Pietro Zanuttigh · Giulio Marin
Carlo Dal Mutto · Fabio Dominio
Ludovico Minto · Guido Maria Cortelazzo

Time-of-Flight and Structured Light Depth Cameras

Technology and Applications

Springer

Time-of-Flight and Structured Light Depth Cameras

Pietro Zanuttigh • Giulio Marin • Carlo Dal Mutto
Fabio Dominio • Ludovico Minto
Guido Maria Cortelazzo

Time-of-Flight and Structured Light Depth Cameras

Technology and Applications

Springer

Pietro Zanuttigh
Department of Information Engineering
University of Padova
Padova, Italy

Giulio Marin
Department of Information Engineering
University of Padova
Padova, Italy

Carlo Dal Mutto
Aquifi Inc.
Palo Alto, CA, USA

Fabio Dominio
Department of Information Engineering
University of Padova
Padova, Italy

Ludovico Minto
Department of Information Engineering
University of Padova
Padova, Italy

Guido Maria Cortelazzo
3D Everywhere s.r.l.
Padova, Italy

ISBN 978-3-319-30971-2 ISBN 978-3-319-30973-6 (eBook)
DOI 10.1007/978-3-319-30973-6

Library of Congress Control Number: 2016935940

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

“Cras ingens iterabimus aequor” (Horace, Odes, VII)
In memory of Alberto Apostolico (1948–2015)
unique scholar and person

Preface

This book originates from three-dimensional data processing research in the Multimedia Technology and Telecommunications Laboratory (LTTM) at the Department of Information Engineering of the University of Padova. The LTTM laboratory has a long history of research activity on consumer depth cameras, starting with Time-of-Flight (ToF) depth cameras in 2008 and continuing since, with a particular focus on recent structured light and ToF depth cameras like the two versions of Microsoft Kinect™. In the past years, the students and researchers at the LTTM laboratory have extensively explored many topics on 3D data acquisition, processing, and visualization, all fields of large interest for the computer vision and the computer graphics communities, as well as for the telecommunications community active on multimedia.

In contrast to a previous book by some of the authors, published as Springer Briefs in Electrical and Computer Engineering targeted to specialists, this book has been written for a wider audience, including students and practitioners interested in current consumer depth cameras and the data they provide. This book focuses on the system rather than the device and circuit aspects of the acquisition equipment. Processing methods required by the 3D nature of the data are presented within general frameworks purposely as independent as possible from the technological characteristics of the measurement instruments used to capture the data. The results are typically presented by practical exemplifications with real data to give the reader a clear and concrete idea about the actual processing possibilities.

This book is organized into three parts, the first devoted to the working principles of ToF and structured light depth cameras, the second to the extraction of accurate 3D information from depth camera data through proper calibration and data fusion techniques, and the third to the use of 3D data in some challenging computer vision applications.

This book comes from the contribution of a great number of people besides the authors. First, almost every student who worked at the LTTM laboratory in the past years gave some contribution to the know-how at the basis of this book and must be acknowledged. Among them, in particular, Alvise Memo must be thanked for his help with the acquisitions from a number of different depth cameras and for his review of this book. Many other students and researchers have contributed, and we would like to thank also Mauro Donadeo, Marco Fraccaro, Giampaolo Pagnutti, Luca Palmieri, Mauro Piazza, and Elena Zennaro. We consider a major contribution to this book the proofreading by Ilene Rafii which improved not only the quality of the English language but also the readability of this book in many parts. The authors would like to acknowledge 3DEverywhere which in 2008 purchased the first ToF camera with which the research about depth sensors at the LTTM laboratory began. Among the 3DEverywhere people, a special thank goes to Enrico Cappelletto, Davide Cerato, and Andrea Bernardi. We would also like to thank Gerard Dahlman and Tierry Oggier for the great collaboration we received from Mesa Imaging, Arrigo Benedetti (now with Microsoft, formerly with Canesta), and Abbas Rafii (with Aquifi) who helped the activity of the LTTM laboratory in various ways.

This book also benefited from the discussions and the supportive attitude of many colleagues, among which we would like to recall David Stoppa and Fabio Remondino (with FBK), Roberto Manduchi (with U.C.S.C.), Stefano Mattoccia (with the University of Bologna), Marco Andreetto (with Google), and Tim Droz and Mitch Reifel (with SoftKinetic).

Padova, Italy
January 2016

Pietro Zanuttigh
Giulio Marin
Carlo Dal Mutto
Fabio Dominio
Ludovico Minto
Guido Maria Cortelazzo

Contents

Chapter 1
Introduction

The acquisition of the geometric description of dynamic scenes has traditionally been a challenging task which required state of the art technology and instrumentation only accessible by research labs or major companies until professional-grade and consumer-grade depth cameras arrived in the market. Both professional-grade and consumer-grade depth cameras mainly belong to two technological families, one based on the *active triangulation* working principle and the other based on the *Time-of-Flight* working principle. The cameras belonging to the active triangulation family are usually called *structured light* depth cameras, while the cameras belonging to the second family are usually called *matricial Time-of-Flight* depth cameras, or simply ToF depth cameras, as in the remainder of this book.

Structured light depth cameras are the most diffused depth cameras in the market. Among them, the most notable example is the Primesense camera used in the first generation of Microsoft Kinect$^{\text{TM}}$. ToF depth cameras have historically been considered professional-grade (e.g., Mesa Imaging SwissRanger), however, recently they have also appeared as consumer-grade products, such as the first and second generation of Microsoft Kinect$^{\text{TM}}$, from now on called Kinect$^{\text{TM}}$ v1 and v2.

In several technical communities, especially those of computer vision, artificial intelligence, and robotics, a large interest has risen for these devices, along with the following questions: "What is a ToF camera?", "How does the Kinect$^{\text{TM}}$ work?", "Are there ways to improve the low resolution and high noise characteristics of ToF cameras data?", "How far can I go with the depth data provided by a 100–150 dollar consumer-grade depth camera with respect to those provided by a few thousand dollars professional-grade ToF camera?". This book tries to address these and other similar questions from a data user's point of view, as opposed to a technology developer's perspective.

This first part of this book describes the technology behind structured light and ToF cameras. The second part focuses on how to best exploit the data produced

P. Zanuttigh et al., *Time-of-Flight and Structured Light Depth Cameras*,
DOI 10.1007/978-3-319-30973-6_1

by structured light and ToF cameras, i.e., on the processing methods best suited to depth information. The third part reviews a number of applications where depth data provide significant contributions.

This book leverages on the depth nature of the data to present approaches that are as device-independent as possible. Therefore, we refer as often as possible to *depth cameras* and make the distinction between structured light and ToF cameras only when necessary. We focus on the depth nature of the data, rather than on the devices themselves, to establish a common framework suitable for current data from both structured light and ToF cameras, as well as data from new devices of these families that will reach the market in the next few years. Although structured light and ToF cameras are functionally equivalent depth cameras, i.e, providers of depth data, there are fundamental technological differences between them which cannot be ignored. These differences strongly impact noise, artifacts and production costs.

The synopsis of distance measurement methods in Fig. 1.1, derived from [17], offers a good framework to introduce these differences. For the purposes of this book, the reflective optical methods of Fig. 1.1 are typically classified into *passive* and *active*. Passive range sensing refers to 3D distance measurement by way of radiation (typically, but not necessarily, in the visible spectrum) already present in the scene. Stereo-vision systems are a classical example of this family of methods. Active sensing refers instead to 3D distance measurement obtained by projecting some form of radiation in the scene, as done for instance by structured light and ToF depth cameras.

The operation of structured light and ToF depth cameras involves a number of different concepts about imaging systems, ToF sensors and computer vision. These

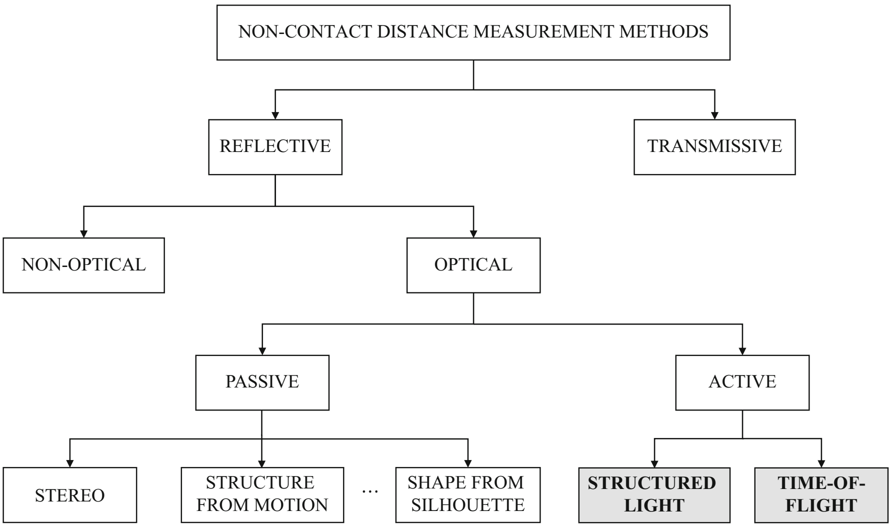


Fig. 1.1 Taxonomy of distance measurement methods (derived from [17])

concepts are recalled in the next two sections of this chapter to equip the reader with the notions needed for the remainder of the book; the next two sections can be skipped by readers already acquainted with structured light and ToF systems operation.

The depth or distance measurements taken by the systems of Fig. 1.1 can typically be represented by depth maps, i.e., data with each spatial coordinate (u, v) associated with the corresponding depth value z, and the depth maps can be combined into full all-around 3D models [14] as will be seen in Chap. 7. Data made by a depth map together with the corresponding color image are also referred to as RGB-D data.

1.1 Basics of Imaging Systems

1.1.1 Pin-Hole Camera Model

Let us consider a 3D reference system with axes x, y and z, called *Camera Coordinates System (CCS)*, with origin at O, called *center of projection*, and a plane parallel to the (x, y)-plane intersecting the z-axis at negative z-coordinate f, called *sensor* or *image plane* S as shown in Fig. 1.2. The axes' orientations follow the so called right-hand convention. Consider also a 2D reference system

$$\begin{aligned} u &= x + c_x \\ v &= y + c_y \end{aligned} \tag{1.1}$$

associated with the sensor, called *S-2D reference system*, oriented as shown in Fig. 1.2a. The intersection c of the z-axis with the sensor plane has coordinates $\mathbf{c} = [c_x, c_y]^T$. The set of sensor points p, called *pixels*, of coordinates $\mathbf{p} = [u, v]^T$ obtained from the intersection of the rays connecting the center of projection O with all the 3D scene points P with coordinates $\mathbf{P} = [x, y, z]^T$, is the scene footprint on the sensor S.

The relationship between P and p, called *central* or *perspective projection*, can be shown by triangle similarity (see Fig. 1.2b, c) to be

$$\begin{cases} u - c_x = f\dfrac{x}{z} \\ v - c_y = f\dfrac{y}{z} \end{cases} \tag{1.2}$$

where the distance $|f|$ between the sensor plane and the center of projection O is typically called *focal length*. In the adopted notation, f is the negative coordinate of the location of the sensor plane with respect to the z-axis. The reader should be aware that other books adopt a different notation, where f denotes the focal length, hence it is a positive number and the z coordinate of the sensor plane is denoted as $-f$.

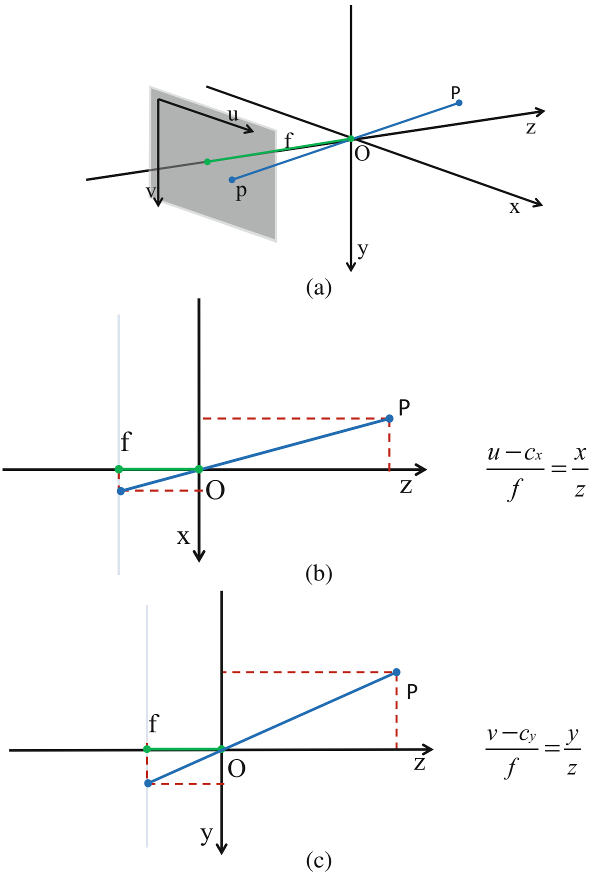


Fig. 1.2 Perspective projection: (**a**) scene point P projected to sensor pixel p; (**b**) horizontal section of (**a**); (**c**) vertical section of (**a**)

The perspective projection (1.2) is a good description of the geometric relationship between the coordinates of the scene points and the corresponding location in an image obtained by a pin-hole imaging device with the pin-hole positioned at center of projection O. Such a system allows a single light ray to go through the pin-hole at O. For a number of reasons, in imaging systems it is more practical to use optics, i.e., suitable sets of lenses, instead of pin-holes. Quite remarkably, the ideal model of an optical system, called *thin-lens model*, maintains the relationship (1.2) between the coordinates of P and of p if the lens' optical center (or *nodal point*) is in O and the lens' optical axis, i.e., the line orthogonally intersecting the lens at its nodal point, is orthogonal to the sensor. If a thin lens replaces a pin-hole in Fig. 1.2c, the optical axis coincides with the z-axis of the CCS.

1.1.2 Camera Geometry and Projection Matrix

Projective geometry associates to each 2D point p with Cartesian coordinates $\mathbf{p} = [u, v]^T$ of a plane a 3D representation called *2D homogeneous coordinates* $\tilde{\mathbf{p}} = [hu, hv, h]^T$, where h is any real constant. The usage of $h = 1$ is rather common and $[u, v, 1]^T$ is often called the *extended vector* of p [57].

The coordinates $\mathbf{p} = [u, v]^T$ can be obtained by dividing $\tilde{\mathbf{p}} = [hu, hv, h]^T$ by its third coordinate h. Vector $\tilde{\mathbf{p}}$ can be interpreted as the 3D ray connecting the sensor point p with the center of projection O.

In a similar way each 3D point P with Cartesian coordinates $\mathbf{P} = [x, y, z]^T$ can be represented in 3D homogeneous coordinates by a 4D vector $\tilde{\mathbf{p}} = [hx, hy, hz, h]^T$ where h is any real constant. Vector $[x, y, z, 1]^T$ is often called the *extended vector* of P.

The coordinates $\mathbf{P} = [x, y, z]^T$ can be obtained by dividing $\tilde{\mathbf{P}} = [hx, hy, hz, h]^T$ by its fourth coordinate h. An introduction to projective geometry suitable to computer vision applications can be found in [32].

The homogeneous coordinates representation of p allows one to rewrite the non-linear relationship (1.2) in a convenient matricial form:

$$z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}. \tag{1.3}$$

Note that the left side of (1.3) represents p in 2D homogeneous coordinates but the right side of (1.3) represents P in 3D Cartesian coordinates. It is straightforward to add a column with all 0 entries at the right of the matrix in order to represent P in homogeneous coordinates as well. This latter representation is more common than (1.3), which nevertheless is often adopted for its simplicity [57].

Digital sensor devices are typically planar matrices of rectangular sensor cells hosting photoelectric conversion systems based on CMOS or CCD technology in the case of digital cameras or video cameras, or single ToF receivers in the case of ToF cameras, as explained in Sect. 1.4. Customarily, they are modeled as a rectangular lattice Λ_S with horizontal and vertical step-size k_u and k_v respectively, as shown in Fig. 1.3a.

Given the finite sensor size, only a rectangular window of Λ_S made by N_C columns and N_R rows is of interest for imaging purposes.

In order to deal with normalized lattices with origin at $(0, 0)$ and unitary pixel coordinates $\mathbf{u_S} \in [0, \ldots, N_C - 1]$ and $\mathbf{v_S} \in [0, \ldots, N_R - 1]$ in both the u and v direction, relationship (1.3) is replaced by

$$z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{1.4}$$

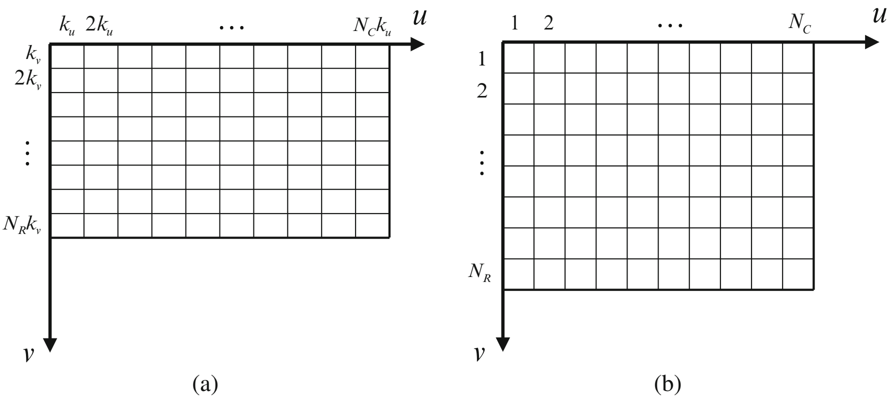


Fig. 1.3 2D sensor coordinates: (**a**) rectangular window of a non-normalized orthogonal lattice; (**b**) rectangular window of a normalized orthogonal lattice

where **K** is the intrinsic parameters matrix defined as

$$\mathbf{K} = \begin{bmatrix} f_x & \alpha & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1.5}$$

with $f_x = fk_u$ the x-axis focal length of the optics, $f_y = fk_v$ the y-axis focal length of the optics, c_x and c_y the (u, v) coordinates of the intersection of the optical axis with the sensor plane. All these quantities are expressed in [pixel], i.e., since f is in [mm], k_u and k_v are assumed to be [pixel]/[mm]. Notice also that an additional parameter α (*axis skew*) is sometimes used to account for the fact that the two axes in the sensor lattice are not perfectly perpendicular, however since it is typically negligible we will not consider it in the rest of the book and approximate **K** by the r.h.s. of (1.5). The symbol $\approx$ within (1.5) denotes approximation.

In many practical situations it is convenient to represent the 3D scene points not with respect to the CCS, but with respect to a different easily accessible reference system conventionally called *World Coordinate System (WCS)*, in which a scene point denoted as P has coordinates $\mathbf{P}_W = [x_W, y_W, z_W]^T$. The relationship between the representation of a scene point with respect to the CCS, denoted as **P**, and its representation with respect to the WCS, denoted as $\mathbf{P}_W$, is

$$\mathbf{P} = \mathbf{R}\mathbf{P}_W + \mathbf{t} = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix} \mathbf{P}_W + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \tag{1.6}$$

where **R** and **t** are a suitable rotation matrix and translation vector, respectively. For future usage let us introduce an explicit notation for the rows $\mathbf{r}_i^T$, $i = 1, 2, 3$ of **R** and the components t_i, $i = 1, 2, 3$ of **t**. By representing $\mathbf{P}_W$ at the right side in homogeneous coordinates $\tilde{\mathbf{P}}_W = [hx_W, hy_W, hz_W, h]^T$ and choosing $h = 1$, the relationship (1.6) can be rewritten as

$$\mathbf{P} = [\mathbf{R} \mid \mathbf{t}]\tilde{\mathbf{P}}_W. \tag{1.7}$$

In this case, the relationship between a scene point represented in homogeneous coordinates with respect to the WCS and its corresponding pixel in homogeneous coordinates, from (1.4), becomes

$$\tilde{\mathbf{p}} \cong \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \cong \frac{1}{z}\mathbf{KP} \cong \frac{1}{z}\mathbf{K}[\mathbf{R} \mid \mathbf{t}]\tilde{\mathbf{P}}_W \cong \frac{1}{z}\mathbf{M}\tilde{\mathbf{P}}_W \cong \frac{1}{z}\mathbf{M}\begin{bmatrix} x_W \\ y_W \\ z_W \\ 1 \end{bmatrix} \tag{1.8}$$

where the 3×4 matrix

$$\mathbf{M} = \mathbf{K}[\mathbf{R} \mid \mathbf{t}] = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix} \tag{1.9}$$

is called *projection matrix*. Projection matrix $\mathbf{M}$ depends on the intrinsic parameters matrix $\mathbf{K}$ and on the extrinsic parameters $\mathbf{R}$ and $\mathbf{t}$ of the imaging system. A projection matrix $\mathbf{M}$ is said to be in normalized form if its bottom row is exactly $\mathbf{m}_3^T = [\mathbf{r}_3^T \mid t_3]$. It is straightforward to see that if $\mathbf{M}$ is in normalized form, (1.8) holds with the equality sign: in this case $z = \mathbf{r}_3^T\tilde{\mathbf{P}}_W + t_3$ assumes the value of the depth of P_W with respect to the camera reference system. By denoting with $\mathbf{m}_i^T$, $i = 1, 2, 3$ the rows of $\mathbf{M}$, the image coordinates (u, v) of point P from (1.8) can be written as

$$\begin{cases} u = \dfrac{\mathbf{m}_1^T\tilde{\mathbf{P}}_W}{\mathbf{m}_3^T\tilde{\mathbf{P}}_W} = \dfrac{\tilde{\mathbf{P}}_W^T\mathbf{m}_1}{\tilde{\mathbf{P}}_W^T\mathbf{m}_3} \\ v = \dfrac{\mathbf{m}_2^T\tilde{\mathbf{P}}_W}{\mathbf{m}_3^T\tilde{\mathbf{P}}_W} = \dfrac{\tilde{\mathbf{P}}_W^T\mathbf{m}_2}{\tilde{\mathbf{P}}_W^T\mathbf{m}_3} \end{cases}. \tag{1.10}$$

The symbol $\cong$ within (1.8) denotes that in general, the equality holds up to a multiplicative constant since it involves homogeneous coordinates. In this sense $\mathbf{M}$ is also defined up to a multiplicative constant since it has 12 parameters but just 11 degrees of freedom: 5 from $\mathbf{K}$ (4 excluding the skew parameter), 3 from $\mathbf{R}$ and 3 from $\mathbf{t}$.

From a set of J known 2D-3D correspondence values $(p^j, P^j), j = 1, \ldots, J$ from (1.10) one may derive a set of $2J$ homogeneous linear equations

$$\begin{cases} \mathbf{m}_3^T\tilde{\mathbf{P}}_W^j u^j - \mathbf{m}_1^T\tilde{\mathbf{P}}_W^j = 0 \\ \mathbf{m}_3^T\tilde{\mathbf{P}}_W^j v^j - \mathbf{m}_2^T\tilde{\mathbf{P}}_W^j = 0 \end{cases} \quad j = 1, \ldots, J \tag{1.11}$$

from which $\mathbf{M}$ can be computed. In principle, $J = 6$ correspondences suffice since $\mathbf{M}$ has 12 entries; in practice, one should use $J \gg 6$ in order to effectively deal with noise and non-idealities. However, this method, typically called *Direct Linear Transform* (DLT), only minimizes a target with algebraic significance, and is not invariant with respect to Euclidean transformations. Therefore the result of the DLT is typically used as starting point for a nonlinear minimization either in L_2 or L_∞ directly addressing Eqs. (1.10), for example

$$\min_{\mathbf{K},\mathbf{R},\mathbf{t}} \sum_{j=1}^{J} |p^j - f(\mathbf{K}, \mathbf{R}, \mathbf{t}, P^j)|^2 \tag{1.12}$$

where $f(\mathbf{K}, \mathbf{R}, \mathbf{t}, P^j)$ is a function that given $\mathbf{K}$, $\mathbf{R}$ and $\mathbf{t}$, projects P in the image plane, as in (1.8). More details on the estimation of $\mathbf{K}$, $\mathbf{R}$ and $\mathbf{t}$ will be provided in Chap. 4.

1.1.3 Lens Distortions

As a consequence of distortions and aberrations of real optics, the coordinates $\hat{\mathbf{p}} = (\hat{u}, \hat{v})$ of the pixel actually associated with scene point P with coordinates $\mathbf{P} = [x, y, z]^T$ in the CCS system do not satisfy relationship (1.4). The correct pixel coordinates (u, v) of (1.4) can be obtained from the distorted coordinates $(\hat{u}, \hat{v})$ actually measured by the imaging system, by inverting suitable distortion models, such as

$$\mathbf{p}_T = \Psi^{-1}(\hat{\mathbf{p}}_T) \tag{1.13}$$

where $\Psi(\cdot)$ denotes the distortion transformation.

Anti-distortion model (1.14), also called the *Heikkila model* [33], has become popular since it adequately corrects the distortions of most imaging systems and effective methods exist for computing its parameters:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \Psi^{-1}(\hat{\mathbf{p}}_T) = \begin{bmatrix} \hat{u}(1 + k_1 r^2 + K_2 r^4 + k_3 r^6) + 2d_1\hat{u}\hat{v} + d_2(r^2 + 2\hat{u}^2) \\ \hat{v}(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + d_1(r^2 + 2\hat{v}^2) + 2d_2\hat{u}\hat{v} \end{bmatrix} \tag{1.14}$$

where $r = \sqrt{(\hat{u} - c_x)^2 + (\hat{v} - c_y)^2}$, parameters k_i with $i = 1, 2, 3$ are constants accounting for radial distortion and d_i with $i = 1, 2$ account for tangential distortion. A number of other more complex models, e.g. [18], are also available.

Distortion parameters

$$\mathbf{d} = [k_1, k_2, k_3, d_1, d_2] \tag{1.15}$$

are intrinsic camera parameters to be considered together with $[f, k_u, k_v, c_x, c_y]$. Equation (1.12) can be modified to also account for distortion, in this case, the projection function f becomes $f(\mathbf{K}, \mathbf{R}, \mathbf{t}, \mathbf{d}, P^j)$.

The estimation of intrinsic and extrinsic parameters of an imaging system by suitable methods such as [16] and [6] is called *geometric calibration* and is discussed in Chap. 4.

1.2 Stereo Vision Systems

This section and the previous one summarize basic computer vision concepts necessary for understanding the rest of this book and can be skipped by readers familiar with computer vision. Readers interested in a more extensive presentation of these topics are referred to computer vision textbooks such as [15, 20, 22, 24, 26, 27, 32, 45, 48, 55, 57, 61].

1.2.1 Two-view Stereo Systems

A stereo vision, or *stereo*, system is made by two standard cameras partially framing the same scene, namely the left camera L, also called *reference camera*, and the right camera R, also called *target camera*. Each camera is assumed to be calibrated, with calibration matrices $\mathbf{K}_L$ and $\mathbf{K}_R$ for the L and R cameras respectively. As previously seen, each has its own 3D CCS and 2D reference systems, as shown in Fig. 1.4. Namely, the L camera has CCS with coordinates (x_L, y_L, z_L), also called *L-3D reference system*, and a 2D reference system with coordinates (u_L, v_L). The R camera has CCS with coordinates (x_R, y_R, z_R), also called *R-3D reference system*,

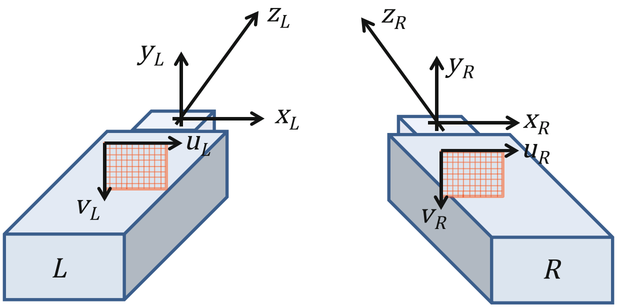


Fig. 1.4 Stereo vision system coordinates and reference systems

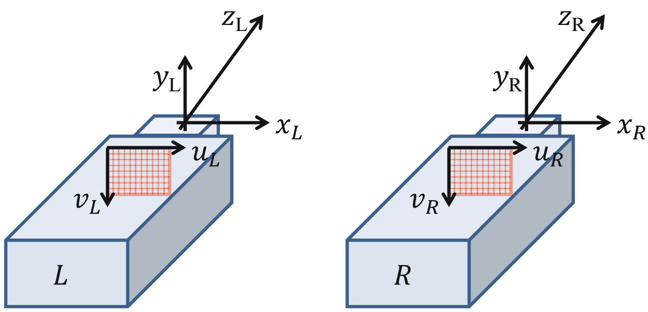
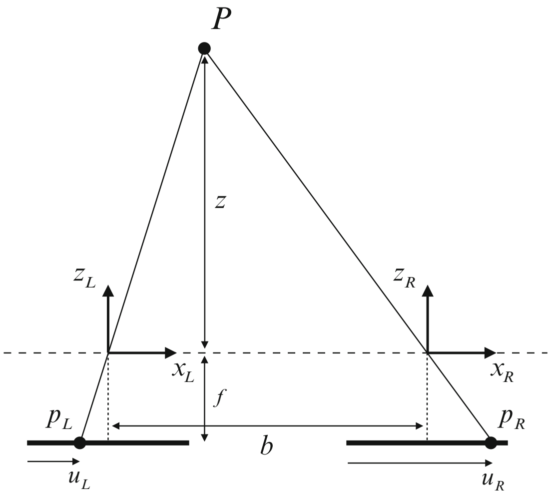


Fig. 1.5 Rectified stereo system

Fig. 1.6 Triangulation with a rectified stereo system



and a 2D reference system with coordinates (u_R, v_R). The two cameras may be different, but in this book they are assumed to be identical, with $\mathbf{K} = \mathbf{K}_L = \mathbf{K}_R$, unless explicitly stated. A common convention is to consider the L-3D reference system as the reference system of the stereo vision system and to denote it as *S-3D reference system*.

Let us momentarily consider the case of a calibrated and rectified stereo vision system, i.e., a stereo vision system made by two identical standard cameras with coplanar and aligned imaging sensors and parallel optical axes as shown in Fig. 1.5. In rectified stereo vision systems points p_L and p_R have the same vertical coordinates. By denoting

$$d = u_L - u_R \tag{1.16}$$

the difference between their horizontal coordinates, called *disparity*, a 3D point P with coordinates $\mathbf{P}_L = [x_L, y_L, z_L]^T$ with respect to the S-3D reference system, is projected to the pixels p_L and p_R of the L and R cameras with coordinates

$$\mathbf{p}_L = \begin{bmatrix} u_L \\ v_L \end{bmatrix} \qquad \mathbf{p}_R = \begin{bmatrix} u_R = u_L - d \\ v_R = v_L \end{bmatrix} \tag{1.17}$$

respectively. Furthermore, let $\mathbf{P}_R = [x_R, y_R, z_R]^T$ denote the coordinate of P with respect to the R-3D reference system and let $(\mathbf{R}, \mathbf{t})$ denote the rigid transformation mapping the R-3D reference system to the L-3D reference system, which is also the S-3D reference system, i.e.,

$$\mathbf{P}_R = \mathbf{R}\mathbf{P}_L + \mathbf{t}. \tag{1.18}$$

By introducing normalized image coordinates

$$\begin{aligned}
\tilde{\mathbf{q}}_L &= \begin{bmatrix} u'_L \\ v'_L \\ 1 \end{bmatrix} = \mathbf{K}^{-1}\tilde{\mathbf{p}}_L = \begin{bmatrix} \frac{1}{f} & 0 & -\frac{c_x}{f} \\ 0 & \frac{1}{f} & -\frac{c_y}{f} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{u_L - c_x}{f} \\ \frac{v_L - c_y}{f} \\ 1 \end{bmatrix} \\
\tilde{\mathbf{q}}_R &= \begin{bmatrix} u'_R \\ v'_R \\ 1 \end{bmatrix} = \mathbf{K}^{-1}\tilde{\mathbf{p}}_R = \begin{bmatrix} \frac{1}{f} & 0 & -\frac{c_x}{f} \\ 0 & \frac{1}{f} & -\frac{c_y}{f} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_R \\ v_R \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{u_R - c_x}{f} \\ \frac{v_R - c_y}{f} \\ 1 \end{bmatrix}
\end{aligned} \tag{1.19}$$

the Cartesian coordinates of P with respect to the L and R 3D reference system can be written as

$$\begin{aligned}
\mathbf{P}_L &= z_L \mathbf{K}^{-1}\tilde{\mathbf{p}}_L = z_L \tilde{\mathbf{q}}_L \\
\mathbf{P}_R &= z_R \mathbf{K}^{-1}\tilde{\mathbf{p}}_R = z_R \tilde{\mathbf{q}}_R
\end{aligned} \tag{1.20}$$

and (1.18) can be rewritten as

$$z_R \tilde{\mathbf{q}}_R - z_L \mathbf{R}\tilde{\mathbf{q}}_L = \mathbf{t} \tag{1.21}$$

or

$$\begin{cases} z_R u'_R - z_L \mathbf{r}_1^T \tilde{\mathbf{q}}_L = t_1 \\ z_R v'_R - z_L \mathbf{r}_2^T \tilde{\mathbf{q}}_L = t_2 \\ z_R - z_L \mathbf{r}_3^T \tilde{\mathbf{q}}_L = t_3 \,. \end{cases} \tag{1.22}$$

By substituting in the first equation of (1.22) $z_R = t_3 + z_L \mathbf{r}_3^T \tilde{\mathbf{q}}_L$, derived from the third equation, one obtains

$$z_L = \frac{t_1 - u'_R t_3}{u'_R \mathbf{r}_3^T \tilde{\mathbf{q}}_L - \mathbf{r}_1^T \tilde{\mathbf{q}}_L} \,. \tag{1.23}$$

Equation (1.23) shows that the depth, i.e., the z coordinate, of $3D$ point P with respect to the L-3D reference system denoted z_L can be obtained upon knowledge of the left image coordinate $\tilde{\mathbf{p}}_L$ and of the right image coordinate $\tilde{\mathbf{p}}_R$ of point P, assuming the stereo system calibration parameters are known. These parameters are the external calibration parameters $(\mathbf{R}, \mathbf{t})$, relating the position of the right camera to the left camera, and the internal calibration parameters $\mathbf{K}$, concerning both cameras of the rectified stereo system. The procedure computing the stereo system calibration parameters will be seen in Chap. 4. Such procedure delivers as output the left and right camera projection matrices, which within the assumed conventions respectively result in

$$\mathbf{M}_L = \mathbf{K}_L \big[\, \mathbf{I} \mid \mathbf{0} \,\big] \qquad \mathbf{M}_R = \mathbf{K}_R \big[\, \mathbf{R} \mid \mathbf{t} \,\big] . \tag{1.24}$$

The methods indicated in Sect. 1.2.1.3 can be used to solve the so-called correspondence problem, i.e., the automatic determination of image points $\tilde{p}_L$ and $\tilde{p}_R$, called conjugate points. *Triangulation* or *computational stereopsis* is the process by which one may compute the 3D coordinates $\mathbf{P}_L = [x_L, y_L, z_L]^T$ of a scene point P from (1.20), from the knowledge of conjugate points $\tilde{\mathbf{p}}_L$ and $\tilde{\mathbf{p}}_R$, obtained by solving the correspondence problem, i.e., as

$$\mathbf{P}_L = \begin{bmatrix} x_L \\ y_L \\ z_L \end{bmatrix} = \mathbf{K}_L^{-1} \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix} z \tag{1.25}$$

where $\mathbf{K}_L^{-1}$ is the inverse of the intrinsic parameters matrix (1.5) of camera L (or R) of the stereo system.

In the case of a rectified system, where $\mathbf{K} = \mathbf{K}_L = \mathbf{K}_R$, as shown in Fig. 1.6, the parameters entering $\mathbf{M}_R$ in (1.24) are $\mathbf{R} = \mathbf{I}$ and $\mathbf{t} = [-b, 0, 0]^T$ and it can be readily seen that from $\mathbf{r}_3^T \tilde{\mathbf{q}}_L = 1$ and $\mathbf{r}_1^T \tilde{\mathbf{q}}_L = u'_L$, expression (1.23) becomes

$$z_L = \frac{-b}{u'_R - u'_L} = -\frac{bf}{u_R - u_L} = \frac{bf}{d} \tag{1.26}$$

where d is the disparity defined in (1.16). Equation (1.26), which shows that disparity is inversely proportional to the depth value z of P, can be directly obtained from the similarities of the triangles inscribed within the triangle with vertices at P, p_L and p_R of Fig. 1.6. Indeed, from the established CCS conventions, one can write for the L camera

$$\frac{u_L - c_x}{x_L} = \frac{f}{z_L} \tag{1.27}$$

and for the R camera

$$\frac{u_R - c_x}{x_L - b} = \frac{f}{z_R} = \frac{f}{z_L} \tag{1.28}$$

since in the case of rectified stereo systems $x_R = x_L - b$ and $z_R = z_L$. By substituting (1.27) in (1.28) one obtains

$$z_L = \frac{u_L - c_x}{u_R - c_x} z_L - \frac{bf}{u_R - c_x} \tag{1.29}$$

which gives (1.26). The above derivation is what justifies the name of *triangulation* for the procedure adopted to infer the 3D coordinates of a scene point P from its conjugate image points p_L and p_R.

The procedure actually used for triangulation or stereopsis can be summarized in very general terms as follows. Since

$$\begin{cases} \tilde{\mathbf{p}}_L \cong \frac{1}{z}\mathbf{M}_L \tilde{\mathbf{P}}_L \\ \tilde{\mathbf{p}}_R \cong \frac{1}{z}\mathbf{M}_R \tilde{\mathbf{P}}_L \end{cases} \tag{1.30}$$

where $\mathbf{M}_L = [\mathbf{m}_{1L}^T,\ \mathbf{m}_{2L}^T,\ \mathbf{m}_{3L}^T]$ and $\mathbf{M}_R = [\mathbf{m}_{1R}^T,\ \mathbf{m}_{2R}^T,\ \mathbf{m}_{3R}^T]$ are the perspective projection matrices of the L and R camera of (1.24) and $\tilde{\mathbf{P}}_L$ represents the coordinates of P with respect to the S-3D reference system, assumed to be the L-3D reference system, by (1.10) expression (1.30) can be rewritten as

$$\begin{bmatrix} \mathbf{m}_{3L}^T u_L - \mathbf{m}_{1L}^T \\ \mathbf{m}_{3L}^T v_L - \mathbf{m}_{2L}^T \\ \mathbf{m}_{3R}^T u_R - \mathbf{m}_{1R}^T \\ \mathbf{m}_{3R}^T v_R - \mathbf{m}_{2R}^T \end{bmatrix} \tilde{\mathbf{P}}_L = \mathbf{0}_{4\times 1} \tag{1.31}$$

which, since $\mathbf{p}_L$, $\mathbf{p}_R$, $\mathbf{M}_L$, and $\mathbf{M}_R$ are assumed known, corresponds to a linear homogeneous system of four equations in the unknown coordinates of P. Clearly

(1.31) gives a non-trivial solution only if the system matrix has rank 3. This condition may not always be verified because of noise. The so-called *linear-eigen* method [31] based on singular value decomposition overcomes such difficulties. As already seen for the estimate of **M** by the DLT method, since the estimate of P returned by (1.31) complies only with an algebraic criterion, it is typical to use it as a starting point for the numerical optimization of (1.31), in terms of

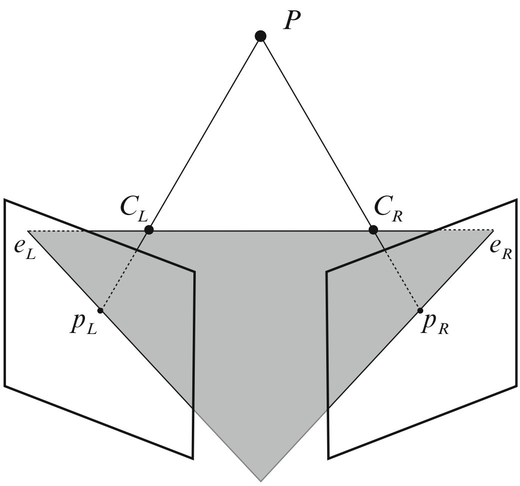
$$\min_{\tilde{\mathbf{P}}_L} \left\{ \left(u_L - \frac{\mathbf{m}_{1L}^T \tilde{\mathbf{P}}_L}{\mathbf{m}_{3L}^T \tilde{\mathbf{P}}_L} \right)^2 + \left(v_L - \frac{\mathbf{m}_{2L}^T \tilde{\mathbf{P}}_L}{\mathbf{m}_{3L}^T \tilde{\mathbf{P}}_L} \right)^2 + \right.$$
$$\left. + \left(u_R - \frac{\mathbf{m}_{1R}^T \tilde{\mathbf{P}}_L}{\mathbf{m}_{3R}^T \tilde{\mathbf{P}}_L} \right)^2 + \left(v_R - \frac{\mathbf{m}_{2R}^T \tilde{\mathbf{P}}_L}{\mathbf{m}_{3R}^T \tilde{\mathbf{P}}_L} \right)^2 \right\}. \tag{1.32}$$

Equation (1.32) can be interpreted as a variation of (1.12), where the reprojection error is jointly minimized in both cameras. Note that the goal of (1.32) is to find the coordinates of P, rather than **M**, as in (1.12).

1.2.1.1 Epipolar Geometry

Figure 1.7 schematically represents the stereo system of Fig. 1.5 and evidences only some elements of special geometric significance, such as the optical centers C_L and C_R and the image planes of the two cameras. It shows that given p_L, its conjugate p_R must lie on the plane defined by p_L, C_L, and C_R, called *epipolar plane*, and similarly for p_R. This geometric constraint implies that given p_L, its conjugate point p_R can only be sought along the intersection of the epipolar plane through p_L, C_L,

Fig. 1.7 Epipolar geometry



and C_R with the right image plane, which is a line, called the *right epipolar line* of p_L. Similar reasoning applies to p_R. Epipolar geometry, which reduces the search for the conjugate point from planar to linear, is formalized by the Longuet-Higgins equation

$$\tilde{\mathbf{p}}_R^T \mathbf{F} \tilde{\mathbf{p}}_L^T = 0 \tag{1.33}$$

where 3×3 matrix $\mathbf{F}$ is called the *fundamental matrix* [44]. In practical settings due to noise and inaccuracies the equation does not perfectly hold and it can be replaced by the search for the conjugate points that minimize (1.33). The homogeneous equation of the epipolar line of $\mathbf{p}_L$ from (1.33) is $\tilde{\mathbf{p}}_R^T \mathbf{F}$ and similarly $\tilde{\mathbf{p}}_L^T \mathbf{F}$ is the equation of the epipolar line of $\mathbf{p}_R$.

Since the epipolar plane is defined by P, C_L, and C_R, it varies with P. Therefore, there are infinite epipolar planes forming infinite epipolar lines on the left and right image. It is worth noting that since every epipolar plane, i.e., the epipolar plane defined by any P, includes C_L and C_R, all epipolar planes include the baseline connecting C_L and C_R. Furthermore, the baseline intersects the left and right image planes at two points called *left epipole* e_L and *right epipole* e_R. Indeed, e_L and e_R belong to the bundle of all the left and right epipolar lines, since every epipolar plane defined by any P must include rays $p_L\, P$ and $p_R\, P$.

1.2.1.2 Epipolar Rectification

A stereo system is called rectified if it has parallel image planes, as in Fig. 1.6. This configuration is of special interest, since the epipoles become points at infinity; therefore, the epipolar lines, bounded to intersect the epipoles, become parallel lines as shown in Fig. 1.8. Such a geometry further simplifies the search for the conjugate of $\mathbf{p}_L = [u_L, v_L]^T$ in the right image, which epipolar geometry already turns from a 2D search to a 1D search, to a search on the horizontal right image line of equation $y = v_L$.

Figure 1.8 emphasizes that the projection matrices $\mathbf{M}_L$ and $\mathbf{M}_R$ and the left and right images I_L and I_R of the stereo system with vergent cameras differ from the those of the rectified system, respectively denoted as $\mathbf{M}'_L, \mathbf{M}'_R$ and I'_L, I'_R. In a rectified stereo system (Fig. 1.6), the left and the right projection matrices are

$$\mathbf{M}_L = \mathbf{K}\left[\,\mathbf{I} \mid \mathbf{0}\,\right] \qquad \mathbf{M}_R = \mathbf{K}\left[\,\mathbf{I} \mid [b, 0, 0]^T\,\right]. \tag{1.34}$$

There exist methods for computationally rectifying vergent stereo systems, such as the algorithm of [28] which first computes $\mathbf{M}'_L$ and $\mathbf{M}'_R$ from $\mathbf{M}_L$ and $\mathbf{M}_R$ and then rectifies the images, i.e., it computes I'_L and I'_R upon $\mathbf{M}'_L$ and $\mathbf{M}'_R$. Figure 1.9 shows an example of image rectification. In current practice, it is typical to apply computational stereopsis to rectified images, which is equivalent to computationally turning actual stereo systems into rectified stereo systems.

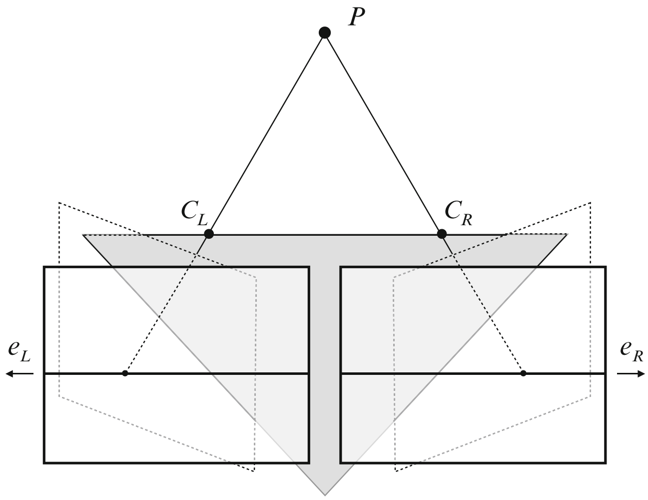


Fig. 1.8 Epipolar rectification

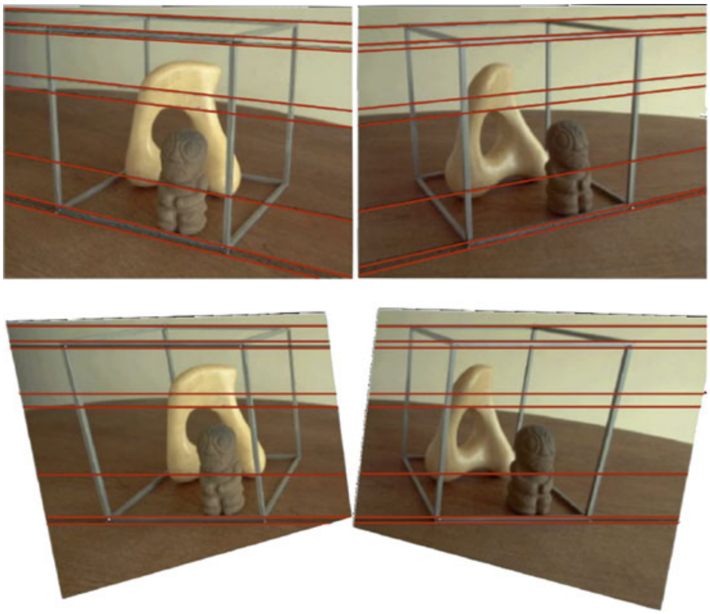


Fig. 1.9 *Top:* pair of images acquired by a vergent stereo system [41]. *Bottom:* rectified images. *Red lines* highlight some epipolar lines

1.2.1.3 The Correspondence Problem

The triangulation procedure assumes the availability of a pair of conjugate points p_L and p_R. This represents a delicate and tricky assumption for the triangulation procedure, first of all because such a pair may not exist due to occlusions. Even if it exists, it may not be straightforward to find it.

Indeed, the *correspondence problem*, i.e. the detection of conjugate points between the stereo image pairs, is one of the major challenges of stereo vision algorithms. The methods proposed for this task can be classified according to various criteria. A first distinction concerns dense and sparse stereo algorithms. The former, representing current trends [51], are methods aimed at finding a conjugate point for every pixel of the left (right) image, of course within the limits imposed by occlusions. The latter are methods which do not attempt to find a conjugate for every pixels.

A second distinction concerns the methods suited for short baseline and wide baseline stereo systems. The former implicitly assume the two images share considerable similarity characteristics hence, in principle, can adopt simpler methods with respect to the latter.

The third distinction concerns *local* and *global* approaches. Local methods consider only local similarity measures between the region surrounding p_L and regions of similar shape around all the candidate conjugate points p_R of the same row. The selected conjugate point is the one which maximizes the similarity measure, a method typically called *Winner Takes All (WTA)* strategy. Conversely, global methods do not consider each couple of points on their own, but instead estimate all of the disparity values at once, exploiting global optimization schemes. Global methods based on Bayesian formulations are currently receiving great attention in dense stereo. Such techniques generally model the scene as a Markov Random Field (MRF), and include within a unique framework clues coming from local comparisons between the two images and scene depth smoothness constraints. Global stereo vision algorithms typically estimate the disparity image by minimizing a cost function made by a *data term* representing the cost of local matches, similar to the computation of local algorithms (e.g., covariance) and a *smoothness term* defining the smoothness level of the disparity image by explicitly or implicitly accounting for discontinuities [57].

Wide baseline stereo methods traditionally rest on salient point detection techniques such as Harris corner detector [29]. *Scale Invariant Feature Transform* (SIFT) [42], which offers a robust salient point detector and an effective descriptor of the detected points, gave a truly major contribution to this field [47] and inspired a number of advances in related areas. An application of wide baseline matching which recently received major attention, as reported below, is 3D reconstruction from a generic collection of images of a scene [54].

It is finally worth recalling that although specific algorithms may have a considerable impact on the solution of the correspondence problem, the ultimate quality of 3D stereo reconstruction inevitably also depends on scene characteristics. This can be readily realized considering the case of a scene without geometric