$. Under this setting, the sampling rate N in the angular domain is sufficient to recover the texture signal. As shown in Figure 3.11, the 2-D light field, the 1-D depth map and the texture signal are well recovered with a final SNR of 34.5 dB and 35.6 dB, respectively.

We also experiment on light fields with different bandwidths from $L = 32$ to 96 as shown in Figure 3.12. As the bandwidth of the signal increases, the rendered light fields deviate from the acquired light field. Here we keep the number of the Fourier coefficients to be constant for a fair comparison of the depth recovery results. Thus, the quality of the rendered light fields naturally decrease as the high frequency components increase. However, the quality of the depth map remains consistent in our experiments. The high frequency components within a range can remove the ambiguity in the disparity estimation and improves the accuracy of the depth map. But as the bandwidth L continues increasing, the quality of the rendered light field decays accordingly. Thus, for scenes with high frequency textures, we must increase the number of the Fourier coefficients for improving the quality of the rendered light-fields.

As shown in Figure 3.13, we present a plot of how errors change after iterations when the bandwidth L of the surface light-field $g(t)$ is 64. We use l_2 norm to measure the errors of the recovered light field and depth map. The errors are normalized by the l_2 norm of the signal. Thus, we use a green dashed-line to show an error boundary that is equivalent to a signal with a SNR of 30 dB. We can clearly observe that after 40 and 50 iterations, respectively, the SNR

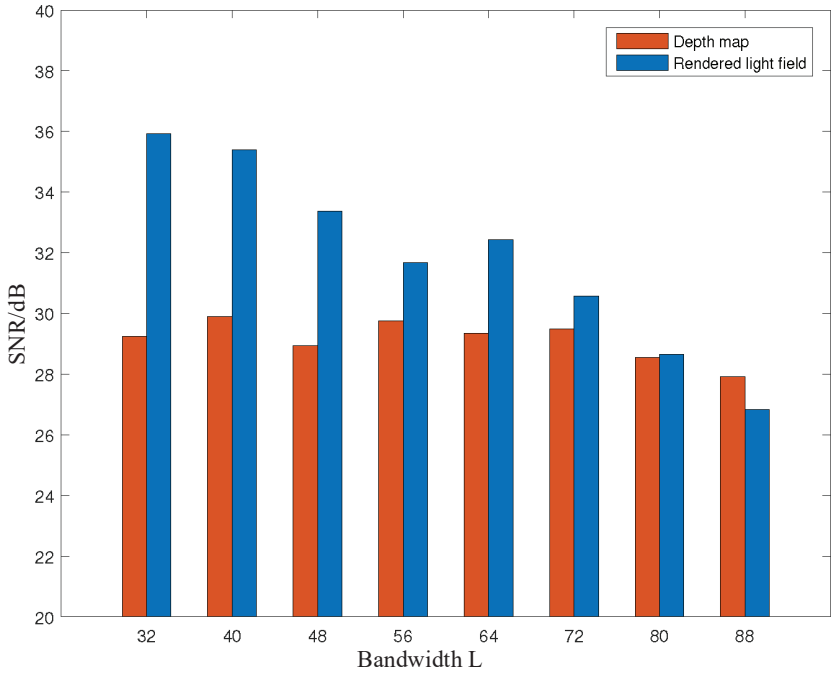

Figure 3.12: The algorithm performance on 2-D light fields with varying bandwidth. The variable L denotes the bandwidth and varies from 32 to 96 while the sampling rates in the spatial and angular domain remain constant as $M = 101$ and $N = 128$, respectively. The SNR of the rendered light field and the depth map are demonstrated respectively.

values of the recovered light field and depth map are already more than 30 dB.

Then we show how noise affects the algorithm in two ways. The first case is the quality of the acquired light field. By adding Gaussian noise on the original data, we observe the SNR of the rendered light field and the depth map compared to the ground truth. The second case refers to the quality of the initial depth map. Instead of estimating the depth map, we add Gaussian noise on the correct depth map to simulate the initial depth maps with different qualities. The quality of the acquired light-field data and the depth map are described with SNR. As shown Figure 3.14, the final results of the rendered light field always converge to more than 30 dB. As for the depth map, since the initial depth map is only affected by Gaussian noise, the final results also converge to a high SNR around 34 dB. As shown in the simulation, although the quality of the light-field data affects the final results, it is not as important as the initial depth map. A robust initialization of the depth map can guarantee good results even with low quality light-field data.

In conclusion, we demonstrate the performance of the algorithm with toy models. We test it with light-field data under varying noise level and aliasing effect. The results degrade accordingly but still achieve decent results. We also notice that the quality of the initialization is critical to improve the final results. In conclusion, the sampling rate of the light camera and the initialization of the depth map are crucial factors to the final results whereas the noise level of the acquired data is less important.

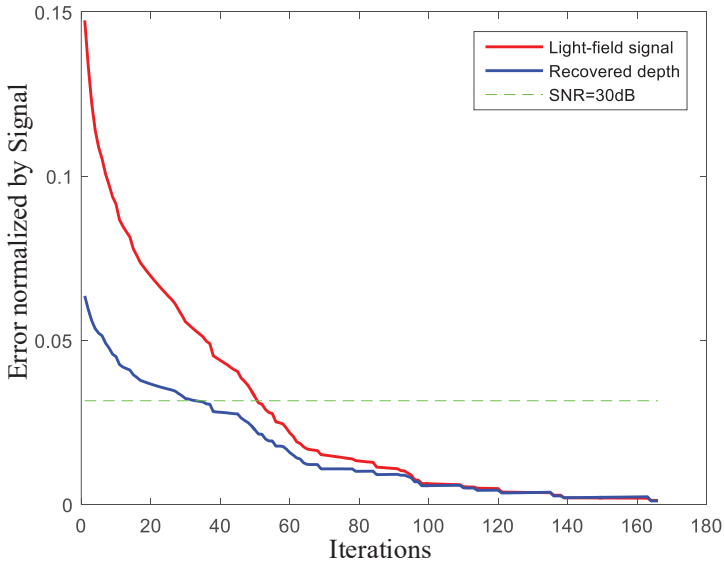

Figure 3.13: Error convergence rate. We show a error convergence rate for bandwidth $L = 64$. The error is measured with l_2 norm that is normalized by the l_2 norm of the signal. For a clear comparison, we also use a green dashed-line to show the error rate of a signal with a SNR of 30 dB. We can observe that after 40 and 50 iterations, respectively, the recovered light field and depth map have SNR values more than 30 dB.

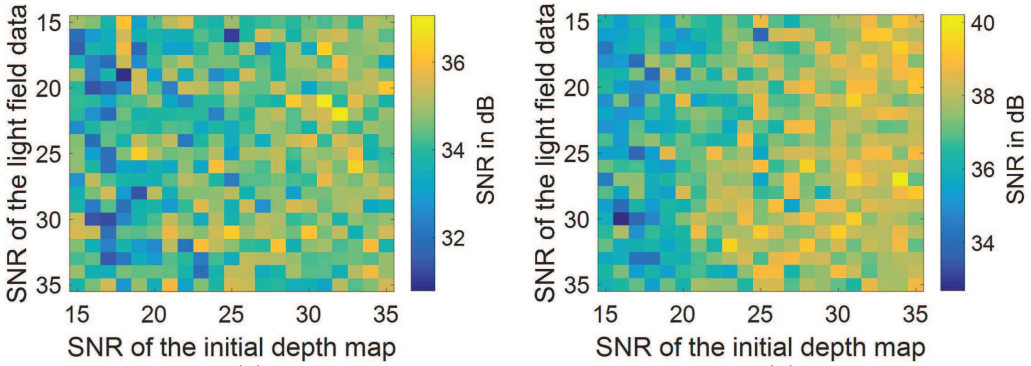

Figure 3.14: Simulation results under different qualities of the light field and initial depth map. Figure (a) shows the quality of the final rendered light field whereas Figure (b) shows the quality of the final depth map comparing to the ground truth.

3.4.2 Experiments on public data-sets

In this section, we test our algorithm on the synthetic light-field datasets from [53]. As shown in Figure 3.15, we choose two 4-D light-fields: the *Budda* and *MonasRoom*. And our results are close to the ground truth under visual inspections. We also present a numerical analysis in Table 3.1, comparing our algorithm with the other three state-of-the-art algorithms: fast denoising and

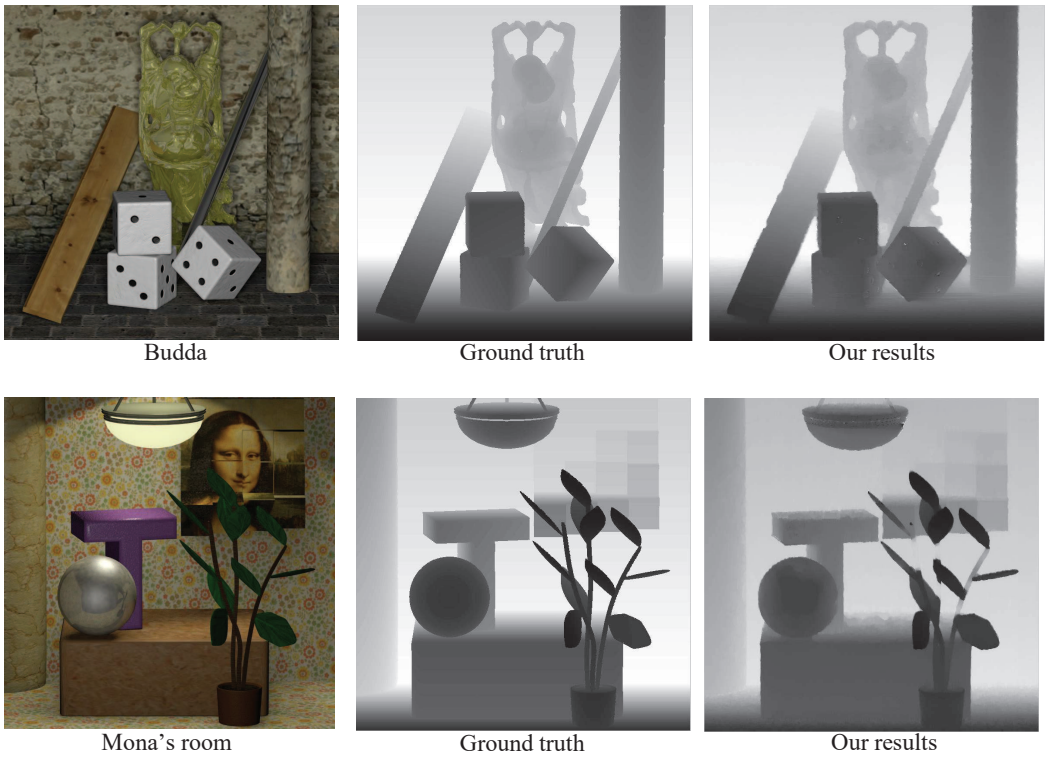

Figure 3.15: Results of synthetic datasets from [53]. From the left to the right, the columns are the center views of the 4-D light fields, the ground truths of the depth maps and our depth-recovery results.

global optimization from [52] and Continuous reconstruction from [33]. The results of the three algorithms are from [33].

The metrics used in Table 3.1 are the percentage of missed pixels of which the relative disparity errors are beyond the threshold (measured with pixel disparity). Our algorithm achieves comparable results to the state-of-the-art algorithms and especially works well in the category where the disparity errors are smaller than 0.1.

Our algorithm has two disadvantages when applied to these light-field datasets. First, the scenes have a large depth range that violates the assumption that we use to model the surface light-field. We apply our algorithm to recover the depth map row by row without segmentation or special treatment for the discontinuities in the scene. Thus, for the dataset of Budda where many discontinuities are present around the object boundaries, our algorithm has less accurate estimations. But for the dataset of MonasRoom, the scene has many large and continuous surfaces, thus we achieve better results compared with the other algorithms.

Second, we assume the depth map to be continuous. But the ground truths are discrete as they are synthetic data without any low-pass filtering. There are aliasing effects in the ground truth of the depth maps. Thus, our assumptions that there are continuous lines in the depth map of the 2-D light field are also violated.

To sum up, although our algorithm faces difficulties due to contradictions between our as-

	Budda			MonasRoom		
Algorithm	>1	>0.5	>0.1	>1	>0.5	>0.1
Fast Denoising [52]	0.13	**0.52**	2.45	0.29	0.86	3.95
Global optimization [52]	0.18	0.56	**2.03**	0.32	0.86	4.33
Continuous Reconstruction [33]	**0.056**	0.60	4.57	**0.033**	0.85	5.00
Proposed algorithm	0.12	0.83	2.91	0.24	**0.65**	**3.68**

Table 3.1: Percentage of Wrong Disparity Estimations. The values in the table are the percentage of the number of missed pixels whose pixel-disparity errors are greater than a certain value (1, 0.5 and 0.1) over the total number of pixels.

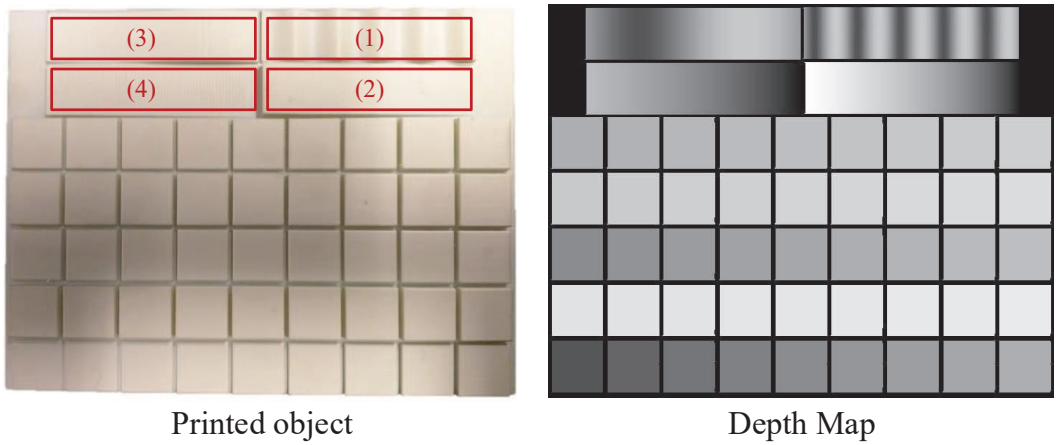

Figure 3.16: The printed 3-D object and its depth map. The 3-D printer that we use has a resolution of 0.033 mm. We designed the slanted planes (marked with (3) and (4)), sinusoidal planes (marked with (1) and (2)) and a set of 5×9 fronto-parallel planes on the bottom. From the top view, the slanted planes and sinusoidal planes have the same dimension, $12\ mm \times 50\ mm$. The sinusoidal surfaces have the periods of 50 mm and 10 mm, respectively. Both amplitudes of the sinusoidal signals are 1.32 mm. The two slanted planes have the maximum heights at 1.98 mm and 3.96 mm, respectively and both have the same minimum heights at 0 mm.

sumptions and the properties of the datasets, we still achieve comparable results to the state-of-the-art algorithms. To further improve our performance, segmenting operations can be introduced to separate the scene to pieces that satisfy our assumptions used to model the surface light-field.

3.4.3 Experiments on acquired datasets

Our depth-recovery algorithm focuses especially on scenes with geometry details in a small depth-range. To verify our algorithm, we use a 3-D printer to generate a planar target as shown in Figure 3.16. There are three different types of carvings: piece-wise constant, slanted plane and sinusoidal plane. We mark four areas with the number from (1) to (4) and show their depth-recovery results in Figure 3.17, Figure 3.19, Figure 3.21 and Figure 3.22, respectively.

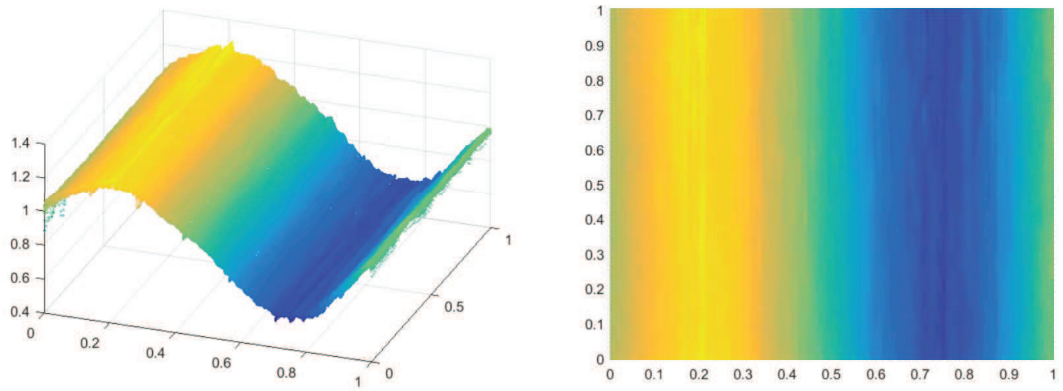

Figure 3.17: The depth-recovery result of area (1) in Figure 3.16. We draw the point cloud in 3-D on the left and show its top view on the right.

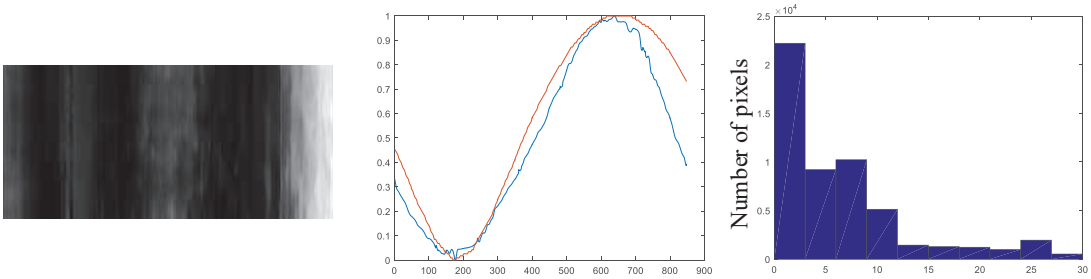

Figure 3.18: The error analysis of depth-recovery result in Figure 3.17. We show the error of the central view of the light field, the side view of the results compared with the ground truth and a histogram of the depth errors in terms of the resolution of the 3-D printer. As 0.033 *mm* is the resolution of the 3-D printer, we use 0.033 *mm* as the step-size when we create the histogram of the depth errors.

In Figure 3.18, we show an error analysis of the depth-recovery result in Figure 3.17. The error map of the center view of the light field is normalized for visual inspection. We can observe that most of the apparent errors locate at the boundaries, from both the error map of center view and the side view comparison to the ground truth. We also show the histogram of the errors. Here we use 0.033 *mm* as the step size when counting the errors because 0.033 *mm* is the resolution of the 3-D printer. We can clearly observe that most errors are within the range of ten times the step size.

In Figure 3.20, we show an error analysis of the depth-recovery result in Figure 3.19. Similarly, even when the frequency of the surface increases, we can still observe a consistent good results of our depth recovery algorithm.

In Figure 3.21 and Figure 3.22, we show the depth-recovery results and the histograms of the

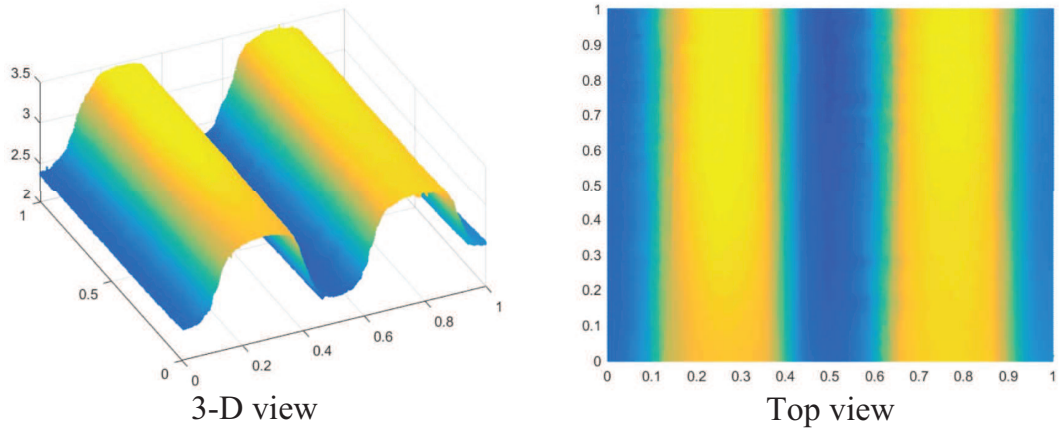

Figure 3.19: The depth-recovery result of area (2) in Figure 3.16. We draw the point cloud in 3-D on the left and show its top view on the right.

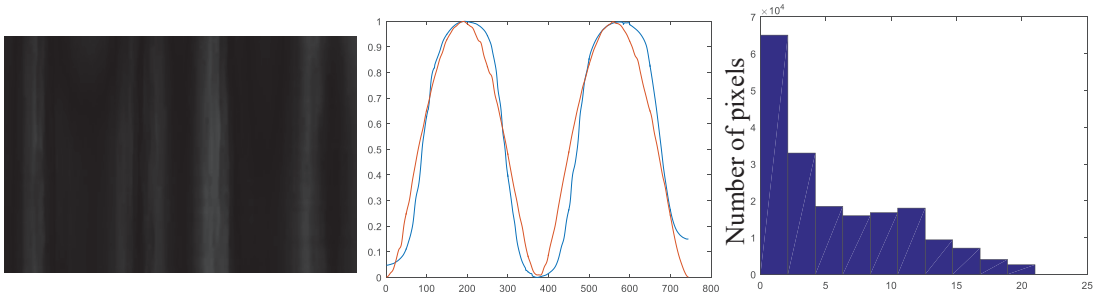

Figure 3.20: The error analysis of depth-recovery result in Figure 3.19. We show the error of the central view of the light field, the side view of the results compared with the ground truth and a histogram of the depth errors in terms of the resolution of the 3-D printer. As 0.033 mm is the resolution of the 3-D printer, we use 0.033 mm as the step-size when we create the histogram of the depth errors.

errors from area (3) and (4) from Figure 3.16. We observe no errors larger than ten times of the resolution of the 3-D printer.

In Figure 3.23, we show a more challenging case, where we reconstruct the geometry structure of an oil painting. We show a depth map of the central view and some 3-D plot of the local patches from the oil painting. We can observe that the reconstructed results can capture most of the apparent structures of the oil painting. We can also see that due to the complex property of the materials of the oil painting, this is still a challenging case for depth recovery from light fields.

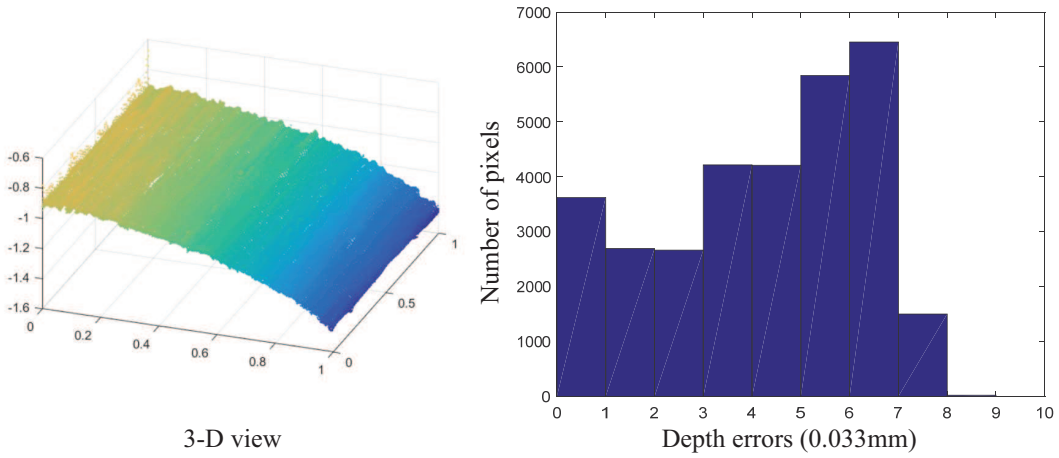

Figure 3.21: The depth-recovery result of area (3) in Figure 3.16. We draw the point cloud in 3-D on the left and show the historam of the errors on the right.

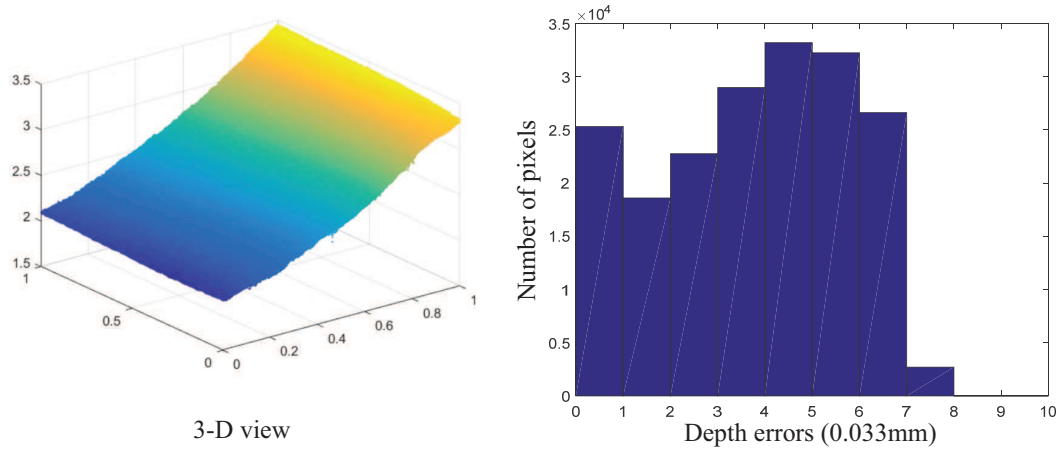

Figure 3.22: The depth-recovery result of area (4) in Figure 3.16. We draw the point cloud in 3-D on the left and show the historam of the errors on the right.

3.5 Conclusions

In this chapter, we model a high dimensional light field as a low dimensional surface light-field modulated by the geometry structure. We see the traditional depth reconstruction problem as a problem of bandlimited signal recovery from unknown sampling locations. We have proposed a novel and practical framework to exploit the properties of the surface light-field for depth recovery with high accuracy. To validate the proposed algorithm, we run experiments on synthetic public datasets and achieve comparable results to the state-of-the-art algorithms. The proposed algorithm best suits geometry structure in a small depth range with fine details. Thus we use a constructed light-field camera from Chapter 2 and run the algorithm on datasets of a 3-D printed object as well as an oil painting.

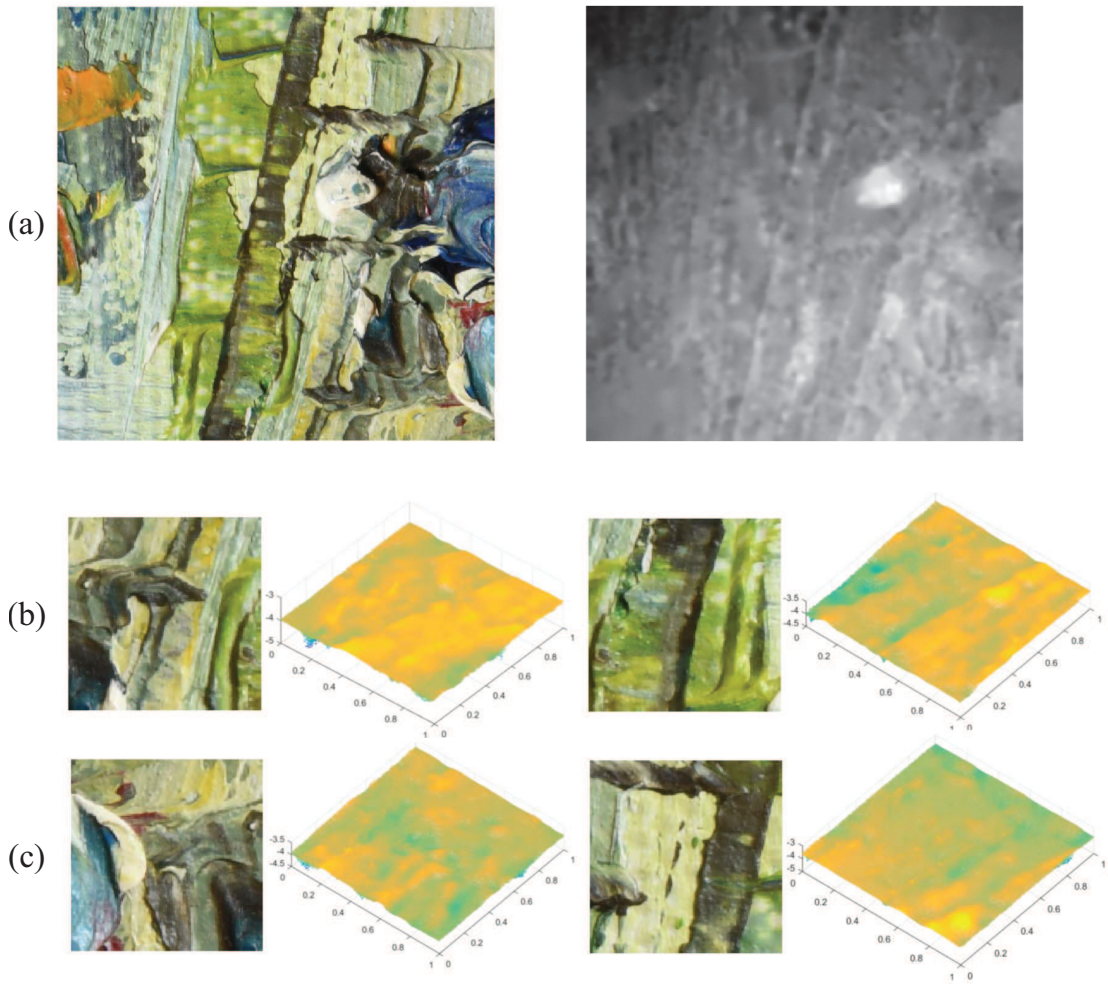

Figure 3.23: Depth recovery results of part of an oil painting. In row (a), we show the central view of the light field and its depth map. In row (b) and (c), we show the recovery results plotted in 3-D of some local patches.

Chapter 4

Light-Field Registration and its Applications

> I guess you could call it a "failure", but I prefer the term "learning experience".
>
> *Andy Weir, The Martian*

The consumer-grade light-field camera, Lytro, has drawn much interest from both the academic and industrial worlds. However, its low resolution in both the spatial and angular domains prevents it from being used for fine and detailed light-field acquisition in a single shot. We discuss the general problem of light-field registration and propose to use a light-field camera as an image scanner and perform light field stitching to increase the size of the acquired light-field data.

After introducing the general problem and a review of the existing work in Section 4.1, we describe general motion models of a light-field camera and show that classic image alignment and stitching is a sub-problem of a more general stitching problem of light fields in Section 4.2. By using the motion models, we describe how to perform light-field acquisition and stitching under two different scenarios: by camera translation, or by camera translation and rotation in Section 4.3. In both cases, we assume the camera motion to be known. In the case of camera translation, we show how the acquired light fields should be re-sampled to increase the spatial range and ultimately to obtain a wider field of view. In the case of camera translation and rotation, the camera motion is designed precisely such that the light fields can be directly stitched and extended in the angular domain. Finally in Section 4.4, we discuss the problem of light-field registration in practice and conclude our work.

4.1 Introduction and Related Work

The first consumer-grade light-field camera, Lytro, generated much interest when it appeared in 2012. Light-field cameras enable many new imaging applications including depth estimation, digital refocusing and perspective shift. They also provide new perspectives for many of the standard problems in image processing such as denoising, super-resolution and image stitching. However, take Lytro as an example, its major limitation is the low resolution of the rendered images: the camera's sensor is indeed used to record both the spatial and angular information of the scene, resulting in a rendered image with a resolution equal to that of the micro-lens array.

To benefit from the great potential of light field cameras, increasing the resolution in both the spatial and angular domains is a crucial step for many applications. As increasing the resolution of the micro-lens array can be very expensive, taking multiple light fields and merging the data is an appealing way to tackle the problem. Thus, in this section, to increase the size of the light-field data, we propose to align and stitch multiple light fields with overlapping areas.

There are many studies on image alignment and stitching. Interested readers are referred to [46] for more information. Here we focus only on the creation of wide-view light fields instead of images. A standard 4-D light field is captured with a camera array as described in [32]. To increase the field of view, we can certainly enlarge the size of the whole camera array, which is bulky and inefficient. We can also increase the view angle of each camera, as proposed in [29, 47]. For very large scenes such as streets, a light-field panorama is very challenging to capture. Kawasaki, et al. use an omnidirectional camera, and introduce the geometry structure of the scene to fully explore the viewing freedom in light fields in [29]. For small scenes, Taguchi, et al. use a spherical mirror and a moving camera to capture a light field with a wide field of view in [47].

There are also methods that directly align and stitch multiple light fields for a wider field of view. Stern, et al. shift the imaging system exactly perpendicular to the optical axis and combine these acquisitions for an extension of the field of view in [44]. Birklbauer, et al. present the first approach to constructing light-field panorama with off-the-shelf commercial light field cameras by using all-in-focus images for registration and focal-stack images for view synthesis in [11].

The main goal of this Chapter is to increase the usability of compact light-field cameras. The work of Birklauer shares the most similarity to our work as they use a simple framework to achieve a light-field panorama with Lytro. However, due to the intermediate focal stack representation, the final light-field panorama cannot handle occlusions and anisotropic light effects.

Our work is different from theirs in two aspects: First, we give a complete motion model of the light-field camera, including both translations and rotations. Second, we directly perform stitching in the 4-D light field, whereas both all-in-focus and focal-stack images are projections of the original light-field into low dimensional data, as mentioned in the work of Levin and Durand in [31].

4.2 Motion Models in Light Fields

Before we can register light fields, we need to first establish the mathematical relationships that map light-field entries from one dataset to another. Although there are many studies on the similar problem of establishing the relations that map pixel coordinates from one image to

another, the relationships between light fields captured with a light-field camera are not well studied. Therefore, we first revisit the motion model of images in the perspective of the light field and then extend our discussion to the motion model of light fields.

We are particularly interested in two types of the camera motion: planar motion and rotation, both of which are commonly used to create photographic panoramas. To be more specific, in this section we focus on deriving the motion model of the light field in two scenarios: a translation perpendicular to the optical axis of the light-field camera and a rotation around the optical center of the light-field camera.

4.2.1 Motion models of standard cameras

Aligning images and stitching them into wide-angle images or panoramas are among the most widely used algorithms in computer vision. A practical stitching algorithm requires robust solutions for image alignment, warping, artifacts removal, etc. Here we focus only on the mathematical model relating pixel coordinates in one image to the pixel coordinates in another. A pinhole-camera model is also used for the sake of simplicity.

There are various camera motions in real life such as 2-D planar translations, 3-D translations, 3-D camera rotations and even random combinations of these motions. Among these scenarios, the two most-used camera motions to create a photographic panorama as shown in Figure 4.1 are planar translations and rotations.

To be more specific, the camera translation is a planar motion perpendicular to the optical axis, whereas the camera rotation is around the optical center. Without loss of generality, we carry out the analysis with 1-D images, which means both the translation and rotation become 1-D motions. Furthermore, all the motions of the 1-D images are modeled in the corresponding 2-D light fields.

Camera rotations in 2-D light fields

We first define a 2-D light field $L(x, p)$ that is created by moving a pinhole camera on the axis x. A 1-D image taken at position x_0 is then denoted with $L(x_0, p)$ where p denotes the coordinates on the image sensor. In the 2-D light field, a 3-D camera rotation around the optical center is reduced to a tilt of the pinhole camera around its optical center.

The camera rotation is usually used to extend the field of view that is defined by the range of the pixel coordinate p. Here we give a definition of the 1-D image $I(p)$ that is captured by the pinhole camera as follows:

$$I(p) = L(x_0, p), \quad p \in \left[-\frac{S}{2f}, \frac{S}{2f}\right],$$

where x_0 represents the location of the camera, S represents the size of the camera sensor and f represents the focal length of the pinhole camera. By rotating the camera, we actually extend the range of the sensor such that even light rays with angles larger than $\frac{S}{2f}$ or angles smaller than $-\frac{S}{2f}$ can also be recorded as shown in Figure 4.2. In the light field, we can clearly see how several 1-D photos merge to one image with larger field of view.

The relationship between the original image $I(p)$ and the image $I'(p)$ after a camera rotation of θ can be formulated

$$I(p) = I'(\tan(\arctan(p) - \theta)).$$

Figure 4.1: Creating a wide view photo by translating a mobile phone on the top and a 360 deg panorama photo camera rotations on the bottom.

In conclusion, the rotation of the 1-D camera corresponds to a shift in the p dimension in the 2-D light field, where no constraints have to be posed on the scene. By rotating the camera, acquired images are shifted in the p dimension and a wider field of view is achieved by merging these images directly.

Camera translations in 2-D light fields

When registering images by camera translation, we usually assume the observed scene to be far enough to be modeled as a plane. This assumption is also widely used in image super-resolution by camera translation.

This problem can be better understood in the 2-D light field. The camera translation is a shift in the x direction, as the 2-D light field itself is defined as a stack of 1-D images taken by a moving camera on the x axis. Taking a photo captured at x_0 as an example, we can represent

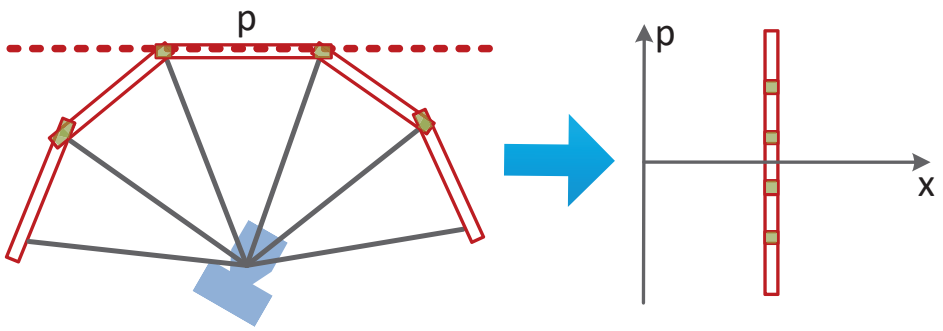

Figure 4.2: Image stitching by camera rotations and their representations in the 2-D light field. The red slices on the right denote the 1-D images by the camera rotations. Then the green areas are used to represent overlapping areas among the observed scenes.

the image $I_{x_0}(p)$ as

$$I_{x_0}(p) = L(x_0, p) \quad p \in \left[-\frac{S}{2f}, \frac{S}{2f}\right].$$

Another image $I_{x_0+\Delta x}(p)$ is taken by a translation of Δx on the x axis.

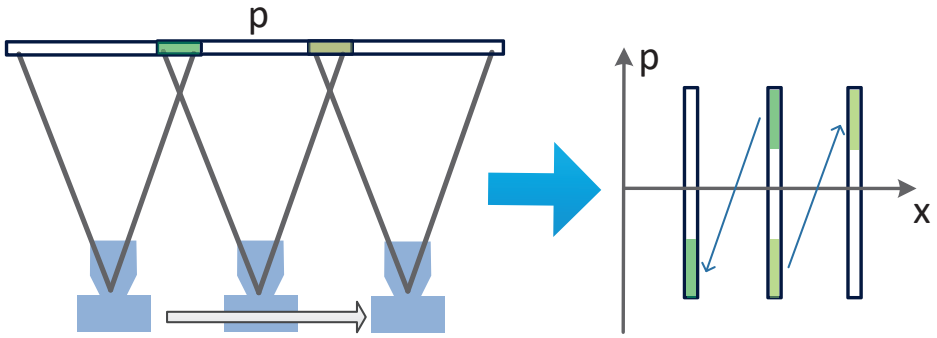

Figure 4.3: Image stitching by camera translation and their representations in 2-D light field on the right. The red slices on the right denote the 1-D images by the camera translations. Then the green areas are used to represent overlapping areas among the observed scenes.

The registration between different images is achieved by matching the overlapping areas in each image. The success of the matching process relies on the fact that this particular area should keep the same appearance for different viewing angles. Unlike the matching pixels for camera rotation, pixels from different images by translations correspond to different light rays. Only when the surface in the scene satisfies the Lambertian property, we can successfully match pixels that correspond to the same location in the scene.

Furthermore, creating the panorama photo requires the scene to have a constant depth value. The creation of the panorama photo is to extend the field of view that is the p dimension in the light field. As shown in the 2-D light field, the translation in the x dimension is equivalent to shift in p direction given that the surface is flat and obeys the Lambertian assumption.

For a scene located z meters away from the camera, the relationship between two images can

be formulated as

$$I_{x_0}(p) = I'_{x_0}(p + \Delta p), \quad \Delta p = \Delta x \frac{f}{z}.$$

In conclusion, the translation of the 1-D camera corresponds to a shift in the x dimension in the 2-D light field. Furthermore, under the assumption that the captured scene is a planar surface with Lambertian-reflectance properties, these 1-D images by translations are equivalent to the images captured by camera rotations. Thus, a wider field of view can also be achieved by camera translations.

4.2.2 Motion models of light-field cameras

We consider a simplified light-field camera model comprising a pinhole camera moving behind a thin lens of focal length f, which is similar to the one used in Chapter 2. We give an illustration of the camera system, as well as the sampling grid of the camera system in Figure 4.4.

We use $I(m, n)$ to denote the discrete 2-D light field of a light field camera, which is a set of measurements on the continuous light field $L(x, p)$. Here m and n respectively stand for the position of the pinhole camera and the pixel index within a pinhole-camera image. The 1-D image obtained by fixing m is referred to as the pinhole-camera image, whereas the 1-D image obtained by fixing n is referred to as the sub-aperture image.

In this setup, we assume a perfect optical system, thus the point-spread function of the main lens becomes a Dirac function. We also assume each pixel of the pinhole camera to be infinitely small and free of noise. Then from the discussion from previous chapters, the imaging process can be formulated as follows:

$$I(m, n) = \langle L([x\ p]^T), \delta_{\mathbf{A}^{-1}\mathbf{T}\cdot[m\ n]^T}([x\ p]^T) \rangle, \tag{4.1}$$

where $\mathbf{A}$ describes the ray propagation from the main lens to the moving pinhole camera and the refraction of the main lens and $\mathbf{T}$ represent the sampling periods on the plane of the moving pinhole-camera as

$$\mathbf{T} = \begin{bmatrix} T_x & 0 \\ 0 & T_p \end{bmatrix},$$

with T_x and T_p defined by the moving step size of the pinhole camera and the normalized pixel size of the pinhole camera's sensor, respectively.

Then we can derive the sampling grid with $\mathbf{A}^{-1}\mathbf{T}$ as

$$\mathbf{A}^{-1}\mathbf{T} = \begin{bmatrix} 1 & -b \\ \frac{1}{f} & 1 - \frac{b}{f} \end{bmatrix} \begin{bmatrix} T_x & 0 \\ 0 & T_p \end{bmatrix} = \begin{bmatrix} T_x & -bT_p \\ \frac{1}{f}T_x & -\frac{b}{a}T_p \end{bmatrix}. \tag{4.2}$$

Here, $\mathbf{A}^{-1}\mathbf{T}$ specify the sampling periods on the continuous light-field $L(x, p)$ that is defined on the plane of the light field camera. To be more specific, the columns of the matrix $\mathbf{A}^{-1}\mathbf{T}$ are the sampling periods in terms of the pinhole camera and pixel, respectively. The first column $[T_x\ f^{-1}T_x]^T$ determines the distance between pixels within the same pinhole-camera, and the second column $[bT_p\ -ba^{-1}T_p]^T$ determines the distance between the same pixel under two consecutive pinhole cameras in the light field.

With the light-field camera as a reference, we demonstrate the rectangular sampling-grid on $L'(x, p)$ defined on the pinhole camera plane and the parallelogram sampling-grid on $L(x, p)$ defined on the main lens in Figure 4.4.

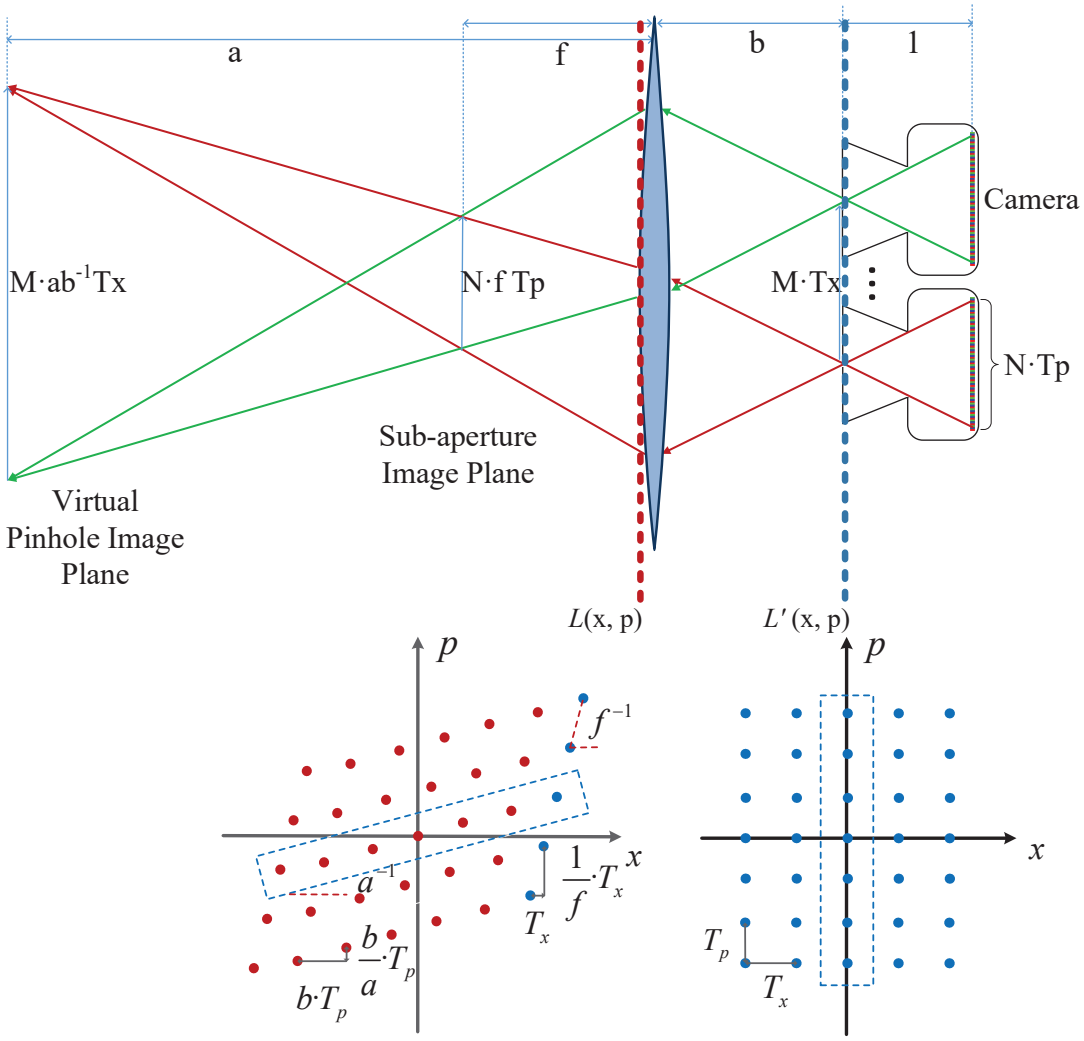

Figure 4.4: An illustration of the sampling grids with the reference of the simplified light-field camera. The light-field camera is defined by a pinhole camera moving behind a thin lens with a step size of T_x. The light field $L(x,p)$ is defined on the main lens plane whereas $L'(x,p)$ is defined on the pinhole camera plane. The transformation between $L(x,p)$ and $L'(x,p)$ is a combination of the refraction $\mathbf{A}_f$ and the propagation $\mathbf{A}_b$.. The two sampling grids are showed for $L(x,p)$ and $L'(x,p)$, respectively. In the sampling grid, the pixels from the same pinhole camera are shown inside the dashed rectangles.

With the sampling grid of a light field camera defined, we can now further identify its motion models. Here we focus on two basic motions: translations and rotations.

Translations of light-field cameras

In this section, we derive the motion model for camera translation, specifically when the light-field camera is shifted in a plane parallel to the main lens and pinhole-camera plane. The 2-D light-field data $L_0(x,p)$ and $L_1(x+\Delta x,p)$ are acquired by moving the light-field camera for Δx meters in the world coordinates. Then the discrete translation $(\Delta m, \Delta n)$ between these two acquired light-fields is as follows

$$I_0(m,n) = I_1(m+\Delta m, n+\Delta n).$$

By using the sampling period $\mathbf{A}^{-1}\mathbf{T}$ from (4.2), we can map the camera translation to the pixel shift in the discrete light field as follows

$$\mathbf{A}^{-1}\mathbf{T}\begin{bmatrix} m \\ n \end{bmatrix} = \begin{bmatrix} x \\ p \end{bmatrix} \mathbf{A}^{-1}\mathbf{T}\begin{bmatrix} m+\Delta m \\ n+\Delta n \end{bmatrix} = \begin{bmatrix} x+\Delta x \\ p \end{bmatrix}.$$

This can be further simplified by removing the common terms to derive

$$\boldsymbol{A}^{-1}\boldsymbol{T}\begin{bmatrix} \Delta m \\ \Delta n \end{bmatrix} = \begin{bmatrix} \Delta x \\ 0 \end{bmatrix},$$

which also leads to the linear relation between Δm and Δn as

$$\Delta n = \frac{a}{f\cdot b}\frac{T_x}{T_p}\Delta m. \tag{4.3}$$

Therefore, the 1-D camera translation results in a 2-D shift in the discrete light field data with only one degree of freedom. The ratio between the pixel shift in angular and spatial dimension is determined by the parameters of the light-field camera as shown in (4.3).

Rotations of light-field cameras

In this section, we derive the motion model for camera rotations around the optical center of the main lens. As shown in Figure 4.5, a 2-D example is used to show how a light-field camera is rotated around its optical center by an angle ϕ clockwise. Here the blue line with two arrows is used to represent the original position of the light-field camera's main lenses, and the green one is used to represent the position of the main lens after rotation. The dashed line is used to denote the same light-ray captured through these two main lens. The variable $L_r(x_r,p_r)$ is used for the light field after rotation to distinguish it from the original light-field $L(x,p)$.

By using geometric relations between (x_r,p_r) and (x,p), we conclude that

$$p = \frac{p_r - \tan\phi}{1+p_r\tan\phi} = \tan(\arctan(p_r)-\phi), \tag{4.4}$$

and

$$x = x_r\sin\phi\cdot p + x_r\cos\phi. \tag{4.5}$$

In Equation (4.4) and (4.5), we observe that the light-field transformation by camera rotation is not a linear operation in the p dimension. As for the transformation in the x dimension, the shift depends on both the camera rotation and the direction p of the light ray. Only when the rotation angle ϕ is very small, then the light-field transformation by camera rotation can be

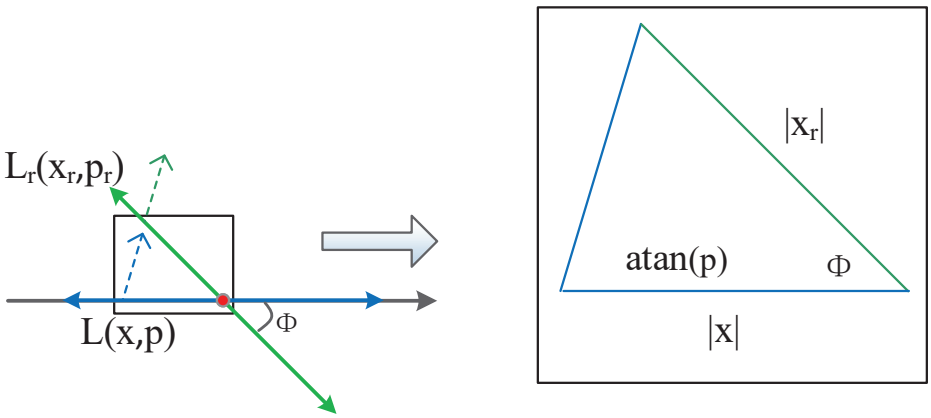

Figure 4.5: The light-field camera is rotated by ϕ around the optical center of its main lens. The dashed lines $L(x,p)$ and $L_r(x_r,p_r)$ denote the same light ray captured by the light-field camera before and after rotation. The geometric relations between (x_r,p_r) and (x,p) are shown on the left.

approximated as a linear transformation. Therefore, extending the light field in p dimension by camera rotation can only work for small rotation angles due the non-linearity of the rotations of the light-field camera. As ϕ increases, the parallelogram shape of the sampling pattern after rotation cannot hold when aligning to the original light-field. Hence, light-field stitching by camera rotations can only work for the view extension in a small range.

4.3 Experiments and Discussions

With the guidance of the derived motion model, we use a light-field camera as a scanner and perform light-field stitching to increase the size of the acquired light-field data. We report two applications in light field stitching: stitching by camera translations and stitching with a combination of camera translations and rotations. The main goal of our simulations is to verify the motion model and demonstrate its potential for light-field acquisition.

4.3.1 Light-field stitching by camera translations

By using Equation (4.1), we establish the relations between the camera motion and the acquired light field data defined by the pinhole-camera index m and the pixel index n. Then we can simply capture multiple light-fields as shown in Figure 4.6. To fully use each entry of the captured light-fields, we design the translation such that there are no overlapping areas between neighboring light-fields.

A direct stitching approach is demonstrated with the blue dashed parallelogram. Each pinhole-camera image is extended at the cost of reducing the size of the sub-aperture image (same pixel index in all pinhole cameras). The reduction in spatial range is actually due to the fact that the captured data has the shape of a parallelogram in the 2-D light field defined on the main lens plane of the light-field camera.

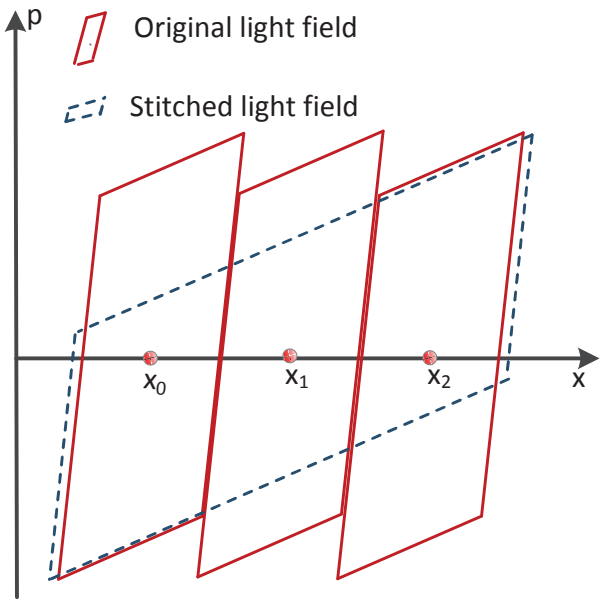

Figure 4.6: The red parallelograms represent the light field taken by light-field cameras at x_0, x_1 and x_2 and the blue parallelogram with dashed line represents a direct stitched light field by extending the pinhole camera image.

To avoid reducing the size of the sub-aperture image, we propose to re-sample the captured light-field to increase the field of view. As mentioned in Chapter 2, slices in the 2-D light field can be seen as 1-D images that are captured by virtual cameras positioned at planes, where depth values correspond to the directions of the slices. By carefully choosing the slicing slopes in the light field, the sub-aperture image represented with the blue dashed line obtains a wider field of view as shown in Figure 4.7.

Based on the strategy in Figure 4.7, a synthetic light-field camera designed in Matlab is used to capture multiple light fields at a close distance to a slanted plane painted with pink circles. The texture is chosen to be a band-limited signal. As the texture is painted on a slanted plane, the light field is also a band-limited signal as shown in [21]. The Nyquist condition is satisfied by the sampling periods in the 4-D light field. The dimension (n_p, n_q, m_x, m_y) of each acquired light-field data is $21 \times 21 \times 15 \times 15$. A total number of 31×31 light field acquisitions are used for light-field stitching. One stitched sub-aperture image of the final panorama light-field is shown in Figure 4.8 with a resolution of $(31 \times 21) \times (31 \times 21)$.

4.3.2 Light-field stitching by camera rotations and translations

We can also perform light field stitching by combining camera rotations and translations. Hence, we can extend each pinhole image directly without resampling.

As shown in Figure 4.6, pinhole-camera images at the boundaries of the light-field acquisition cannot be extended because of the parallelogram shape. By rotating the light-field camera, acquired light-fields are shifted in the p dimension in order to extend each pinhole camera image,

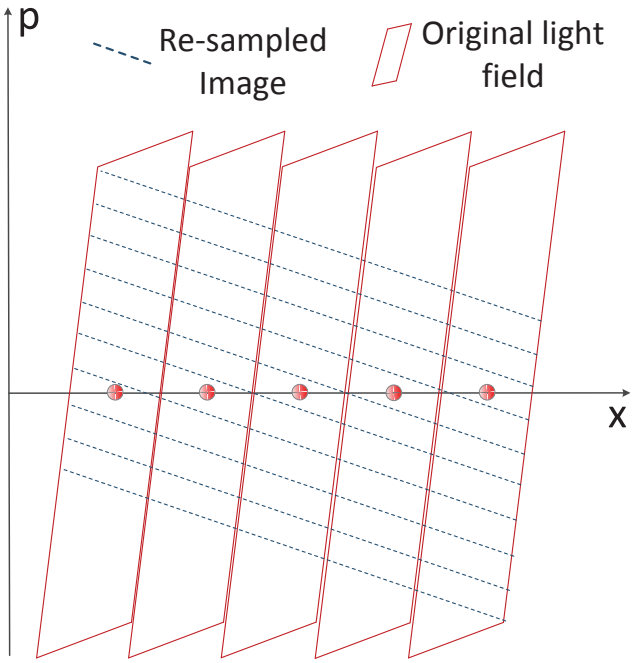

Figure 4.7: Resampling multiple light fields to create a virtual light field with larger field of view.

as shown in Figure 4.9.

The simulation results of a 2-D light-field stitching by both camera rotations and translations are given in Figure 4.9. Each light-field camera with a different translation is rotated around the optical center of its main lens. The specific camera parameters for the simulation is shown in Table 4.1. The parameters are similar to those of a Lytro.

Then we first use Equation (4.2) to calculate the range of each pinhole-camera image in x dimension as follows

$$\Delta x = N \cdot T_p \cdot b,$$

where N represents the size of each pinhole-camera image, T_p represents the sampling period in p for each pixel and b represents the distance between the main lens and the pinhole-camera

light-field camera parameters for simulation			
Main Lens	f	a	b
	6.45 mm	0.3 m	6.9 mm
Microlens Array	pitch	f	b
	1.4×10^{-5} m	2.5×10^{-3} mm	2.5×10^{-3} mm
Sensor	pitch	angular dimension	spatial dimension
	1.4×10^{-6} m	11× 11	301 × 301

Table 4.1: Specific camera paramters for light-field stitching simulation.

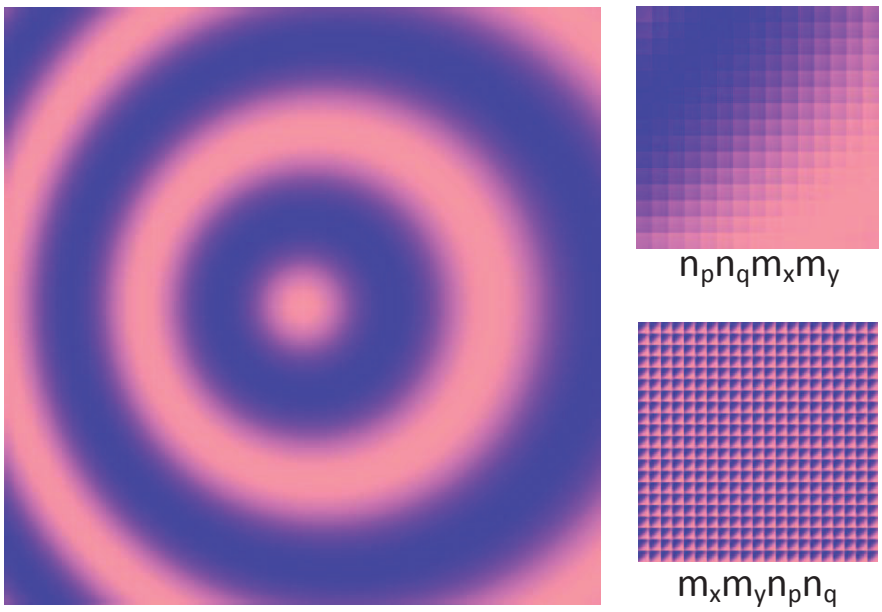

Figure 4.8: A panorama image created from 31×31 light field images. One sub-aperture image of the final panorama light field is on the left and one acquired light field image with different display orders (n_p, n_q, m_x, m_y) (an array of microlens image) and (m_x, m_y, n_p, n_q) (an array of sub-aperture image) is on the right.

plane. Then the camera is translated for Δx to maximize the stitched range in x.

As was shown in Figure 4.4, each pinhole-camera image is a slice in the light field with a slope $1/a$. To compensate for the parallelogram shape, each pinhole camera image should be shifted by $\Delta x/a$ in the p dimension. By using Equation (4.4) and (4.5), the light-field camera is rotated around the main lens' optical center by an angle Φ as follows

$$\Phi = \arctan(\frac{\Delta x}{a}). \tag{4.6}$$

As shown in Figure 4.9, five light fields are generated with different translations ($-2\Delta x$, $-\Delta x$, 0, Δx and $2\Delta x$). To compensate for the shift in the p dimension, each camera is rotated for an angle calculated by Equation (4.6). From the simulation results in Figure 4.9, we can observe that the required range of the light field, to extend each pinhole-camera image, is covered by the acquired data. Further treatments for interpolation are also needed, especially when ϕ increases.

In conclusion, light field stitching by camera rotations and translations effectively increases the size of each pinhole camera image of the light field. However, camera rotations are usually more difficult to implement. Due to the non-linearity of the light field transformation by camera rotation, this stitching method works effectively only for when the rotation angle ϕ is small.

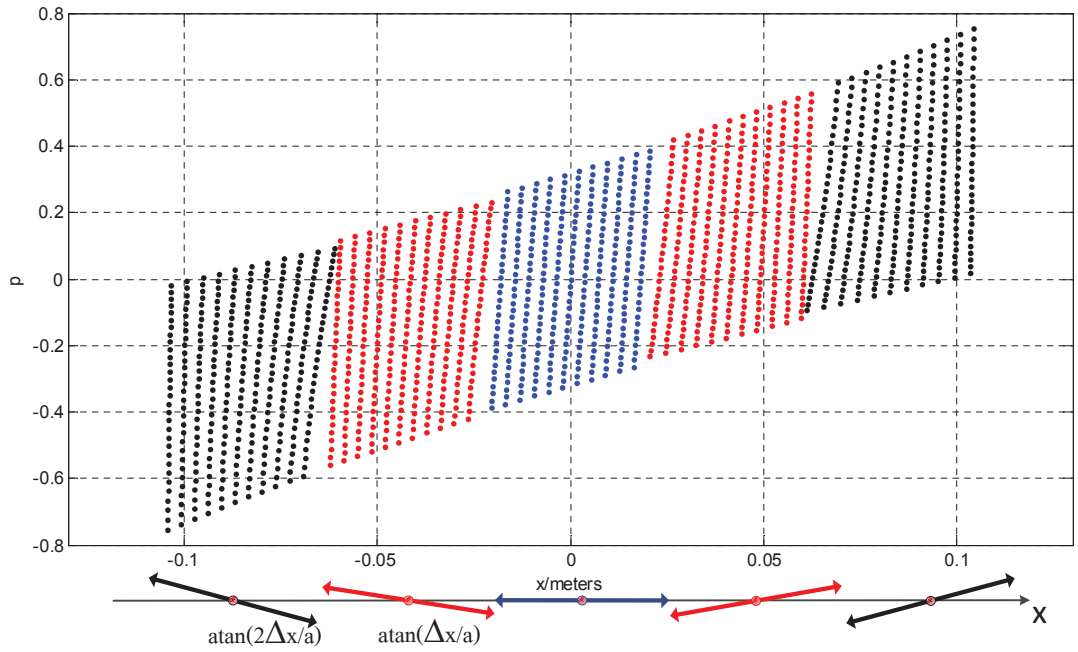

Figure 4.9: Five light-field cameras with different translations $(-2\Delta x, -\Delta x, 0, \Delta x, 2\Delta x)$ and rotations are used to sample the light field. As ϕ increases, the distortion of the parallelogram shape becomes more obvious.

4.4 Discussions of Light-Field Registration and Conclusions

In our simulations in Section 4.3, we assume we know the exact position of the light-field camera. In practice, we need to devise algorithms based on motion models, to estimate the exact motion.

In this section, we propose and discuss two possible registration algorithms for the linear motion models described in Section 4.2 and then conclude this whole chapter.

4.4.1 Extensions to registration algorithms

In this section, we propose two different types of approaches: one direct approach is to warp the light fields relative to each other and to check how much the pixels agree; the other is a feature-based approach that seeks unique features to register light fields with large motions.

Light-field registration in the frequency domain

Here we only estimate the planar shift between the reference light-field L_1 and an other light-field L_2, by using a frequency domain algorithm. Although the frequency domain methods are limited to global motion, they can be computationally efficient and capable of dealing with aliasing in the acquired light-field.

In addition to the constraints that there are only the planar motions, we also assume the shift

to be very small such that there are large overlapping areas between light fields. This constraint makes the frequency-based registration more suitable for light field super-resolution rather than the creation of panorama.

Two light fields L_1 and L_2 are taken on a plane parallel to the sensor plane of the light-field camera. The motion between two camera positions is a planar shift $(\Delta x, \Delta y)$ without any rotation. We use $\mathbf{n}$ to represent 4-D index vector (m_x, n_x, m_p, n_p) and $\Delta\mathbf{n}$ to represent the 4-D shift $(\Delta m_x, \Delta n_x, \Delta m_p, \Delta n_p)$ between these two light-fields as follows

$$L_2(\mathbf{n}) = L_1(\mathbf{n} + \Delta\mathbf{n}).$$

Then the linear operator of the 4-D light-field camera $\mathbf{A} = \mathbf{A}_f\mathbf{A}_b$ can be represented respectively as

$$\mathbf{A}_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -f^{-1} & 0 & 1 & 0 \\ 0 & -f^{-1} & 0 & 1 \end{bmatrix}$$

and

$$\mathbf{A}_b = \begin{bmatrix} 1 & 0 & b & 0 \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where b denotes the distance between the main lens and the moving pinhole camera and f denotes the focal length of the main lens. Then the sampling period $\mathbf{T}$ is represented as

$$\mathbf{T} = \begin{bmatrix} T_x & 0 & b & 0 \\ 0 & T_y & 0 & b \\ 0 & 0 & T_p & 0 \\ 0 & 0 & 0 & T_q \end{bmatrix},$$

where T_x and T_y represent the moving step size of the pinhole camera and T_p and T_q represent the normalized pixel size of the pinhole camera.

With these operators, we can establish the relationships between the planar shift of the camera and the pixel shift in the 4-D light fields as

$$\mathbf{A}^{-1}T \begin{bmatrix} \Delta m_x \\ \Delta n_x \\ \Delta m_p \\ \Delta n_p \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \\ 0 \\ 0 \end{bmatrix}. \tag{4.7}$$

Then we formulate the registration problem in the frequency domain as follows:

$$\mathcal{L}_2(\mathbf{k}) = e^{j2\pi\mathbf{k}^T(\Delta\ \mathbf{n./N})}\mathcal{L}_1(\mathbf{k}) \tag{4.8}$$

where $\mathcal{L}_1(\mathbf{k})$ and $\mathcal{L}_2(\mathbf{k})$ are used to represent the Fourier transforms of L_1 and L_2. The constant N is the length of the data for normalization.

By applying Equation (4.7) to Equation (4.8), we have a set of linear equations to estimate the shift between two light fields as follows:

$$\mathbf{k}(T^{-1}\mathbf{A})_{[:,1:2]} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \angle\left(\frac{\mathcal{L}_2[\mathbf{k}]}{\mathcal{L}_1[\mathbf{k}]}\right),$$

where **[:,1:2]** indicates that only the first two columns of the matrix are kept and $\angle$ denotes an operator to estimate the angle of the phase.

In 4-D light fields, the spatial domain shifts only affect the phase values of the Fourier transforms. The properties of the light-field camera map the 4-D shifts to the 2-D motion vector. By estimating the phase difference between the Fourier transforms of the two light fields, we can directly derive the planar motions with high accuracy.

However, this method only works for scenarios when small translations occur. Hence, in the following section, we also propose a feature-based registration method for large camera motions.

Light-field registration by using the intersection points

In this section, we discuss how to stitch multiple light-fields with planar motions into a light field with a larger spatial and angular range. They usually do not share overlapping areas in the angular domain. Therefore, the frequency-based registration algorithm is no longer feasible.

Let us consider a special case when we register light fields of a fronto-parallel plane with a Lambertian surface as shown in Figure 4.10. For these two light fields, the ambiguity in the angular domain makes it impossible to register the positions of the light-field cameras.

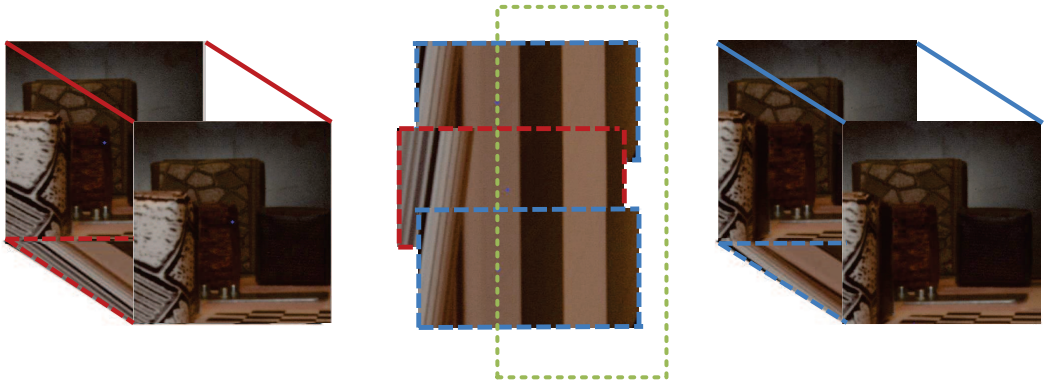

Figure 4.10: An example of registering two light-fields. The 3-D light fields are captured by two light field cameras with a translation. For the sake of simplicity, we illustrate the registration between two 2-D light fields shown in the middle. In the area within the green dashed rectangle, the scene is fronto-parallel planes with Lambertian surfaces. We can observe that the 2-D light field from the right dataset can be registered either to the top or the bottom of the 2-D light field from the left dataset. We cannot achieve a unique registration result because of the ambiguity in the angular domain.

Quite the contrary to a fronto-parallel plane, occlusions in the scene can remove the ambiguity and enable us to find the correct motion. When an occluded area is present in the scene, we can observe line intersections in the 2-D light field as shown in Figure 4.11.

The intersection in the 2-D light field indicates one specific view from which two spatial samples at different depths are perfectly aligned. The view direction is unique for these two spatial samples, and we define the intersection as a feature descriptor **INT** as

$$\mathbf{INT}(m, n) = (m, n, s_0, s_1, \mathbf{t}_0, \mathbf{t}_1),$$

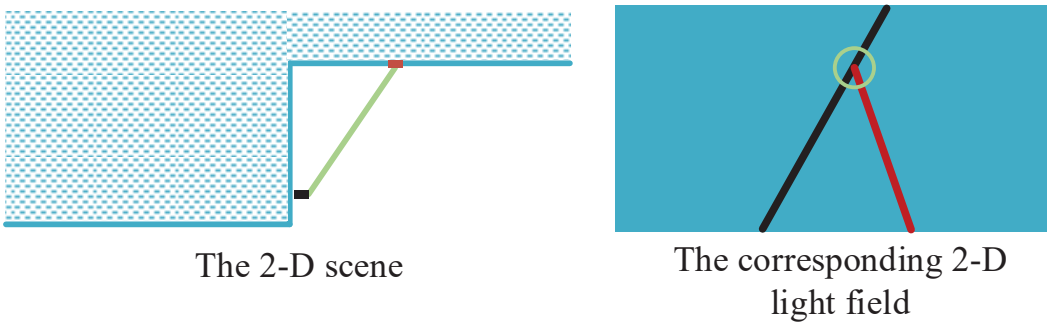

Figure 4.11: An example of a scene with occlusion and its corresponding light-field. On the left, we show a 2-D scene with two fronto-parallel planes. The surfaces are both blue except for two points painted with black and red. On the right, we show the corresponding 2-D light field and use a green circle to emphasize the intersection, which also means the black point occludes the red point. Furthermore, the intersection actually represents the green line that connects the black and red points.

where (m, n) denotes the position of the intersection in the 2-D light field, s_0 and s_1 denote the slopes of the two intersected lines and $\mathbf{t}_0$ and $\mathbf{t}_1$ denote the texture information along these two lines. The texture information can be pixel values, local patches and feature descriptors such as scale-invariant feature transform (SIFT) [35], histogram of oriented gradients(HOG) [19] and speeded up robust features (SURF) [8]. Then we can estimate the camera motions by matching the intersection feature **INT**.

However, the occlusions are not always presented in the 2-D light field because one light field corresponds only to one horizontal or vertical line in the scene. To detect more feasible features, we extend the intersections into 3-D light field that is a sequence of 2-D images instead of 1-D line samples. Instead of finding intersections within one light-field, we also detect virtual intersections between lines from different 2-D light fields. These intersections are no longer view directions, but unique planes in a given 3-D world that can also be used to register two light-field cameras.

As for the 4-D light field, we can just use the 3-D light fields by fixing the angular dimension n_p or n_q. Then we have two sets of virtual planes that can be used to estimate the 2-D planar shift of the light-field camera.

4.4.2 Conclusions

In this chapter, we have derived the motion models of a simplified light-field camera and presented the applications in light-field stitching. The light-field camera is used as a scanner and multiple light-fields are merged into one light-field with larger spatial range or angular range.

We have presented two applications. First, we capture multiple light fields by camera translations and then resample them to create light field panoramas. By slicing the acquired multiple light-fields with different slopes, each sub-aperture image of the new light-field has a wider field of view.

Second, by combining camera rotations and translations, the angular range of the light field can be increased directly. The motion model of the camera rotation is described and verified

by the simulation of a 2-D light field, which can be easily operated in 4-D. But this type of acquisition is limited to a certain angular range because of the non-linearity of the light field transformation by camera rotations.

However, light-field registration is still a challenging problem. To deal with this problem in practice, we have proposed two plausible algorithms: a frequency-based global algorithm and a feature-based algorithm. In the future, we will further explore the possibility to utilize these methods for real-life registration-problems of light field cameras.

Chapter 5

Circular Light-Fields for Virtual Reality

> There are painters who transform the sun to a yellow spot, but there are others who with the help of their art and their intelligence, transform a yellow spot into sun.
>
> *Pablo Picasso*

In this chapter, we propose a novel representation, a circular light-field to model light rays in a given 3-D space for virtual-reality applications. The circular light-field is a simplification of the 7-D plenoptic function, which is specifically designed to render novels views with a 360-degree field of view at any chosen location within a given area.

The circular light-field is a powerful tool for the acquisition and rendering of indoor environments. It can be created simply with a set of standard cameras mounted on a circular rig. It also has good extensibilities. By registering and stitching multiple circular light-fields, we can easily create the data for rendering novel views in a much larger area compared with the size of the circular rig. The circular light-field shows great potential in image-based rendering for virtual-reality applications.

This chapter is organized as follows: We first introduce the 2-D circular light-field in Section 5.2. In Section 5.3, we demonstrate how to perform circular light-field registration and super-resolution, both of which are important techniques for exploiting the properties of the circular light-field in practice. We then extend the 2-D circular light-field to 3-D and 4-D circular light-field in Section 5.4 and conclude the chapter in Section 5.5.

5.1 Introduction and Related Work

As technology improves, users are able to interact with visual displays to experience new locations, new views, new activities, etc. through virtual-reality systems [45]. This is usually realized by users wearing a virtual-reality goggle, which combines a screen, a gyroscopic sensor and an accelerometer. In the current market, there are many excellent options for this type of devices such as Oculus Rift, HTC vive, Playstation VR and Google Cardboard [15]. By using them, in real time, we can render an interactive photo/video that corresponds to the head and body movement of the user.

In our work, we focus on the virtual-reality applications for real scenes. Contents generated with computer-graphics techniques are beyond the scope of this chapter. The very idea to use light field to create interactive 3-D experiences originates from the computer graphics community when they discovered it to be capable of using a collection of images to interpolate intermediate views [16, 17, 20, 25, 32, 37, 41, 43]. And the techniques they use are usually referred to as image-based rendering.

Although image-based rendering has been a hot topic for more than a decade, in both the industrial and academic worlds, generating the contents of real scenes for virtual reality is still a challenging problem.

Here, we refer interested readers to a detailed review on this topic from Shum and Kang [42]. As they stated, the various rendering techniques are classified into three categories: rendering with explicit geometry, rendering with implicit geometry, and rendering with no geometry.

The techniques of rendering with explicit geometry can be tracked back to the work of Debevec, et al. [20] in 1996. The geometric and photometric model is reconstructed from a collection of photographs. Many efforts have been put into this topic to increase the accuracy of the models and reduce the consuming time of the algorithm as sometimes it is quite difficult to fully recover some real environments. In addition to this, it is still very challenging to render a high quality 3-D model as it comprises a large amount of vertices.

The techniques of rendering with implicit geometry require features and correspondences between images for view interpolation. Without the explicit geometric models, novel views are generated by interpolating optical flow between corresponding points [17] or by in-between camera matrices along the line of two original camera centers from view morphing [41].

For both of these techniques, as the scenes become more complicated, not only the cost for content generation but also the cost for the rendering devices (graphics card) increases substantially.

However, is geometry information really necessary for this task? As for virtual-reality contents, the primary application is only to render images at a collection of view points, in which case a complete 7-D plenoptic function [1] is all that is needed.

As we introduce in Section 2.1.1, although the high dimensionality of the plenoptic function makes it very difficult for the acquisition, there are many ways to simplify the 7-D function to capture and render efficiently. [1] Here we illustrate three different types of simplifications for creating novel views for virtual reality.

The most straightforward way is to simplify the application scenario, where the user only changes the view angle, without moving the viewing position. Then the plenoptic function becomes a 2-D panorama video or photo (if fixing time) that is widely used for virtual-reality

1. Note that the wavelength is usually simplified to three color channels.

applications. Nowadays, it is very popular to create movies, animations and even to broadcast sports events in 360 degree. The panorama photo is a well studied research topic. We refer the interested readers to the technical report by Szeliski [46] for a comprehensive review of this topic. This kind of technology enables the interactive rendering corresponding to the head motions of the user. In the recent Google Jump Project, a 16-camera rig is used to capture a panorama video. To render a stereo panorama video, the depth map of the scene is also estimated for each pixel. The whole operation requires high computational power, thus the captured videos are processed by remote servers from Google instead of a local computer. However, it is still not ideal for creating virtual reality contents as the users can only change the viewing angle but not the viewing location.

Another simplification of the plenoptic fucntion is the 4-D light field that we discussed in Chapter 2. The 4-D data is captured by a set of cameras that are positioned on a pre-defined camera plane, which means the scene is constrained to a bounding box. With a 4-D light field, the position of a rendered novel view can be moved freely on the camera plane and moved towards or away from the scene. All the rendered views are from the bounded box captured by the cameras on the pre-defined plane. We also discussed how to create a light-field panorama in Chapter 4. It is quite challenging to both acquire and render the light field. As the freedom of the movements increases, the complexity of the acquisition increases dramatically.

The third simplification, a concentric mosaic shares the most similarity to our proposed method. He and Shum proposed the concentric mosaic for simplifying the plenoptic function without compromising its rendering ability in 2001 [43]. The concentric mosaics are created by constraining the camera motion to planar concentric circles and taking slit images at different locations along each circle. The acquisition device is a rotating robot arm, on which a set of cameras are positioned towards the tangent direction. Each pixel on the slit image corresponds to a light ray parameterized by three parameters: the radius, rotation angle and vertical elevation. Whenever a novel view is chosen, the corresponding light rays are selected based on their parameters to render the image. The rendered views usually covers the circular region with a radius that equals to the maximum radius of the slit camera. Although the system is still bulky, it provides more viewing freedom compared to the light-field acquisitions with similar complexity. And compared with panorama photos, this technology provides a much better visual experience.

Nevertheless, there are still some limitations of this method. First, to render the off-the-plane light rays, the depth information is required to correct the vertical distortions. Second, the slit camera is not a standard and efficient device for capturing images. Finally, the observing area depends on the maximum radius of the slit camera, which is also the length of the robot arm. It substantially limits the range of novel views.

In a short conclusion, the ideal virtual-reality contents should satisfy the following requirements. First, it should provide the ability to render novel views at any chosen location for any chosen viewing angles. Second, it should be captured in an efficient way without depth reconstruction because not only the creation but also the rendering of a 3-D model requires high computational power for both the software and hardware. Finally, it should be captured with a standard or easy-to-build imaging system.

In this section, we propose a novel model for simplifying the plenoptic function, a circular light-field. As the 4-D light field is parameterized with two parallel planes, we use two concentric circles to parameterize the light rays. By using this, we can render novel views within an area much larger than the circles themselves, without any 3-D reconstruction. To fully record a given scene and create high quality contents for virtual-reality applications, we propose a framework

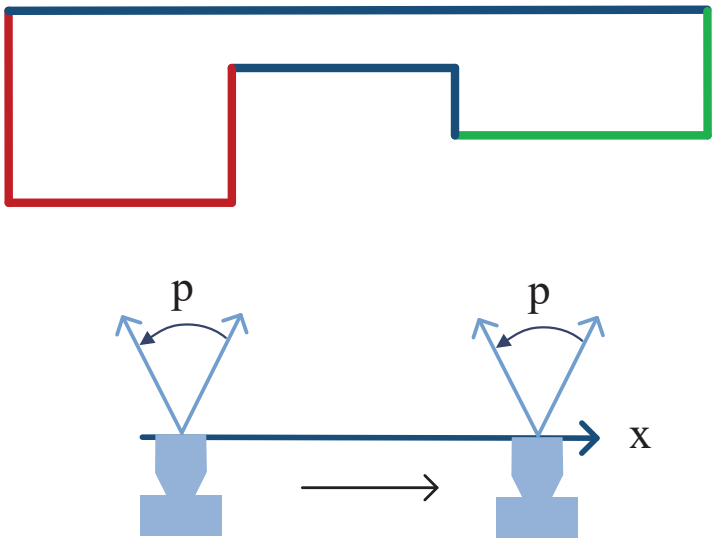

Figure 5.1: Creating a 2-D light field with a moving 1-D pinhole camera.

for capturing, registering and merging multiple circular light-fields.

5.2 2-D Circular Light-Fields

In this section, we introduce a novel representation, the circular light-field for simplifying the plenoptic function for the rendering of virtual-reality applications. Similarly to the standard light field, the circular light-field is also used to represent light rays in a given 3-D space. However, the standard light-field is used to render views of a scene within a bounded box that is determined by the field of view of the light-field camera, whereas the circular light-field is proposed to render novel views with a 360-degree viewing angle at any chosen location in a given area.

In this section, for the sake of simplicity, we carry out the analysis in a 2-D world, in which a 2-D circular light-field is used to represent light rays. Without loss of generality, we later extend the 2-D circular light-field to 3-D and 4-D circular light-field in the following sections.

5.2.1 Motivations

A standard 4-D light field is represented with a two-plane parameterization, in which each light ray is uniquely determined by its intersection with two predefined planes parallel to each other. The 4-D index of each light ray is represented with the coordinates of the two intersections on these planes, respectively.

Here we demonstrate the rendering problem with the 2-D light field and then the two parallel planes become two lines. The 3-D space also becomes a 2-D plane that can be seen as a top view of the original scene. The most simple setup to capture a 2-D light field is to use a 1-D pinhole camera that moves on a line as shown in Figure 5.1. Both the pinhole camera's field of view and its moving range determine the field of view of the acquired 2-D light field.

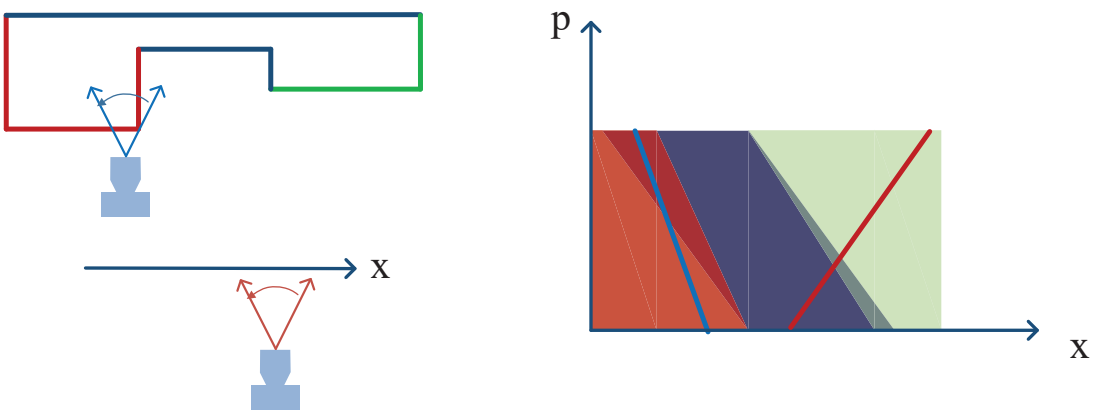

Figure 5.2: The novel views and the corresponding slicing lines in the 2-D light field. We use the blue view in the scene to represent a novel viewing position that is closer to the scene, compared with the original camera plane. The novel view is created by slicing the 2-D light field with the blue line, as shown on the right. The slope of the line is determined by how far the novel view moves from the original camera plane. We use the red view and a red slicing line to represent a novel viewing position behind the original camera plane.

As shown in Figure 5.2, by using the acquired 2-D light field, we can render novel views, and we can create a walk-through for users to interactively view the captured scene. A user can move freely on the line and has the same field of view as the pinhole camera. The user can also move towards the scene to observe the details (illustrated by the blue view in the scene and blue slices in the 2-D light field), or move away from the scene to see the whole scene (illustrated by the red view in the scene and red slices in the 2-D light field). When the user is close to the scene, the field of view becomes narrower compared with the pinhole camera; whereas when the user is far away from the scene, the field of view becomes wider compared with the pinhole camera.

In conclusion, with the standard light field, the novel views can be rendered for areas in front of the scene within a limited range. However, this rendering technique only provides a limited view that is clearly smaller than 180 degree. The walk-through path is heavily constrained by the range of the line on which the pinhole camera moves. Therefore, to address the rendering problem, we propose a novel representation in which novel views with a 360-degree viewing angle can be rendered at any chosen location in a given area.

5.2.2 Definitions and notations

Our goal is to design a representation for light rays such that we can render novel views at various positions with a 360-degree viewing angle. Hence, we move away from the standard approach that uses two lines to parameterize the 2-D light field or uses two planes to parameterize the 4-D light field. Instead, we turn to an omnidirectional representation and propose a novel representation, the circular light-field in which we represent a given light ray with its intersections with two concentric circles.

Although using circles to represent a light field seems to be quite similar to the creation of a panorama photo, there are fundamental differences between our proposed representation and

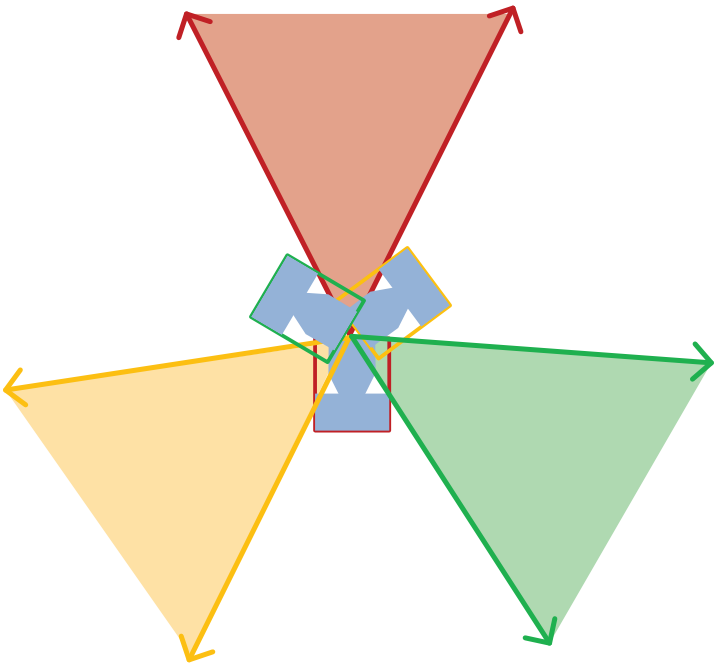

Figure 5.3: Creating a panorama photo by rotating a pinhole camera around the pinhole. The rotation is around the pinhole that is the optical center of the camera such that the depth of scene does not affect the registration and stitching process. Note that as we use the pinhole camera here, the optical center locates at the pinhole.

the panorama technique. When creating a panorama photo, the camera is rotated around its optical center such that the depth of the scene does not affect the registration and stitching process, as shown in Figure 5.3. Hence, a panorama image can not specify the directions of the recorded light rays. But by using the two concentric circles, the direction of each light rays can be uniquely specified.

The scene is a 2-D space that is represented with the $x-y$ plane. The object or light origin in the 2-D space can be either represented with (x, y) or (z, ψ) where

$$z = \sqrt{(x^2 + y^2)}, \ \psi = \arccos \frac{x}{\sqrt{x^2 + y^2}}.$$

The index (z, ψ) is actually the polar coordinates of the 2-D space.

By using the two-concentric-circles parameterization, as shown in Figure 5.4, each light ray is represented by its two intersections ϕ and θ with the two concentric circles, respectively. To be more specific, ϕ is the angle of the intersection that is determined by the light ray and the outer circle, and θ is the relative angle of the intersection on the inner circle to the line that connects the intersection ϕ on the outer circle and the circle center, as shown in Figure 5.9. Thus, each light ray in 2-D space is represented with a circular light-field $L(\phi, \theta)$.

Note that the variables x, y, z and ψ are all used to describe the locations in the 2-D space, and θ and ϕ are used to represent the index of the circular light-field.

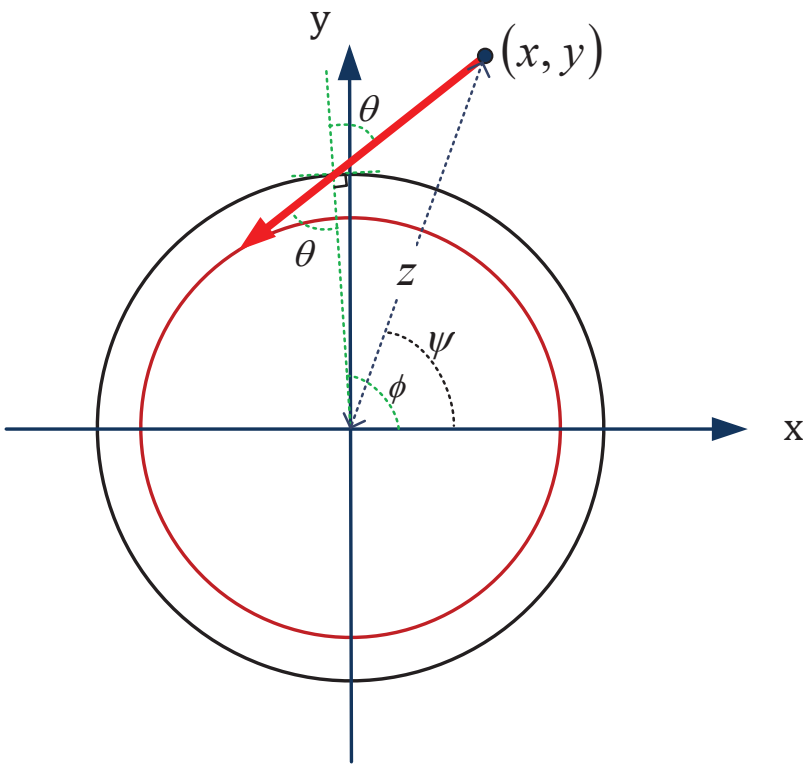

Figure 5.4: Illustration of the 2-D circular light-field representation of a red light ray. For the sake of clarity, we use the blue dashed-lines to illustrate the information of the origin of the red light ray whereas we use the green dashed-lines to illustrate the intersections of the light ray with the concentric circles. The red light ray comes from the position (x, y), or (z, ψ) in the polar coordinates. It intersects with the outer circle at the angle ϕ. To record its intersection with the inner circle, we use θ to represent the angle between the line that connects the two intersections and the line that connects the circle center and the intersection on the outer circle.

We draw a comparison between the standard light-field $L(x, p)$ and the circular light-field $L(\phi, \theta)$. In the standard 2-D light field $L(x, p)$, we see the index x as the intersection on the camera plane, whereas we see the index p as the direction of the corresponding light ray. In the 2-D circular light-field $L(\phi, \theta)$, the index ϕ is the intersection of the light ray with the outer circle, whereas $\tan\theta$ is the direction of the recorded light ray. Thus the circular light-field $L(\phi, \theta)$ is much more flexible for rendering novel views because circles, unlike the planes or lines, do not face towards a specific direction.

5.2.3 Creation of circular light-fields

In this section, we discuss how to create a circular light-field. Based on the definition of the circular light-field, the variable ϕ defines the location, whereas the variable θ denotes the direction. To capture a given light ray, we can simply put a pinhole camera at the angle ϕ on the outer circle. By aligning the optical axis of the pinhole camera such that it passes through

the center of these two concentric circles, the light ray $L(\phi, \theta)$ is actually the pixel located at $f \tan\theta$ on the camera sensor, where f denotes the focal length of the pinhole camera.

To sum up, the 2-D circular light-field is created by capturing a set of photos with multiple 1-D cameras that are positioned on a circular rig, as shown in Figure 5.5. For static scenes, we can also use only one 1-D camera that moves along the circular rig and takes the 1-D images sequentially. Note that for both cases, the 1-D cameras face outwards and its optical axis passes through the center of the circular rig.

In this setup, the radius of the circular rig is denoted with r, and the focal length of the camera is denoted with f. As for the images captured by the pinhole camera, their coordinates are denoted with the variable p where

$$p = f \tan\theta.$$

Thus we can also use $L(\phi, p)$ to represent the circular light-field. Then the 1-D image captured at location ϕ_0 on the circular rig can be represented as

$$L(\phi_0, p) = L(\phi_0, f \cdot \tan\theta).$$

For the sake of clarity, we use the two concentric circles simply to make a comparison to the two-plane parameterization. For any given light ray in the 2-D circular light-field, we are only interested in the angle of the light ray. The location of the light ray's intersection with the inner circle is not necessary for our representation.

We do not need to give any specific definitions on the inner circle. For a more clear illustration, we set the radius of the inner circle to be $r - f$. Under this setting, the image plane of the pinhole camera becomes a tangent of the inner circle. For the sake of simplicity, we also normalize the focal length f to 1. Therefore, we simply use $L(\phi, \tan\theta)$ and $L(\phi, p)$ to represent the circular light-field.

5.3 Applications of Circular Light-Fields

In this section, we propose to use the circular light-field for rendering novel views with a 360-degree viewing angle at any chosen location in a given area. This new representation of light rays is much more flexible, compared with the panorama photo and the standard light field.

We first demonstrate the rendering of the circular light-field. The positions of the rendered view can cover the full space when we assume that no occlusions are present in the scene. However, the field-of-view of the rendered image decreases as the distance between the viewing position and the center of the circular rig increases.

Obviously, a single circular light-field is not enough to achieve the goal of rendering novel views with a 360-degree viewing angle at any chosen location. Hence, in practice, we need to capture and exploit multiple circular light-fields in an efficient manner. We address this problem in two ways. First, we demonstrate how to register multiple circular light-fields to increase the field of view for rendering. Second, we show that by using super-resolution techniques, we can reduce the required resolution of the acquired circular light-fields, thus improving the efficiency of the circular light-field acquisition.

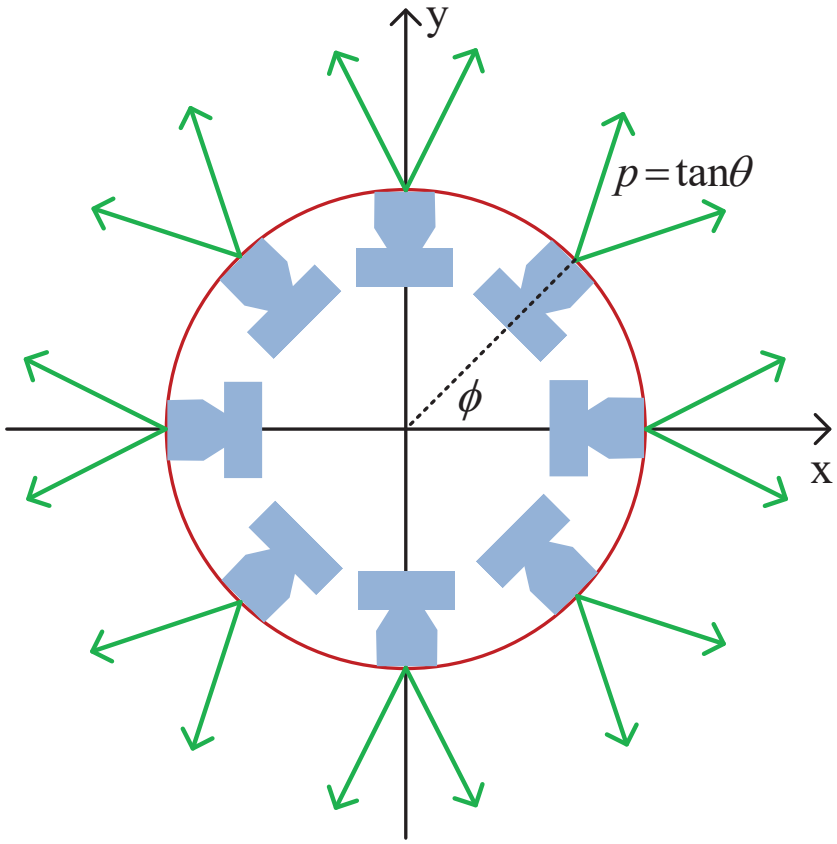

Figure 5.5: Illustration of the creation of a 2-D circular light-field. We position a set of cameras on a circular rig and capture multiple images. The position of the camera is the variable ϕ and the pixel coordinates on each image is the variable $\tan\theta$.

5.3.1 Circular light-field rendering

To render novel views, we need to first model the light rays in the circular light-field. A standard light field comprises lines with different slopes that are determined by their corresponding depths. Then by slicing the 2-D light field, we render novel views at different depths. As shown in Section 5.2.3, the vertical slices correspond to novel views at the camera plane. The slices with negative slopes correspond to novel views in front of the camera plane, whereas the slices with positive slopes correspond to novel views behind the camera plane

We show that a point in a given space still corresponds to a line in the 2-D circular light-field. To verify this, we formulate the problem as follows: The circular rig is positioned in a 2-D space that is defined as the $x-y$ plane. For the sake of simplicity, we define the center of the circular rig to the origin of the $x-y$ plane. Any location in the 2-D space can also be defined by its polar coordinates (z,ψ) where z denotes the distance to the origin and ψ denotes the angle relative to the x axis. Then the 2-D circular light-field is represented with $L(\phi,\theta)$ or $L(\phi,p)$.

Thus, to model the light rays in the circular light-field, we formulate the line structure in the 2-D data (ϕ,p) of all the light rays that pass through a given location (z,ψ) in the 2-D space.

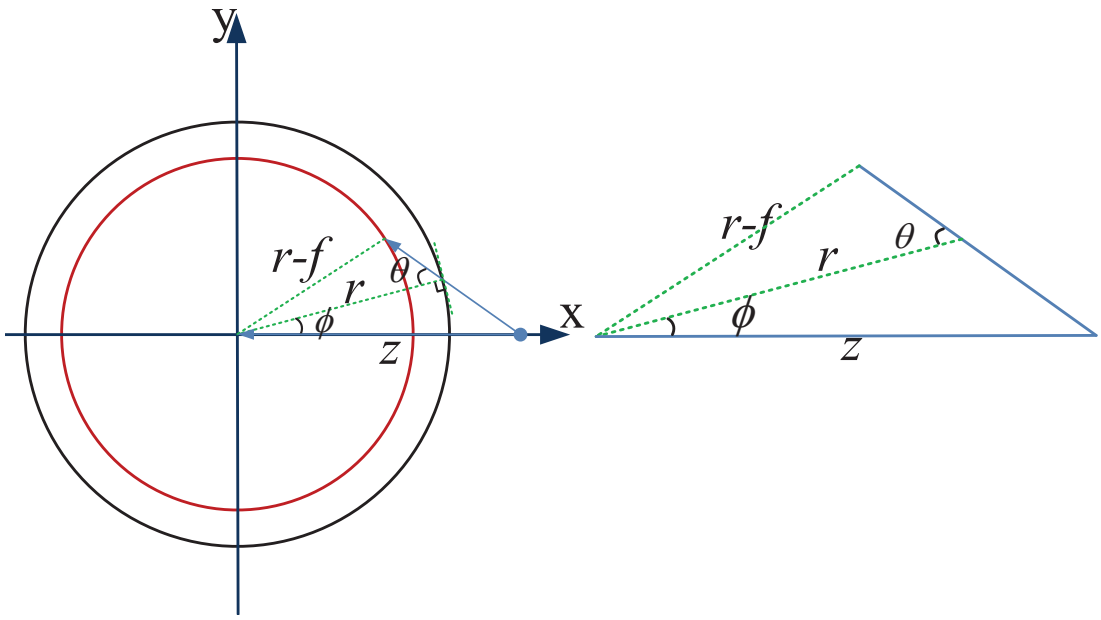

Figure 5.6: Illustration of the light rays emitted from the location $(z, 0)$. In the 2-D circular light-field, we establish the relationship between the light ray $L(0, 0)$ and the light ray $L(\phi, \theta)$, both of which are from the same location $(z, 0)$.

Without loss of generality, we choose two special light rays that emit from a point on the x axis z meters away from the center of the circular rig as shown in Figure 5.6.

The radius of the outer circle is defined as r whereas the focal length of the camera is normalized to 1. For the sake of simplicity, we assume that one of the light rays intersects with the outer circle at 0 degree and it passes the center of the circular rig. Thus both the intersection angle and relative angle of this light ray are 0 degree and the light ray is represented with $L(0, 0)$.

As for the other light ray, its two unknown coordinates are denoted with (ϕ, θ). We illustrate the relationships between these two light rays in Figure 5.6. By applying the Sine Law to the triangle in Figure 5.6 on the right, the relationships between the two light rays that come from the same point z meters away can be formulated as

$$\tan\theta = \frac{\sin\phi}{-\cos\phi + r \cdot z^{-1}}. \tag{5.1}$$

As the focal length is 1, $\tan\theta$ is also the pixel index p of the captured image. Then we can consider Equation (5.1) in a more general way: (ϕ, θ) denotes all the light rays that pass a given location $(z\cos\psi, z\sin\psi)$, which can also be represented with the polar coordinates (z, ψ). Under this assumption, the coordinates of the previous light ray $L(0, 0)$ naturally becomes $(\psi, 0)$. By introducing the pixel index p, we formulate the relationships as

$$p = \frac{\sin(\phi - \psi)}{-\cos(\phi - \psi) + r \cdot z^{-1}}. \tag{5.2}$$

We usually refer to Equation (5.2) as the parametric function of the novel view located at

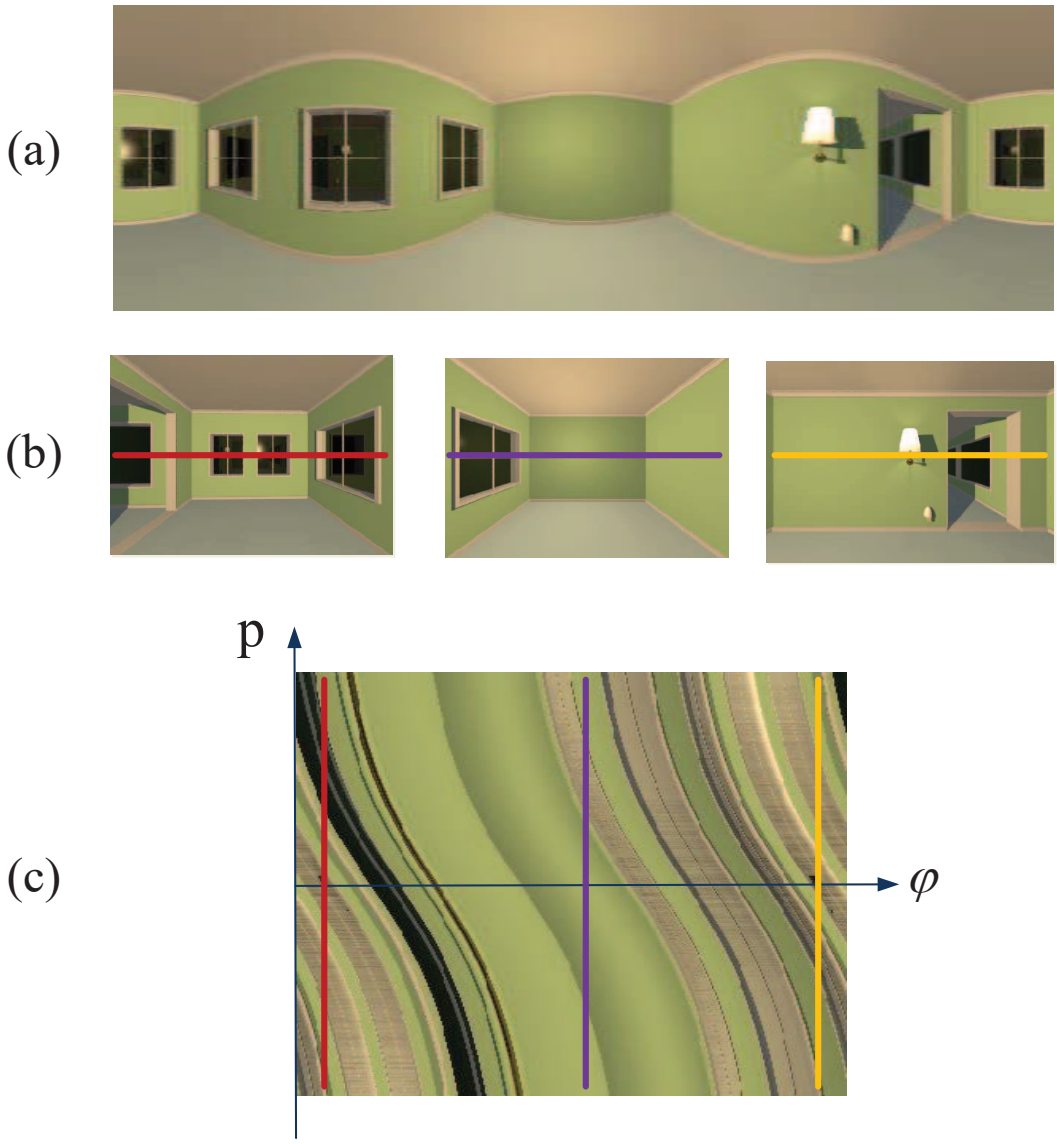

Figure 5.7: An example of the 2-D circular light-field. In row (a), we show the panorama photo of the scene. In row (b), we show three photos from the dataset to create the circular light field. We use line slices to identify the 1-D images used to create the 2-D circular light-field. In row(c), we show the 2-D circular light-field and identified the three 1-D images from row (b).

(z, ψ). This equation represents all light rays that are emitted from or passing through the point $(z\cos\psi, z\sin\psi)$ in the 2-D space. By changing the parameters ψ and z, we render novel views at different locations.

For a better understanding of the circular light-field rendering, we give an example, as shown in Figure 5.7. The scene is a room and its panorama photo is showed in Figure 5.7(a). A total

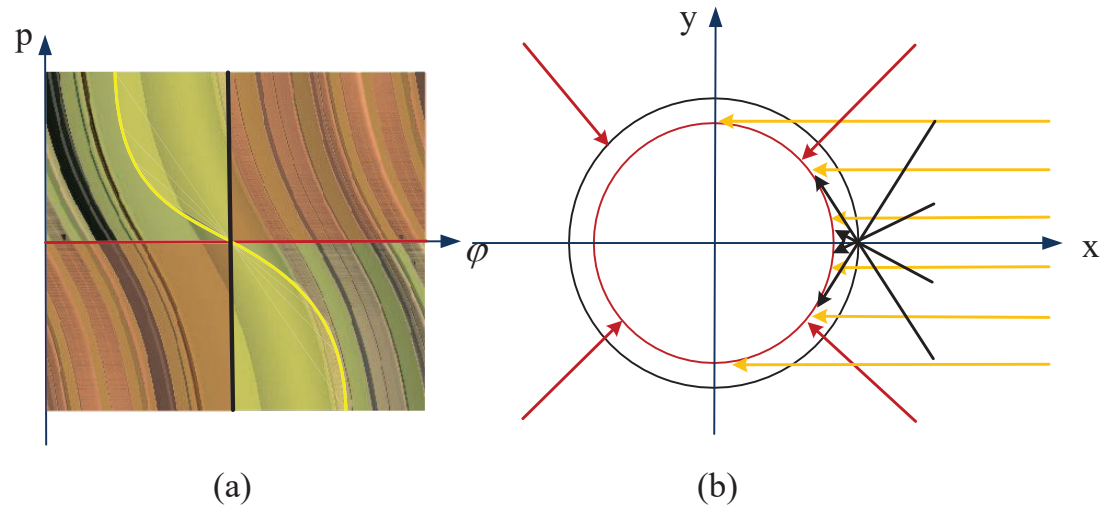

Figure 5.8: Illustration of the line structure in the circular light-field. (a) We show three special slicing curves: the yellow one, black one and red one, which correspond to the functions when $z = +\infty$, r and 0, respectively. (b) We also show the corresponding light rays in the same color in the 2-D space. The yellow lines represent a set of parallel light rays, the black ones represent a set of light rays converging on the outer circle and the red ones represent a set of light rays converging at the center of the circle.

number of 360 photos are taken of the room, and we show three of them in Figure 5.7(b). The central row of each photo is identified by line slices (red, purple and yellow slices, respectively). Then, by stacking these line slices that are the central row from each photo, we create a 2-D circular light field of the room as shown in Figure 5.7(c). The 2-D circular light field is represented with $L(\phi, p)$. Here ϕ denotes where the photos are taken whereas p denotes the pixel coordinates. Thus, each column in the 2-D circular light field corresponds to a 1-D image that is the central row from each photo.

By using this 2-D circular light-field, we demonstrate three special slicing lines in the 2-D circular light-field and their corresponding novel views, as shown in Figure 5.8. These curves correspond to novel views located at different depths. We demonstrate more specifically, three extreme cases of the slicing curves: a yellow curve, a black curve and a red curve, which correspond to the novels views when $z = +\infty$, r and 0, respectively.

More specifically, when $z = +\infty$, the parametric function can be simplified as

$$p = -\tan\phi. \tag{5.3}$$

This slicing curve represents the parallel light-rays from infinity. By sliding this curve horizontally, we render parallel light-rays that come from different directions.

When $z = r$, the parametric function is a vertical line that is actually the original image captured by the camera on the circular rig. By choosing different vertical lines, we render the novel views on the circular rig.

When $z = 0$, the parametric function becomes $p = 0$, which is a horizontal line across the whole circular light-field. This particular novel view is created by choosing all the light rays that

pass the center of the circular rig, which is a 360-degree panorama photo.

The two curves when $z = 0$ and $z = +\infty$ are the boundaries of variable z. As z increases from 0 to $+\infty$, the corresponding curve changes from a horizontal line to a vertical line, and finally to a tan() function. We use these three curves to segment the rendered novel views, as shown in Figure 5.8(a). When the slicing curve is between the yellow and the black curve (identified by the yellow color), the position of rendered views is outside the circular rig. When the slicing curve is between the red and black curve (identified by the red color), the position of rendered views is inside the circular rig. We also show the corresponding light rays of these curves in Figure 5.8(b). The red light rays, which form a panorama correspond to the horizontal curve. The black light rays that converge on the rig correspond to the vertical black curve. The yellow light rays are parallel to each other and they correspond to the yellow curve.

In conclusion, by slicing the 2-D circular light-field with the parametric function (5.2), we render a novel view at $(z\cos\psi, z\sin\psi)$ in the 2-D space. Of course the field of view of the rendered view depends on the dimension of the original data. When $z = 0$, we achieve a full 360-degree field of view whereas when $z = +\infty$, we have a 0-degree field of view. To sum up, the maximum achievable field-of-view of the rendered image decreases as z increases from 0 to $+\infty$.

This could become problematic in practice, as we expect a 360-degree field of view at any chosen locations. In the following two section, we address this problem by introducing the registration, stitching and super-resolution of the circular light-field.

5.3.2 Circular light-field registration

Without loss of generality, we consider a simple scenario for the registration between two circular light-fields in a given 2-D space. Both circular light-fields are captured with the same system, thus the radius of the circular rig and the focal length of the camera are the same. Furthermore, the center of the circular rig in the 2-D space can be represented with two parameters: the variable z that is the distance from the center to the origin of the 2-D space and ψ that is the relative angle to the x axis.

For the sake of simplicity, we assume one circular light-field is captured with the circular rig located at the origin of the 2-D space. Then we only need to estimate the position of the second circular light-field. To be more specific, we need to estimate the distance z between the two centers of the circular rigs and the relative angle ψ between the two centers, as shown in Figure 5.9. Based on the parametric function of the circular light-field, we propose to estimate the two unknown parameters sequentially.

Registration of ψ

First, we estimate the relative angle ψ of the second circular rig. We use two sets of parallel light rays that pass through both circular rigs, as shown in Figure 5.10. The yellow beam is from the bottom left part of the scene, whereas the green beam is from the top right part of the scene. Both of the beams are recorded in the circular light-fields and they correspond to the same parametric function (5.3), thus the same curves in the circular light-fields. Therefore, the estimation problem of the unknown parameter ψ becomes an alignment problem of two curves that can be described with the parametric function (5.3).

We present an example of the registration between two circular light-fields $L_0(\phi, p)$ and $L_1(\phi, p)$. We use the variable ψ_0 to denote the relative angle between these datasets. As shown

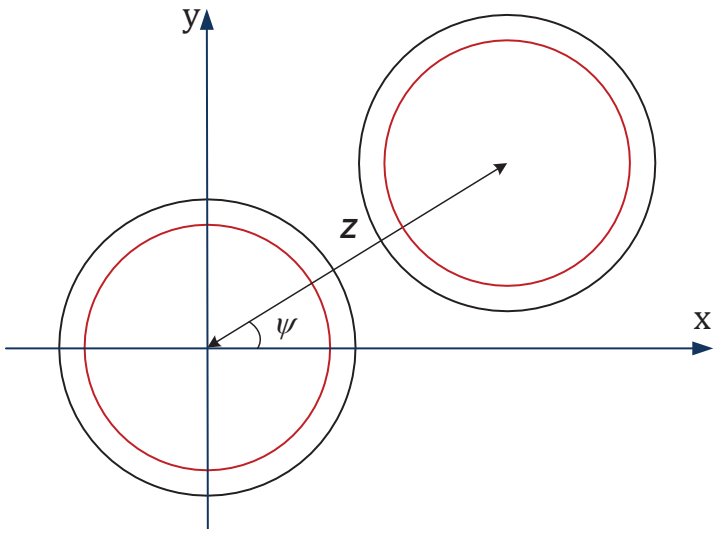

Figure 5.9: Illustration of two CLFs.

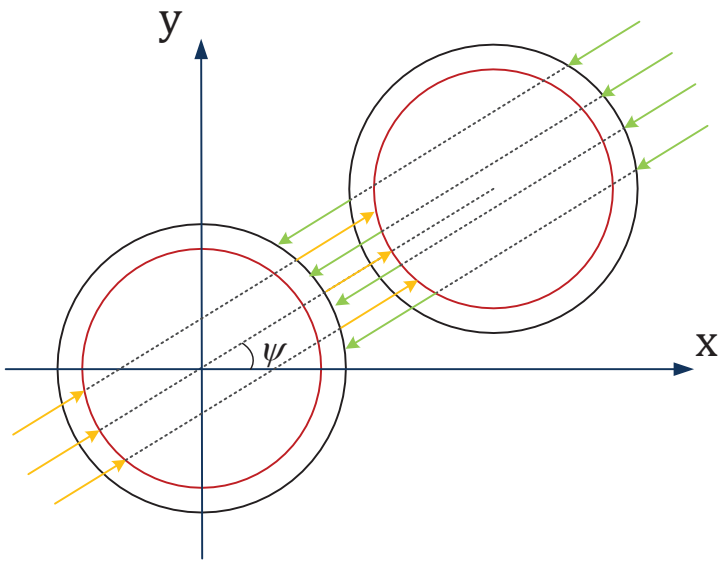

Figure 5.10: The registration of the relative angle ψ. Here we illustrate two sets of parallel light rays shared by both of the circular light-fields.

in Figure 5.11, we show the two circular light-fields $L_0(\phi, p)$ and $L_1(\phi - \psi_0, p)$ on the top row. Given the correct shift ψ_0 in the ϕ dimension, we can observe many similarities between them.

In practice, we can optimize discretely over the ϕ dimension by calculating

$$Err_\psi(\phi, p) = |L_0(\phi, p) - L_1(\phi - \psi, p)|.$$

Then the corresponding entries of the same light rays should be zero, as shown in Figure 5.9.

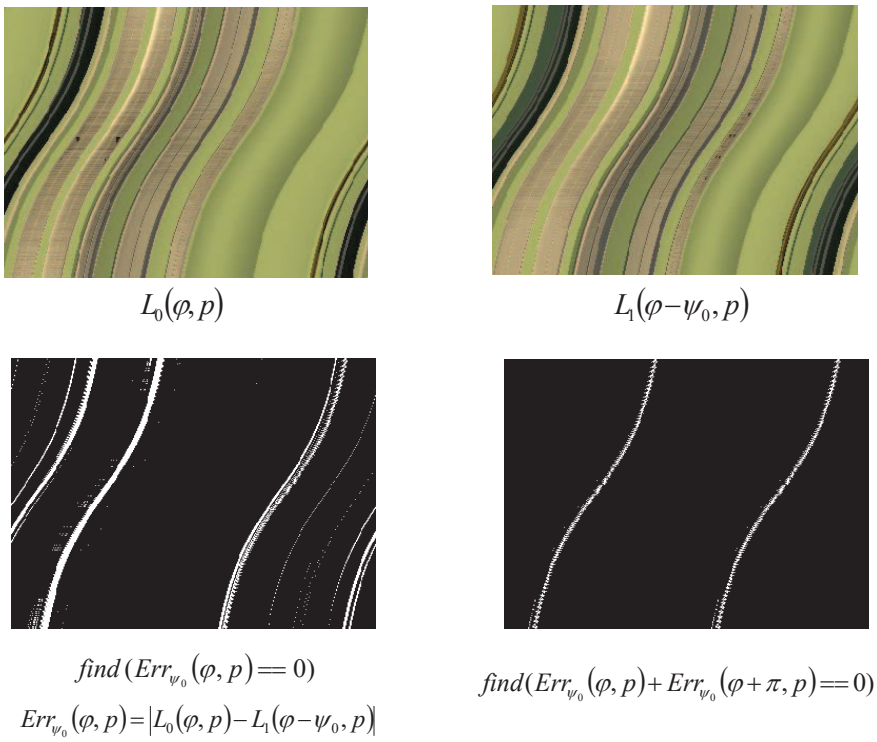

Figure 5.11: The registration of the parameter ψ by subtraction. We demonstrate the two circular light-fields $L_0(\phi, p)$ and $L_1(\phi, p)$ on the top row. We demonstrate the direct subtraction on the bottom left, in which there are many false detected entries. We demonstrate the summation of the subtraction and a 180 deg shift of the subtraction on the bottom right, where we can clearly observe the corresponding light rays.

We demonstrate the direct results in Figure 5.11 on the bottom left. As many similar textures are presented in the scene, we observe many false detected areas.

To address this problem, we use the knowledge that the two sets of parallel light rays have a 180-degree shift in the ϕ dimension. Hence, instead of detecting zero entries in $Err_\psi(\phi, p)$ directly, we analyze

$$Err_\psi(\phi, p) + Err_\psi(\phi + \pi, p),$$

as shown in Figure 5.11 on the bottom right. Finally, we can clearly observe the two sets of parallel light rays that are defined by the parametric function (5.3).

Registration of z

The registration of the distance between two circular rigs is much more difficult because the parametric function is complicated for a random view. Here we only give a description of the problem and postpone the algorithm derivation to the future work.

As shown in Figure 5.12, the distance between two circular rigs is denoted with z. We propose to register the variable z in the same way as we register the relative angle ψ, by using

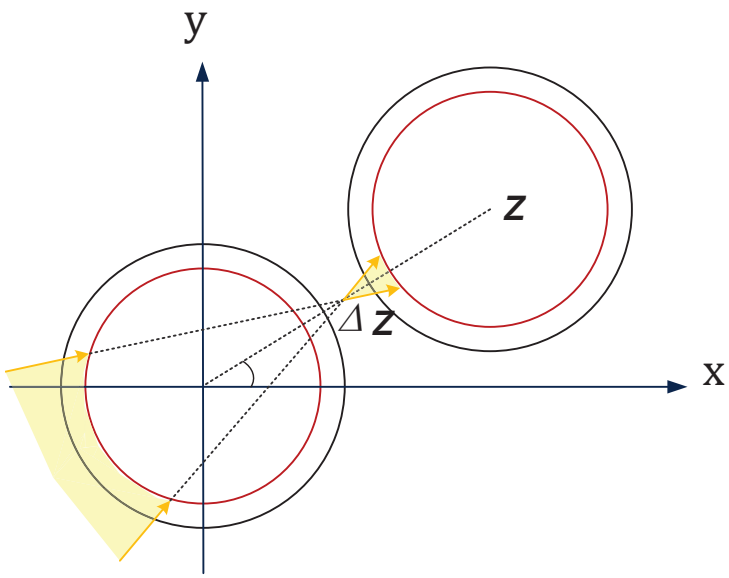

Figure 5.12: The registration of the relative angle ψ. Here we illustrate two sets of parallel light rays shared by both of the circular light-fields.

light rays recorded in both circular light-fields. More specifically, we can use a set of light rays that correspond to a novel view that connects a line between the two centers.

We give an example in Figure 5.12: a novel view Δz meters away from the origin. The two curves, as shown in Figure 5.11 on the bottom right, are two special cases of the novel views on the connecting line. By changing the parameter z of the slicing curves in the circular light-field, we move the position of the novel view on the connecting line.

Take the novel view represented with yellow light-rays as an example. In the circular light-field that is at the origin, the parametric function is created by putting $-\Delta z$ into the parametric function (5.2). Then the same light rays can be selected by using the parametric function with the slicing variable $z - \Delta z$. To sum up, given the correct z, we can find the matching slice-pairs in the two circular light-fields by changing the variable Δz. We can design an energy function between these slicing pairs, with respect to the unknown distance z, and solve the registration problem by minimizing the energy function.

5.3.3 Circular light-field super-resolution

The quality of virtual-reality content largely depends on the resolution of the acquired data. As for the circular light-field, it comes down to how many photos we need to take around the circular rig. To increase the efficiency of the acquisition system, we propose a super-resolution algorithm to increase the resolution of the acquired circular light-fields. Thus we achieve a high quality acquisition at a lower cost for both the hardware and time consumption.

In this section, we give a formulation of the super-resolution problem of the circular light-field. We propose to use a model that is similar to the surface light-field in Chapter 3. As discussed in Section 5.2, by fixing $p = 0$, the circular light-field $L(\phi, 0)$ is a panorama photo formed by all

the light rays passing through the center of the circular rig. We assume the panorama image to be a 1-D band-limited signal $f(t)$. We use the function $g(\phi, p)$ to represent the depth value of the origin of the corresponding light ray $L(\phi, p)$. Under the Lambertian assumption of the scene, the light ray $L(\phi, x)$ can be mapped onto the 1-D signal $f(t)$ with its depth information $g(\phi, x)$ as

$$L(\phi, x) = f(t(g(\phi, x))).$$

To be more specific, the geometry information $g(\phi, p)$ determines the sampling location of the sample $L(\phi, p)$ on the 1-D signal $f(t)$ that can be represented with a Fourier series as

$$f(t) = \sum_{k=-L}^{k=L} a_k \exp(2\pi jkt), \qquad t \in [0, 1)\,.$$

Therefore, the 2-D circular light-field $L(\phi, x)$ is fundamentally the 1-D signal $f(t)$ modulated by the geometry structure $g(\phi, p)$ of the scene. Given the depth information $g(\phi, p)$, we can represent the circular light in a continuous domain, thus increasing the resolution of the data.

In conclusion, the problem of the circular light-field super-resolution can be modeled as a signal recovery problem. Even without the depth map $g(\phi, p)$, we demonstrate how to alternatively recover the signal and the depth map, using a similar model in Chapter 3.

5.4 From 2-D to Higher Dimensional Circular Light-Field

So far, we have proposed a novel representation of the light rays: the circular light-field; and we demonstrate its creation and rendering in a given 2-D space. In this section, we extend the 2-D space to a 3-D space and demonstrate the representations of the 3-D and 4-D circular light-field, respectively.

5.4.1 3-D circular light-field

In Section 5.2, the scene is in 2-D and we represent the 2-D circular light-field as a stack of 1-D images that are captured by cameras located on a circular rig. In a given 3-D world, standard cameras are positioned on the same circular rig. By stacking these 2-D images, we create a 3-D dataset that is defined as the 3-D circular light-field. The plane on which the circular rig is located is perpendicular to each sensor plane of the cameras, With the 3-D circular light-field, we render new views on the same horizontal plane on which the circular rig is located.

Note that the 3-D circular light-field is an extension of the 2-D circular light-field. By choosing and stacking the central row of each 2-D image, we create a standard 2-D circular light-field, as shown in Figure 5.13.

In the 3-D circular light-field, light rays are represented with three parameters: the angle ϕ relative to the x axis and pixel index (p, q) on the image plane. In the acquisition setup, all the optical axes of the cameras converge at the center of the circular rig; and we define the center as the origin of the 3-D space. Then a novel view in the 3-D world is represented with $(z\cos\psi, z\sin\psi, h)$ where h denotes the relative height to the $x-y$ plane, z denotes the distance from the novel view to the center, which is a projected distance on the plane of the circular rig and ψ denotes the angle of the novel view relative to the origin.

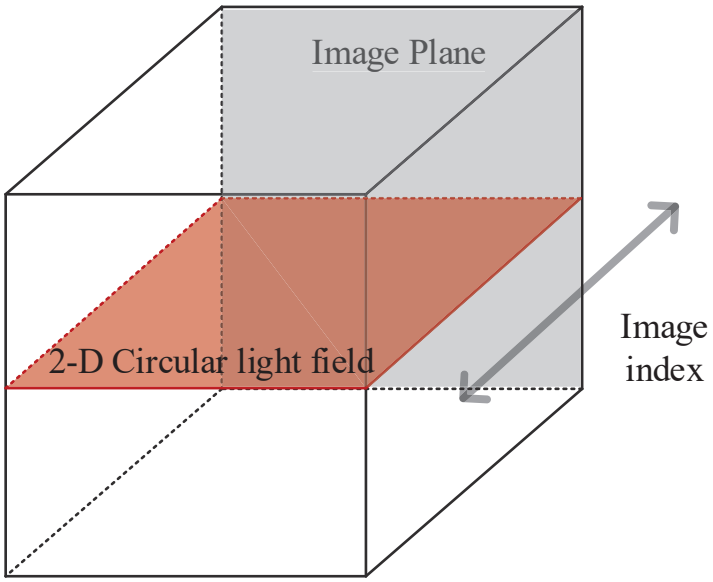

Figure 5.13: The 3-D circular light-field. The 3-D circular light-field is created by stacking a sequence of 2-D image. The grey plane represents the image plane, and the red plane represents a 2-D circular light-field that is created by stacking the central row from each image.

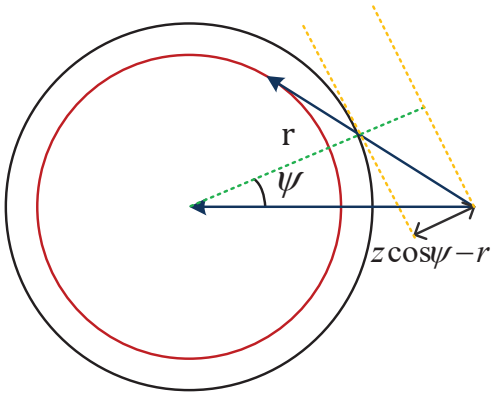

Figure 5.14: The perpendicular distance from the object (z, ψ) to the observing camera at $(\phi = 0)$.

When we render novel views at different locations, the transformation in the vertical direction has to be considered. As shown in Figure 5.14, to model the transformation in the vertical direction, we introduce a variable d that is the perpendicular distance from the object to the closest camera that observes the object. It is calculated as follows:

$$d = z \cos \psi - r,$$

where r is the radius of the circular rig. When the observing camera ψ changes, the distance d

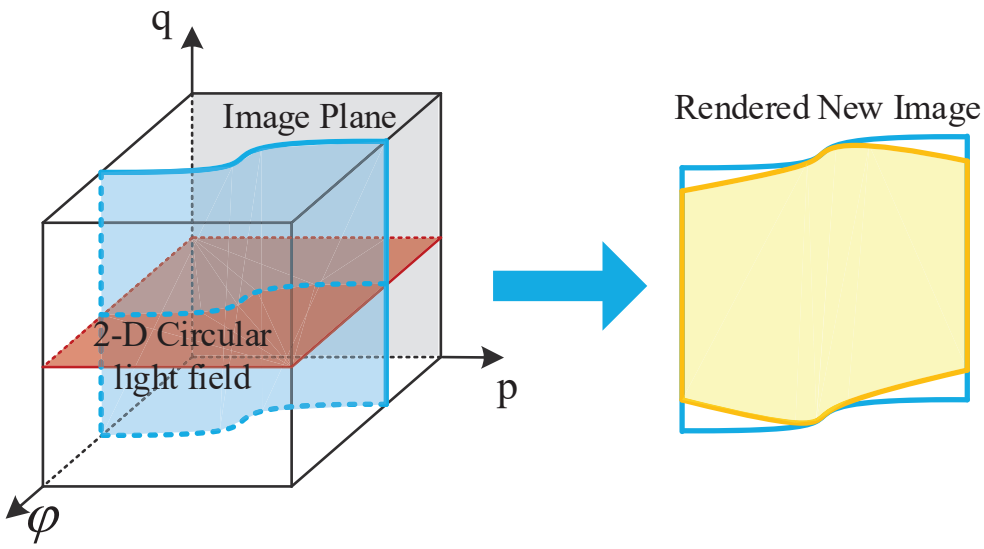

Figure 5.15: The rendering of 3-D circular light-field. The blue line on the 2-D circular light-field is calculated according to the novel view (z, ψ). We directly extend the blue line to the blue plane and then apply the correction in the q direction to map the blue plane to the final rendered view, the yellow plane.

changes accordingly.

Our analysis of the 2-D circular light-field can be applied directly to the analysis of the 3-D circular light-field in the horizontal p dimension. But we need to address the properties of the circular light-field in the vertical q by using the distance d that determines how the object is projected on the image sensor. More specifically, the projection of the object $(z\cos\psi, z\sin\psi, h)$ in the vertical dimension q can be calculated for the camera located at angle ϕ as

$$q = \frac{h}{z\cos(\psi - \phi) - r}, \tag{5.4}$$

where h is the height of the object and the focal length is normalized to 1. Note that, except for the objects that are on the same plane as the circular rig, all the other objects are projected on different rows on the image sensor for different cameras.

Then to render a novel view from the 3-D circular light-field, we still use the parametric curves to slice the 3-D data in the x dimension. As shown in Figure 5.15, we first slice the blue plane on the left, which is a direct extension of the slicing curve in the 2-D circular light-field. Furthermore, by using Equation (5.4), the q coordinate of the rendered view changes accordingly to the angle ϕ.

For the sake of simplicity, we calculate the transformation on the image plane, instead of estimating the original height h. We use the variable $\hat{q}$ to denote the original coordinates on the blue plane and the variable q to represent the warped coordinates on the yellow plane, both as shown in Figure 5.15. The relation between q and $\hat{q}$ can be formulated as

$$q = \hat{q}\frac{z\cos(\psi - \phi) - r}{z - r}. \tag{5.5}$$

There are two steps for rendering the 3-D circular light-field. After choosing the novel view (z, ψ), we first select the slicing lines from each 2-D circular light-field and form the blue plane, as shown in Figure 5.15. Second, we warp the blue plane with Equation (5.5) to deal with distortion in the vertical direction and obtain the final novel view as the yellow plane.

Finally, we show a rendered sequence of the room in Figure 5.16. With the 3-D circular light field, we rendered a sequence of novel views moving towards the window. The image identified with red rectangle is the original view. The rendered views behind the original view have a larger field of view whereas the rendered views in front of the original view have a smaller field of view.

The rendering framework is quite straightforward. We first determine the position and viewing direction of a novel view, then we calculate the corresponding slicing curve. With a bilinear interpolation in the 3-D data, we obtain the novel views without the vertical transformation. Finally, by applying Equation (5.5) to rendered image, we finally rendered the novel views as shown in Figure 5.16.

5.4.2 4-D circular light-field

There are two ways to define and create a 4-D circular light-field.

Two concentric cylinders

In the first definition, any given light ray in the 3-D space is represented by its intersection with two concentric cylinders. This definition is a direct extension of the 3-D circular light-field $L(\phi, p, q)$ by adding a t dimension. For a static scene, the whole camera rig is moved vertically in the t dimension to capture the 4-D circular light-field $L(t, \phi, p, q)$. Thus the index (t, ϕ) denotes the camera position, and the index (p, q) denotes the pixel coordinate.

On one hand, when we fix the variable t, the 4-D circular light-field becomes exactly the same as a 3-D circular light-field described in the previous section. On the other hand, by fixing the variable ϕ, the 4-D circular light-field becomes a standard 3-D light field $L(h, p, q)$ that is an image sequence by moving a camera vertically.

Two concentric spheres

In the second definition, any given light ray in the 3-D space is represented by its intersection with two concentric spheres. The 4-D circular light-field can be acquired by mounting cameras on a sphere. The optical axis of each camera passes through the sphere center. For a static scene, the 4-D circular light-field can also be captured with a circular rig. Instead of moving the circular rig vertically, we rotate the rig around its center. The optical axis of each camera still passes the center, while the camera positions are extended from a circular rig to a sphere.

The 4-D circular light-field is represented with $L(\phi_0, \phi_1, x, y)$. The angle pair ϕ_0 and ϕ_1 can be seen as the elevation and azimuth of each camera on the sphere.

By fixing the variable ϕ_1, the 4-D circular light-field becomes a 3-D circular light-field described in the previous section. By fixing the variable ϕ_0, the 3-D data is still a set of images captured on a circular rig. However, each camera faces toward the center of the sphere instead of the center of the rig. As the optical center of the each camera is still located on the rig, when we apply a homography transformation to each image, the 3-D data can be transformed into a standard 3-D circular light-field.

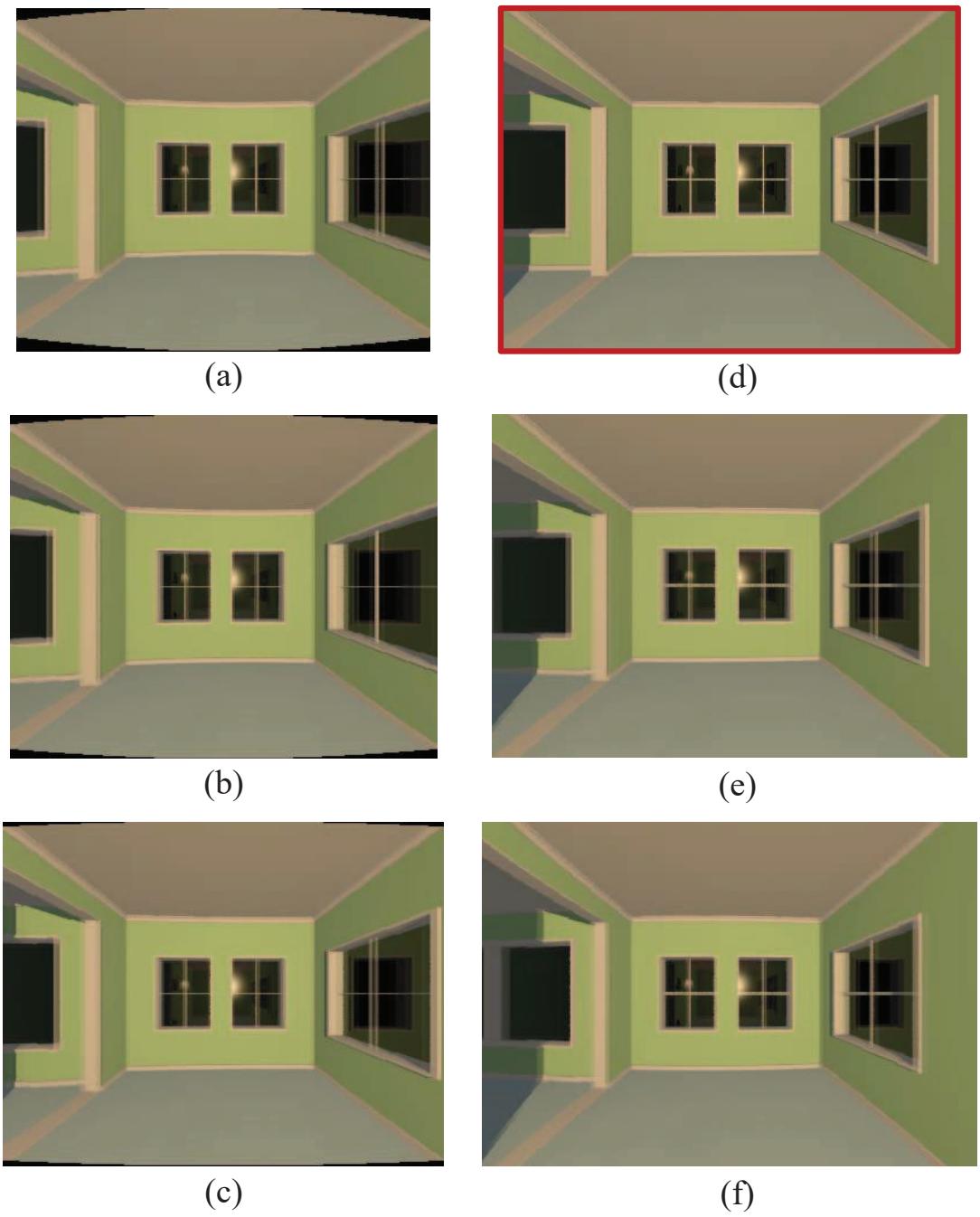

Figure 5.16: A sequence of novel views (from (a)-(f)) moving towards the window. The image identified with the red rectangle is the original view.

5.5 Conclusions

In this chapter, we proposed a circular light-field to represent light rays and showed its great potential for rendering a novel view at any chosen location in a given space. In practice, for

an efficient acquisition framework, we discussed the problem formulation for circular light-field registration and super-resolution. We also showed how to extend the 2-D circular light-field to 3-D and 4-D.

A circular light-field is a powerful tool for the acquisition and rendering of indoor environments. It can be acquired simply with a camera mounted on a circular rig. The registration and stitching of multiple circular light-fields are straightforward to formulate and implement. It shows great potential in image-based rendering for virtual-reality applications.

Furthermore, we can also create a circular light-field by pointing the camera towards the center of a circular rig. Then instead of capturing the environment, we capture a particular object position in the center. Thus, we can render this object for different viewing angles at different distances. This type of dataset is ideal for augmented reality; we refer the interested reader to a detailed survey by Billinghurst [10].

Chapter 6

Conclusions and Future Works

Though I'm past one hundred thousand miles
I'm feeling very still
And I think my spaceship knows which way
to go

David Bowie

Hundreds of years ahead of his time, Leonardo da Vinci described and sketched ideas for many inventions such as tanks, helicopters, and mechanical calculators. The central topic of this thesis, the light-field camera, is also one of his flights of imagination: an imaging device capable of capturing every optical aspect of the scene in front of it.

In this thesis, we have investigated the sampling models in the light field and have proposed novel methods to exploit the acquired data for the reconstruction and rendering of a scene. Our central interests lie in both aspects of light-field analysis: the reconstruction of a scene to its finest details and the rendering of a scene to its complete information. Hence, we have addressed these two problems with a surface light-field and a circular light-field, respectively. Both the theoretical and practical extensions for the future work are built around these two novel representations.

6.1 Theoretical Extensions

Compared with standard light-field, the surface light-field and the circular light–field have their advantages and disadvantages. The effectiveness of the surface light-field relies on the assumption that its essential bandwidth is limited, thus it can be represented with a finite Fourier series. This assumption is significantly violated when the scene has large depth variations and discontinuities. As for the circular light-field, although it is capable of rendering 360-degree novel views within an area, the number of the required circular light-fields and the sampling rate of

each circular light-field is largely determined by the size of the area.

Surface Light-Fields

In future work, we need first to address the constraints of the assumption that the scene is within a small depth range without discontinuities. Although the experimental results in Chapter 3 show that the depth recovery from surface light-field still has consistent performance, even with the presence of discontinuities in the scene, we need to propose alternative ways to treat a more complicated scene more elegantly.

The most straightforward way to treat this problem is to segment the scene to different areas. By identifying the discontinuities and separating an area with large depth variations into areas with small depth variations, the assumptions used for the surface light-field will be kept for these pieces. This method can guarantee the limited essential bandwidth for the surface light-field of each piece, at the cost of the complexities for processing the data of the whole scene.

There is also an alternative way, inspired by the work of Tosic et al.[48]. They propose a scale-depth space transform for depth estimation. They design a special kernel that corresponds to the different depth ranges with different scales, then they convolve the light field with different kernels and by detecting local extrema in the scale-depth space, they estimate the depth map. We will investigate the possibility of applying a similar framework in the surface light-field. We will model both the texture and depth map in multiple scales, and the surface light-field will become a summation of the modulated textures in multiple scales.

In addition, we also plan to apply the surface light-field to a circular-light field. When there are no occlusions observed by the camera, the circular light-field can be seen as a panorama photo modulated by the its depth map. With the parametric model of the curves in the circular light field, we will be able to map the circular light-field to a signal with a lower dimension. This extension involves the bandwidth analysis of the circular light-field that is another a direction our theoretical extensions could take.

Circular Light-Fields Sampling

We have two different problems to solve in sampling the circular light-field. First, we need to address the sampling rate of each circular light-field, with regard to the complexity of the scene. Compared with the line structure in a standard light-field, the 2-D circular light-field has a more complicated curve-structure. We propose to model the curves with third-order polynomials, which will largely reduce the complexity of the analysis on circular light-field. Then using the texture and geometry structure of the scene, we can derive the required sampling rate to fully capture one circular light-field without aliasing.

Second, we need to address the sampling rate for a given area within which the novel views are rendered. As we state in Chapter 5, one circular light-field can render novel views at any given location but the rendered field-of-views decrease as the locations of the rendered views are further away from the center of the circular rig. Thus, for a given area, we propose to capture multiple circular light-fields and merge these data. The interesting question is how many circular light-fields are required for a given area. By assuming that there is no aliasing in each acquired circular light-field, the sampling rate here is only referred to the acquisition of multiple circular light-fields in the given area.

6.2 Practical Extensions

The studies presented in all of our chapters are related to practical applications. The most interesting application is the circular light-field for virtual-reality applications. To make the circular light-field a practical concept, we need to implement both the acquisition and rendering system. These are the two topics for our practical extensions for future work.

Building Circular Light-Field Cameras

The straightforward way to build a circular light-field camera, according to its definition, is to mount a standard camera on a robotic arm. The facing direction of the camera should be along the robotic arm. By precisely rotating the arm, we can create circular light-fields directly.

As we mention in Chapter 5, when the camera faces towards the center of a circle, we can create augmented-reality contents of the object placed in the center. This type of acquisition can be implemented with a rotating stage and a single camera that is pointed at the object in the center of the rotating stage. By placing green curtains behind the object and covering the stage, we can easily perform background subtraction to create viable datasets.

There is also an alternative way to acquire the data with a much simpler device, compared with previous methods. We can simply combine a omnidirectional camera and several depth cameras, such as Kinect and Primesense. While we capture a 360-degree panorama photo, we also capture the depth map for each entry in the photo. We then apply the model of the surface light-field by modulating the panorama photo with its depth map and then create the corresponding circular light-field.

Circular Light-Field Rendering in Practice

In practice, the main concern of rendering is the required time. On one hand, to reduce the rendering time, we can take advantage of the 3-D texture in OpenGL. By loading the circular light-field into the graphic card, we can render novel views by slicing the 3-D texture directly. Although there are memory issues for the size of the data that can be loaded, we can break the scene into small areas. Then by using the current view, we predict the expected areas to be loaded and accelerate the rendering process. There are also many other techniques and skills for us to explore.

6.3 Conclusions

In this thesis, we have built our work around the sampling models in light fields proposed in Chapter 2. In Chapter 3, we introduce the concept of surface light field and design a novel algorithm around it to achieve high-accuracy results in depth estimation.

In Chapter 4 and Chapter 5, we have focused on extending the field of view of the light field for acquisition and rendering. We first work on the standard light-field cameras and derive their motion models for light-field stitching and registration. However, as the standard light field is represented with the two-plane-parameterization, it is intrinsically not suitable for rendering novel views with a 360-degree field of view. Thus, we have proposed a new representation for the light rays called the circular light-field.

The surface light-field and circular light-field are two novel concepts proposed in our thesis and our future work will also be built around these two novel representations. On one hand, we will explore the sampling problems in circular light-fields and apply the surface light-field model to the circular light-field. On the other hand, we will build acquisition systems and rendering devices for the circular light-field for virtual-reality applications.

Bibliography

[1] E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision," in Proc. *Computational Models of Visual Processing*, pp. 3–20. MIT Press, 1991.

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

[3] J. Aloimonos, "Shape from texture," *Biological cybernetics*, vol. 58, no. 5, pp. 345–360, 1988.

[4] H. H. Baker, "Building surfaces of evolution: The weaving wall," *International Journal of Computer Vision*, vol. 3, no. 1, pp. 51–71, 1989. [Online]. Available: http://dx.doi.org/10.1007/BF00054838.

[5] H. H. Baker and R. C. Bolles, "Generalizing epipolar-plane image analysis on the spatiotemporal surface," *International Journal of Computer Vision*, vol. 3, no. 1, pp. 33–49, 1989. [Online]. Available: http://dx.doi.org/10.1007/BF00054837.

[6] S. Baker, T. Kanade, *et al.*, "Shape-from-silhouette across time part i: Theory and algorithms," *International Journal of Computer Vision*, vol. 62, no. 3, pp. 221–247, 2005.

[7] ——, "Shape-from-silhouette across time part ii: Applications to human modeling and markerless motion tracking," *International Journal of Computer Vision*, vol. 63, no. 3, pp. 225–245, 2005.

[8] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer vision–ECCV 2006*. Springer, 2006, pp. 404–417.

[9] J. Berent and P. L. Dragotti, "Segmentation of epipolar-plane image volumes with occlusion and disocclusion competition," in Proc. *Multimedia Signal Processing, 2006 IEEE 8th Workshop on*, pp. 182–185. IEEE, 2006.

[10] M. Billinghurst, A. Clark, and G. Lee, "A survey of augmented reality," *Found. Trends Hum.-Comput. Interact.*, vol. 8, no. 2-3, pp. 73–272, Mar. 2015. [Online]. Available: http://dx.doi.org/10.1561/1100000049.

[11] C. Birklbauer and O. Bimber, "Panorama light-field imaging." in Proc. *SIGGRAPH Posters*, p. 61. ACM, 2012.

[12] R. C. Bolles, H. H. Baker, and D. H. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *International Journal of Computer Vision*, vol. 1, no. 1, pp. 7–55, 1987.

[13] J. Browning, "Approximating signals from nonuniform continuous time samples at unknown locations," *Signal Processing, IEEE Transactions on*, vol. 55, no. 4, pp. 1549–1554, 2007.

[14] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in Proc. *European Conference on Computer Vision (ECCV)*, Lecture Notes in Computer Science, vol. 3024, pp. 25–36. Springer, May 2004. [Online]. Available: http://lmb.informatik.uni-freiburg.de//Publications/2004/Bro04a.

[15] L. Castaneda and M. Pacampara, "Virtual reality in the classroom: An exploration of hardware, management, content and pedagogy," in Proc. *Society for Information Technology & Teacher Education International Conference*, vol. 2016, no. 1, pp. 527–534, 2016.

[16] C.-F. Chang, G. Bishop, and A. Lastra, "Ldi tree: A hierarchical representation for image-based rendering," in Proc. *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pp. 291–298. ACM Press/Addison-Wesley Publishing Co., 1999.

[17] S. E. Chen and L. Williams, "View interpolation for image synthesis," in Proc. *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pp. 279–288. ACM, 1993.

[18] A. Criminisi, S. B. Kang, R. Swaminathan, R. Szeliski, and P. Anandan, "Extracting layers and analyzing their specular properties using epipolar-plane-image analysis," *Comput. Vis. Image Underst.*, vol. 97, no. 1, pp. 51–85, Jan. 2005. [Online]. Available: http://dx.doi.org/10.1016/j.cviu.2004.06.001.

[19] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pp. 886–893. Washington, DC, USA: IEEE Computer Society, 2005. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2005.177.

[20] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach," in Proc. *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pp. 11–20. New York, NY, USA: ACM, 1996. [Online]. Available: http://doi.acm.org/10.1145/237170.237191.

[21] M. N. Do, D. Marchand-Maillet, and M. Vetterli, "On the bandwidth of the plenoptic function." *Image Processing, IEEE Transactions on*, vol. 21, no. 2, pp. 708–717, 2012.

[22] W. C. D. Fredwill, "Apparatus for making a composite stereograph," Dec. 15 1936, uS Patent 2,063,985.

[23] 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 Proc. *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pp. 3965–3968. IEEE, 2010.

[24] C. Gilliam, P.-L. Dragotti, and M. Brookes, "On the spectrum of the plenoptic function," *Image Processing, IEEE Transactions on*, vol. 23, no. 2, pp. 502–516, 2014.

[25] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in Proc. *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pp. 43–54. New York, NY, USA: ACM, 1996. [Online]. Available: http://doi.acm.org/10.1145/237170.237200.

[26] F. E. Ives, "Parallax stereogram and process of making same." Apr. 14 1903, uS Patent 725,567.

[27] H. E. Ives, "A camera for making parallax panoramagrams," *JOSA*, vol. 17, no. 6, pp. 435–437, 1928.

[28] ——, "The projection of parallax panoramagrams," *JOSA*, vol. 21, no. 7, pp. 397–403, 1931.

[29] H. Kawasaki, K. Ikeuchi, and M. Sakauchi, "Light field rendering for large-scale scenes," in Proc. *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 2, pp. II–64. IEEE, 2001.

[30] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, "Scene reconstruction from high spatio-angular resolution light fields," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 73:1–73:12, July 2013. [Online]. Available: http://doi.acm.org/10.1145/2461912.2461926.

[31] A. Levin and F. Durand, "Linear view synthesis using a dimensionality gap light field prior," in Proc. *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 1831–1838. IEEE, 2010.

[32] M. Levoy and P. Hanrahan, "Light field rendering," in Proc. *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pp. 31–42. New York, NY, USA: ACM, 1996. [Online]. Available: http://doi.acm.org/10.1145/237170.237199.

[33] J. Li, M. Lu, and Z.-N. Li, "Continuous depth map reconstruction from light fields," *Image Processing, IEEE Transactions on*, vol. 24, no. 11, pp. 3257–3265, 2015.

[34] G. Lippmann, "Epreuves reversibles donnant la sensation du relief," *Journal de Physique*, vol. 7, no. 4, pp. 821–825, 1908.

[35] D. G. Lowe, "Object recognition from local scale-invariant features," in Proc. *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2, pp. 1150–1157. Ieee, 1999.

[36] A. Lumsdaine and T. Georgiev, "The focused plenoptic camera," in Proc. *Computational Photography (ICCP), 2009 IEEE International Conference on*, pp. 1–8. IEEE, 2009.

[37] W. R. Mark, L. McMillan, and G. Bishop, "Post-rendering 3d warping," in Proc. *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, pp. 7–ff. New York, NY, USA: ACM, 1997. [Online]. Available: http://doi.acm.org/10.1145/253284.253292.

[38] M. Matoušek, T. Werner, and V. Hlavác, "Accurate correspondences from epipolar plane images," in Proc. *Proc. Computer Vision Winter Workshop*, pp. 181–189, 2001.

[39] G. Miller, S. Rubin, and D. Ponceleon, "Lazy decompression of surface light fields for precomputed global illumination," in *Rendering Techniques 98*. Springer, 1998, pp. 281–292.

[40] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," *Computer Science Technical Report CSTR*, vol. 2, no. 11, pp. 1–11, 2005.

[41] S. M. Seitz and C. R. Dyer, "View morphing: Uniquely predicting scene appearance from basis images," in Proc. *Proc. Image Understanding Workshop*, pp. 881–887, 1997.

[42] H. Shum and S. B. Kang, "Review of image-based rendering techniques," in Proc. *Visual Communications and Image Processing 2000*, pp. 2–13. International Society for Optics and Photonics, 2000.

[43] H.-Y. Shum and L.-W. He, "Rendering with concentric mosaics," in Proc. *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pp. 299–306. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1999. [Online]. Available: http://dx.doi.org/10.1145/311535.311573.

[44] A. Stern and B. Javidi, "3-d computational synthetic aperture integral imaging (compsaii)," *Optics express*, vol. 11, no. 19, pp. 2446–2451, 2003.

[45] J. Steuer, "Defining virtual reality: Dimensions determining telepresence," *Journal of communication*, vol. 42, no. 4, pp. 73–93, 1992.

[46] R. Szeliski, "Image alignment and stitching: A tutorial," Microsoft Research, Tech. Rep. MSR-TR-2004-92, October 2004. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=70092.

[47] Y. Taguchi, A. Agrawal, S. Ramalingam, and A. Veeraraghavan, "Axial light field for curved mirrors: Reflect your perspective, widen your view," in Proc. *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 499–506. IEEE, 2010.

[48] I. Tosic and K. Berkner, "Light field scale-depth space transform for dense depth estimation," in Proc. *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*, pp. 441–448. IEEE, 2014.

[49] M. Vetterli and P. Marziliano, "Reconstruction of irregularly sampled discrete-time bandlimited signals with unknown sampling locations," *IEEE Transactions on Signal Processing*, vol. 48, no. 12, pp. 3462–3471, Dec. 2000. [Online]. Available: http://dx.doi.org/10.1109/78.887038.

[50] S. Wanner, S. Meister, and B. Goldluecke, "Datasets and benchmarks for densely sampled 4d light fields," in Proc. *Vision, Modelling and Visualization (VMV)*, 2013.

[51] S. Wanner and B. Goldluecke, "Globally consistent depth labeling of 4d light fields," in Proc. *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 41–48. IEEE, 2012.

[52] ——, "Variational light field analysis for disparity estimation and super-resolution," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 3, pp. 606–619, 2014.

[53] S. Wanner, S. Meister, and B. Goldluecke, "Datasets and benchmarks for densely sampled 4d light fields." in Proc. *VMV*, pp. 225–226. Citeseer, 2013.

[54] C. Wheatstone, "Contributions to the physiology of vision.–part the first. on some remarkable, and hitherto unobserved, phenomena of binocular vision," *Philosophical transactions of the Royal Society of London*, vol. 128, pp. 371–394, 1838.

[55] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: a survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 8, pp. 690–706, 1999.

Curiculum Vitæ

Zhou Xue

Audiovisual Communications Laboratory (LCAV)
Swiss Federal Institute of Technology (EPFL)
1015 Lausanne, Switzerland

Email: zhou.xue@epfl.ch

Personal Information

Date of birth:	January, 5, 1985
Nationality:	China

Education

2011–2016	PhD candidate in School of Computer and Communication Sciences, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.
2008–2011	Master of Engineering in Automation Tsinghua University, China.
2004-2008	Bachelor of Engineering in Information Engineering Beijing University of Posts and Telecommunications, China

Professional experience

2011–present	**Research and teaching assistant**, Audiovisual Communications Laboratory (LCAV), Swiss Federal Institute of Technology (EPFL).
2008–2011	**Research assistant**, Broadband Network and Digital Media Lab, Tsinghua University.

Research Interests

— Light field sampling
— Image-based rendering
— Depth recovery

Publications

Papers

[3]. **Z. Xue**, L. Baboulaz, P. Prandoni and M. Vetterli, *Depth recovery from surface light-fields*, in submission.
[2]. **Z. Xue**, L. Baboulaz, P. Prandoni and M. Vetterli, *Light field panorama by a plenoptic camera*, IS&T/SPIE Electronic Imaging 2014.
[1]. **Z. Xue**, J. Yang, Q. Dai and N. Zhang, *Multi-view image denoising based on graphical model of surface patch*, The 3DTV Conference, June 2010.

Patents

[2]. **Z. Xue**, L. Baboulaz and M. Vetterli. *Method, apparatus and program for processing a circular light field*, US patent, in application.
[1]. Q. Dai and **Z. Xue**. *Method, apparatus and program for video denoising based on wavelet-transform and block-search.* China patent, 2011.

Skills

Programming:	Java and Matlab. Working knowledge of Android, C++, Python. Experienced in LaTeX.
Libraries:	OpenCV, OpenGL C++.