DEPARTMENT OF
INFORMATION
ENGINEERING
UNIVERSITY OF PADOVA

Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria delle Telecomunicazioni

Depth Super-Resolution with Hybrid Camera System

Relatore:
Chiar.mo Prof. Giancarlo Calvagno

Correlatori:
Ing. Francesco Michielin
Ing. Piergiorgio Sartor

Laureando:
Alessandro Vianello

Padova, 14 Ottobre 2013

Abstract

An important field of research in computer vision is the 3D analysis and reconstruction of objects and scenes. Currently, among all the the techniques for 3D acquisition, stereo vision systems are the most common. These systems provide an estimation of the three-dimensional geometry of the scene by using two or more cameras and exploiting the triangulation principle. More recently, Time-of-Flight (ToF) range cameras have been introduced. They allow real-time depth estimation in condition where stereo does not work well. Unfortunately, ToF sensors still have a limited resolution (e.g., 200 × 200 pixel). The focus of this thesis is to combine the information from the ToF with one or two standard cameras, in order to obtain a high-resolution depth image. To this end, ToF cameras and stereo systems are described in Chapters 2 and 3 respectively. Then, a comparison between the two systems is provided in Chapter 4, in order to show their complementarity. Chapters 5 and 6 present two approaches to up-sample a low-resolution depth image, which exploit the additional information coming from a single camera color image or a two cameras disparity map. In Chapter 7 the camera rig calibration procedure and some tests for the noise characterization of the ToF camera are presented. Finally, Chapter 8 presents the entire framework of the method for fusing the ToF data with a single camera color image. Furthermore, an optional refinement based on the stereo matching is proposed. The algorithm allows to enhance the resolution of the ToF camera depth image up to 1920 × 1080 pixel. Moreover, the image noise is reduced in the super-resolution, thanks to a filtering procedure based on the ToF camera noise measurements.

Sommario

L'analisi e la ricostruzione tridimensionale di scene ed oggetti è uno dei più importanti settori di ricerca nell'ambito della visione artificiale. Al giorno d'oggi, tra tutti i sistemi di acquisizione 3D, i sistemi di visione stereoscopica sono quelli più comuni. Tali sistemi permettono di stimare la configurazione geometrica della scena utilizzando due o più telecamere e sfruttando il principio di triangolazione. Più recentemente sono stati introdotti sensori basati sul principio del tempo di volo. Come i sistemi stereoscopici, anche questi apparecchi permettono l'acquisizione 3D, e forniscono i risultati migliori proprio nelle condizioni in cui i sistemi stereoscopici danno i maggiori problemi. Sfortunatamente il principale svantaggio dei sensori a tempo di volo è la bassa risoluzione, la quale è nell'ordine dei 200 × 200 pixel. L'obiettivo di questa tesi è di ottenere una mappa di profondità ad alta risoluzione, partendo dall'informazione fornita dal sensore a tempo di volo e combinandola con l'informazione proveniente da una o due telecamere. A questo scopo, i due sistemi di acquisizione 3D sono analizzati nei Capitoli 2 e 3. Dopodichè i due sistemi vengono confrontati nel Capitolo 4, dal quale emerge chiaramente la loro complementarità. Nei Capitoli 5 e 6 vengono descritti due approcci per aumentare la risoluzione della mappa di profondità, i quali sfruttano l'informazione addizionale proveniente da una o due telecamere. Nel Capitolo 7 vengono presentate la procedura di calibrazione delle telecamere e i risultati di alcuni esperimenti per la caratterizzazione del rumore della telecamera a tempo di volo. Infine nel Capitolo 8 viene descritto l'intero sistema per la super-risoluzione della mappa di profondità del sensore a tempo di volo usando l'immagine a colori proveniente da una telecamera. L'algoritmo permette di raggiungere una risoluzione massima di 1920 × 1080 pixel. Viene inoltre proposta una procedura di rifinitura basata sull'uso di una seconda telecamera.

To my Family.

Acknowledgements

First of all I would like to thank Prof. Giancarlo Calvagno who gave me the possibility to work on this interesting and exciting project. I am also grateful to Oliver Erdler for the chance to work at Sony EuTEC in Stuttgart. I want to express my gratitude to both my supervisors Francesco Michielin and Piergiorgio Sartor for the advice given, the support, and for being so helpful every day. Thank you Francesco also for the help and the patience, I learned a lot of things under your guidance.

I would like to thank my colleagues from Sony EuTEC, especially Christian, Yalcin, Roman, Martin, Fabian, Jens, Thimo and Paul for the great time we spent together at the Stuttgart Technology Center. It has been a pleasure meeting and working with you. A special thank also to Sony's students, in particular Gevorg, Yi, Alex, and Oliver, my mates of coffee breaks and kicker sessions.

My seven months in Stuttgart have been so good also thanks to Corsano and Münzenmaier families who helped me feeling at home day by day.

Last, but not least, I would like to thank my family. My parents Fabio and Antonietta for supporting and believe in me during all my studies. A special thank to my sister Ilaria for all the help she gave me with the correction of this thesis. Without your dedication I would have never reached this goal.

Grazie di tutto!

Spinea, October 9th 2013 Alessandro Vianello

Contents

Chapter 1

Introduction

One of the most important field of research in computer vision is the 3D analysis and reconstruction of objects and scenes. The aim of 3D reconstruction is to produce a 2D image showing the distance to surface points on objects in a scene from a known reference point, normally associated with some type of sensor device. These information are fundamental inputs data for position determination, object recognition, or collision prevention. Therefore they can be applied in areas like robotics, automotive assistance, human machine interfaces and many more. Currently there exists a variety of systems for acquiring three-dimensional information about the target scene. For example *laser range scanners* and *structured light cameras*, which can provide extremely accurate and dense 3D measurements over a large working volume. However, laser range cameras can measure a single point at a time, limiting their applications to static scenes only. In a similar way, structured light techniques are not suited for real-time applications.

The most common approaches for 3D reconstruction are the well-known *stereo vision systems.* Stereo systems are able to deliver high-resolution range images by using two or more standard cameras, hence without any energy emission or moving parts. These systems works well on textured scenes, but have difficulties in textureless homogeneous regions. Moreover, depth discontinuities in stereo vision systems causes *occlusions.*

In recent years *time-of-flight* (*ToF*) range cameras have been introduced. These new sensors are active systems, which compute the depth at video frame rates (up to $80\,[fps]$). They exploit the known speed of light and measure for each point of the scene the time that an emitted *infrared* (*IR*) signal need to cover the distance from the emitter to the object and from the object to the camera. Compared to stereo systems, ToF systems have a greater power consumption since they are active systems. Moreover, these particular range cameras are, in contrary to stereo systems, not sensitive to the scene peculiarity, as for instance textureless surfaces. The main issues of ToF cameras are the provided low-resolution depth image and the several noise sources affecting the measurement. The most significant noise sources are the so-called flying pixels at depth boundaries and the wrong depth estimation in region with low infrared reflectance.

From their characteristics, stereo systems and ToF cameras can be considered as complementary systems. Therefore, a collaborative approach which exploits the best characteristics of the two systems may allow to reach a better quality of the final three-dimensional scene sensing. The aim of this thesis is to exploit the information coming from a camera rig composed by a ToF camera and three standard video-cameras in order to increase the low-resolution ToF depth up to the camera image resolution.

Chapter 2 starts with a model that describes the ToF acquisition process. Furthermore, a detailed description of the ToF noise sources is given. Chapter 3 introduces the pinhole camera model and the two-view geometry, which are fundamental concepts for the description of the used camera rig. Moreover, stereo vision systems are presented, with a focus on the three main approaches for depth estimation. Chapter 4 provides a comparison between all the characteristics of ToF and stereo systems, in order to show their complementarity. Then in Chapter 5 a super-resolution algorithm based on *compressive sensing* theory is presented. Chapter 6 presents another approach for the depth super-resolution, which exploits the *joint bilateral filter*. Specifically, a color image is used to guide the up-sampling of a depth map. Moreover, it is proposed to use a modified joint bilateral filter which adapts its behavior according to the area characteristics (edge or flat). All the super-resolution algorithms have been tested on the Middlebury dataset [17], which provides stereo images together with the respective ground truth disparity images. In Chapter 7 a description of the camera calibration procedure and of the ToF noise measurement is given. Finally, Chapter 8 presents the real depth super-resolution algorithm, which combines the high-resolution color image from one video-camera and the low-resolution depth image from the ToF to obtain a high-resolution depth.

Chapter 2

Time-of-Flight (ToF) Cameras

Matricial *Time-of-Flight* (*ToF*) range cameras are relatively new active sensors which allow the acquisition of 3D point clouds at video frame rates. One of the advantages of these sensors with respect to laser scanners or stereo cameras is that they can acquire 3D images without any scanning mechanism and from just one point of view. Among all the ToF manufacturers, the more known ones are *PMDTec* [39], *Mesa Imaging* [29], *SoftKinetic* [48] and *Microsoft* [30]. Recently, new camera models were released, such as MESA SR4000 [29] and PMD CamCube 3.0 [39] (Figure 2.1), and researchers have started to work extensively with these recent models. Depth measurements are based on the well-known time-of-flight principle. Time-of-flight τ_d is the time that the light needs to cover the distance d from a light source to an object and from this object back to the camera. If the light source is assumed to be located near the camera, τ_d can be

Figure 2.1: Examples of ToF camera models. Left side: CamCube 3.0 from PMD Technologies GmbH, Germany. Right side: SR4000 from MESA Imaging AG, Switzerland.

computed as

$$\tau_d = \frac{2d}{c}, \tag{2.1}$$

where c is the light speed ($c = 3 \cdot 10^8 [m/s]$). According to the camera technology, the ToF method is suitable for ranges starting from some centimeters to several hundreds of meters with relative accuracies of 0.1%. This means that standard deviations in the millimeter range are realistically achievable at absolute distances of some meters, corresponding to a time-resolution of $6.6[ps]$ [3].

Two types of ToF cameras are available: pulse-based and phase-based, the latter better known as *Continuous Wave* (*CW*) ToF [47]. This Chapter describes the CW technology following the model presented by Dal Mutto in his Ph.D. thesis [33]. In Section 2.1 the working principle is presented. Section 2.2 describes the major components of a ToF camera. Then, Section 2.3 examines CW ToF technology, which is the approach that all the majors ToF manufacturers are currently using for their products. In Section 2.4 a detailed description of all the distance measurement errors is given. Finally, Section 2.5 concludes the Chapter with a short overview of the main specifications of the PMD[vision]® CamCube 3.0, which is the ToF camera used during the development of this thesis.

2.1 ToF cameras: working principle

As previously stated, two different variations of ToF cameras exist, which are pulse-based and CW. The simplest version are the pulse-based ToF cameras, which directly evaluate τ_d using discrete pulse of light emitted by a light source and backscattered by the object. In these devices each pixels has an independent clock, used to measure the time of travelled laser pulse [37]. Pulse-based ToF cameras can be implemented by arrays of *Single-Photon Avalanche Diodes* (*SPADs*) [1, 43] or an optical shutter technology [12]. The SPADs high sensitivity enables the sensor to detect low level of reflected light, therefore inexpensive laser sources with milliwatt power can be used for ranges up to several meters [37]. The advantage of using pulsed light is the possibility of transmitting a high amount of energy in a very short time. Thus, the influence of background illumination can be reduced. The common drawback of these systems is that they must be able to produce very short light pulses with fast rise and fall times, which are necessary to assure an accurate detection of the incoming light pulse.

Given that the time-of-flight is very short and the reflected signal very weak, direct measurement is difficult. Indirect methods that imply modulation and demodulation of light are used, like in the continuous wave ToF cameras, where the depth is determined by measuring the phase shift between the emitted and the received optical signal [25]. As shown in Figure 2.2, the emitted light is reflected by the objects in the scene and travel backs to the camera. Here the returning *Radio Frequency* (*RF*) modulated signal is demodulated by each pixel of the image sensor, producing a per-pixel range and intensity measurement. For a periodical modulation of frequency f_{mod} the phase shift $\Delta\phi$ corresponds to a temporal shift

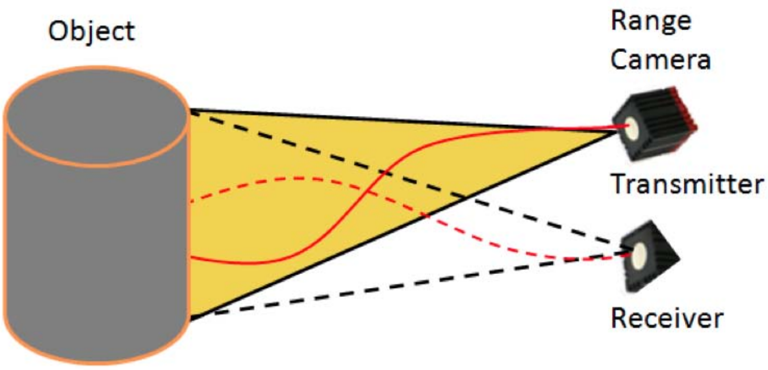


Figure 2.2: Principle of continuous wave ToF measurement.

$$\tau_d = \frac{\Delta\phi}{2\pi f_{mod}}. \tag{2.2}$$

From this, the distance is calculated by inverting (2.1)

$$d = \frac{c}{4\pi f_{mod}}\Delta\phi. \tag{2.3}$$

2.2 ToF cameras: components

So far, ToF cameras where considered as a single sensor composed by a single emitter and a co-positioned single receiver. This simplification was used to explain the working principle, but a real ToF camera is more complex. The camera structure can be described in three major components: the *image sensor*, the *illumination unit* and the *camera optics*. In the following Sections a description about these components and the data delivered by ToF cameras is reported.

2.2.1 Image sensor

The first component is the *image sensor*. This is conceptually an array of collectors matching the type of emitters of the illumination unit as considered so far. Currently, the integration in a single chip of $N_R \times N_C$ (where R means row, and C means column) emitters and the corresponding $N_R \times N_C$ receivers is not possible, especially for high values of N_R and N_C as needed to achieve a high image resolution. However, a single emitter can provide an irradiation that is reflected back by the target object and collected by many neighbors pixels. These receiver pixels are based on *Complementary Metal Oxide Semiconductor* active pixel sensor architecture fabricated with *Charge Coupled Device* technology (*CCD/CMOS*) [3, 24] and integrated in a $N_R \times N_C$ matrix (e.g., the sensor of the PMD CamCube 3.0 is made by 200 × 200 lock-in pixels). The so-called *Photonic Mixer Device* (*PMD*) sensor, also implemented on the PMD CamCube 3.0, consists of two quantum wells for every pixel, which store the electrons generated by the incident photons (Figure 2.3). Those photons generate electrons which are sorted by an electronic switch, implemented as a variable electrical field, into the one or the other quantum well. This switch is synchronized with the reference modulated signal, thus the number of accumulated electrons in each quantum well corresponds to one sample of the light signal [46].

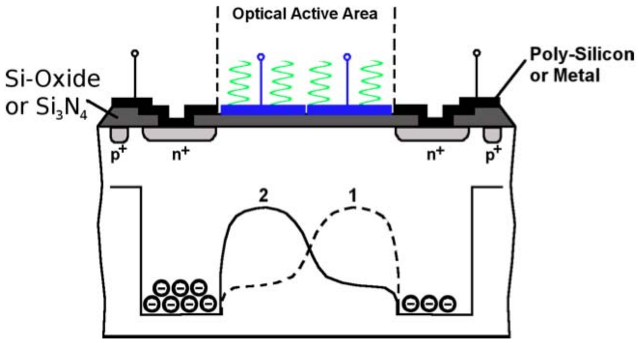


Figure 2.3: Schematic representation of the PMD two-tap ToF sensor. Incident photons generate electrons which are sorted by an electric field into two quantum wells. The switch is synchronized with the modulated light source, thus the number of electrons in each tap corresponds to a sample of the correlation function [46].

2.2.2 Illumination unit

The *illumination unit* is the component responsible of sending the modulated signal toward the scene. Different kind of emitters can form the illumination unit, but the most frequently employed IR emitters are the *Light Emitting Diods* (*LEDs*) arrays. The issue of the LEDs is that they cannot be integrated in the sensor. However, it is possible to simulate the presence of a single emitter co-positioned with the center of the receiver. Figure 2.4 shows this configuration for the case of the PMD CamCube 3.0. All the IR signals emitted by the LEDs can be considered as a unique spherical wave emitted by a single emitter, called *simulated emitter*. This approximation leads to some artifacts, one of which is a systematic distance measurement offset larger for closer than for further scene points [33].

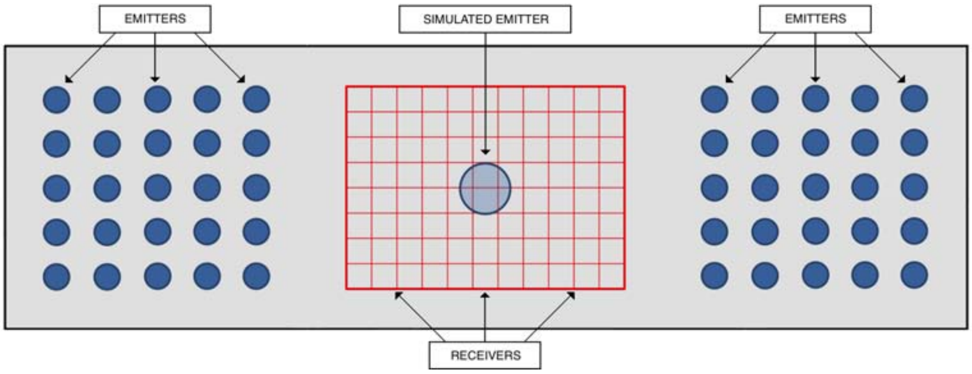


Figure 2.4: Scheme of a ToF sensor. The CCD/CMOS matrix of lock-in pixels is red. The emitters (blue) are distributed on the two sides of the the lock-in pixels and mimic a simulated emitter co-positioned with the center of the sensor (light-blue).

2.2.3 Camera optics

The third component of a ToF sensor is the *camera optics*, which consist of a lens and a filter. The lens collects a mix of reflected light and ambient lights and projects the image onto the image sensor. The narrow band pass filter only propagates the light with the same wavelength as the illumination unit. This is the optimal combination to minimize sunlight entering to the sensor (background light suppression) and to maximize transmission of light from the active light source.

2.2.4 Other components

Beyond the three fundamental components already mentioned, a ToF camera is composed of two more elements: the *driver electronics* and the *computation unit.* Both the illumination unit and the image sensor have to be controlled by high speed signals. These signals have to be very accurate to obtain a high-resolution. For example, if the signals between the illumination unit and the sensor shift by only $10[ps]$, the distance changes by $1.5[mm]$. This accuracy requirement is given by the driver electronics, designed to transmit the amplitude modulated signal output to the computation unit. Finally the computation unit uses the reference signal and the measurements from the image sensor to calculate the distance directly in the camera.

2.3 CW ToF cameras

ToF cameras can be described with a physical model, whose aim it is to give a realistic reproduction of the sensor data. In literature, complete and accurate models exist, such as the CW ToF camera model described in [46]. This model is focused on an precise reproduction of the camera noise. In this thesis the chosen model follows the version reported in [33], which describes the acquisition process and all the noise sources of CW ToF cameras.

Continuous wave ToF cameras emit an IR optical signal $s_E(t)$ of amplitude A_E modulated by a sinusoid of frequency f_{mod}

$$s_E = A_E \left[1 + \sin\left(2\pi f_{mod} t\right)\right]. \tag{2.4}$$

The signal $s_E(t)$ is reflected by the target scene surface and travels back to the camera sensor positioned near the emitter. The received signal $s_R(t)$, which is detected by the sensor, is offset-shifted in intensity with respect to the emitted signal mainly because of additional background light [34]. This signal results in a mean intensity of B_R. Furthermore, there is also a phase delay $\Delta\phi$ due to the energy absorption generally associated to the reflection, to the free-path propagation attenuation (proportional to the square of the distance) and to the time needed for the propagation of IR optical signals. The received signal is described by the following equation

$$s_R = A_R \left[1 + \sin\left(2\pi f_{mod} t + \Delta\phi\right)\right] + B_R, \tag{2.5}$$

where A_R is the amplitude attenuated by a factor k taking into account all optical losses in the transmission path. Figure 2.5 shows a scheme of the phase shift measurement principle. Even though they are both IR radiation amplitudes (measured in $[V]$), it

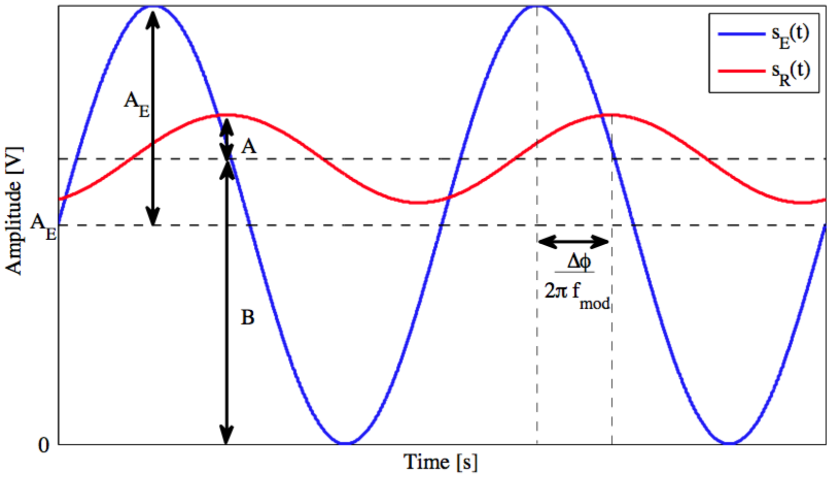


Figure 2.5: Scheme of the phase shift measurement principle (emitted signal $s_E(t)$ in blue and received signal $s_R(t)$ in red) [33].

is common to call the quantity A_R (denoted by A) *amplitude* and $A_R + B_R$ (denoted by B) *intensity* or *offset*; the latter is the average of the received signal, with the component B_R due to the background light and the component A_R due to the non-perfect demodulation [3]. With these notations, Equation (2.5) becomes

$$s_R = A \sin\left(2\pi f_{mod} t + \Delta\phi\right) + B. \tag{2.6}$$

The unknowns are the two amplitudes A, B (measured in volt) and the phase delay $\Delta\phi$ (pure number). The former amplitudes are important for SNR evaluation, whereas the latter phase delay is essential for the estimation of the depth. In fact the relation between $\Delta\phi$ and the time-of-flight τ_d is

$$\Delta\phi = 2\pi f_{mod}\tau_d = 2\pi f_{mod}\frac{2d}{c}. \tag{2.7}$$

As stated in Equation (2.3) the inversion of Equation (2.7) gives the distance.

The three unknowns are extracted using a suitable demodulation device, which can perform either a correlation or a sampling process. As described in [3], the received signal $s_R(t)$ is usually sampled at least 4 times per modulation period, with each sample shifted by 90°. Every sample represents the integration of the photo-generated charge carriers over a fraction of the modulation period: this technique is the so-called natural sampling process. Adding up the samples over several modulation periods increases the signal-to-noise ratio. For instance, using a modulation frequency of $20[MHz]$ and a summation over $10[ms]$ means that each sampling can be integrated over 200.000 modulation periods. From these four samples (s_R^0, s_R^1, s_R^2, s_R^3), the amplitude (A), the phase shift ($\Delta\phi$) and the intensity offset (B) can be calculated:

$$A = \frac{\sqrt{\left(s_R^0 - s_R^2\right)^2 + \left(s_R^1 - s_R^3\right)^2}}{2} \tag{2.8}$$

$$B = \frac{s_R^0 + s_R^1 + s_R^2 + s_R^3}{4} \tag{2.9}$$

$$\Delta\phi = \arctan\left(\frac{s_R^0 - s_R^2}{s_R^1 - s_R^3}\right). \tag{2.10}$$

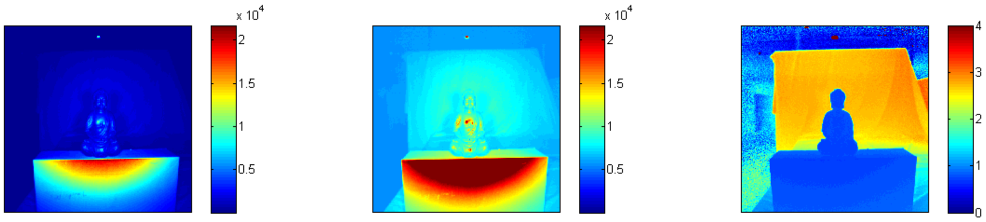
The amplitude is a measure of the achieved depth resolution and it is also used to generate a grayscale image of the observed scene. The offset describes the total intensity of the detected signal, also including the additional background light, and it may be used to generate another 2D grayscale intensity image. The final distance d is obtained from the phase information combining (2.3) and (2.10) as

$$d = \frac{c}{4\pi f_{mod}}\Delta\phi. \tag{2.11}$$

Sometimes, a sort of confidence map or flag matrix is also delivered, which contains information about the quality of the acquired data (i.e., saturated pixels, low signal amplitudes, invalid measurement, etc.). In order to give an idea of the data acquired with the PMD CamCube 3.0, an example is showed in Figure 2.6.

2.4 CW ToF cameras: typical distance measurement errors

This Section provides an overview about the typical distance measurement errors of CW ToF cameras as well as the currently available compensation methods. There are two type of errors: random errors (like *photon-shot noise, internal scattering, multi-path effect, motion blur, flying pixels*) and systematic errors (like *harmonic distortion, phase wrapping, amplitude-related errors*). Generally, the former can be managed by calibration and the latter by filtering.



(a) Amplitude image. (b) Intensity image. (c) Range image.

Figure 2.6: Visualization of data acquired with the PMD CamCube 3.0 camera: (a) amplitude image A, (b) intensity image B, (c) range image Z (scale in meters).

2.4.1 Harmonic distortion

Unfortunately, it is technically very difficult to create perfect sinusoids with the required frequency. In practice [4], most ToF cameras modulate the light with a square wave rather than a sinusoidal wave. Precisely, the sinusoids are obtained as low-pass filtered versions of squared wave-forms emitted by the LEDs. Since a harmonic sinusoidal illumination is the basic assumption for the measurement process, the estimated phase shift $\Delta\phi$ and consequently the corresponding distance d results distorted. This distortion leads to a systematic offset component which depends only on the measured depth for each pixel. A metrological characterization of this *harmonic distortion* can be found in [19] and [53]. As reported in [33] the harmonic distortion offset usually varies with a cyclic influence in the whole working range of the camera and it can assume values up to some tens of centimeters. This systematic offset, sometimes referred to as *wiggling* or *circular error*, reduces the accuracy of distance measurements, but it can be limited using a compensation via *look-up-table* (*LUT*) correction. A LUT has been proposed in [20], this method stores the depth errors depending on the measured depth distance using only one central pixel.

Furthermore, as shown in Figure 2.7, the sampling of the received signal is not ideal, but it is performed in finite time intervals Δt (for ideal sampling the system would need an infinite bandwidth). However, the natural sampling process has no influence on the phase as long as the integration time of each sampling point is shorter than the modulation period of the sampled signal. Only the measured amplitude is

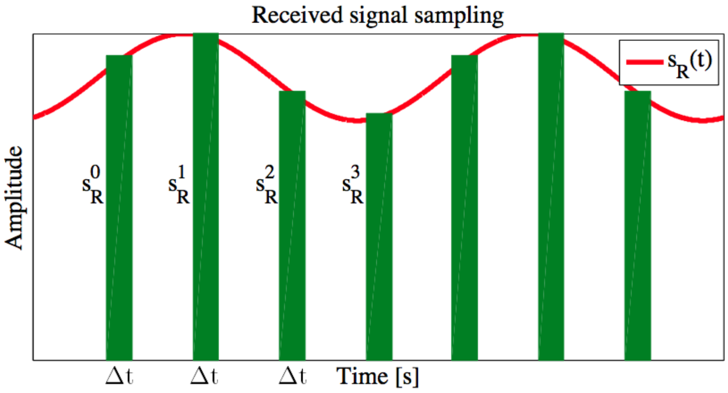


Figure 2.7: Pictorial illustration of non instantaneous sampling of the received signal $s_R(t)$ [33].

attenuated, but if the integration time is chosen as one half of the modulation period $\Delta t = T/2 = 1/2f_{mod}$ the measured amplitude is 64% of the real amplitude [24].

2.4.2 Phase wrapping

The second systematic distance measurement error is the *phase wrapping*. While with pulse-based ToF cameras no problem of measurement ambiguity occurs, with CW ToF the phase shift estimation can give some issues. In fact, the phase shift $\Delta\phi$ is obtained from the arctangent function in Equation (2.10), which is only unambiguously defined for values in the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$. However, since the physical delays in Equation (2.7) can only be positive, with the usage of $arctan2(\cdot, \cdot)$ it is possible to have a larger interval available for $\Delta\phi$, extending the codomain to $[0, 2\pi]$. Hence, from Equation (2.11) the non-ambiguity range for the estimated distance is $[0, \frac{c}{2f_{mod}}]$. For instance, with $f_{mod} = 20[MHz]$, the distance measurement range of the camera is $[0, 7500][mm]$. For this reason, objects located at a distance $d > d_{MAX} = \frac{c}{2f_{mod}}$ appear to be at the distance modulo(d/d_{MAX}), in fact the corresponding estimated phase shift $\Delta\phi$ is greater than 2π, so the distance of these objects is wrongly estimated. In practice, the measured distance corresponds to the remainder of the division between the actual $\Delta\phi$ and 2π, multiplied by $\frac{c}{2f_{mod}}$. This is the well-known phenomenon called *phase wrapping* since it may be regarded as a periodic wrapping around 2π of phase values $\Delta\phi$ [33]. Several methods have been proposed in order to resolve the phase wrapping problem and one of them uses multiple modulation frequencies to extend the non-ambiguity range.

With this approach two frequencies f_1 and f_2 are used either in two subsequent measurements or as a superposed signal in one measurement. The beat frequency, which is given by the difference of the two single modulation frequencies ($f_b = f_2 - f_1$), extends the maximum measurable distance to

$$d_{MAX} = \frac{c}{2\left(f_2 - f_1\right)}. \tag{2.12}$$

Figure 2.8 shows the non-ambiguity augmentation by using two signals with different modulation frequencies. The phase can be unambiguously reconstructed from the point where both phases are zero to the next point of the same condition [4].

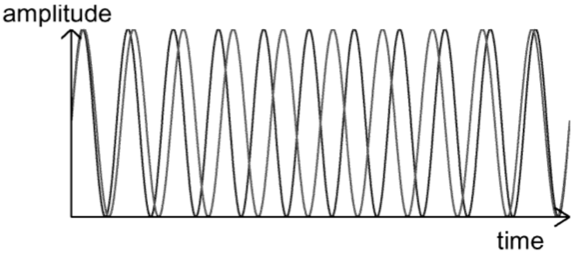


Figure 2.8: The measurement with two frequencies allows the unambiguous extraction of the phase within a significantly enhanced range compared to the single frequency distance measurement.

2.4.3 Photon-shot noise

According to [3], because of the dark electron current and the photon-generated electron current, the four acquired samples s_R^0, s_R^1, s_R^2, s_R^3 are affected by *photon-shot noise.* Whereby the dark current shot-noise component in the pixel can be reduced by lowering the sensor temperature or by improving the technology, there is no way to eliminate the photon-shot noise. Therefore, this noise is the most dominating noise source of CW ToF cameras. Photon-shot noise describes the statistical Poisson-distributed nature of the arrivals process of photons on the sensor and the generation process of electron hole-pairs, hence it is statistically characterized by a Poisson distribution. However, as reported in [32], the probability density function of the noise affecting the estimated distance d can be approximated by a Gaussian with standard deviation

$$\sigma_d = \frac{c}{4\pi f_{mod}\sqrt{2}} \frac{\sqrt{B}}{A}. \tag{2.13}$$

The precision (repeatability) of the measured distance is determined by the standard deviation in Equation (2.13). This standard deviation depends on three factors: the amplitude A, the offset B and the modulation frequency f_{mod}. Clearly, the higher the amplitude of the received signal A is, the better is the measurement precision (i.e., lower standard deviation of the measured distance). This means that the measurement precision decreases with the distance. Differently, if the intensity B increases, the measurement precision decreases. Since $B = A_R + B_R$, it may increase both with an increment of the background illumination B_R, which worsen the precision, or with an increment of the received signal amplitude A, which gives a slight precision improvement (considered the squared root dependence of B in Equation (2.13)). Last but not least, the modulation frequency f_{mod} is in inverse proportion with respect to the maximum measurable distance but, at the same time is in direct proportion with respect to the standard deviation of the distance measurements. In fact, if f_{mod} increases σ_d decreases, while the maximum measurable distance decreases (and vice-versa). For all these reasons, the modulation frequency is the first fundamental parameter of ToF cameras.

Usually, in the case of ToF cameras which allows to change the modulation frequency of the signal (e.g., PMD[vision]® CamCube 3.0 allows 4 choices of f_{mod} between 18 and $21[MHz]$), this parameter should be adjusted based on the desired maximum measurable distance and precision. Moreover, the usage of more than one modulation frequency allow the cooperation of multiple cameras.

2.4.4 Additional noise sources

Beyond *photon-shot noise, phase wrapping* and *harmonic distortion*, other noise sources such as *flicker noise*, *thermal noise* and *kTC noise* need to be considered. These noises are modeled as a constant noise floor N that is added to the demodulated signal independently from the exposure time [3]. For this reason the offset component B in Equation (2.13) can be written as

$$B = A_R + B_R + B_{dark} + N \tag{2.14}$$

where B_{dark} is the intensity of the dark electron current, which contributes, together with N, to a constant offset of the demodulated signal. In order to operate at the physical limitation given by the photon-shot noise, the main target of all the ToF cameras manufacturers is to reduce these additional noise sources using high quality components. Concerning the thermal noise, internal camera temperature affects the depth processing, hence some cameras include an integrated fan. The measured depth values suffer from a drift in the whole image until the temperature of the camera is stabilized [7]. The general approach to reduce temperature depth errors is to switch on the camera and wait until it takes a stable temperature before starting the acquisition. In [38] the authors suggest to wait up to 40 minutes to obtain distance measurement stability, both with the SR4000 and the CamCube 3.0. Especially in the case of the CamCube 3.0, for which warm-up distance measurement variations up to $0.12[m]$ have been founded (the test was performed on a white wall at a distance of $1[m]$ from the camera).

As stated in [33] it is possible to further reduce the additional noise effects by averaging distance measurements over several periods. However, the noise effects cannot be completely eliminated. The averaging interval length is called *integration time* (IT) and represents the length of time that the pixels are allowed to collect light. This is, together with the modulation frequency, the second fundamental measurement parameter of ToF cameras. In fact, increasing the integration time (maintaining all other factors constants, such as modulation frequency, distance of the object, room temperature, etc.) leads to better ToF distance measurement repeatability. Generally, a higher integration time is recommended to achieve a high distance measurement accuracy. On the other hand, long integration times introduce severe impact on the depth sensing quality, which are explained in the next Subsection.

2.4.5 Amplitude-related errors and motion blur

Amplitude-related errors occur due to low or overexposed reflected amplitudes. Increasing the integration time can lead to *saturation* of the pixels due to the increase in the amplitude of the measured signal. In fact, the longer is the integration time, the higher is the number of collected photons. When the received photons exceed the maximum quantity that the receiver can collect, saturation occurs. The saturation is often reached due to surfaces with high reflectivity to the camera signal or due to the presence of high levels of background light (i.e., sunlight). While in the case of high object reflectivity it could be sufficient to reduce the integration time, for background light recent ToF cameras, such as the PMD CamCube 3.0, have introduced the *Suppression of Background Illumination* (*SBI*) modality, allowing the usage of the devices also in outdoor applications. This technique allows precise 3D measurements even in environments with huge uncorrelated signals at high temperatures thanks to additional compensation sources. These sources inject additional charges in both quantum wells in order to instantaneously compensate for the saturation effects of uncorrelated signals during the integration process. Another solution is to automatically adjust the integration time, depending on the maximum amplitude present in the current scene, using a suitable software for data acquisition. This is, for instance, the case for the MESA SR4000 ToF camera. The opposite problem is the low amplitude error, which is mainly due to low illumination of the scene with objects at different distances and differences in object reflectivity's (non-specular materials for example retain energy and modify consequently the reflected light phase, depending on their refraction indices). Low amplitude errors can be avoided by filtering pixels with lower amplitude than a threshold [54], but this solution may discard a large region of the image [7].

The second phenomenon linked to the averaging over multiple periods is the *motion blur*. This error is due to the physical motion of the objects or the camera during the integration time used for sampling and leads to a blurring of the image across the direction of the motion. In fact, a computed value does not correspond to the state of the scene before nor after the event. Furthermore it is normally not between these values, but lays somewhere in the available range. Longer integration times usually allow higher accuracy of depth measurement so, for static objects, one may want to decrease the frame rate in order to obtain higher measurement accuracies from longer integration times. On the other hand, capturing a moving object at fixed frame rate leads to the motion blur problem. This imposes a limit on the integration time. Device manufacturers are trying to reduce the latency between the individual exposures for the phase samples, which is mainly caused by the data readout from the chip, but the problem still remains and might be solved by motion-compensated integration of the individual measurements [37].

2.4.6 Flying pixels

As explained in Section 2.2.1, since a sensor pixel is associated to a finite scene area (Figure 2.9) and not to a single scene point, it receives the radiation reflected from all the points of this area.

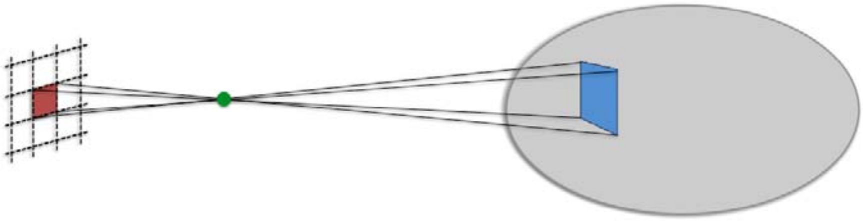


Figure 2.9: Finite size scene area (blue) associated to a ToF sensor pixel (red).

These approximations lead to the so-called problem of *flying pixels*, which are errant 3D data resulting from the way ToF cameras estimate the depth in edge areas. When the acquired area is a flat region, the association of a scene area to a single pixel does not introduces any artifact in the estimated depth. On the other hand, if the area crosses a depth discontinuity the estimated depth $Z(p_T)$ for the correspondent pixel p_T is a convex combination of the two different depth levels, where the combination coefficients depend on the percentage of the area at closer depth and the area at farther depth respectively reflected on p_T. As shown in Figure 2.10, flying pixels lead to severe depth estimation errors and they should be discarded from acquired data. However, flying pixels elimination is a difficult task. One possible solution in presented in [18, 27], where an heavy reduction of the flying pixels is achieved by exploiting the RGB acquired data with an external digital camera associated to the ToF camera.

2.4.7 Internal scattering

Internal scattering effects arise due to multiple light reflections between the camera lens and its sensor. This effect produces a depth underestimation over the affected pixels because of the energy gain produced by its neighboring pixels reflections [7]. The underestimation is particularly strong when the weak signal from far objects (background) is affected by the strong scattering signal from foreground objects. Moreover, the internal scattering is highly scene dependent. A schematic representation of this effect is reported in Figure 2.11. The signal emitted by the illumination unit is reflected

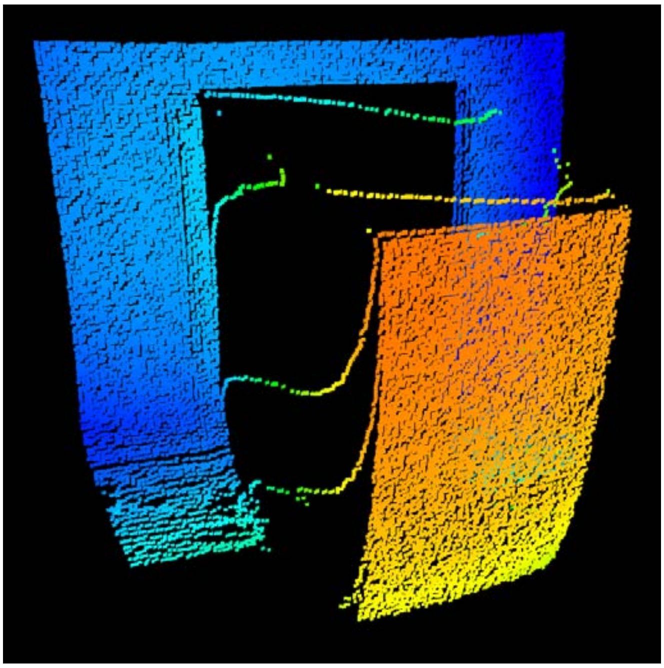


Figure 2.10: An example of flying pixels at the depth edge between object and background.

by the foreground object (red) and the background object (blue), which are positioned at distances r_1 and r_2 from the sensor. Due to the high amplitude of the signal reflected by the foreground object (S_1), multiple internal reflections can occur between the lens and the sensor (dotted red arrows in Figure 2.11). These reflections can superimpose the signal reflected by the background object (S_2), resulting in some changes in both amplitude and phase [37]. However, this problem is negligible in some recent camera models.

2.4.8 Multi-path propagation

One of the main errors sources of CW ToF cameras distance measurements is *multi-path propagation*. Essentially, this problem occurs due to the so-called *scattering*, schematically represented in Figure 2.12, where the orange optical ray incident to a non-specular surface is reflected in multiple directions (green and blue). In the ideal scenario, only the green ray of Figure 2.12 exists and it returns to the camera sensor, so that the total distance travelled by the light is twice the distance from the camera to the object. However, also the others (blue) rays should be taken into account. In fact, these

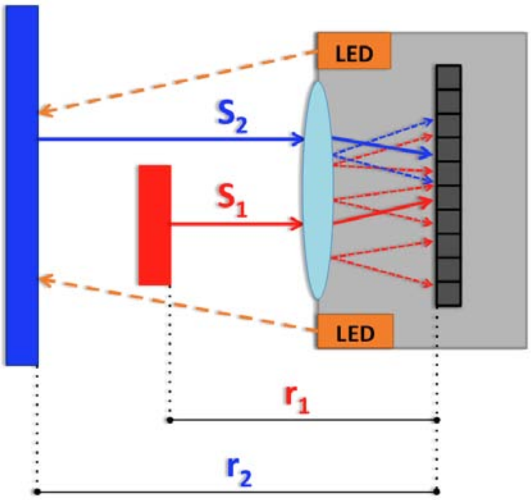


Figure 2.11: Schematic representation of the multiple internal reflections inside the camera.

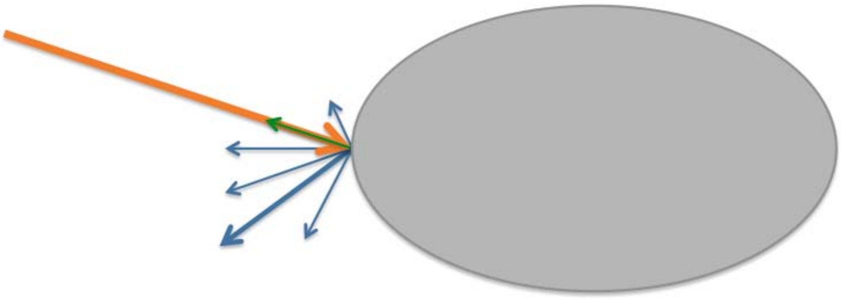


Figure 2.12: Scattering effect.

rays could first hit others scene objects and then travel back to the ToF sensor: in this case the light travels both by the direct and the indirect paths, affecting therefore the distance measurements of others scene points. Hence, the apparent distance is a weighted average of the path distances, where the weights depends on the strength of signal returned via each path. The final result is an over-estimation of the distance measurements. An example of multi-path phenomenon is showed in Figure 2.13. Since multi-path is a scene depend error, it is very hard to model. Hence, currently there is no method for its compensation [33].

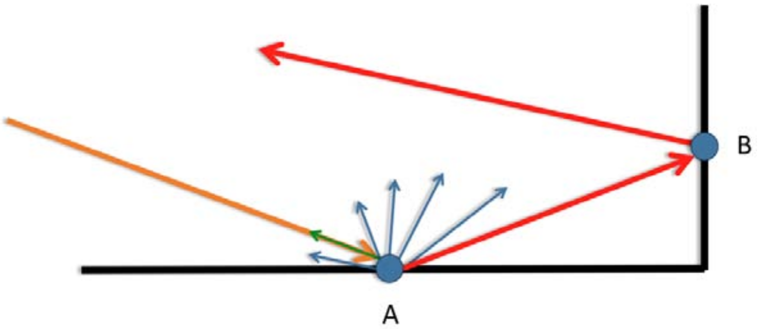


Figure 2.13: Multi-path effect: the emitted ray (orange) hit the surface at point A and is reflected in multiple directions (blue and red rays). The red ray reaches then B and travels back to the ToF camera affecting the distance measured at the sensor pixel relative to B.

2.5 ToF camera PMD CamCube 3.0

The PMD[vision]® CamCube 3.0 [39] is a phase shift based ToF camera composed by two LEDs illumination units and a central camera unit with a sensor resolution of 200 × 200 pixels at frame rate 40 $[fps]$. The “crop utility” delivered by PMD allows cropping of pixel columns and rows, therefore it is possible to get frame rate up to 80 $[fps]$ at 160×120 pixels resolution. The CamCube 3.0 provides three types of images all captured at once, namely intensity image, amplitude image and range image. The reason for the low image resolution are the multiple detection units resulting into larger pixels. The use of larger pixels is to enable the collection of a higher amount of incoming light which increases the depth precision [46]. The camera is more robust to sunlight than other ToF cameras based on phase shift measurements thanks to its PMD 41k-S2 sensor with integrated SBI technology. The declared measurement repeatability is

$0.003\,[m]$, while the working range with standard settings is $[0.3, 7.0]\,[m]$. Finally, the *field of view* (FoV) of the camera is increased up to $60° \times 60°$ by using a different optics.

In this thesis the camera is always used with a $f_{mod} = 20[MHz]$ which is a very common modulation frequency that leads to a non-ambiguity range of $7.5[m]$. There are two reasons for choosing this frequency. Firstly, no high-power IR-LEDs are available with higher modulation frequency at a low price tag. Secondly, the non-ambiguity range suits most of the indoor and outdoor applications. However, the suggested measurement range for this camera is $[0.3, 7][m]$. Eventually, the integration time is set to $1[ms]$.

Chapter 3

Stereo Systems

A stereo vision system uses two standard cameras in order to simulate the human binocular vision and estimate the depth distribution of an acquired scene. The basis of this system is the passive triangulation principle: when a 3D object is viewed from two coplanar cameras, the image as observed from one position is shifted laterally when viewed from the other one. The offset between the points in the two images, known as disparity, is inversely proportional to the distance between the object and the cameras. This distance can be calculated from the estimated disparity when the intrinsic and extrinsic parameters of the cameras are known. The hardware implementation of this system is made by a pair of identical video-cameras (or a single camera with the ability to move laterally) and an optional synchronization unit, used in case of dynamic scenes. The final estimated depth is always relative to the point of view of one of the two cameras, usually called reference camera, while the other one is called target camera [33].

In this Chapter a detailed description of the camera model used for the modeling of stereo vision systems is given in Sections 3.1 and 3.2. Then, Section 3.3 presents the image rectification procedure and the definition of stereo disparity. Finally, in Section 3.4 the three main categories of stereo correspondence algorithms are described.

3.1 Pinhole camera model

The pinhole camera model defines the geometric relationship between a 3D point and its 2D corresponding projection onto the camera. This model is described by its *optical* *center* **C** (also known as *camera projection center*) and the *image plane* $\mathcal{I}$ (also known as *focal plane*). Let the optical center be the origin of an Euclidean coordinate system, and let the upper left corner of $\mathcal{I}$ be the origin of the *image coordinate system*. The image plane is located at a distance $z = f$ from **C**, where f is the camera focal length. With the pinhole camera model it is possible to map a point in world coordinates $\mathbf{W} = (x, y, z)^\top$ onto a point in the image plane $\mathcal{I}$. This projected point $\mathbf{w} = (u, v)^\top$ lies where the line joining **W** and the camera center **C** meets the image plane. Using these prerequisites along with similar triangles, it is possible to deduce the so-called perspective projection mapping from world to image coordinates, showed in Figure 3.1 and described by

$$(x, y, z)^\top \mapsto \left(f\frac{x}{z}, f\frac{y}{z}\right)^\top . \tag{3.1}$$

This is a mapping from an Euclidian 3D-space $\mathbb{R}^3$ to an Euclidean 2D-space $\mathbb{R}^2$. The line from the camera center perpendicular to the image plane is called the *principal axis* or *optical axis* of the camera, whereas the intersection of the optical axis with the image plane $\mathcal{I}$ is the so-called *principal point* **p**. Eventually, the origin of the *camera coordinate system* (*CCS*) with the principal point coordinates (u_c, v_c) is located at the center of the image plane $\mathcal{I}$. If the world and the image points are represented

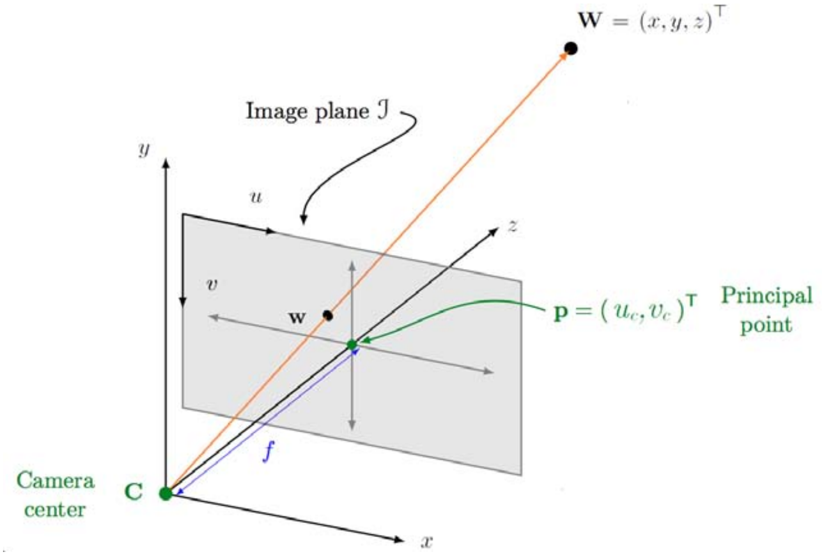

$\mathbf{W} = (x, y, z)^\top$
Image plane $\mathcal{I}$
y
u
z
v
$\mathbf{w}$
$\mathbf{p} = (u_c, v_c)^\top$
Principal point
Camera center
C
f
x

Figure 3.1: Pinhole camera model. Projection of the point **W** on the image plane by drawing the line through the camera center **C** and the point to be projected. Note that the image plane is placed in front of the camera center for illustration purposes only [49].

by homogeneous vectors, then the perspective projection can be expressed in terms of matrices multiplication as

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fx \\ fy \\ 1 \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \tag{3.2}$$

where $(x, y, z, 1)^\top$ are the homogeneous coordinates of the 3D point, and $(u, v, 1)^\top$ are the homogeneous pixel coordinates of the image point. The matrix describing the mapping is called the *camera projection matrix* $\mathtt{P}$. It is a 3×4 full-rank matrix with 3 degrees of freedom [13], and it can be written as

$$\mathtt{P} = \operatorname{diag}(f, f, 1)\,[\mathtt{I}|\mathbf{0}] \tag{3.3}$$

where $\operatorname{diag}(f, f, 1)$ is a diagonal matrix, and $[\mathtt{I}|\mathbf{0}]$ is the identity matrix $\mathtt{I}$ concatenated with a null vector $\mathbf{0}$. With this notation, Equation (3.2) becomes

$$\mathbf{w} = \mathtt{P}\mathbf{W}. \tag{3.4}$$

The projection matrix $\mathtt{P}$ represents the simplest possible case, as it only contains information about the focal distance f.

3.1.1 Intrinsic parameters

Equation (3.1) assumes that the principal point $\mathbf{p}$ corresponds to the origin of the *image coordinate system* in the image plane $\mathcal{I}$. In practice, this is not always true, hence an offset vector $(u_c, v_c)^\top$ corresponding the principal point coordinates has to be considered. Moreover, in the case of CCD cameras there is the possibility of having non-square pixels. Therefore, the camera's focal length in terms of pixel dimensions in u and v directions has to be considered. For these reasons, the general formulation of the central projection mapping is

$$(x, y, z)^\top \mapsto \left(k_u f \frac{x}{z} + u_c, k_v f \frac{y}{z} + v_c\right)^\top \tag{3.5}$$

where k_u and k_v are the effective number of pixels per millimeter along the u and v-axes of the image plane $\mathcal{I}$, and (u_c, v_c) are the coordinates of the principal point. Now, the camera projection matrix becomes

$$\mathtt{P} = \begin{bmatrix} fk_u & & u_c & 0 \\ & fk_v & v_c & 0 \\ & & 1 & 0 \end{bmatrix}. \tag{3.6}$$

From this formulation, it is possible to decompose $\mathtt{P}$, using the QR factorization, in the product of two matrices

$$\mathtt{P} = \mathtt{K}\left[\mathtt{I}|\mathbf{0}\right], \tag{3.7}$$

where

$$\mathtt{K} = \begin{bmatrix} \alpha_u & & u_c \\ & \alpha_v & v_c \\ & & 1 \end{bmatrix} \tag{3.8}$$

is the *camera calibration matrix* which encodes the transformation in the image plane from the normalized camera coordinates to pixel coordinates. This matrix depends from the so-called *intrinsic parameters*, which are the focal lengths $\alpha_u = fk_u$, $\alpha_v = fk_v$ in horizontal and vertical pixels, respectively (f is the focal length in millimeters, k_u and k_v are the number of pixels per unit distance in image coordinates), and the principal point coordinates (u_c, v_c) in pixel. Moreover, there exists a further parameter, the *skew coefficient* γ between the u and v-axes, but for high-quality build cameras it can be approximated to zero.

3.1.2 Extrinsic parameters

In general, the camera coordinate system and the world coordinate system are different Euclidean coordinates systems. These two systems are related via a 3 × 3 *rotation matrix* $\mathtt{R}$ and a *translation vector* $\mathbf{t}$, which are the so-called *extrinsic parameters*. If $\mathbf{W}$ represents a point in the world coordinate system, and $\mathbf{W_c}$ is the same point in the camera coordinate system then the relation between these two points in homogeneous coordinates is

$$\mathbf{W_c} = \begin{bmatrix} \mathtt{R} & -\mathtt{R}\mathbf{C} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{bmatrix} \mathtt{R} & -\mathtt{R}\mathbf{C} \\ 0 & 1 \end{bmatrix} \mathbf{W}, \tag{3.9}$$

where $\mathbf{C}$ are the coordinates of the camera center in the world coordinate system, and $\mathtt{R}$ is the rotation matrix that describes the position and orientation of the camera with respect to the world coordinate system. With the notation $\mathbf{t} = -\mathtt{R}\mathbf{C}$, the perspective projection mapping of Equation (3.4) becomes

$$\mathbf{w} = \mathtt{K}\left[\mathtt{R}|\mathbf{t}\right]\mathbf{W} = \mathtt{P}\mathbf{W}. \tag{3.10}$$

This is the general mapping from a point $\mathbf{W} = (x, y, z)^\top$ in world coordinate system to a point $\mathbf{w} = (u, v)^\top$ in the camera image plane. The new resulting camera projection matrix $\mathtt{P} = \mathtt{K}\,[\mathtt{R}|\mathbf{t}]$ has 11 degrees of freedom: 5 for the intrinsic parameters in $\mathtt{K}$ (the elements f, k_u, k_v, u_c, v_c), and 6 for the extrinsic parameters $\mathtt{R}$ and $\mathbf{t}$ [13].

3.1.3 Lens distortion

So far, the introduced camera model assumes that the acquisition process of the camera can be described as a linear model. Therefore, the optical center, the world point and the image point are collinear. This means that world lines are imaged as lines, but for real (non-pinhole camera) lenses this assumption will not hold. In fact, a realistic camera model must take into account some nonlinear distortions introduced by the lenses (e.g., an imperfect centering of the lens components and other manufacturing defects). The most significant distortion is generally a radial distortion, but there exists also a tangential distortion. The effect is that incoming light rays are bended more or less depending on the distance to the principal point $\mathbf{p}$. The distortion becomes more significant as the focal length, the FoV (and the price) of the lens decreases [13]. The target of lens undistortion is to remove this error in order to obtain a camera that works as a linear device. To do this, the pinhole camera model described in Section 3.1 is usually improved with an additional lens distortion model. The radial distortion is modeled as a nonlinear transformation from the ideal (undistorted) pixel coordinates (u, v) to the observed (distorted) pixel coordinates $(\hat{u}, \hat{v})$. In the implementation of [2], the latter distorted pixel coordinates are

$$\hat{\mathbf{w}}_{\mathbf{d}} = \underbrace{\left(1 + \rho_1 r^2 + \rho_2 r^4 + \rho_3 r^6\right)}_{\text{radial distortion}} \mathbf{w} + \underbrace{\begin{bmatrix} 2\varrho_1 u_c v_c + \varrho_2 \left(r^2 + 2\left(u_c\right)^2\right) \\ \varrho_1 \left(r^2 + 2\left(v_c\right)^2\right) + 2\varrho_2 u_c v_c \end{bmatrix}}_{\text{tangential distortion}}, \tag{3.11}$$

where ρ_1, ρ_1 and ρ_3 are the radial distortion coefficients, ϱ_1 and ϱ_2 are the tangential distortion coefficients, (u_c, v_c) are the coordinates of the image center, and $r = \sqrt{(u_c)^2 + (v_c)^2}$.

3.2 Two-view geometry

The two-view geometry is essentially the intrinsic geometry of two different perspective views of the same scene. The two perspective views may be acquired simultaneously, for example in a stereo rig, or sequentially, for example by moving a single camera.

From the geometric viewpoint the two situations are equivalent, but in the second case the scene might change between successive acquisitions. Figure 3.2 shows a two-view geometry, where each camera has his own camera projection matrix $\mathtt{P}_1$ and $\mathtt{P}_2$. Most of the 3D scene points must be simultaneously visible in both the views. This is not true in case of occlusions (i.e., points visible only in one camera). Each unoccluded 3D scene point $\mathbf{W} = (x, y, z)^\top$ is projected to the image plane $\mathcal{I}_1$ in the point $\mathbf{w_1} = \mathtt{P}_1\mathbf{W}$, and to the image plane $\mathcal{I}_2$ in the point $\mathbf{w_2} = \mathtt{P}_2\mathbf{W}$. These two points are called *corresponding points* (or conjugate points).

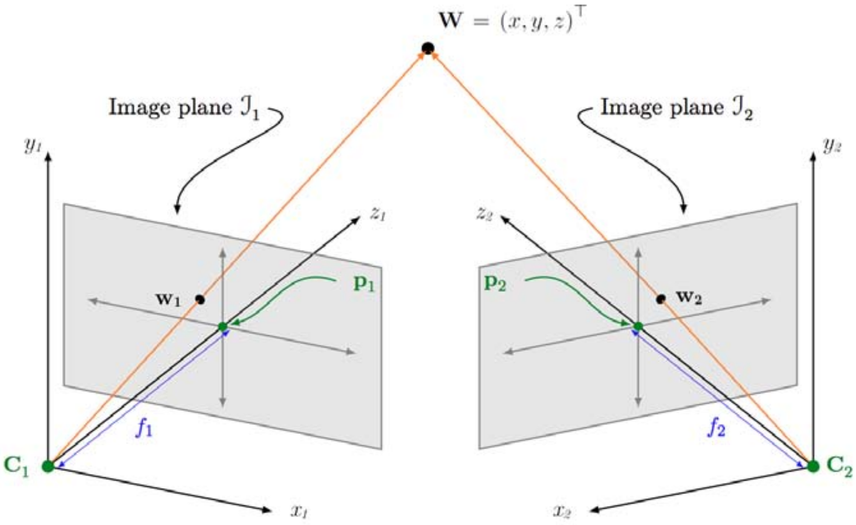


Figure 3.2: Two-view geometry for a stereo camera rig. The points $\mathbf{w_1}$ and $\mathbf{w_2}$ are corresponding points, as they are the projection of the same 3D point $\mathbf{W}$ [49].

3.2.1 Epipolar geometry

The *epipolar geometry* describes the geometric relationship between two perspective views of the same 3D scene. The key point is that corresponding image points must lie on particular image lines. Therefore, a point $\mathbf{w}$ in one image defines an epipolar line ℓ. With the epipolar geometry, instead of searching the corresponding point in the whole 2D region of the other image, it is possible to reduce the search range to only the epipolar line ℓ. This is the reason why the epipolar geometry is widely used for the search of corresponding points in stereo matching. Figure 3.3 illustrates the epipolar constraint for the mapping $\mathbf{w_1} \mapsto \ell_2$ of image point $\mathbf{w_1}$ in the first view to the epipolar line ℓ_2 in the second view. The 3D point $\mathbf{W}$, the two corresponding image points $\mathbf{w_1}$ and $\mathbf{w_2}$, and the camera projection centers $\mathbf{C_1}$ and $\mathbf{C_2}$ are coplanar and define a plane π called *epipolar plane*. The conjugate epipolar lines ℓ_1 and ℓ_2 are the intersections of the epipolar plane π with the image planes. The line connecting the camera centers $\mathbf{C_1}$ and $\mathbf{C_2}$ is called the *baseline*. The intersection of the baseline with the image planes $\mathcal{I}_1$ and $\mathcal{I}_2$ defines the two *epipoles* $\mathbf{e_1}$ and $\mathbf{e_2}$. For each view, all the epipolar lines ℓ intersect at the corresponding epipole $\mathbf{e}$. As the position of the 3D point $\mathbf{W}$ varies, the epipolar planes "rotate" about the baseline, therefore any plane containing the baseline is an epipolar plane [13] (Figure 3.4).

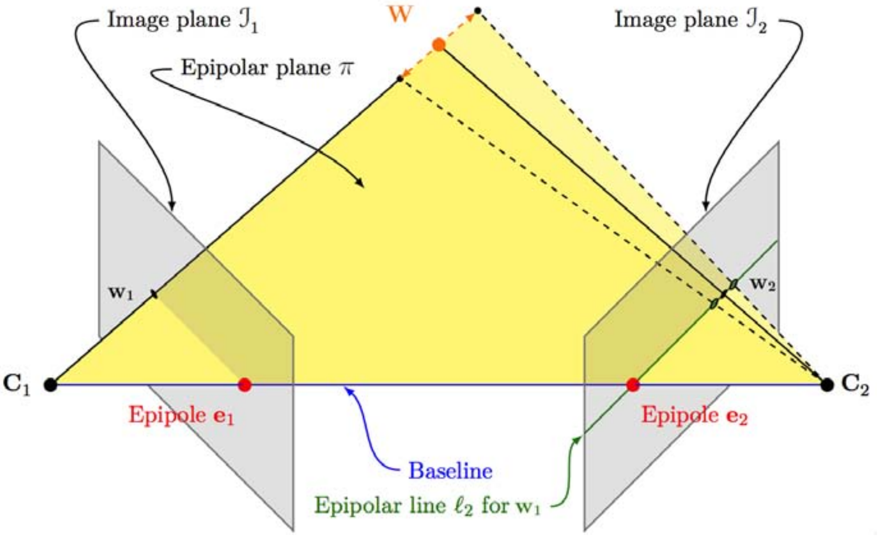


Figure 3.3: Epipolar constraint for the mapping $\mathbf{w_1} \mapsto \ell_2$. The camera centers, 3D point $\mathbf{W}$, and its images $\mathbf{w_1}$ and $\mathbf{w_2}$ lie in the common epipolar plane π [49].

3.2.2 Fundamental matrix

The *fundamental matrix* is the algebraic representation of epipolar geometry. Given two corresponding points, $\mathbf{w_1}$ and $\mathbf{w_2}$, the fundamental matrix $\mathtt{F}$ describes the relation between one point and the epipolar line on which his correspondent point on the other image must lie. This means that, for all pairs of corresponding points $\mathbf{w_1} \leftrightarrow \mathbf{w_2}$, the fundamental matrix $\mathtt{F}$ is a 3×3 matrix that satisfies the epipolar constraint

$$\mathbf{w_1}^\top \mathbf{F} \mathbf{w_2} = 0. \tag{3.12}$$

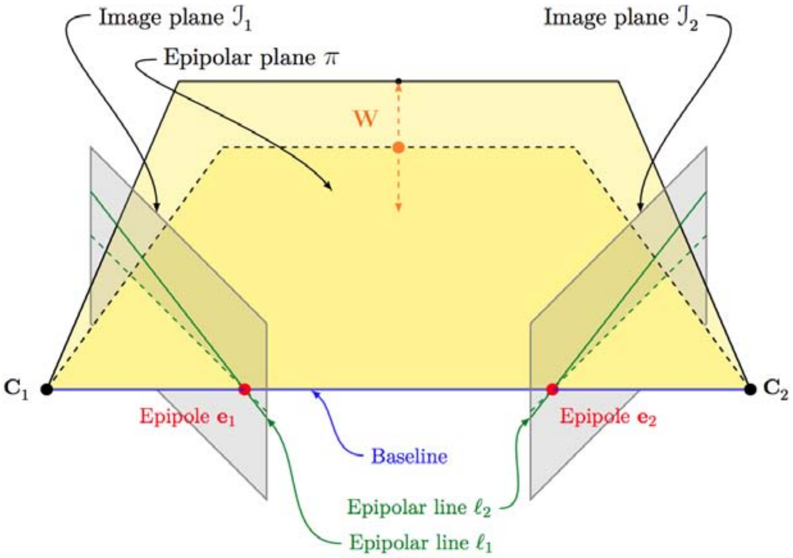


Figure 3.4: Epipolar geometry where any plane containing the baseline is an epipolar plane π. The epipolar plane π intersects the image planes $\mathcal{I}_1$ and $\mathcal{I}_2$ at the corresponding epipolar lines ℓ_1 and ℓ_2. Each epipolar plane for any point correspondence defines the respective epipolar lines [49].

For any point $\mathbf{w_1}$ in the first image, the corresponding epipolar line is

$$\ell_2 = \mathtt{F}\mathbf{w_1}. \tag{3.13}$$

In the same way, for the image point $\mathbf{w_2}$ in the second image, the corresponding epipolar line is defined by

$$\ell_1 = \mathtt{F}\mathbf{w_2}. \tag{3.14}$$

Geometrically, $\mathtt{F}$ represents the mapping from the 2D projective plane of the first image to the pencil of epipolar lines through the epipole $\mathbf{e_2}$. Therefore, it must have rank 2 [13]. Moreover, $\mathtt{F}$ has 7 degrees of freedom hence it can be estimated given at least 7 points correspondences. If the fundamental matrix $\mathtt{F}$ represents the pair of camera projection matrices $\{\mathtt{P}_1, \mathtt{P}_2\}$, then the transposed fundamental matrix $\mathtt{F}^\top$ represents the camera projection matrices in the opposite order $\{\mathtt{P}_2, \mathtt{P}_1\}$. Another important property is that the two epipoles $\mathbf{e_1}$ and $\mathbf{e_2}$ are, respectively, the right null-vector and the left null-vector of $\mathtt{F}$. Hence, they satisfy the relations

$$\mathtt{F}\mathbf{e_1} = 0, \tag{3.15}$$

$$\mathtt{F}^\top\mathbf{e_2} = 0. \tag{3.16}$$

3.2.3 Essential matrix

The *essential matrix* is the specialization of the fundamental matrix to the case of normalized image coordinates for the corresponding points $\hat{\mathbf{w}}_{\mathbf{1}} \leftrightarrow \hat{\mathbf{w}}_{\mathbf{2}}$, where $\hat{\mathbf{w}} = \mathtt{K}^{-1}\mathbf{w}$. This matrix represents the epipolar geometry considering calibrated cameras

$$\mathtt{K}^{-1}\mathtt{P} = [\mathtt{R}|\mathbf{t}]\,, \tag{3.17}$$

where the camera matrix $\mathtt{K}^{-1}\mathtt{P}$ is called a *normalized camera matrix*. The relationship between the two corresponding image points in normalized coordinates $\hat{\mathbf{w}}_{\mathbf{1}} \leftrightarrow \hat{\mathbf{w}}_{\mathbf{2}}$ is expressed by the defining equation for the essential matrix

$$\hat{\mathbf{w}}_{\mathbf{2}}^{\top}\mathtt{E}\hat{\mathbf{w}}_{\mathbf{1}} = 0. \tag{3.18}$$

From this equation it is possible to obtain the relationship between the fundamental and the essential matrices

$$\mathtt{E} = \mathtt{K}_2{}^{\top}\mathtt{F}\mathtt{K}_1, \tag{3.19}$$

where $\mathtt{K}_1$ and $\mathtt{K}_2$ are the camera calibration matrices of the two cameras. Summing up, the essential matrix is a 3 × 3 matrix with only 5 degrees of freedom. This because of the constraint that two of its singular values are equal and the third singular value is zero [49]. For the rest, it shares the basic properties with the fundamental matrix $\mathtt{F}$.

3.3 Image rectification

As explained in the previous Sections, for normal video-cameras the mapping between a scene point $\mathbf{W}$ and his projection $\mathbf{w}$ on the two image planes is determined by the cameras' *intrinsic* and the *extrinsic parameters*. All the parameters of such model can be estimated with a camera calibration procedure (described in Section 7.1.1). After the calibration, the two images are usually rectified, in order to simplify the stereo matching algorithm. Image rectification transforms each image plane such that pairs of conjugate epipolar lines become collinear and parallel to one of the image axes (usually the horizontal one). The rectified images can be thought of as acquired by a new stereo rig, obtained by rotating the original cameras in order to generate two coplanar image planes that are in addition parallel to the baseline. With this expedient, the search domain of corresponding pixels is restricted from horizontal and vertical lines of the rectified images to only the horizontal lines (Figure 3.5). Image rectification also allows to correct the projective distortion introduced by the camera lenses and compensate the focal length differences between left and right camera; for more details see [9, 31, 50]. After the rectification both left image (reference image I_1) and right image (target image I_2) are referred to the 2D CCS of the reference image, with horizontal axis u and vertical axis v. Hence a scene point $\mathbf{W} = (x, y, z)^\top$ expressed with respect to the left 3D CCS, if visible from the two cameras, is projected to point $\mathbf{w_1}$ with coordinates $(u_1, v_1)^\top$ onto reference image plane and to point $\mathbf{w_2}$ with coordinates $(u_2, v_2)^\top = (u_1 - d, v_1)^\top$ onto target image plane. With this notation, the horizontal distance d between the coordinates of $\mathbf{w_1}$ and $\mathbf{w_2}$ is called *disparity* and is expressed in *pixel*. Since the depth information is inversely proportional the the corresponding disparity, the depth value z of $\mathbf{W}$ can be computed as

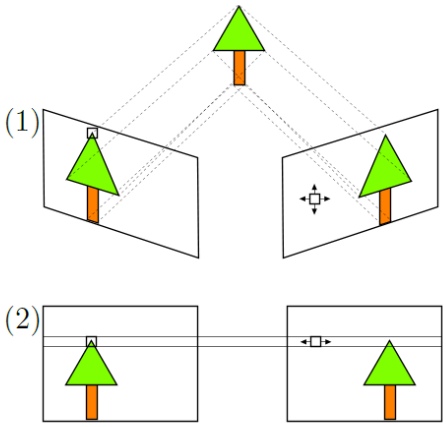


Figure 3.5: The search space before (1) and after (2) rectification [55].

$$z = \frac{bf}{d}, \tag{3.20}$$

where b is the baseline in *meters* (i.e., the distance between the two camera nodal points) and f is the focal length in *pixel*, which is assumed equal in both directions (u, v) of the image plane and in both the cameras. High values of disparity d correspond to low depth z (i.e., points close to the camera), whereas low values of d correspond to high z. Usually the distance $z = \infty$ is associated with disparity $d = 0$. Moreover it is customary to limit the range of values which d may take using the minimum and maximum depth vales (z_{MIN} and z_{MAX}) of the scene, if they are known. With this convention d can take values between $d_{MIN} = bf/z_{MAX}$ and $d_{MAX} = bf/z_{MIN}$.

The accuracy of stereo vision systems decreases quadratically with respect to z, this can easily proved by deriving Equation (3.20) with respect to d

$$\Delta z \cong -\frac{bf}{d^2}\Delta d, \tag{3.21}$$

where Δd is the estimation precision of d. Finally, the consequent depth estimation error Δz is obtained by combining 3.21 and 3.20

$$\Delta z \cong -\frac{z^2}{bf}\Delta d. \tag{3.22}$$

Summarizing, neglecting the sign, the depth accuracy Δz of a traditional stereo system decreases quadratically with the distance, which means that the accuracy in the near range far exceeds that of the far range.

The major problem to solve is the computation of the disparity map, this means find a set of points in the reference image which can be identified as the same points in the target image. For achieving this, points or features in I_1 are matched with the corresponding points or features in I_2. This is the so-called correspondence problem, which is treated in the next Section.

3.4 Stereo correspondence algorithms

Stereo correspondence has traditionally been one of the most heavily investigated topics in computer vision. The goal of these algorithms is to provide, starting from two images I_1 and I_2 taken by a stereo system, a *disparity image* D_S. Since the focal length f and the baseline b are known, this disparity image can be used to compute the depth of the acquired scene with Equation (3.20). In order to detect conjugate points (w_1, w_2), for every pixel $w_1 \in I_1$ a linear search along each horizontal line of I_2 is performed. The reliability of point matching is fundamental, because mismatching correspondences lead to wrong depth estimation and gaps in the reconstruction. In literature many stereo correspondence algorithms have been proposed, and they can be divided into three main categories, described in the following Subsections.

3.4.1 Local methods

With local methods the disparity at a given point w_1 of the reference image is computed by exploiting the local similarity of the intensity values, within a finite window, in the correspondent row of the target image. One of the most used algorithm is the so-called *Fixed Window* (*FW*) stereo algorithm. With this technique, for each pixel $\mathbf{w_1} = (u, v)^\top$ on the reference image its conjugate $\mathbf{w_2} = (u - d^*, v)^\top$ is searched using a window W_1 of size $(2n + 1)(2m + 1)$ pixel centered around w_1. This window is compared with other windows of the same size W_2^i centered around each possible candidate $w_2^i (u - i, v)$ with $i = 1, ..., d_{MAX} - d_{MIN}$ in the target image, as shown in Figure 3.6. The computed disparity is the shift associated at the maximum *similarity* between the values of each pair of windows W_1 and W_2^i. In order to evaluate this similarity many *cost functions* can be used, one of the most employed is the *Sum of Squared Differences* (*SSD*):

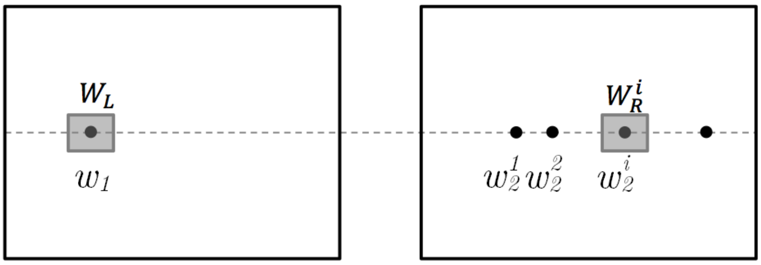


Figure 3.6: Fixed Window stereo algorithm [33].

$$SSD(u, v, d) = \sum_{(k,l)} (I_1 (u + k, v + l) - I_2 (u + k + d, v + l))^2, \tag{3.23}$$

where $k \in (-n, n)$, $l \in (-m, m)$ and $I(u, v)$ is the gray-level of pixel $k \in (u, v)$. Clearly many other different measures can be used, common window-based matching costs include *Sum of Absolute Differences* (*SAD*), *Normalized Cross Correlation* (*NCC*) or the *census transform*. Small values of SSD indicate that two image portions are similar. For each pixel (u, v) the algorithm choses the disparity associated with the minimum cost value by using

$$d^* (u, v) = \arg \min_d SSD(u, v, d). \tag{3.24}$$

Thus, the computed disparity has a precision of one pixel, however it is possible to obtain a sub-pixel precision by interpolating the cost function SSD in proximity of the minimum. Such a local method considers a single pixel of I_1 at the time, it performs a *Winner-Takes-All* (*WTA*) strategy for disparity optimization and it does not explicitly impose any model on the depth distributions. As most local algorithm, this approach performs poorly in textureless regions. Another issue is that uniqueness of matches is only enforced for one image (the reference image), while points in the other image might get matched to multiple points. However, the main limitation of FW algorithms lies in selecting an appropriate windows size, which must be large enough to include sufficient intensity variation for matching, but should be small enough to avoid the effect of projective distortion. If the window is too small and does not cover enough intensity variation, it gives a poor disparity estimation, because the ratio between intensity variation and noise is low. On the other hand, if the window is too large and include a scene depth discontinuity, then the position of minimum SSD may not represent the correct matching due to different projective distortions in the left and right images. For all these reasons, modifications of FW have been proposed, like using multiple coupling windows for a single pair of candidate conjugate points [8] or select the windows size adaptively depending on local variations of intensity and disparity [21]. These variations of the classical FW approach lead to performance improvements, especially in presence of depth discontinuity, but they cause a significant increase in terms of computation time.

3.4.2 Global methods

In contrast to local methods, algorithms based on global correspondences overcome some of the aforementioned problems. They compute the whole disparity image D_S at once by imposing smoothness constraints on the scene depth in the form of regularized energy functions. The result is an optimization problem which typically has a greater computational cost as compared to local methods. The target of global algorithms is to find a disparity image D_S that minimizes the global energy cost function

$$\hat{I}_D = \arg\min_{D} E\left(D_S\right) = \left(E_{data}\left(I_1, I_2, D_S\right) + E_{smooth}\left(D_S\right)\right). \qquad (3.25)$$

The quantity $E_{data}\left(I_1, I_2, D_S\right)$, called *data term*, measures how well the disparity image D_S agrees with the input image pair (it is a cost function similar to the one of local algorithms). The *smoothness term* $E_{smooth}\left(D_S\right)$ encodes the smoothness assumptions made by the algorithm. To make the optimization computationally tractable, the smoothness term is often restricted to only measure the differences between neighboring pixels disparities. Other terms can be added to Equation (3.25), in order to consider occlusions and other a-priori knowledge on the scene depth distribution.

Once the global energy has been defined, a variety of algorithms can be used to find a (local) minimum of Equation (3.25). Since a high number of variables are involved, this minimization is not trivial, i.e., $n_{row} \times n_{col}$ disparity values of D_S which can assume $d_{MAX} - d_{MIN} + 1$ possible values within range (d_{MIN}, d_{MAX}). Therefore, there are $(n_{rows} \times n_{col})^{d_{MAX} - d_{MIN} + 1}$ possible configurations of D_S, and a greedy search of the minimum over all these configurations is not feasible [33]. Traditional minimization approaches associated with regularization and *Markov Random Fields* (*MRF*) including continuation, highest confidence first, and mean-field annealing have been presented in the past [45]. More recently, max-flow and graph-cut methods have been proposed to solve a special class of global optimization problems [44].

As previously stated, global methods are computationally more expensive than local methods, but the main advantage of these algorithms is that they are able to cope with depth discontinuity and are more robust in textureless areas.

3.4.3 Semi-global methods

Semi-global algorithms use a global disparity model like the global methods, but they impose constraints only on part of the scene, in order to reduce the computational cost. In detail, for each point of D_S, the minimization of the cost function is computed on a reduced model. These algorithms offer a good tradeoff between accuracy and runtime, therefore they are well suited for many practical applications.

One of the most classical semi-global algorithm is the so-called *Semi-Global Matching* (*SGM*) technique [16], which is based on the idea of pixel-wise matching by using *Mutual Information* (*MI*) as matching cost. This algorithm computes many 1D energy functions along different paths (usually 8 or 16), then the functions are minimized and finally their costs are summed up. For each point, the chosen disparity correspond to the minimum aggregated cost. As compared to local and global methods, this algorithm is very fast and works well even with textureless regions.

Chapter 4

ToF Cameras and Stereo Systems: Comparison and Combination

From the previous Chapters it is clear that both ToF and stereo vision systems are good techniques for depth estimation of a 3D scene. However, each one operates under specific conditions, hence they are suitable for different applications. In this Chapter a description of these two range cameras is provided, in order to show that they can be considered complementary systems. Moreover, a review of literature regarding their fusion is presented. Finally, in Section 4.4 the ideal model considered for test and development of the super-resolution algorithms is described.

4.1 Stereo systems analysis

Stereo systems are the most common schemes for 3D image acquisition. Even if they have been greatly improved during the last years, they still cannot handle all scene situations. One first important parameter for their characterization is the *accuracy*, that is the degree of closeness of measurements of a quantity to its actual (true) value. Since stereo vision systems completely rely on the correct identification of corresponding points, their accuracy depends on scene geometry and texture characteristics. For example, in the FW algorithm the matching consists of searching the correspondent conjugate pixel onto the epipolar line in the target image for every pixel in the reference image. This matching works well on textured scenes, but it has difficulties in homogeneous regions and when repetitive patterns along the epipolar line of the target image introduce ambiguities. Non-textured and featureless regions of a scene are particularly challenging for stereo system since there is an insufficient visual information for establishing a correspondence across the two cameras. This can lead to unknown disparity pixels, where the value is simply propagated from the neighbor disparity. Another parameter that influences stereo matching accuracy is the illumination of the target scene. Local stereo are the most scene-dependent algorithms, whereas global and semi-global methods are less scene-dependent. On the other hand, global and semi-global methods are characterized by very time-consuming algorithms and are not suited for real-time applications.

The second parameter for stereo vision system evaluation is the *precision*, also called reproducibility or repeatability, that is the degree to which repeated measurements under unchanged conditions show the same result. As regards the FW algorithms, if the same scene is acquired N times under the same conditions, the estimated disparity images can change according to noise fluctuations in the acquired image pairs. In fact, because of this noise, the same pixel w_1 of the reference image can be matched with different pixels w_2^i, with $i \in \{1, 2, ..., N\}$, in the target image for each acquisition. The noise effect depends on the amount of texture in the acquired scene: high-textured scenes are less noise affected with respect to low-textured scenes. In general, global and semi-global methods are less noise-dependent as compared to local methods, due to the imposed smoothness model.

Finally, the third evaluation parameter is the *resolution*, which is the smallest change in the underlying physical quantity that produces a response in the measurement system. There are two different resolutions for range image systems: *spatial resolution*, i.e., the number of pixels in the sensor, and *depth resolution*, i.e., the smallest depth variation detectable. The spatial resolution of a stereo vision system is simply the number of pixels of one of the two cameras (e.g., 1920 × 1080). However, usually some pixels of the image cannot be matched, especially in presence of depth discontinuities and occlusions. For these reasons the real spatial resolution of a stereo vision system is less or equal than the total number of pixels. Regarding the depth resolution, as stated in Chapter 3 and in particular from Equation (3.22), this decreases quadratically with the distance from the objects.

4.2 Time-of-flight cameras analysis

Differently from stereo vision systems, ToF cameras can extract depth information in real-time and are not very sensitive at scene peculiarity such as textures. In fact, ToF cameras excels on textureless surfaces for which stereo systems perform poorly. On the other hand, they have in practice limited spatial resolution, limited accuracy due to several noise sources and they are very sensitive to background illumination.

Accuracy on ToF cameras is strongly influenced by all the errors affecting the depth measurement. As broadly described in Section 2.4, the measured depth is affected by random errors, such as flying pixels and internal-scattering. Furthermore, also systematic errors like harmonic distortion and phase wrapping are present. Although the effect of all these noise sources can be mitigated, especially the one from systematic errors, the accuracy of the measurement remains one of the main issue for these devices. In fact, as stated in [19], accuracy can be up to some hundreds of millimeters (e.g., $400[mm]$).

Regarding the precision, noise on ToF cameras can be modeled to be Gaussian [33] with standard deviation given by Equation (2.13). From that equation it can be seen that the measurement precision increases as the amplitude of the received signal A increases (i.e., high-reflective objects or closer distance). Moreover, both a lower background illumination B and a higher modulation frequency f_{mod} allow a precision improvement. Although from the producer specifications the precision (repeatability) of the CamCube 3.0 is less than $3[mm]$ (typical value, central sensor area at $4[m]$ distance and 75% reflectivity), under normal conditions the precision is lower. In fact, in Chapter 7 experiments performed with the CamCube 3.0 show that its precision is about some centimeters.

Resolution is the last evaluation parameter. As explained in Section 2.2.1, spatial resolution is one of the main limitation of ToF cameras. Currently, one of the ToF cameras with highest spatial resolution is the PMD CamCube 3.0, which has a 200 ×

200 pixel sensor, whereas the MESA SR4000 has 176 × 144 pixel. Regarding the depth resolution, at first approximation it can be considered constant within the non-ambiguity range (e.g., $[0, 7500][mm]$ for the CamCube 3.0 with $f_{mod} = 20[MHz]$). However, many factors must be considered to estimate the real depth resolution of the camera. The most important is the noise affecting the measurement, which increases with the distance. For this reason depth resolution cannot be considered constant in the whole range. Since the resolution of the CamCube 3.0 is not given from PMDTec, the only way to estimate it is through some evaluation tests. For example, measuring the depth of a plane object placed parallel to the camera and then moving the plane closer to the camera until the measured depth change value. The difference between these two values is the searched depth resolution.

4.3 Comparison and combination

ToF cameras and stereo vision systems compensate for each other deficiencies, hence their combination aims at creating information that exceeds the quality of each individual source. Compared to stereo systems the accuracy of ToF cameras is higher, especially for flat and/or non-textured areas, in which a two cameras setup cannot provide a reliable disparity estimation. Moreover, since ToF cameras are monocular systems, they do not suffer from occlusions problem. On the other hand, details and discontinuities in intensity and/or depth decrease the performance of the ToF depth measurement, while they typically increase the performance of stereo. Furthermore, traditional cameras have in common much higher spatial resolution and stereo setups have a larger working range. In fact, stereo systems can work in both, indoor and outdoor environment, whereas ToF is suited for indoor acquisition. Regarding the depth resolution, stereo systems perform better than ToF for closer objects, whereas ToF cameras depth resolution is less dependent on the object distance. The advantage and disadvantage of the two systems are listed in Table 4.1. As it can be seen, they complement each other.

Many approaches consider the fusion of ToF generated depth with a standard camera image or a stereo vision system estimated disparity. In [41] a hardware-based realization is presented. This setup combines a traditional CCD and a PMD chip in a

	Stereo	**ToF**
Image resolution	✓	X
Depth discontinuities	✓	X
Flat areas	X	✓
Occlusions	X	✓
Depth resolution	$\Delta z \cong \frac{z^2}{bf}\Delta d$	$\Delta z \sim C \in \left[0, \frac{c}{2f_{mod}}\right]$

Table 4.1: Advantage and disadvantage of stereo and ToF cameras. The two systems can be considered complementary.

monocular device using an optical splitter. Due to the resulting monocular camera, no special mapping transformation between both images has to be done as both consist of the same view. However, a know disadvantage of this approach consists in the used beam splitter. The incoming optical signal is separated into two parts, and this leads to using only half of the optical power for the PMD sensor, which influences the depth estimation measurement process. In [27], a binocular camera setup is used, which combines single high-resolution RGB images with PMD distance data in order to project color information onto the depth image and refine it. Another binocular setup is presented in [57], where the authors, inspired by stereo matching, create a cost volume from the depth map, filter it joint-bilaterally using the color images, and then extract an improved depth map. This approach has unfortunately a high computation time. In fact it takes several seconds per frame. Three cameras setups are proposed in [10, 11], where ToF and stereo are combined by converting the ToF depth data into disparity and then using it as an initialization for a hierarchical stereo matching algorithm. In [33], fusion between ToF and stereo is achieved with a probabilistic method based on a *Maximum-a-Posteriori Markov Random Field* (*MAP-MRF*) Bayesian approach. This global approach guarantees good results, at the expense of complexity and computation time.

4.4 Ideal model for depth super-resolution

The target of this thesis is to combine the strength points of ToF and stereo vision systems in order to do super-resolution of a ToF depth image employing a *local* approach. Depth image super-resolution is different from color image super-resolution. For color images blurry edges are perceptually tolerable and expected, but for depth images the presence of blur in depth discontinuities indicates a wrong depth estimation and it is not accepted. Moreover, the ToF estimated depth is very noisy, hence also a depth noise reduction should be achieved together with the super-resolution. Simple interpolation techniques for super-resolution such as *Cubic*, *Lanczos* or *Bilinear interpolation* are prone to errors and suffer from annoying artifacts such as aliasing effect, halo, blur, and other problems, around the edges due to their non-adaptive properties. The aim of the developed algorithms it is to produce a high-resolution depth image starting from a low-resolution ToF depth, by using a color image from a *single* camera or a disparity map from *two* cameras. The resulting depth map must have sharp edges and smooth flat regions. Therefore, the approach consists to use ToF information for flat areas and one camera (color image) or two cameras (stereo disparity) information for edge areas.

The first algorithm is based on *Compressive Sensing* (*CS*) theory [14] and allows to obtain a dense disparity/depth map starting from a sparse disparity/depth map. More in detail, starting from only about 25% of the total pixels, is possible to produce an accurate result. Based on this algorithm, the proposed method uses information from stereo disparity (which can be easily converted in depth by using Equation (3.20)) for the edges and information from ToF depth map for flat regions.

After that, a second approach is presented, which is based on *bilateral filtering* [51]. More in detail, three different *Joint Bilateral Filters* (*JBF*) [23] have been developed. The aim of JBF is to combine a low-resolution ToF depth with a high-resolution color image from a single additional video-camera, in order to obtain a high-resolution depth. In this case edge information from the high-resolution color image is used to guide the up-sampling of the depth.

Since the ground truth (i.e., a high-resolution depth image) is normally not available all the algorithms where first tested in an ideal scenario. For this purpose the Middlebury 2006 Stereo dataset [17] was used, which is composed by 21 pairs of color images representing different scenes and their corresponding ground truth disparity images created using structured light technique. The images are rectified and radial distortion has been removed. For each scene three resolutions are provided: “full-size” (width: 1240..1396, height: 1110 pixel), “half-size” (width: 620..698, height: 555 pixel) and “third-size” (width: 413..465, height: 370 pixel). The algorithms have been tested using a pair of “third-size” disparity and color images for each scene. An *ideal model* is defined, which assumes a perfect mapping between ToF and video-camera images, and a Gaussian ToF depth image noise. Therefore, it is possible to recover the ground truth depth map starting from a down-sampled and noised version of the depth map, which simulates the low-resolution ToF image, and from the original color or disparity image which simulate the information from one or two standard cameras. All the super-resolution algorithms have been tested with several levels of additive Gaussian noise standard deviation, specifically $\sigma_N \in \{0,\ 0.5,\ ...,\ 10\}[cm]$. Then, the quality of each algorithm can be evaluated by computing the *Mean Absolute Error* (MAE) between reconstructed depth D_R and ground truth depth D_{GT}

$$MAE = \frac{1}{N} \sum_{u,v} \left| D_R\left(u,v\right) - D_{GT}\left(u,v\right) \right|, \tag{4.1}$$

where both the depth maps are composed by a matrix of $N_R \times N_C$ pixels, $u \in [0, ..., N_C]$, $v \in [0, ..., N_R]$ and N is the total number of pixels $N = N_C \cdot N_R$.

Chapter 5

Compressive Sensing Depth Super-Resolution

In this Chapter a super-resolution algorithm based on *Compressive Sensing* (*CS*) theory is described. Section 5.1 introduces the compressive sensing fundamentals. In Section 5.2 the reconstruction algorithm based on the work of Hawe et al. [14] is described. Then, Section 5.3 presents the adaptation of the compressive sensing reconstruction for depth images super-resolution purposes. Section 5.4 shows the experimental results, whereas a summary is presented in Section 5.5.

5.1 Compressive sensing

Compressive Sensing (also known as Compressed Sensing, Compressive Sampling, or Sparse Sampling) is a signal processing technique for efficiently acquiring and reconstructing a signal, by finding solutions to underdetermined linear systems. Signals can often be well-approximated as a linear combination of just a few elements from a known basis. When this representation is exact the signal can be called sparse. Mathematically, a signal $\mathbf{x}$ is k-sparse when it has at most k nonzero entries. Typically, CS is used for data compression, hence it deals with signals which are not themselves sparse, but admit a sparse representation in some basis Φ. In this context, the aim of CS is to recover a signal $\mathbf{s} \in \mathbb{R}^n$ starting from a small number of noisy linear measurements $\mathbf{y} = \Phi\mathbf{s} \in \mathbb{R}^m$, with $m < n$. The focus is on the undetermined linear system, in which the operator $\Phi \in \mathbb{R}^{m \times n}$, called *sampling basis*, has unit norm columns and forms an overcomplete basis considering that $m < n$ [40]. The resulting problem is regularized by considering that the unknown signal $\mathbf{s}$ is k-sparse, or compressible with k significant coefficients in the signal $\mathbf{x} \in \mathbb{R}^n$. The corresponding linear transformation is

$$\mathbf{s} = \Psi\mathbf{x}, \tag{5.1}$$

where $\Psi \in \mathbb{R}^{n\times n}$ is an orthonormal basis of $\mathbb{R}^n$, called *representation basis*. Moreover, with the sampling basis Φ, that converts the original signal $\mathbf{s}$ into the measurements vector $\mathbf{y}$, the relation becomes

$$\mathbf{y} = \Phi\mathbf{s} = \Phi\Psi\mathbf{x}. \tag{5.2}$$

Now, the target of CS is to reconstruct $\mathbf{s}$ starting from the measurements $\mathbf{y}$ by computing $\mathbf{x}$ from Equation (5.2) and exploiting the sparse nature of $\mathbf{x}$ [14]. This means find the sparsest vector $\mathbf{x}$ compatible with the acquired measurements $\mathbf{y}$. Mathematically this can be expressed with the following problem

$$\begin{aligned}&\underset{\mathbf{x}\in\mathbb{R}^n}{\arg\min}\, \|\mathbf{x}\|_0\\ &\text{subject to } \mathbf{y} = \Phi\Psi\mathbf{x},\end{aligned} \tag{5.3}$$

where $\|\mathbf{x}\|_0$ is the ℓ_0-pseudo norm of $\mathbf{x}$ (i.e., the number of nonzero entries). Unfortunately, as stated in [14] solving Equation (5.3) is computationally intractable, therefore, under certain conditions on the matrix $\Phi\Psi$, it is possible to use the ℓ_1-norm instead of the ℓ_0-pseudo norm, which leads to

$$\begin{aligned}&\underset{\mathbf{x}\in\mathbb{R}^n}{\arg\min}\, \|\mathbf{x}\|_1\\ &\text{subject to } \mathbf{y} = \Phi\Psi\mathbf{x}.\end{aligned} \tag{5.4}$$

This is a computationally tractable regularized inverse problem, which can be solved in polynomial time. With the CS, if the number of measurements m is big enough compared to the sparsity factor k, it is possible to correctly solve Equation (5.4) and to reconstruct the original signal $\mathbf{s}$ from the computed $\mathbf{x}$ using Equation (5.1).

5.2 CS for dense reconstruction

The concept of sparsity has been exploited in signal processing for compression and denoising applications. In particular, CS can be used for image processing, since the wavelet transform provides nearly sparse representation for natural images as shown in

Figure 5.1. In fact, the wavelet transform consists on recursively dividing the image into its low and high-frequency components. The lowest frequency components provide a coarse scale approximation of the image, while the higher frequency components fill-in the details and resolve edges. The same concepts can be applied onto disparity images. Following the method proposed in [14], let $D \in \mathbb{R}^{h \times w}$ be a disparity map having $n = hw$ entries and assume that only $m < n$ disparities are known. Usually, disparity maps are composed by large homogeneous regions of equal disparity, which correspond to flat areas of the scene and some discontinuities at the transitions between those regions, which are the scene edges. In the wavelet transform of this disparity map large homogeneous regions are represented by only a small number of wavelet coefficients, whereas edge areas are linked to important coefficients cluster. In this way it is possible to assume the wavelet transform of disparity maps to be sparse, and reconstruct D starting from the wavelet transform of some of its samples.

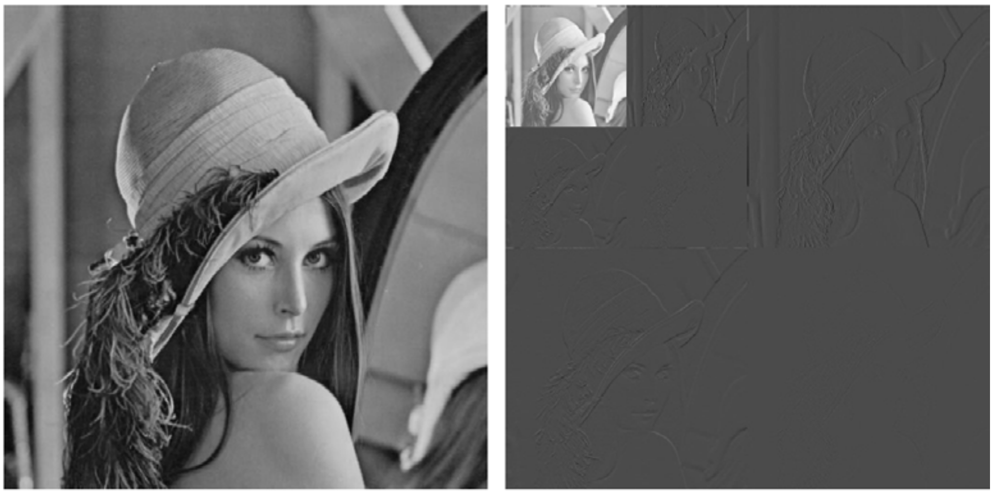


Figure 5.1: Sparse representation of an image via a multi-scale wavelet transform. From left to right: original image; wavelet representation. Large coefficients are represented by light pixels, while small coefficients are represented by dark pixels. Observe that most of the wavelet coefficients are close to zero.

With the notations of Section 5.1, let $\mathbf{s} \in \mathbb{R}^n$ be the vectorized unknown disparity map D and $\mathbf{y} \in \mathbb{R}^m$ the disparity measurements. Furthermore, let denote with Ψ a Daubechies Wavelet basis for the relation $\mathbf{s} = \Psi\mathbf{x}$, with $\mathbf{x} \in \mathbb{R}^n$ the sparse vector of wavelet coefficients. Eventually, the sampling basis Φ depends on the measurement y_i itself, for example if $\mathbf{p} \in \mathbb{N}^m$ is the vector containing the indices of the measured disparities, the sampling basis can be written as

$$\Phi = \left[\mathbf{e}_{\mathbf{p}_{(1)}}, ..., \mathbf{e}_{\mathbf{p}_{(m)}}\right]^{\mathrm{T}}, \tag{5.5}$$

where $\mathbf{e}_i \in \mathbb{R}^n$ is the standard basis vector corresponding to the measurement y_i.

From the fact that in wavelet transform relevant coefficients coincide with discontinuities, the sampling positions are exactly those disparities lying at the discontinuities. In [14] the authors assume that disparity discontinuities coincide with image intensity edges, which means finding the edges directly into the disparity map D by using a Canny filter [5] and taking the positions of the detected edges as the sampling positions. By simply applying the same edge detector into one of the two color images also texture edges can be detected. Unfortunately, texture edges not always correspond to real depth edges, and consequently relevant wavelet coefficients. Regarding the remaining flat areas, some disparity values are selected, in order to control the minimum sampling density.

5.2.1 Reconstruction algorithm

The chosen method for solving the convex optimization problem (5.4) is a first-order method. Key point of this algorithm is the *Total Variation* (*TV*) of the disparity map D. This can be expressed as

$$\|D\|_{\mathrm{TV}} = \sum_{i=1}^{h-1} \sum_{j=1}^{w-1} \|\nabla D(i,j)\|_2 , \tag{5.6}$$

where $\nabla D(i,j)$ is the local variation of D at entry (i,j)

$$\nabla D(i,j) = [D(i,j) - D(i,j+1), D(i,j) - D(i+1,j)] . \tag{5.7}$$

Total variation is often used in image processing, especially for image denoising, where it is remarkably effective at simultaneously preserving edges whilst smoothing away noise in flat region.

With these notations, it is possible to express the TV-norm of the unknown disparity map $\mathbf{s}$ by means of two suitable matrices $\mathcal{G}_x$, $\mathcal{G}_y \in \mathbb{R}^{n \times n}$

$$\|\mathbf{s}\|_{\mathrm{TV}} := \|D\|_{\mathrm{TV}} = \sum_{j=1}^{n} \sqrt{\left(\mathbf{e}_j^{\mathrm{T}} \mathcal{G}_x \mathbf{s}\right)^2 + \left(\mathbf{e}_j^{\mathrm{T}} \mathcal{G}_y \mathbf{s}\right)^2} . \tag{5.8}$$

Moreover, in order to further reduce the number of measurements required, an additional square diagonal weighting matrix $\mathcal{W} \in \mathbb{R}^{n \times n}$ for the wavelet coefficients is introduced. Now the problem (5.4) can be expressed in the unconstrained Lagrangian form

$$\arg\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{2}\|\mathcal{A}\mathbf{x}-\mathbf{y}\|_2^2 + \lambda\left(\|\mathcal{W}\mathbf{x}\|_1 + \gamma\|\Psi\mathbf{x}\|_{\mathrm{TV}}\right) \tag{5.9}$$

where $\mathcal{A} = \Psi\Phi$, λ is the Lagrange multiplier and $\gamma \geq 0$ is a weighting parameter. Then, Equation (5.9) is minimized by a first-order method which requires to compute gradient. The authors propose a *conjugate subgradient* method. For the minimization of $\|\mathcal{W}\mathbf{x}\|_1$ the Euclidean norm is used, hence its sub-differential is the set $\partial\|\mathcal{W}\mathbf{x}\| \subset \mathbb{R}^n$ with

$$\partial\|\mathcal{W}\mathbf{x}\|(i) = \begin{cases} \dfrac{(\mathcal{W}\mathbf{x})(i)}{|(\mathcal{W}\mathbf{x})(i)|} & \text{if } (\mathcal{W}\mathbf{x})(i) \neq 0 \\ [-1,1] & \text{otherwise.} \end{cases} \tag{5.10}$$

Regarding the subgradient for $\|\Psi\mathbf{x}\|_{\mathrm{TV}}$, since the Euclidean norm is computationally unfeasible, the TV-norm is approximated to

$$\|\mathbf{s}\|_{\mathrm{TV}} \approx \|\mathbf{s}\|_{\nu,\mathrm{TV}} := \sum_{j=1}^{n} h_\nu\left(\sqrt{\left(\mathbf{e}_j^{\mathrm{T}}\mathcal{G}_x\mathbf{s}\right)^2 + \left(\mathbf{e}_j^{\mathrm{T}}\mathcal{G}_y\mathbf{s}\right)^2}\right), \tag{5.11}$$

where h_ν is the Huber functional

$$h_\nu = \begin{cases} |x| - \dfrac{\nu}{2} & \text{if } |x| \geq \nu \\ \dfrac{x^2}{2\nu} & \text{otherwise.} \end{cases} \tag{5.12}$$

In this way, using the shorthand notation

$$\mathbf{r} := \Psi^{\mathrm{T}}\sum_{j=1}^{n}\frac{\mathcal{G}_x^{\mathrm{T}}\mathbf{e}_j\mathbf{e}_j^{\mathrm{T}}\mathcal{G}_x\mathbf{s} + \mathcal{G}_y^{\mathrm{T}}\mathbf{e}_j\mathbf{e}_j^{\mathrm{T}}\mathcal{G}_y\mathbf{s}}{\sqrt{\left(\mathbf{e}_j^{\mathrm{T}}\mathcal{G}_x\mathbf{s}\right)^2 + \left(\mathbf{e}_j^{\mathrm{T}}\mathcal{G}_y\mathbf{s}\right)^2}}, \tag{5.13}$$

the gradient of $\|\Psi\mathbf{x}\|_{\nu,\mathrm{TV}}$ is given by $\nabla\|\Psi\mathbf{x}\|_{\nu,\mathrm{TV}}$ with the entries

$$\nabla\|\Psi\mathbf{x}\|_{\nu,\mathrm{TV}}(i) = \begin{cases} \mathbf{r}(i) & \text{if } |x| \geq \nu \\ \mathbf{r}(i)^2/\nu & \text{otherwise.} \end{cases} \tag{5.14}$$

Consequently, the subdifferential of the modified objective

$$f(\mathbf{x}) = \frac{1}{2}\|\mathcal{A}\mathbf{x}-\mathbf{y}\|_2^2 + \lambda\left(\|\mathcal{W}\mathbf{x}\|_1 + \gamma\|\Psi\mathbf{x}\|_{\nu,\mathrm{TV}}\right) \tag{5.15}$$

is the set

$$\partial f\left(\mathbf{x}\right) = \mathcal{A}^{\mathrm{T}}\left(\mathcal{A}\mathbf{x} - \mathbf{y}\right) + \lambda\left(\partial\|\mathcal{W}\mathbf{x}\|_1 + \gamma\nabla\|\Psi\mathbf{x}\|_{\nu,\mathrm{TV}}\right). \tag{5.16}$$

With the notation

$$\mathbf{b} = \lambda^{-1}\mathcal{A}^{\mathrm{T}}\left(\mathcal{A}\mathbf{x} - \mathbf{y}\right) + \gamma\nabla\left\|\Psi\mathbf{x}\right\|_{\nu,\mathrm{TV}}, \tag{5.17}$$

it is possible to verify that the final subgradient with smallest Euclidean norm is given by

$$\mathbf{g}\left(\mathbf{x}\right) = \mathcal{A}^{\mathrm{T}}\left(\mathcal{A}\mathbf{x} - \mathbf{y}\right) + \lambda\left(\nabla\|\mathcal{W}\mathbf{x}\|_1 + \gamma\nabla\|\Psi\mathbf{x}\|_{\nu,\mathrm{TV}}\right), \tag{5.18}$$

where

$$\nabla\|\mathcal{W}\mathbf{x}\|_1\left(i\right) := \begin{cases} \dfrac{\left(\mathcal{W}\mathbf{x}\right)\left(i\right)}{\left|\left(\mathcal{W}\mathbf{x}\right)\left(i\right)\right|} & \text{if } \left(\mathcal{W}\mathbf{x}\right)\left(i\right) \neq 0 \\ -\operatorname{sign}\left(\mathbf{b}\left(i\right)\right)\min\left\{\left|\mathbf{b}\left(i\right), 1\right|\right\} & \text{otherwise.} \end{cases} \tag{5.19}$$

The descendent method is initiated with $\mathbf{x}_0 = \mathcal{A}^{\mathrm{T}}\mathbf{y}$ and iteratively updated

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i\mathbf{h}_i, \tag{5.20}$$

where the scalar $\alpha_i \geq 0$ is the line search parameter or the step length (set to 10^{-5}) and $\mathbf{h}_i$ is the descent direction at the i^{th} iteration. The chosen line-search technique for computing α_i is the *background line-search*, whereas the descent direction $\mathbf{h}_i$ is updated with the Hestenes-Stiefel formula

$$\mathbf{h}_{i+1} = -\mathbf{g}_{i+1} + \frac{\mathbf{g}_{i+1}^{\mathrm{T}}\left(\mathbf{g}_{i+1} - \mathbf{g}_i\right)}{\mathbf{h}_i^{\mathrm{T}}\left(\mathbf{g}_{i+1} - \mathbf{g}_i\right)}\mathbf{h}_i, \tag{5.21}$$

where $\mathbf{g}_i := \mathbf{g}(\mathbf{x}_i)$ and initial value $\mathbf{h}_0 = -\mathbf{g}_0$. Equations (5.20) and (5.21) are iterated either until convergence is achieved, or a maximum number of iterations has been reached. The stopping criterion is the norm of $\mathbf{g}_i$.

Eventually, the output of the algorithm are the reconstructed wavelet coefficients $\mathbf{x}$ from which it is possible to compute the dense disparity map using Equation (5.1).

5.3 CS for depth super-resolution

The authors provide a Matlab code of the algorithm, available at [6], which starts from a ground truth disparity image and samples it by taking ground truth edges from a Canny edge detector and some random samples for flat areas. Then, from this sparse disparity map (around 20% of the original pixels) the reconstruction algorithm is applied, in order to obtain the dense disparity image.

From this code, it is possible to adapt the CS reconstruction for ToF depth and stereo disparity combination. As stated in Chapter 4, ToF information can be used for flat areas, whereas stereo disparity is useful for the scene edges. In the ideal model case, simulations have been made by using the ground truth disparity maps D_{GT} from the Middlebury dataset. The original algorithm was planned to work with disparity images, which means disparity values in units between 0 and 255. Therefore, instead of convert this disparity in depth using Equation (3.20) the ground truth disparity is considered as a depth with quantized values in the range $[0, 255]$. Therefore, from here the ground truth disparity will be denoted as ground truth depth.

An overview of the framework for the CS super-resolution is provided in Figure 5.2. The algorithm starts from a low-resolution ToF depth D_{ToF} and a high-resolution stereo disparity, which is converted to the depth D_{STEREO} using Equation (3.20). Then, these two depths are fused in order to obtain a new depth image D_D with the same resolution of the stereo disparity. The stereo depth is simulated by using the "third-size" Middlebury depth images D_{GT} (i.e., 427 × 370 pixel), whereas ToF depth is reproduced by down-sampling the same depth with a scale factor 2 (i.e., 214 × 185 pixel). The down-sampling for the generation of the ToF depth is accomplished by dividing D_{GT} in 2 × 2 non-overlapping tiles and copying only the upper left pixel in the low-resolution D_{ToF}. Moreover, the ToF camera noise, which is supposed to be

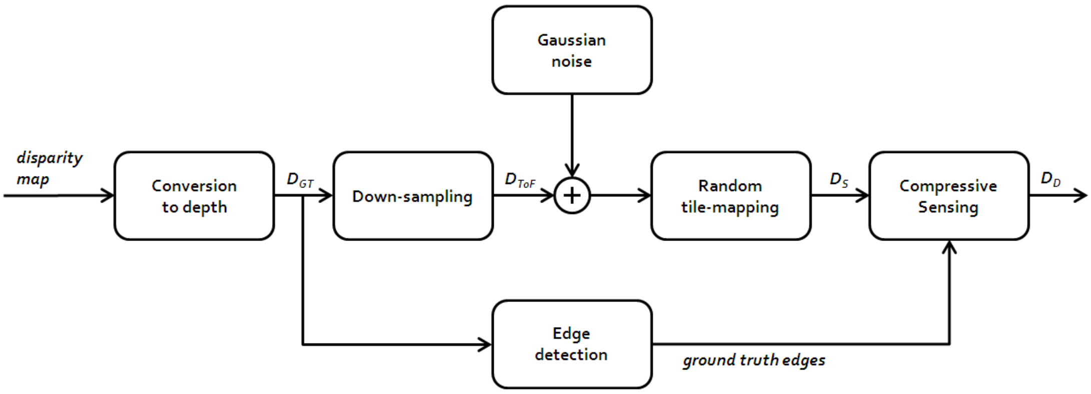


Figure 5.2: CS super-resolution framework.

Gaussian in the ideal model, is generated by adding some Gaussian noise with different standard deviation intensities to the down-sampled depths. In order to create the starting sparse depth image D_S, each pixel from the low resolution ToF depth has to be mapped onto D_S. Since the mapping is supposed to be perfect, starting from the top-left position D_S is divided into 2×2 non-overlapping tiles, and one of the four positions is randomly selected. Then, a depth value from the ToF is copied to the same position. This procedure, from now denoted by *random tile-mapping*, is repeated for all the ToF pixels starting from the top-left one. The random tile mapping allows to reproduce the behavior of the ToF sensor, which associates a single pixel depth value to a finite scene area (see also Section 2.2.1). Figure 5.3 shows an example of random tile-mapping. Afterwards, considering the assumption that depth discontinuities coincide with image intensity edges, stereo edge pixels are mapped by simply applying a Canny edge detector to $D_{STEREO} \equiv D_{GT}$. Then, the depth values corresponding to the edge positions are written in the same locations onto D_S, creating the new sparse depth map with edges $D_{S_{EDGES}}$. If one edge pixel corresponds to one of the previously copied ToF depth values, the new stereo edge value overwrites the old pixel. Furthermore, a random set of sparse samples of the flat areas is removed to keep constant the percentage of the initial collection of pixels (from now denoted as starting pixels). Figure 5.4 shows the three steps for the creation of $D_{S_{EDGES}}$. Once the sparse depth image $D_{S_{EDGES}}$ is made, the CS reconstruction algorithm is started, which provides the final dense depth image D_D.

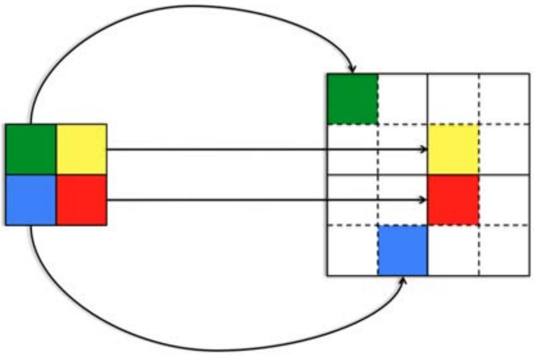
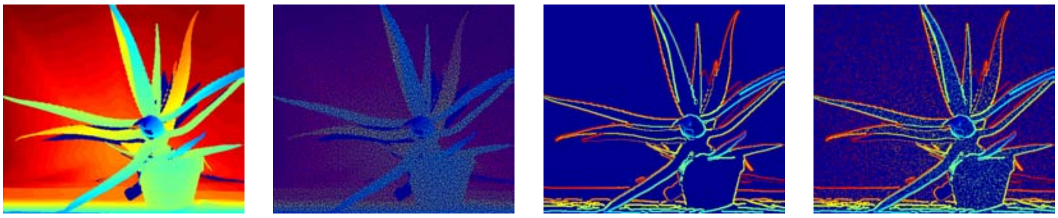


Figure 5.3: An example of random tile-mapping for four pixels. Dense ToF depth (left), and sparse mapped depth D_S (right).



(a) D_{GT}. (b) D_S. (c) Canny edges. (d) $D_{S_{EDGES}}$.

Figure 5.4: Creation of the sparse depth map D_S: (a) ground truth depth (blue pixels are occluded areas), (b) random tile-mapping, (c) edges from Canny filter, (d) final sparse depth D_S.

5.4 Experimental results

The reconstruction algorithm was applied to the Middlebury dataset [17], by adding to each ToF low-resolution depth a Gaussian noise with 21 different levels of standard deviation $\sigma_N \in \{0,\ 0.5,\ ...,\ 10\}[cm]$. Note that the original MATLAB code works with values in the range $[0, 255]$. Therefore, after the noise addition and the creation of the sparse depth, each sparse depth is converted in disparity, and then the CS algorithm is applied to this image. Finally, the output dense disparity is converted again into the depth D_D. Because of the noise addition, after the first depth-disparity conversion some pixels values are greater than 255 or less than 0, hence they are clipped to the range $[0, 255]$. The percentage of pixels over 255 is at most the 2.2% for the *aloe* scene, 14.7% for the *wood2* scene, and 4.9% for the *baby1* scene.

For each reconstruction the maximum number of iterations was fixed up to 1000, whereas the others parameters are the same of the original code. The algorithm's accuracy is then evaluated by computing the mean absolute error of the reconstructed depth D_D with respect to the ground truth depth D_{GT}

$$MAE = \frac{1}{N} \sum_{u,v} \left| D_D\left(u, v\right) - D_{GT}\left(u, v\right) \right|, \tag{5.22}$$

where N is the total pixels number of one depth image. It is important to specify that occluded areas of the Middlebury depth images (i.e., the zero values) are not considered in the MAE calculation.

5.4.1 CS super-resolution without edges

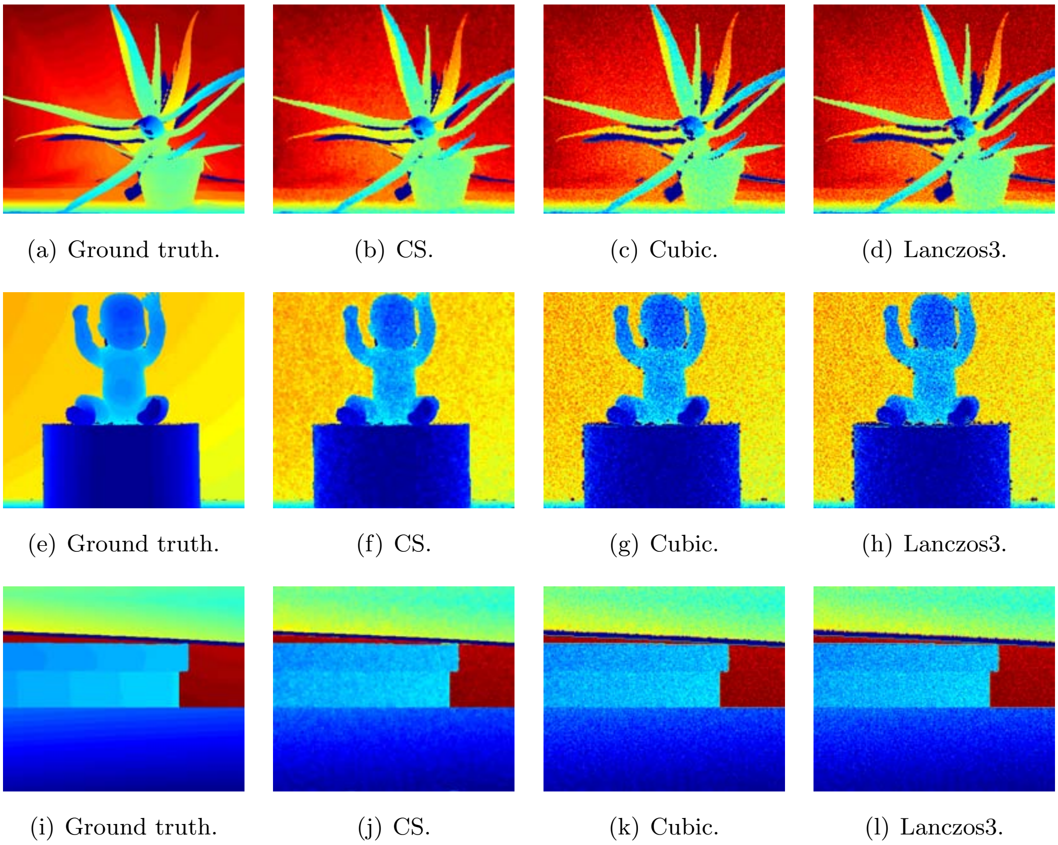
In order to evaluate the performance of CS, the algorithm has been firstly tested starting with the sparse depth D_S, created with the random tile-mapping of the low-resolution ToF depth without any stereo edge information (Figure 5.4(b)). Moreover, the same low-resolution depth D_{ToF} is up-sampled with a scale factor 2 by using the well-known Cubic and Lanczos3 interpolations. Visual and numerical results in Figure 5.5 and 5.6 respectively show that CS, thanks to the total variation minimization, produces depth images that are much more accurate than the simple up-scaling using standard interpolation techniques, especially for high levels of noise. Lanczos3 and Cubic scaling blur depth images which results in greater differences, moreover these techniques cannot reduce the noise in flat areas like the CS do.

5.4.2 CS super-resolution with edges

Once proved the effectiveness of CS, the stereo edge information has been added to the sparse depth, to obtain the final sparse depth $D_{S_{EDGES}}$ as depicted in Figure 5.4(d). Starting from this new sparse depth, the CS reconstruction is applied. The visual results for three scenes of the Middlebury dataset are showed in Figure 5.7. Obviously, now edges are more accurate than the previous reconstruction, and the numerical results showed in Figure 5.8 prove it. From these plots it can be seen how the ground truth edges decrease the final MAE of the reconstructed depth. In the *aloe* scene a greater improvement is achieved because this scene has a lot of edges. In fact, among the 25% of starting pixels in the sparse depth $D_{S_{EDGES}}$, 15% are ground truth edges whereas 10% are noised flat areas samples. On the other hand, for the *wood2* scene, only the 4% of the starting pixels are ground truth edges, and the remaining 21% pixels belongs to flat areas. Therefore, in this case the improvement is less visible.

5.4.3 Starting pixels recomputing

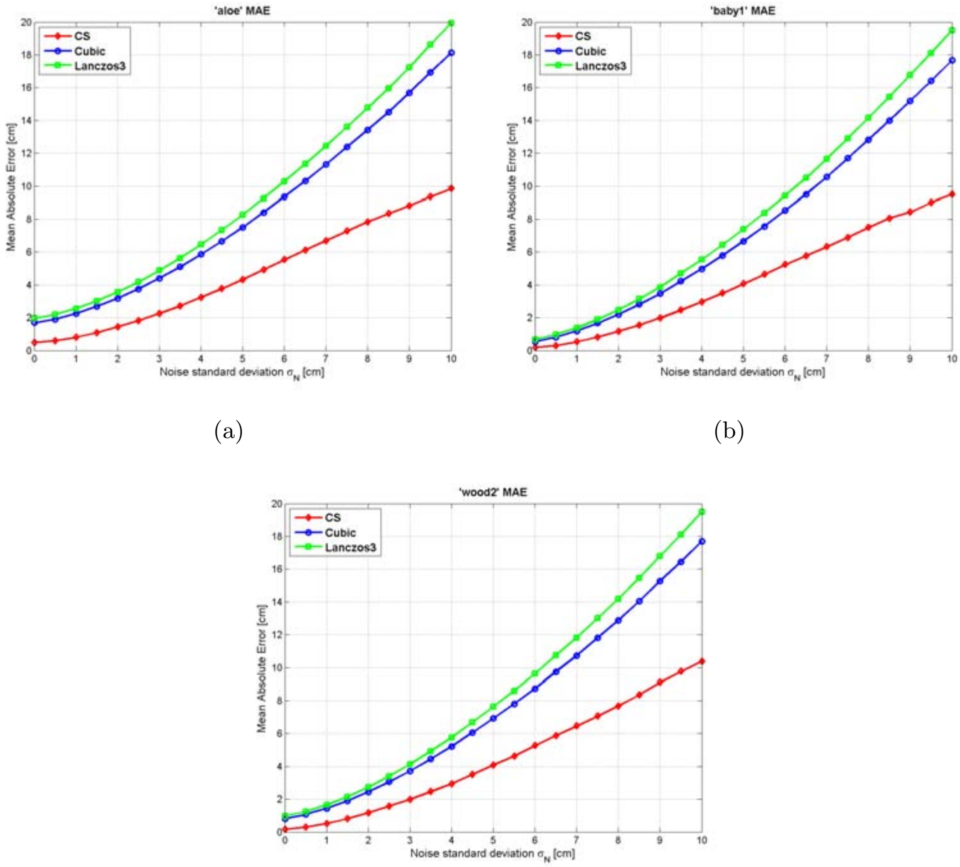
Although Compressive Sensing reconstruction leads to good reconstruction results, its main problem is that that the starting pixels of the sparse depth map D_S are fixed, hence they do not change during the algorithm iterations. Therefore, flat areas of reconstructed depths are smooth except for the starting pixels, which are clearly noticeable especially for high noise standard deviations. In this case the result is a dense depth


(a) Ground truth. (b) CS. (c) Cubic. (d) Lanczos3.

(e) Ground truth. (f) CS. (g) Cubic. (h) Lanczos3.

(i) Ground truth. (j) CS. (k) Cubic. (l) Lanczos3.

Figure 5.5: Visual comparison of the super-resolution between CS, Cubic and Lanczos3 interpolations in case of noise $\sigma_N = 2[cm]$: (a), (b), (c) *aloe* scene; (d), (e), (f) *baby1* scene; (g), (h), (i) *wood2* scene.

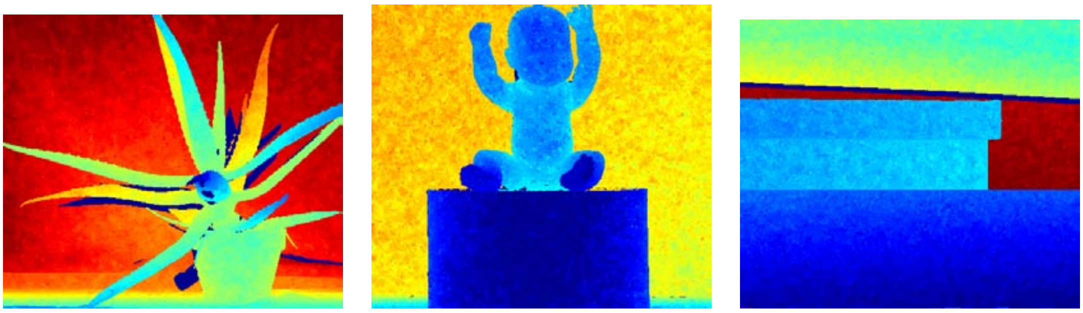


(a)

(b)

(c)

Figure 5.6: Numerical comparison of super-resolution with CS, Cubic, and Lanczos3 interpolation ($\sigma_N = 0$ means noise free samples). The images show the results for: (a) *aloe* scene, (b) *baby1* scene, (c) *wood2* scene.



(a) *aloe.*

(b) *baby1.*

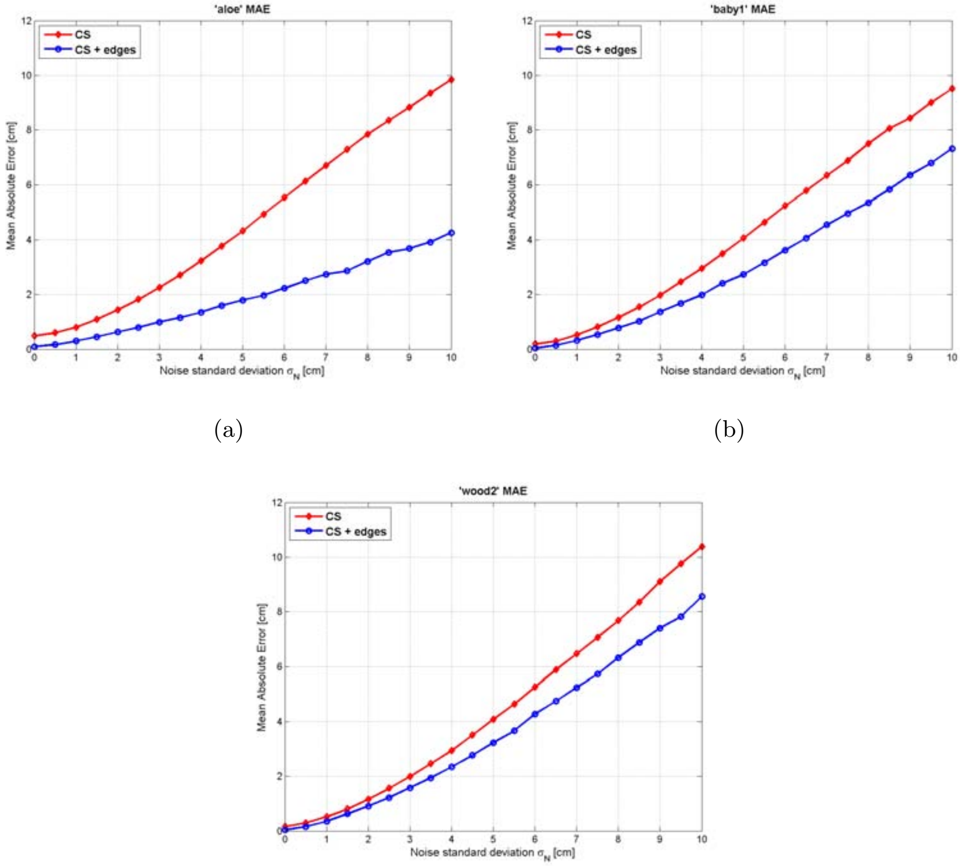
(c) *wood2.*

Figure 5.7: Results of the super-resolution using CS with ground truth edges in case of noise $\sigma_N = 2[cm]$: (a) *aloe* scene, (b) *baby1* scene, and (c) *wood2* scene.

with a *salt and pepper* noise as shown in Figure 5.7(b). For this reason, the depth reconstruction has been improved by removing the starting pixels from the first dense depth D_D, obtaining a new sparse depth $D_{S'}$. Then, starting from $D_{S'}$, a new dense depth map $D_{D'}$ is created by using again CS algorithm and setting the iterations limit to 200. The effect of this second CS reconstruction for the *baby1* scene is clearly visible in Figure 5.9. Moreover, Figure 5.10 shows the behavior of the final MAE as a function of the noise standard deviation. With the exception of $\sigma_N = 0$, in which the results are obviously almost the same, it can be seen that from the second level of noise the starting pixels recomputing reconstruction gives a lower error. The advantage becomes significant when the noise increases. The convergence of this second CS application is reached after around 150 iterations.

5.5 Summary

In this Chapter the first depth super-resolution technique, based on Compressive Sensing, was presented. Numerical results show that it is possible to achieve accurate dense depth by using around 25% of the original depth, which means that a resolution enhancement of scale factor 2 is possible. Moreover, with the starting pixels recomputation the algorithm's robustness to Gaussian noised ToF measurement is increased. Unfortunately, even though the quality of the reconstruction has been improved with the starting pixels recomputation, the CS super-resolution is an iterative and time-

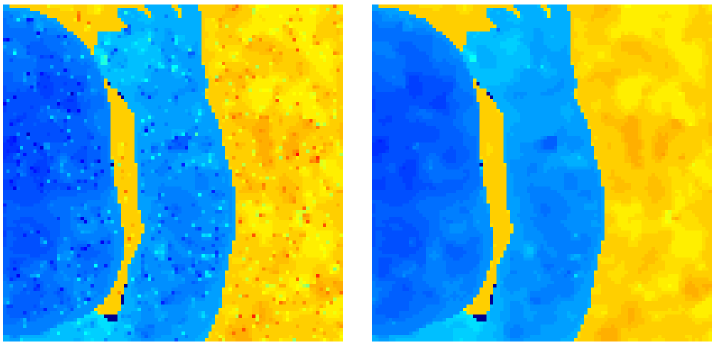


(a)

(b)

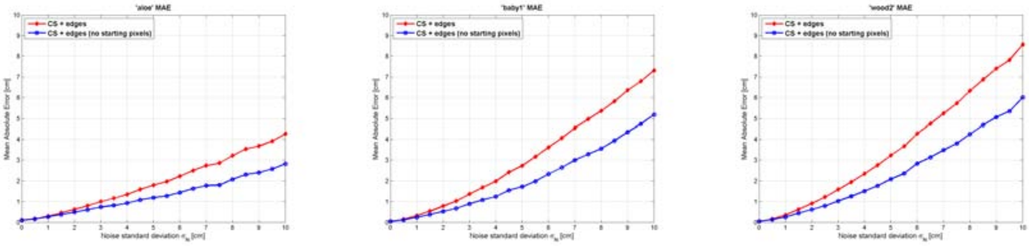
(c)

Figure 5.8: Numerical comparison of super-resolution with CS and CS plus ground truth edges ($\sigma_N = 0$ means noise free samples). The images show the results for: (a) *aloe* scene, (b) *baby1* scene, (c) *wood2* scene.



(a) (b)

Figure 5.9: Image crop on the *baby1* scene to show the difference between the first reconstruction (a) and the second reconstruction without the starting pixels (b). Noise $\sigma_N = 2[cm]$



(a) (b) (c)

Figure 5.10: Numerical comparison of super-resolution with CS plus ground truth edges with and without the starting pixels recomputing ($\sigma_N = 0$ means noise free samples). The images show the results for: (a) *aloe* scene, (b) *baby1* scene, (c) *wood2* scene.

consuming algorithm. In fact, the main problem of this approach is that it needs around 500 iterations before convergence in the first reconstruction, depending from the noise level of the starting sparse depth. Another limitation is that this approach is restricted to the ideal model, which means that CS works only with a perfect edge detection on the stereo disparity/depth map. Since edge detection is still one of the open issues in image processing, in real conditions this algorithm can fail if the edge detector finds an edge in a flat area of the disparity map, or if there are some errors in the stereo disparity estimation. Moreover, this algorithm only considers stereo disparity and not color image from a single camera, an important information that should be used for the super-resolution. In the next Chapter a new joint bilateral filter approach will be presented. It will exploit all the available information (ToF depth, single camera image, two cameras disparity) in order to increase the resolution of the starting ToF depth image.

Chapter 6

Joint Bilateral Filter Depth Super-Resolution

The second super-resolution approach exploits *bilateral filter* techniques for up-sampling the low-resolution ToF depth. Usually, up-sampling is achieved by convolving a low-resolution image with an interpolator kernel, and by resampling the result on a new high-resolution grid. Apply a normal up-sampling, such as Cubic or Lanczos interpolation, on the low-resolution ToF depth leads to edges blurring, because of the smoothness prior inherent in the linear interpolation filters. However, from the combination of ToF camera and stereo camera, additional information are available in the form of the high-resolution color image or of the high-resolution stereo disparity. A *Joint Bilateral Upsampling* (*JBU*) operation can use all these information in order to produce a high-resolution depth image starting from the ToF data.

Section 6.1 introduces the bilateral filter theory. Then, in Section 6.2 the joint use of low and high-resolution images for super-resolution is presented. Specifically, two joint bilateral filters are described and a new *weighted joint bilateral filter* is proposed. In Section 6.3 these up-sampling filters are tested and the results are compared with the CS super-resolution of Chapter 5. Finally, a summary is given in Section 6.4.

6.1 Bilateral filter

Bilateral filtering is a technique with the purpose of smoothing images and, at the same time, preserving edges. This technique was introduced by Tomasi and Manduchi in [52] as a non-linear filter utilizing both (spatial) domain and (intensity) range kernels evaluated on the data values themselves. The key idea of the bilateral filter is that for a pixel to influence another pixel, it should not only occupy a nearby location but also have a similar value. More formally, an output pixel image $BF\left[I\right]_{\mathbf{p}}$ is weighted average of pixels $\mathbf{q}$ from input image $I_{\mathbf{p}}$ in a neighborhood S of pixel $\mathbf{p}$. Mathematically

$$BF\left[I\right]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} G_{\sigma_s}\left(\|\mathbf{p} - \mathbf{q}\|\right) G_{\sigma_r}\left(|I_{\mathbf{p}} - I_{\mathbf{q}}|\right) I_{\mathbf{q}}, \tag{6.1}$$

where $W_{\mathbf{p}}$ is the normalization factor that ensures pixel weights sum to 1

$$W_{\mathbf{p}} = \sum_{\mathbf{q} \in S} G_{\sigma_s}\left(\|\mathbf{p} - \mathbf{q}\|\right) G_{\sigma_r}\left(|I_{\mathbf{p}} - I_{\mathbf{q}}|\right). \tag{6.2}$$

Function G_{σ_s} is the spatial filter kernel, such as a Gaussian centered over $\mathbf{p}$ that decreases the influence of distant pixels, whereas G_{σ_r} is the range filter kernel, a Gaussian that decreases the influence of pixels $\mathbf{q}$ when their intensity values differ from $I_{\mathbf{p}}$.

The bilateral filter is controlled by the two standard deviations σ_s and σ_r. Figure 6.1 illustrates their effects. As the range parameter σ_r increases, the bilateral filter gradually approximates Gaussian convolution more closely because the range Gaussian G_{σ_r} widens and flattens (i.e., is nearly constant over the intensity interval of the image), whereas increasing the spatial parameter σ_s smooths larger features [35].

After the bilateral filter, *Joint Bilateral Filter* has been introduced. This is a filter in which the Gaussian function is applied to a second *guidance image* (e.g., $\tilde{I}$), which gives additional information in order to classify ambiguous regions. This information can be used, for example, to combine the high-frequencies from one image and the low-frequencies from another image [36]. Mathematically

$$JBF\left[I\right]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} G_{\sigma_s}\left(\|\mathbf{p} - \mathbf{q}\|\right) G_{\sigma_r}\left(\left|\tilde{I}_{\mathbf{p}} - \tilde{I}_{\mathbf{q}}\right|\right) I_{\mathbf{q}}. \tag{6.3}$$

The only difference respect to Equation (6.1) is that the range filter uses the guidance image $\tilde{I}$ instead of I. Clearly, also the normalization factor $W_{\mathbf{p}}$ is changed, in order to ensure again that the pixel weights sum to 1.

To sum it up, bilateral filter techniques are an effective way to smooth an image while preserving its discontinuities and also to separate image structures of different

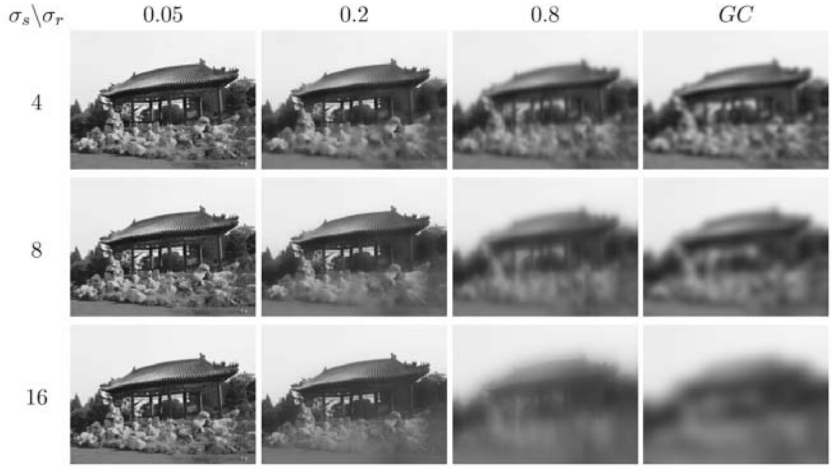


Figure 6.1: The bilateral filter's range and spatial parameters provide more versatile control than Gaussian Convolution (CG). As soon as either of the bilateral filter weights reaches values near zero, no smoothing occurs. As a consequence, increasing the spatial sigma will not blur an edge as long as the range sigma is smaller than the edge amplitude. For example, note that the rooftop contours are sharp for small and moderate range settings σ_r, and that sharpness is independent of the spatial setting σ_s [35].

scales. tinuities and also to separate image structures of different scales. These approaches have many applications, and their central notion of assigning weights that depend on both space and intensity can be tailored to fit a diverse set of applications, such as depth image denosing and super-resolution.

6.2 Joint bilateral filter for depth super-resolution

The bilateral filter was initially used for image denoising. However, in the last period bilateral filter techniques have been developed for super-resolution purposes [22, 23].

6.2.1 Joint bilateral filter

In [23] the authors propose the usage of the joint bilateral filter for up-sampling a low-resolution image D with the guidance of the original high-resolution image $\tilde{I}$. With the random tile-mapping procedure described in Section 5.3, from the low-resolution depth image D_{ToF} a sparse version D_S is created. Then, this sparse depth is interpolated by the JBF. A spatial filter is directly applied to D_S, while a similar range filter is jointly applied on the high-resolution image $\tilde{I}$. With these notations, the up-sampled dense image $JBF\left[D_S\right]_{\mathbf{p}}$ can be written as

$$JBF\left[D_S\right]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} G_{\sigma_s}\left(\|\mathbf{p}-\mathbf{q}\|\right) G_{\sigma_r}\left(\left|\tilde{I}_{\mathbf{p}} - \tilde{I}_{\mathbf{q}}\right|\right) D_{S,\mathbf{q}}, \tag{6.4}$$

where $\mathbf{p}$ and $\mathbf{q}$ denote the coordinates of pixels in $\tilde{I}$ and D_S, $W_{\mathbf{p}}$ is the new normalization factor, and S is the filter aperture. This approach can be used to up-sample low-resolution depth maps guided by their corresponding high-resolution color images.

6.2.2 Joint bilateral filter "Kim"

Direct application of JBF for depth super-resolution frequently exhibits artifacts in reconstructed geometry that can be attributed to erroneous assumption about the correlation of color and depth data. One of the possible artifacts is *texture copying*. This happens when textures from the reference color image are transferred into geometric patterns that appear in the up-sampled depth map, and thus reconstructed geometry. In fact, the bilateral filter adaptively shapes its kernel weights depending upon the values of neighborhood pixels. Hence, in presence of textures in the color image, the Gaussian intensity difference term $G_{\sigma_r}(\cdot)$ of Equation (6.4) takes smaller weight even within the same object and the same depth. To overcome this issue, in [22] the authors propose an up-sampling JBF (from now denoted by JBF_{KIM}) which includes a depth difference parameter in order to give different weights to the color intensity difference term $G_{\sigma_r}(\cdot)$ and to the spatial distance term $G_{\sigma_s}(\cdot)$. This up-sampling filter can be written as

$$JBF_{KIM}\left[D_S\right]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in S} \left(1-\gamma\left(\Delta_S\right)\right) G_{\sigma_s}\left(\|\mathbf{p}-\mathbf{q}\|\right) + \gamma\left(\Delta_S\right) G_{\sigma_r}\left(\left|\tilde{I}_{\mathbf{p}} - \tilde{I}_{\mathbf{q}}\right|\right) D_{S,\mathbf{q}}. \tag{6.5}$$

The novelty with respect to Equation (6.4) is the weighting factor γ, which belongs to the interval $[0,1]$. If γ is near 0 the filter's response depends almost entirely from the Gaussian depth difference, whereas if γ is around 1 the filter's response depends from the Gaussian color intensity difference. The idea is to interpolate pixels of flat areas with the Gaussian depth difference, in order to avoid texture copying, and interpolate pixels of edge areas with the Gaussian color intensity difference. This means that when the pixel belongs to an edge area the weighting factor γ should be around 1; on the contrary, for a flat area γ should be about 0. For this reason the authors define in [22] a blending function $\gamma\left(\Delta_S\right)$

$$\gamma\left(\Delta_S\right) = \frac{1}{1 + e^{-\varepsilon\left(\Delta_S - \tau\right)}} \tag{6.6}$$

where Δ_S is the difference between the maximum and the minimum depth values in the pixels neighborhood S. If the difference Δ_S is under a certain value, the local surface contained in S is most likely to be smooth and only the Gaussian σ_s should be triggered ($\gamma \approx 0$). On the other hand, if the max-min difference lies beyond that value, the local surface S is most likely to be a real geometric edge, hence the Gaussian σ_r is the appropriate choice ($\gamma \approx 1$). The parameters ε and τ of Equation (6.6) are two constants: ε controls the width of the transition area between the two cases, whereas τ determines the min-max difference value at which the blending interval shall be centered. In [22], the authors propose a fixed τ parameter, the value of which depends on the characteristic of the employed depth sensor. Unfortunetly, experiments have shown that with a fixed τ in the JBF_{KIM}, when the noise standard deviation exceeds a certain threshold (i.e., $2[cm]$) the filter founds edge areas in the whole scene (i.e., γ is always 1). Therefore the sparse depth map is interpolated by using only the color intensity difference term $G_{\sigma_r}(\cdot)$ of Equation (6.5). In order to fix this problem, a JBF_{KIM} with a variable τ is proposed. Since with high levels of noise the threshold for edge areas detection must be higher, τ increases together with the noise standard deviation. The trend of τ has been funded by empirical tests. For some values of σ_N, the best threshold has been determined, this means that τ must allow to detect depth discontinuities only at edge areas and not in flat regions. The procedure has been done for *baby1*, *aloe* and *wood2* [17]. The latter two can be considered complementary images. In fact, *aloe* is a high-textured image rich of edges, whereas *wood2* has a lot of flat surfaces and only few textures (see Figure 6.5). The resulting τ is almost the same in all the three images, hence its behavior as a function of the noise standard deviation can be interpolated for the others values of σ_N, and used for the super-resolution of all the sparse depth maps. The resulting function is showed in Figure 6.2.

6.2.3 Weighted joint bilateral filter

The proposed *Weighted Joint Bilateral Filter* (*WJBF*) follows the approach of [22]. The idea is to exploit the information from a high-resolution color image to up-sampling a

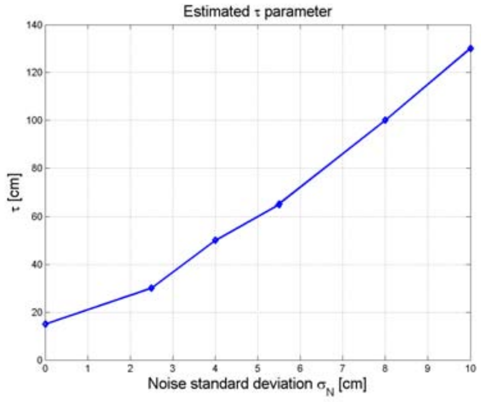


Figure 6.2: Estimation of the parameter τ as a function of the noise standard deviation σ_N.

low-resolution depth image with two different kernels, according to the area characteristics. For flat areas a JBF with a relaxed range parameter $\sigma_{r,FLAT}$ is used in order to have smooth flat regions. On the other hand, for depth edge areas only a Gaussian color intensity difference term with a small and very selective $\sigma_{r,EDGE}$ parameter is applied in order to maintain sharp edges. In fact, depth discontinuities usually coincide with color image intensity edges. The formulation of the WJBF is

$$\begin{aligned} WJBF\left[D_S\right]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q}\in S} \Bigg[& (1-\alpha)\, G_{\sigma_s}\left(\|\mathbf{p}-\mathbf{q}\|\right) G_{\sigma_{r,FLAT}}\left(\left|\tilde{I}_{\mathbf{p}} - \tilde{I}_{\mathbf{q}}\right|\right) + \\ & + \alpha G_{\sigma_{r,EDGE}}\left(\left|\tilde{I}_{\mathbf{p}} - \tilde{I}_{\mathbf{q}}\right|\right)\Bigg] D_{S,\mathbf{q}} \end{aligned} \tag{6.7}$$

where $W_{\mathbf{p}}$ is the normalization factor, and $\tilde{I}$ is the high-resolution color guidance image. Now, the fundamental point is that the filter needs to be able to distinguish between flat and edge areas. For this reason a new weighting factor called α is proposed. This parameter is based on the standard deviation of the sparse depth image.

6.2.3.1 Weighting factor

Edge detection has always been an important discipline in image processing, especially for depth images, where depth edges separate foreground objects from the background. In order to perform a WJBF of the sparse ToF depth D_S, these edge areas firstly have to be found. This task is performed by using a filter that computes the local standard deviation image Σ_S of D_S. More precisely, for each pixel $\mathbf{p}$ of coordinates (u, v) in the sparse image D_S, the standard deviation $\Sigma_S(u, v) \equiv \sigma_S$ is calculated over all its neighborhood non-zero pixels $\mathbf{q}$ within the filter aperture A

$$\sigma_S = \sqrt{\frac{1}{N-1} \sum_{\mathbf{q} \in A} (\mathbf{q} - \bar{\mathbf{q}})^2}, \tag{6.8}$$

where N is the total number of non-zero pixels in A and $\bar{\mathbf{q}}$ is the mean value

$$\bar{\mathbf{q}} = \frac{1}{N} \sum_{\mathbf{q} \in A} \mathbf{q}. \tag{6.9}$$

From the computed standard deviation of D_S, it is possible to derive the new weighting factor α by means of the *three-sigma rule* [42]. This rule states that in a Gaussian distribution, nearly all values lie within 3 standard deviations σ of the mean μ. Specifically, about 68.27% of the values lie within 1σ of the mean, 95.45% of the values lie within 2σ of the mean and almost all (99.73%) of the values lie within 3σ of the mean. A graphical explanation is showed in Figure 6.3. Considering the ToF noise as Gaussian, this rule can be applied for the edge detection in sparse depth D_S. For each standard deviation value σ_S of the image Σ_S the weighting factor α is calculated by comparing σ_S with the standard deviation σ_N of the depth image Gaussian noise. If $\sigma_S < 2\sigma_N$ the depth variation can be considered due to the normal camera noise, hence α is set

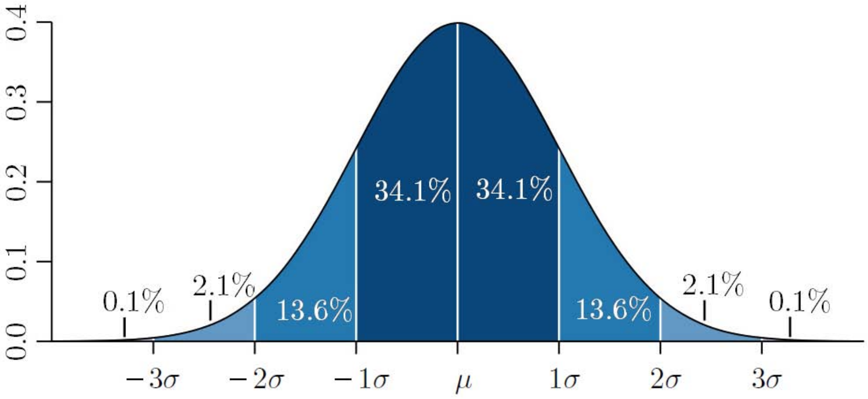
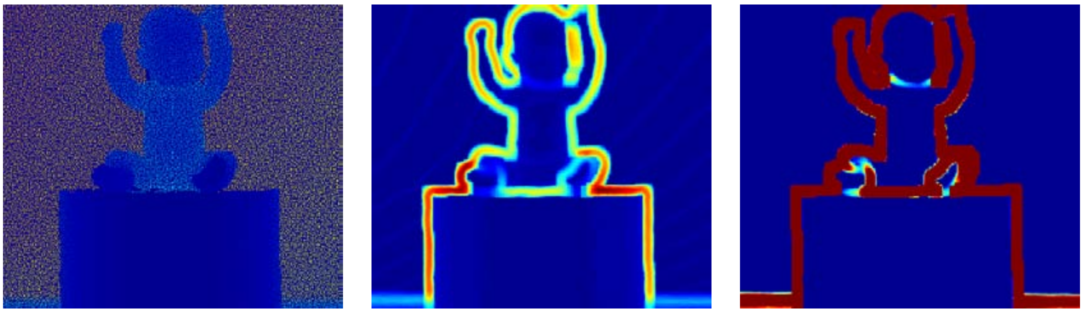


Figure 6.3: Three-sigma rule: most of the values are between $-\sigma$ and σ, almost all of them are no farther than 2σ from μ and there are virtually no observations farther than 3σ from μ [42, 56].



(a) D_S. (b) Σ_S. (c) α.

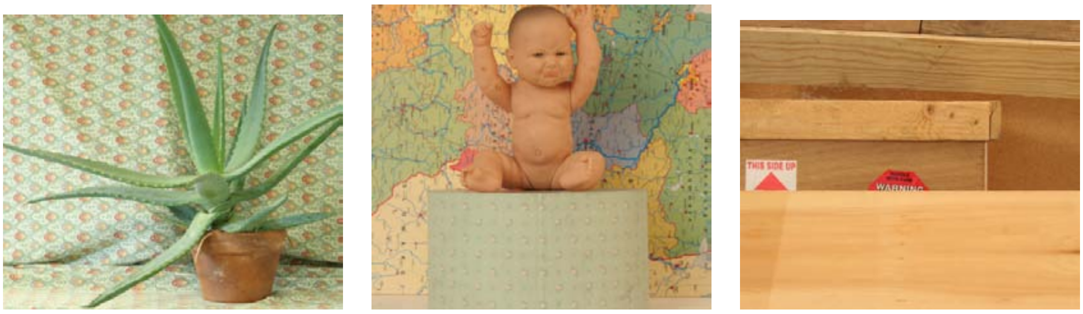
Figure 6.4: Creation of the weighting factor α image: (a) sparse depth D_S, (b) standard deviation image Σ_S, and (c) α image.

to 0. On the other hand, if $\sigma_S > 4\sigma_N$ most likely this high variation is due to a real depth discontinuity, hence α is set to 1. Between $2\sigma_N$ and $4\sigma_N$ the weighting factor α has a linear behavior. Figure 6.4 shows the two steps for the edge detection of a sparse depth D_S.

6.3 Experimental results

All the three up-sampling filters (JBF, JBF$_{\text{KIM}}$ and WJBF) have been tested for the sparse depth map super-resolution using the Middlebury dataset [17], more precisely on the scenes *aloe*, *baby1* and *wood2*. The filters' inputs are the random tile-mapped sparse depth map D_S and the guidance image, which is the color image C (see Figure 6.5). All the inputs images are normalized in the closed interval $[0, 1]$. As in Section 5.4, the starting sparse depth maps D_S are affected by an additive Gaussian noise with 21 different levels of standard deviation $\sigma_N \in \{0,\ 0.5,\ ...,\ 10\}[cm]$. Then, in order to compare the results with the CS reconstruction, the accuracy of the interpolation is evaluated with the MAE of the output dense depth map D_D. The filters' parameters for the super-resolution are:

- **Joint Bilateral Filter:** Filter size 15 × 15 pixels, spatial parameter $\sigma_s = 5$ and range parameter $\sigma_r = 0.03$.
- **Joint Bilateral Filter Kim:** Filter size 15 × 15 pixels, spatial parameter $\sigma_s = 5$ and range parameter $\sigma_r = 0.03$. Regarding the blending function of Equation



(a) *aloe.* (b) *baby1.* (c) *wood2.*

Figure 6.5: The three guidance color images for the joint bilateral filter super-resolution.

(6.6), the parameter $\varepsilon = 0.5$ and τ is set to $15[cm]$. The second value is based on experimental tests performed on the sparse images D_S without noise. With $\tau = 15[cm]$ the detected edge areas correspond to real depth discontinuity.

- **Weighted Joint Bilateral Filter:** Filter size 15 × 15 pixels, spatial parameter $\sigma_s = 5$, range parameter for flat areas $\sigma_{r,FLAT} = 0.1$ and range parameter for edge areas $\sigma_{r,EDGE} = 0.03$. For the calculation of the sparse depth map standard deviation Σ_D the same window size of the filter is used. From Σ_D, the weighting factor α is computed as explained in Section 6.2.3.1; when $\sigma_N = 0[cm]$ the thresholds are calculated using $\sigma_N = 0.5[cm]$.

Figures 6.6, 6.7, and 6.8 show the three scenes up-sampled using the three filters with four different levels of noise. From a visual inspection it can be seen how the proposed WJBF outperforms the other two filters. In fact, the JBF tends to copy textures from the color image to the up-sampled depth and this is increasingly clear as the noise level raises. In like manner, for high noise levels the JBF_{KIM} detect depth discontinuities in flat areas, hence it transfers textures from the guidance color image in the final up-sampled depth. With the WJBF the weighting factor α allows to distinguish between edge and flat areas, hence flat regions are smoother than the other two filters. Moreover, this filter is noise adaptive. Interestingly, for high levels of σ_N, the weighting factor α used for the WJBF can not correctly detect edge areas anymore. In fact, the threshold of $4\sigma_N$ is too high, hence the JBF term for flat area (i.e., the first term of Equation (6.7)) is the dominant term in the depth super-resolution. Anyhow, for these borderline cases the result is an image with slightly blurred edges. The advantage of WJBF over the others filters becomes significantly high when a high noise standard deviation is applied.

The complete numerical results of the experiments are showed in Figure 6.9, which compares the MAE of the up-sampled images D_D as a function of the noise standard deviation σ_N. The curves behaviors shows that $\mathrm{JBF_{KIM}}$ and WJBF always perform better than the normal JBF. Moreover, they give almost the same results for the first 5 levels of noise (i.e., until $\sigma_N = 5[cm]$), and then the WJBF starts to perform better. This is because of the fixed τ in the $\mathrm{JBF_{KIM}}$. With the variable τ, although the performance of the $\mathrm{JBF_{KIM}}$ are improved, from the figure it can be seen that the proposed WJBF still performs best. Finally, also the results of the CS super-resolution are showed in Figure 6.9. Specifically, both the reconstructions with and without ground truth edges (and with the starting pixels recomputation described in Section 5.4.3). For the *aloe* scene, the CS with ground truth edges is the best. This is because this scene has a lot of edges (i.e., 14% out of the 24% of the starting pixels), which are ground truth pixels in the CS reconstruction. In all the others scenes the WJBF gives the best results. In fact, WJBF gives sharp edges and flat areas which are smoother than the CS image.

6.4 Summary

In this Chapter it was presented a sparse depth super-resolution algorithm based on bilateral filter theory. All the experiments have been done using the Middlebury dataset [17]. Like in Chapter 5, the dense depth D_D is reconstructed starting from the sparse D_S, with the 25% of the original depth. Moreover, it was presented a new up-sampling filter, the Weighted Joint Bilateral Filter, and it was compared with the two already existing filters: the JBF of [23] and the $\mathrm{JBF_{KIM}}$ of [22]. Numerical results show that the proposed WJBF outperforms the others for every level of noise. Furthermore, as compared to the compressive sensing super-resolution, the bilateral filter techniques give better results and allow a more general approach. In fact it is possible to combine the low-resolution ToF depth with a single camera image. Moreover, the CS reconstruction is restricted to the ideal scenario, but usually is not always possible to detect the image edges and consequently use the information from the stereo in these areas. The

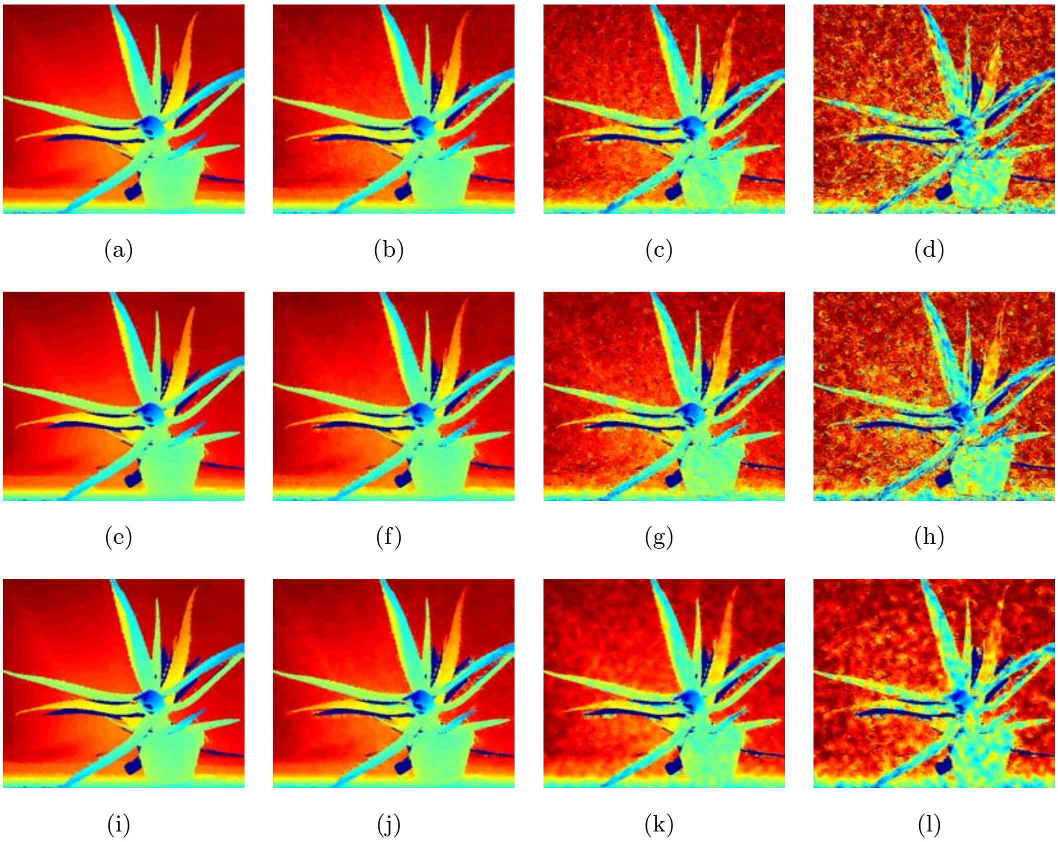


Figure 6.6: Visual comparison of the *aloe* scene super-resolution. The figure shows: (a) JBF $\sigma_N = 0[cm]$; (b) JBF $\sigma_N = 2[cm]$; (c) JBF $\sigma_N = 5[cm]$; (d) JBF $\sigma_N = 10[cm]$; (e) JBF$_{\mathrm{KIM}}$ $\sigma_N = 0[cm]$; (f) JBF$_{\mathrm{KIM}}$ $\sigma_N = 2[cm]$; (g) JBF$_{\mathrm{KIM}}$ $\sigma_N = 5[cm]$; (h) JBF$_{\mathrm{KIM}}$ $\sigma_N = 10[cm]$; (i) WJBF $\sigma_N = 0[cm]$; (j) WJBF $\sigma_N = 2[cm]$; (k) WJBF $\sigma_N = 5[cm]$; (l) WJBF $\sigma_N = 10[cm]$. Blue pixels represent occluded areas.

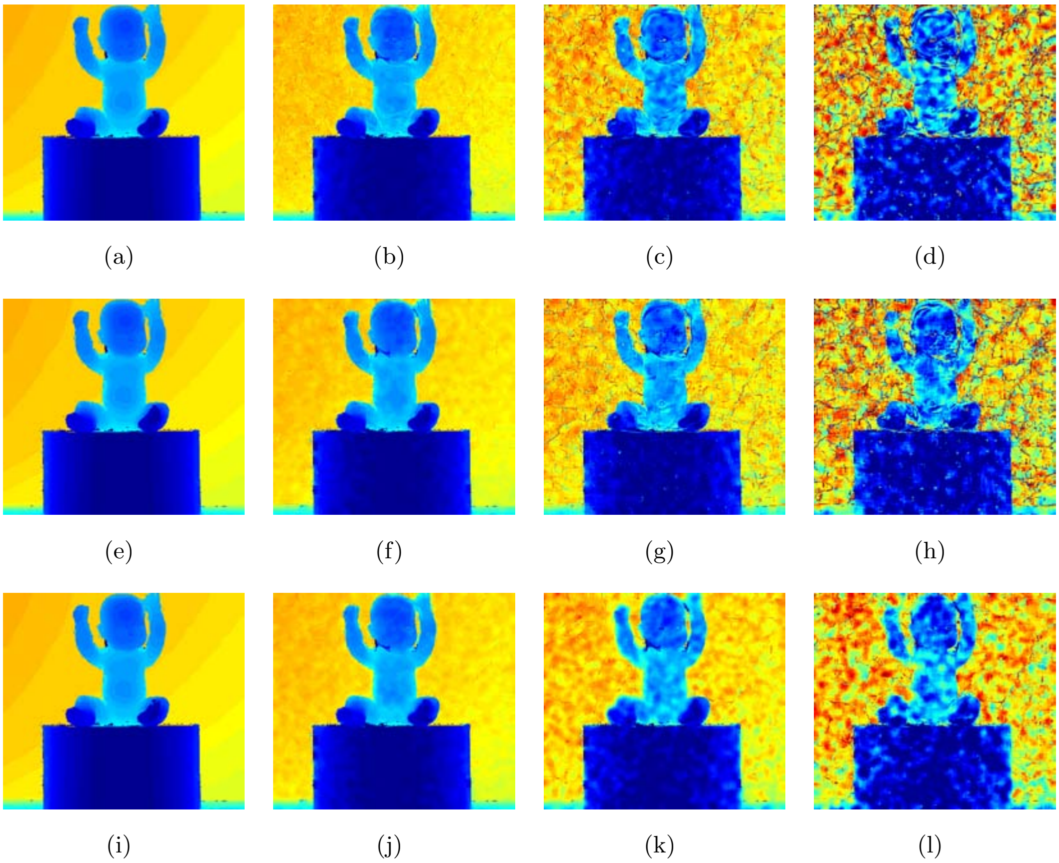
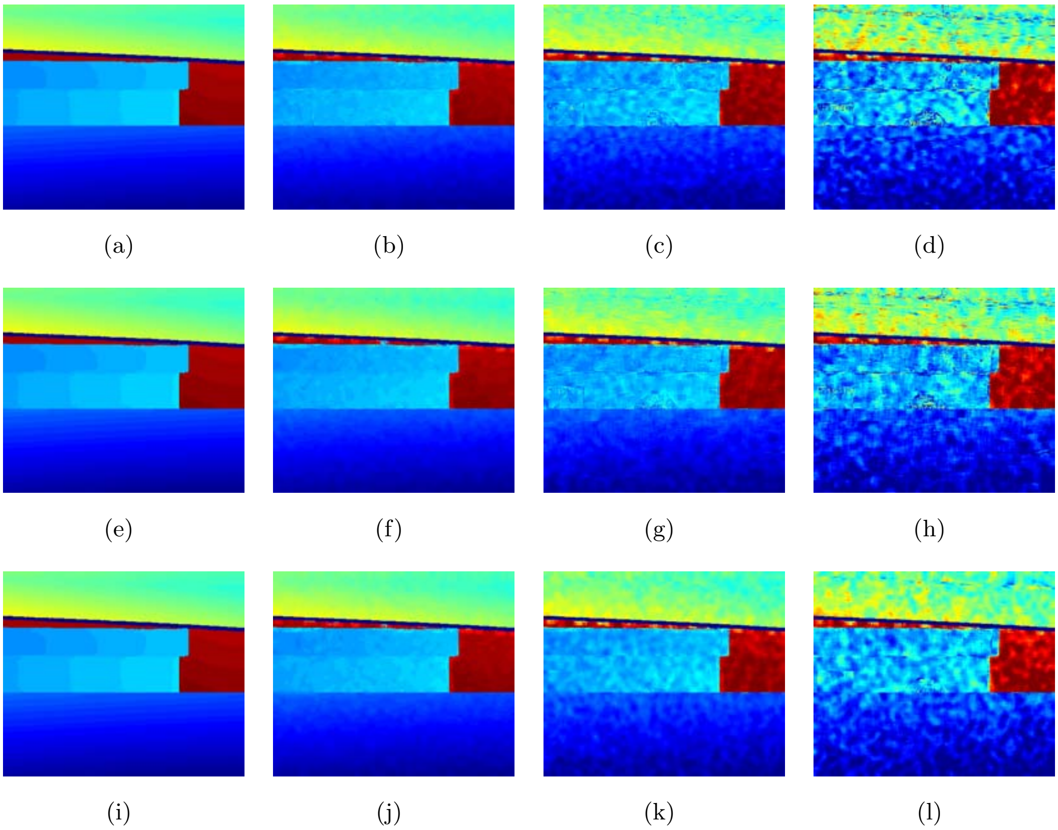


Figure 6.7: Visual comparison of the *baby1* scene super-resolution. The figure shows: (a) JBF $\sigma_N = 0[cm]$; (b) JBF $\sigma_N = 2[cm]$; (c) JBF $\sigma_N = 5[cm]$; (d) JBF $\sigma_N = 10[cm]$; (e) JBF$_{\text{KIM}}$ $\sigma_N = 0[cm]$; (f) JBF$_{\text{KIM}}$ $\sigma_N = 2[cm]$; (g) JBF$_{\text{KIM}}$ $\sigma_N = 5[cm]$; (h) JBF$_{\text{KIM}}$ $\sigma_N = 10[cm]$; (i) WJBF $\sigma_N = 0[cm]$; (j) WJBF $\sigma_N = 2[cm]$; (k) WJBF $\sigma_N = 5[cm]$; (l) WJBF $\sigma_N = 10[cm]$. Blue pixels represent occluded areas.

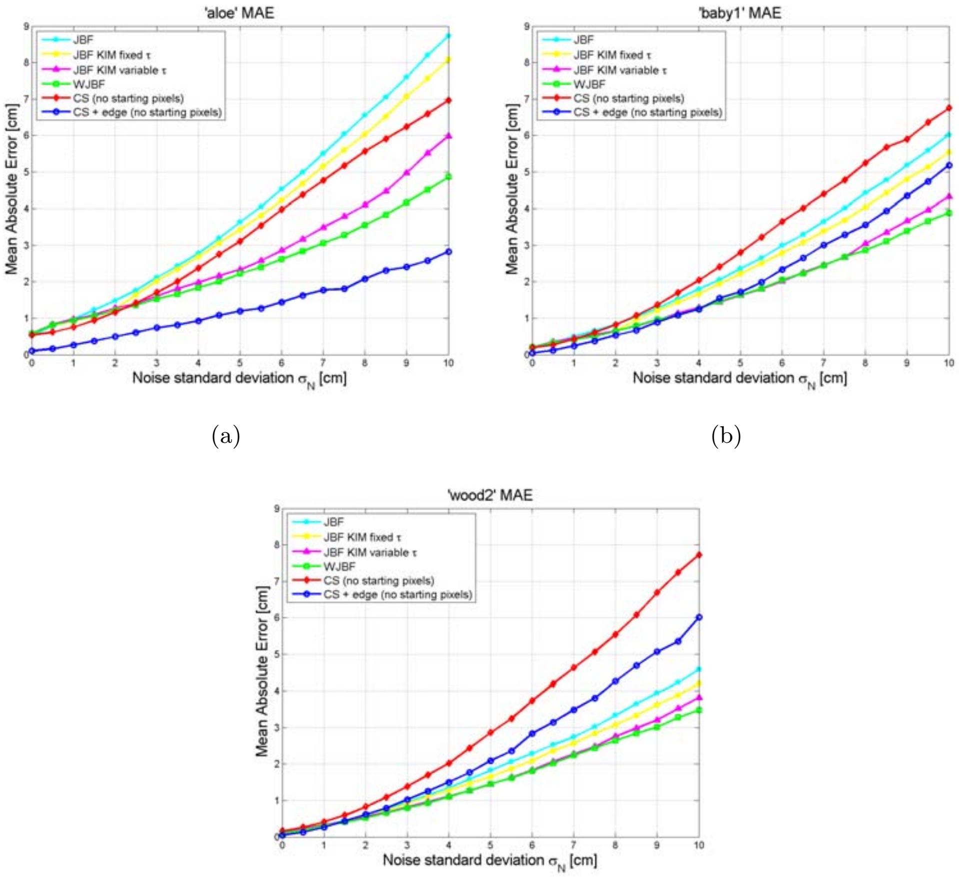


(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

Figure 6.8: Visual comparison of the *wood2* scene super-resolution. The figure shows: (a) JBF $\sigma_N = 0[cm]$; (b) JBF $\sigma_N = 2[cm]$; (c) JBF $\sigma_N = 5[cm]$; (d) JBF $\sigma_N = 10[cm]$; (e) JBF_{KIM} $\sigma_N = 0[cm]$; (f) JBF_{KIM} $\sigma_N = 2[cm]$; (g) JBF_{KIM} $\sigma_N = 5[cm]$; (h) JBF_{KIM} $\sigma_N = 10[cm]$; (i) WJBF $\sigma_N = 0[cm]$; (j) WJBF $\sigma_N = 2[cm]$; (k) WJBF $\sigma_N = 5[cm]$; (l) WJBF $\sigma_N = 10[cm]$. Blue pixels represent occluded areas.



(a)

(b)

(c)

Figure 6.9: Comparison of the super-resolution algorithms JBF, $\mathrm{JBF_{KIM}}$ with fixed τ, $\mathrm{JBF_{KIM}}$ with variable τ, WJBF, CS and CS with ground truth edges. The images show: (a) results for the *aloe* scene, (b) results for the *baby1* scene, (c) results for the *wood2* scene.

execution time is also an important evaluation parameter. The CS is a global and iterative approach which uses the whole image for the reconstruction. On the other hand, the bilateral filtering super-resolution is a local and non-iterative approach. Therefore the latter method requires less time and memory. For all these reasons the bilateral filtering is the approach chosen for the real super-resolution framework.

Chapter 7

ToF Camera Calibration and Characterization

After the development and test of the super-resolution algorithms in the ideal model conditions, the *real model* has to be considered. This means it has to be used a real ToF camera and real video-cameras. Therefore a calibration procedure is needed, since the cameras have different image planes. In this Chapter the procedure for the calibration of the camera rig is described in Section 7.1. Then, Section 7.2 provides a characterization of the ToF camera noise based on some experiments.

7.1 Camera calibration

Camera calibration refers to the problem of recovering the external and internal geometry of an optical acquisition device in order to have a complete description of its image formation process and, therefore, to be able to make accurate 3D measurements from 2D images. The external geometry of a camera is defined by the rotation matrix $\mathtt{R}$ and the translation vector $\mathbf{t}$, which relates the camera orientation and position to the world coordinate system. The internal geometry refers to the camera calibration matrix $\mathtt{K}$ which describes the geometric aberrations produced by the lens system of the camera. Summarizing, the target of camera calibration is to estimate all these parameters describing the cameras imaging process.

7.1.1 Calibration procedure

For the calibration, the Sony's toolbox from [49] has been used. The toolbox ensures to calibrate the stereo camera and the ToF camera simultaneously within subpixel and submillimeter accuracy, respectively, by using a 2.5D dot pattern for the intrinsic and the extrinsic calibration. The used 2.5D dot calibration pattern is a modification of the planar one proposed in [15] and it is represented in Figure 7.1. This planar pattern is composed by 64 dots, uniquely placed in order to ensure their correct identification. In fact, each dot can be identified by observing the closest neighbor dots. The dot pattern is suited for the standard camera calibration, but it does not work with low-resolution ToF camera amplitude images (used for the calibration) which have a strong light fall-off to the image border [49]. For this reason a 2.5D pattern was used, which consists on holes instead of dots. The 2.5D dot pattern has a size of 800 × 600$[mm]$, whereas each hole has a diameter of 40$[mm]$ in order to ensure the IR-rays of the ToF cameras's illumination unit passing through the hole. For the rest, the 2.5D dot pattern has the same properties as the regular dot pattern. After the acquisition of the pattern images, which had to cover the whole FoV of each camera, the cameras were intrinsically calibrated. Eventually, the whole camera rig is calibrated extrinsically by finding the intersection of the already known correspondence point sets. For more details see [49].

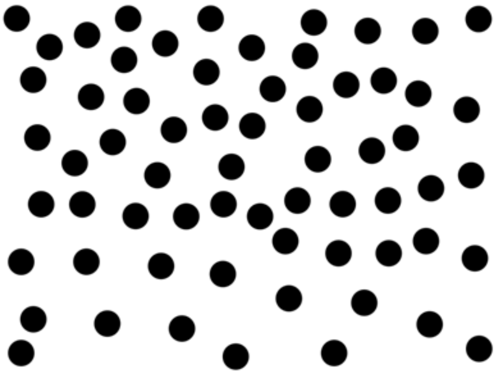


Figure 7.1: The planar dot pattern with 64 uniquely placed black dots.

7.1.2 Camera rig

Figure 7.2 (a) shows the scene in front of the camera rig. The background and the tables are covered with black sheets. Moreover, additional lights are used to illuminate both the scene and the calibration pattern. Figure 7.2 (b) shows the camera rig used for the acquisition. The rig is composed by the PMD CamCube 3.0 ToF camera indicated by ④, and three *Lux Media Plan* (*LMP*) *HD1200* [28] CMOS video-cameras indicated by ①, ②, ③. The ToF camera is positioned on the top of the central standard camera ②. All the cameras are synchronized by an hardware trigger.

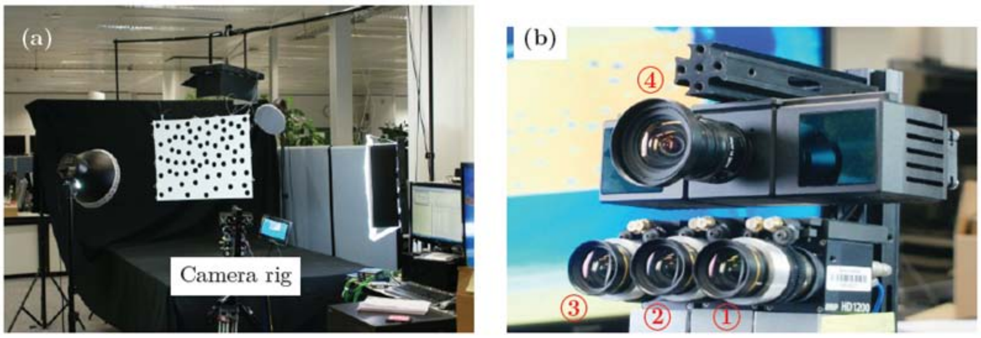


Figure 7.2: Scene in front of the camera rig (a), which is mounted on a tripod. The camera rig (b), composed by the ToF camera and the three standard cameras [49].

7.2 ToF camera noise characterization

As stated in [38], the CamCube 3.0 has a measurement precision of some centimeters. The noise of the ToF camera can be modeled to be Gaussian [33] with standard deviation given by Equation (2.13), and this standard deviation has been estimated with some experimental measurements. A flat panel was placed parallel in front of the ToF camera. More precisely, it was used the planar dot pattern of Figure 7.1 (without holes), since it has both white and black areas which reflect the light differently. In this way it is possible to have a more reliable noise estimation, as black and white areas lie at the same depth but give different depth measurements. The acquisition was carried out at six distances between $0.5[m]$ and $2.5[m]$ with a step of $0.5[m]$ in order to estimate the noise standard deviation as a function of the distance. A higher range of measurements was not possible because of the space limitations in the laboratory. According to [38], before starting the acquisitions a warm-up period of about one hour was performed for the ToF camera, in order to obtain distance measurement stability. After the acquisitions a square central area in the center of the panel was selected for each distance. Moreover, since the camera and the panel cannot be perfectly parallel, the depth values

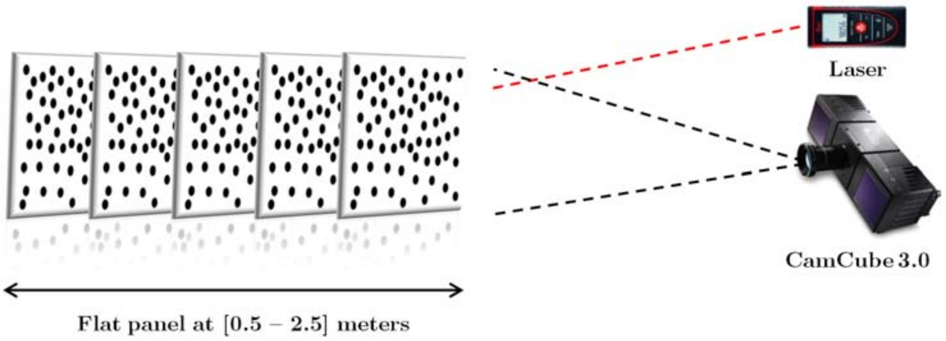


Figure 7.3: Setup for ToF noise standard deviation estimation: ToF camera, laser meter and the flat panel at several distances between $0.5[m]$ and $2.5[m]$.

in the selected area have been pre-processed with the Matlab function *detrend*. This function removes the mean value or the linear trend from the matrix representing the panel area. In this way, errors due to the non-perfect parallelism between camera and panel are eliminated. After the data detrending, the standard deviation of the selected area is computed. The noise standard deviation $\sigma_{ToF}(d)$ behavior is showed in Figure 7.4. From the figure it can be seen that for distances in the range $0 \div 1[m]$ the standard deviation is around $10[cm]$, probably because of the multi-path propagation already described in Section 2.4.8. Therefore, for acquisitions in this range the ToF camera is completely unreliable. After one meter of distance, σ_{ToF} decreases until less than $1[cm]$, then it slightly increases as the distance raises.

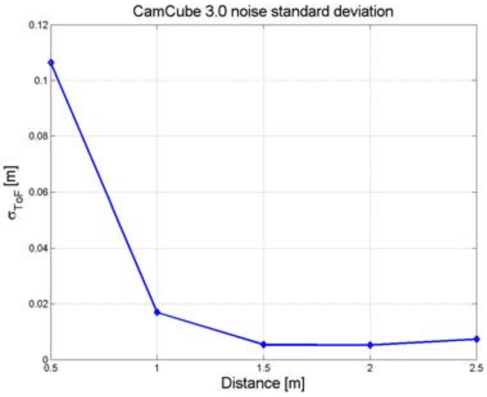


Figure 7.4: Standard deviation $\sigma_{ToF}(d)$ as a function of the distance d.

During the acquisitions, for each distance a ground truth distance value camera-panel was measured by using a *laser distance meter* (Leica DISTO X310 [26]) placed on the edge of the ToF main body. This laser distance was compared with the ToF value and an offset measurement was founded. Furthermore, in order to test the ToF's SBI effectiveness, the acquisitions have been done both with and without additional lights. Numerical results in Table 7.1 shows that the depth value do not depends on the illumination conditions. By interpolating the offset values it is possible to correct the ToF measurement for each distance in the range $1 \div 2.5[m]$. Regarding the range $0 \div 1[m]$, as previously said, the measured depth is not reliable hence the offset is not considered.

Laser distance $[m]$	**ToF distance (light off)** $[m]$	**ToF distance (light on)** $[m]$
0.52	0.38	0.38
0.99	1.13	1.13
1.49	1.59	1.60
1.99	2.08	2.08
2.49	2.59	2.59

Table 7.1: ToF and laser depth measurements comparison. Light on means an illumination intensity value on the panel from 2100 to 420 lux (depending from the distance to the light sources), whereas light off means about 20 lux of illumination intensity.

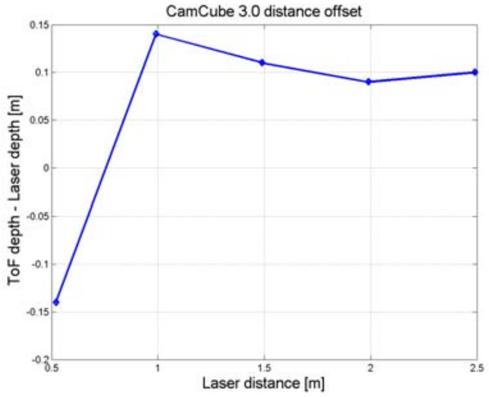


Figure 7.5: CamCube 3.0 distance offset: difference between the ToF measured depth and the ground truth laser depth.

Chapter 8

ToF Depth Super-Resolution

The goal of this thesis is the super-resolution of the 200 × 200 depth image provided by the ToF camera. Figure 8.1 gives an overview of each step in the super-resolution framework. Firstly, the offline cameras calibration procedure described in Section 7.1 is carried out in order to obtain the intrinsic and the extrinsic parameters of all the four cameras in the rig. Then, the sensors fusion for the depth super-resolution is accomplished. This procedure increases the depth resolution and refines it up to the video-camera resolution, by combining information from both the cameras. The first step presented in Section 8.1 is the pre-processing of the ToF raw depth image, the purpose of which is to reduce the camera noise and detect the flying pixels. Section 8.2 describes the pre-processing for the three video-camera color images. Then, Section 8.3 presents the mapping from the ToF to one of the video-cameras. The fusion between the two sensors is explained in Section 8.4. This allows to reach a depth image with a resolution of 1920 × 1080 pixel. In Section 8.5 is proposed a post-processing stereo

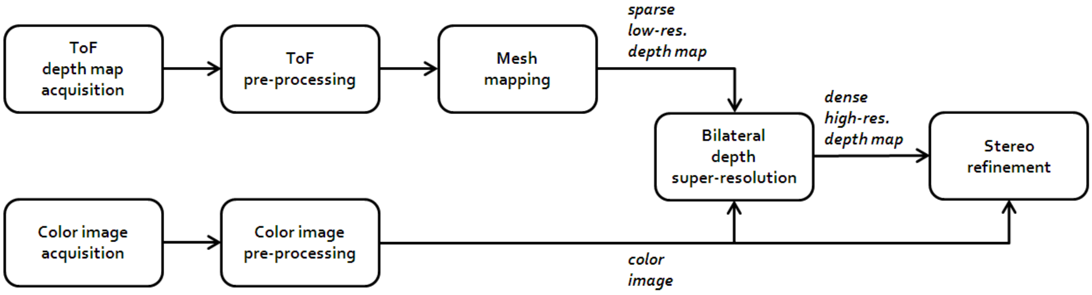

ToF depth map acquisition
ToF pre-processing
Mesh mapping
sparse low-res. depth map
Bilateral depth super-resolution
dense high-res. depth map
Stereo refinement
Color image acquisition
Color image pre-processing
color image

Figure 8.1: Overview of the ToF depth super-resolution framework.

technique for the final super-resolved depth refinement. Section 8.6 shows some results. Concluding, the possible applications of the super-resolution algorithm are discussed in Section 8.7.

8.1 ToF pre-processing

The ToF depth image pre-processing is the first step of the super-resolution framework. A detailed description of this Section is provided in Figure 8.2. Firstly, the *undistortion* of the raw 200 × 200 ToF depth is applied in order to remove the errors introduced by the camera lens. The undistortion is a fundamental operation in the ToF data pre-processing, especially because the used sensor has a wide FoV of $60° \times 60°$. The undistortion is done by using the radial distortion coefficients estimated with the calibration procedure and the MATLAB toolbox from Bouguet [2]. Moreover, the estimated offset correction is also applied. After this, the undistorted image is filtered with a *bilateral filter* (Equation (6.1)), which uses a range standard deviation parameter σ_r adaptively selected with the depth. Specifically, $\sigma_r = 3\sigma_{ToF}(d)$, where $\sigma_{ToF}(d)$ is the ToF noise standard deviation estimated at the distance d (see Section 7.2). In fact, as explained in Section 6.2.3.1, if the depth variation inside the filter aperture is under $3\sigma_{ToF}$ is not possible to differentiate between depth discontinuity or noised flat area. This approach allows to reduce the noise of the depth image and at the same time preserve depth discontinuities. Figure 8.3 shows the effect of undistortion and bilateral filtering onto the raw ToF depth of the acquired scene, from now called *Pyramid*. The last step of the ToF pre-processing is the *flying pixels detection*. The standard deviation of the filtered depth is calculated, then flying pixels are detected with the procedure described in Section 6.2.3.1. Only the pixels having a standard deviation higher than $4\sigma_{ToF}$ are classified as flying pixels. Then the binary flying pixels map is eroded of

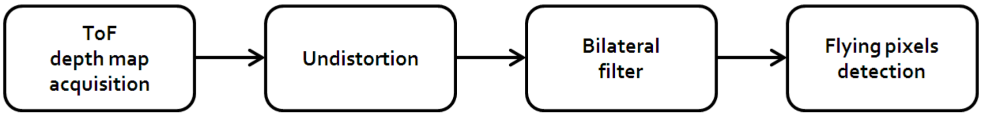


Figure 8.2: Scheme of the ToF depth pre-processing to correct lens distortion, reduce the noise, and find the flying pixels.

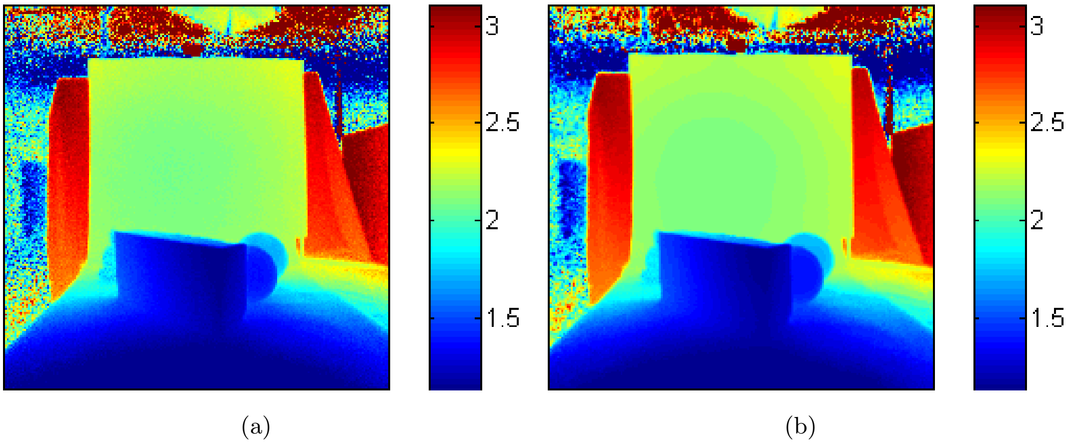


Figure 8.3: *Pyramid* scene ToF depth image pre-processing: (a) is the raw ToF depth image; (b) is the image after undistortion and bilateral filtering. Note: scale in meters.

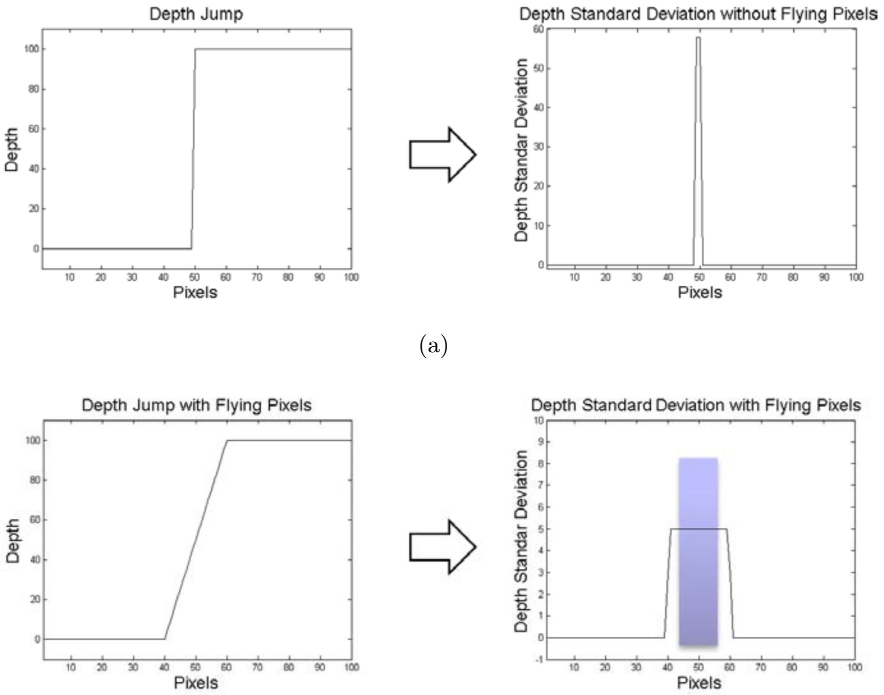
one pixel, in order to select only the flying pixels. In fact, in case of an ideal depth discontinuity the peak of the standard deviation is very narrow (see Figure 8.4(a)). On the other hand, a depth discontinuity in a real ToF image presents the trend showed in Figure 8.4(b), hence the resulting standard deviation peak is much larger. Then, the region corresponding to the flying pixels area can be detected as the two peaks difference, which corresponds to the erosion process.

8.2 Color image pre-processing

As for the ToF camera, also the RGB images from the video-cameras needs an undistortion correction. Therefore, the MATLAB toolbox from Bouguet [2] used for the undistortion of the ToF depth image is applied to each camera with the corresponding radial distortion coefficients. In this case, the resulting image is almost equal to the original, due to the high quality of the used video-cameras. Figure 8.5 shows the resulting RGB color image.

8.3 Mesh mapping

Once the calibration is made, the intrinsic and extrinsic parameters of each camera are available. Therefore, it is possible to project all the pixels from the pre-processed ToF



(a)

(b)

Figure 8.4: Comparison between the standard deviation of an ideal depth discontinuity (a), and the standard deviation of a ToF depth discontinuity (b).

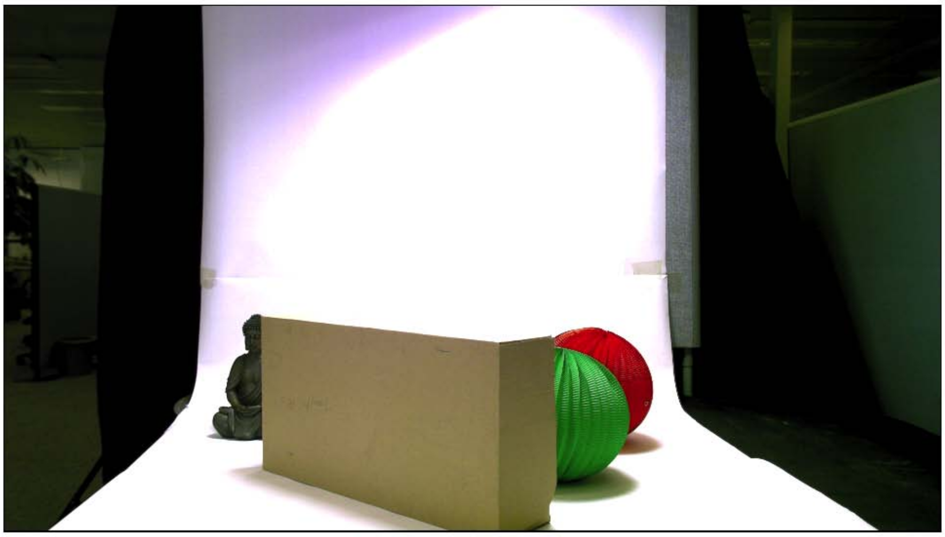


Figure 8.5: *Pyramid* scene high-resolution RGB image from the central video-camera.

depth onto each video-camera image plane. Let (u, v) be the coordinates in the ToF image plane of the pixel $\mathbf{p_{ToF}}$, which has the depth value z_{ToF}. In order to find the corresponding coordinates onto the video-camera image plane the pixel $\mathbf{p_{ToF}}$ is first projected in world coordinates by de-normalizing its coordinates (multiplication by z_{ToF}) and by using the inverse of the ToF camera calibration matrix ${\mathtt{K_{ToF}}}^{-1}$. Through this, the ToF pixel becomes a 3D point in the space consisting on the homogeneous coordinates $\mathbf{P} = (x, y, z, 1)^\top$ referred to the ToF camera coordinate system. Later, the point $\mathbf{P}$ is transformed into the coordinate system of the video-camera by using the roto-translation matrix $[\mathtt{R}|\mathbf{t}]$ between the ToF and the video-camera. Finally, the color image coordinates (x_{RGB}, y_{RGB}) are obtained by projecting the 3D point onto the video-camera image plane. This last step is done by using the video-camera calibration matrix $\mathtt{K_{RGB}}$ and normalizing the resulting coordinates into homogeneous coordinates. The entire mapping procedure is summarized in the matrices multiplication:

$$\begin{pmatrix} x_{RGB} \\ y_{RGB} \\ 1 \end{pmatrix} = \begin{pmatrix} x'_{RGB}/z_{RGB} \\ y'_{RGB}/z_{RGB} \\ 1 \end{pmatrix} = \mathtt{K_{RGB}}\,[\mathtt{R}|\mathbf{t}] \begin{bmatrix} {\mathtt{K_{ToF}}}^{-1} \begin{pmatrix} u \cdot z_{ToF} \\ v \cdot z_{ToF} \\ z_{ToF} \end{pmatrix} \\ 1 \end{bmatrix}. \tag{8.1}$$

Let assume a mapping from the ToF camera ④ to the color image of the central camera ② (from now called reference camera) of the rig in Figure 7.2. Since cameras ④ and ② have different viewpoints, each one can see into areas which are occluded in the other camera's view. Hence, the simple mapping from ToF to the reference camera leads to errors due to the sparsity of the projected pixels. In the case of cameras ④ and ②, the main problem is due to the big vertical occlusion, which causes the overlapping between background areas which are visible only from the ToF, and foreground areas which are visible from both ToF and camera ②. This problem has been solved by the introduction of the *mesh mapping*. Starting from the top-left pixel of the 200 × 200 ToF depth, a square mesh of four pixels is selected (blue pixels in Figure 8.6). Then the four pixels are mapped onto the camera ② image plane in the form of a mesh where the central area is filled with the mean value of the four vertices (red mesh in 8.6). After the mapping, all the meshes which are composed by at least two flying pixels are eliminated. The final result is a *sparse depth map* with the FoV of the chosen reference camera ②. Because of the different FoVs not all the ToF pixels are mapped inside the color image. This high-resolution sparse depth map has to be refined. To do this, in the next Section it is explained a method which exploits the correlation between color image and mapped depth.

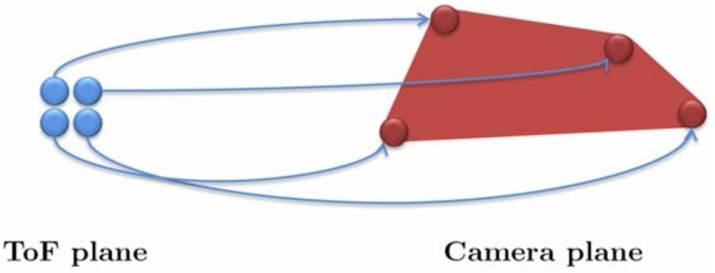


Figure 8.6: Graphical representation of the mesh mapping: four neighbors ToF pixels (blue pixels) are mapped onto the reference camera image plane as a mesh (red mesh).

8.4 Bilateral depth super-resolution

The super-resolution block is the core of the proposed algorithm. The inputs of this Section are the sparse 1920 × 1080 mapped depth map and its corresponding 1920 × 1080 color image. From these, it is possible to refine the sparse depth map by using the *weighted joint bilateral filter* of Equation (6.7), which exploits the information of the guidance image $\tilde{I}$. The output of the WJBF is the refined 1920 × 1080 depth map, called *dense depth map*. Figure 8.7 shows the resulting super-resolved depth image obtained from the low-resolution ToF depth of Figure 8.3. Please note the blue areas on the object borders. These pixels are still undetermined after the super-resolution. The reason and the solution to this issue will be explained in the next Section.

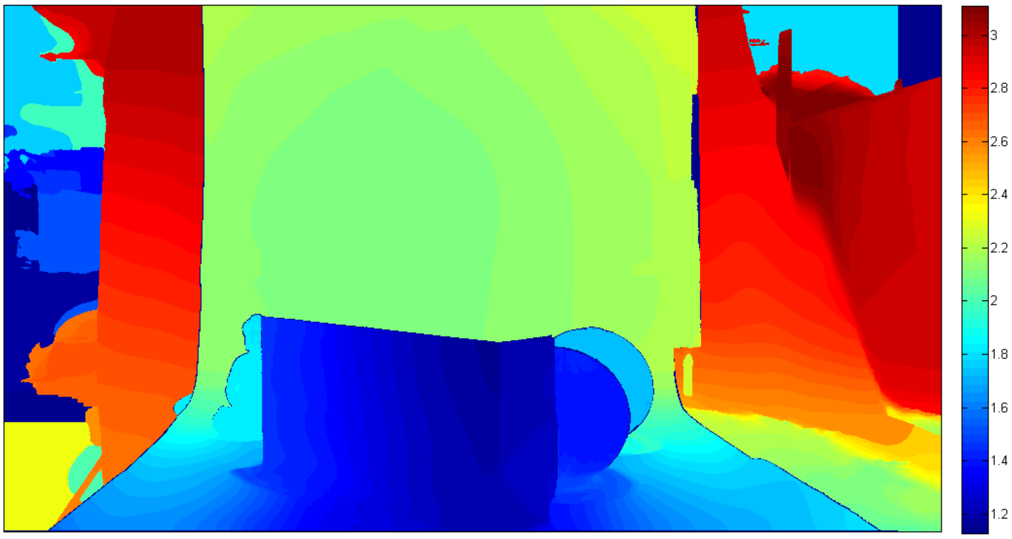


Figure 8.7: *Pyramid* scene super-resolved depth image referred to the central camera (scale in meters).

In addition to the WJBF, the code allows to choose between the JBF of Equation (6.4) and the $\mathrm{JBF}_{\mathrm{KIM}}$ of Equation (6.5) for the sparse depth map interpolation.

8.5 Stereo refinement

The last part of the super-resolution framework is the *stereo refinement*. This is an optional refinement proposed in order to exploit the information coming from the additional video-cameras. The stereo refinement leads to fill the pixels that are still undetermined after the super-resolution. In fact, during the interpolation of some pixels (usually in edge areas) it could happen that the filter cannot find similar color intensity values within its aperture. Therefore, if the filter's coefficients lies under a certain threshold, the filter will not interpolate the pixel. The idea is to fill these undetermined areas by using an approach similar to the local stereo matching described in Section 3.4.1. A graphical explanation of this post-processing procedure is provided in Figure 8.8. Specifically, the purpose of the algorithm is to try to find, for each undetermined pixel $\mathbf{p_R}$ with coordinates (x_R, y_R) in the 1920 × 1080 reference color image, its conjugate $\mathbf{p_L}$ in the 1920 × 1080 target color image provided by another video-camera (e.g., from reference camera ② to target camera ①). Differently from the normal local stereo approaches, the search is limited to the comparison between a window (or block) in the reference image and only few possible windows in the target image. These target windows are defined from the mapping of $\mathbf{p_R}$ onto the target image by using Equation (8.1). Since a depth value z_{ToF} is needed to use this formula, the mapping is performed by assigning to $\mathbf{p}$ all the possible depths $z_1, z_2, ..., z_8$ from the neighbors pixels in a window centered in $\mathbf{p}$ (see Figure 8.8). More precisely, only the unique and available depth values are considered. Then, the similarities between the reference block and all the target blocks are calculated with the SSD (Equation (3.23)). The target block with the lowest SSD gives the match, hence the depth value used to find that block is assigned to the undetermined pixel $\mathbf{p}$. The procedure is applied to all the undetermined pixels.

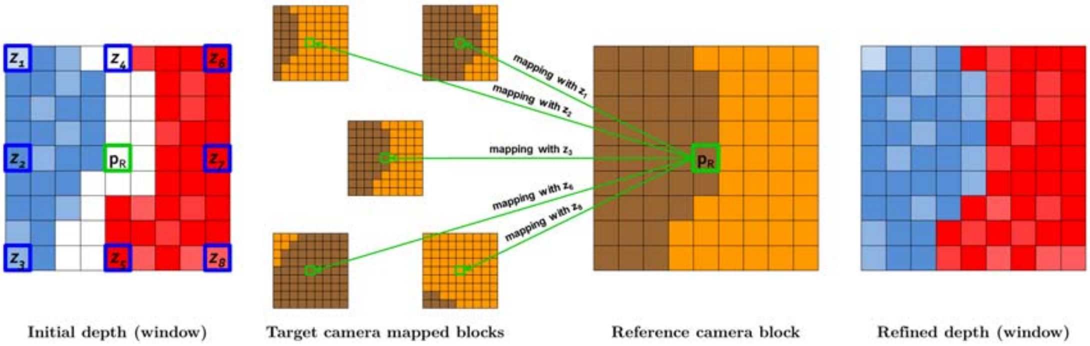


Figure 8.8: Stereo refinement. From left to right: depth map window in an edge area (undetermined pixels in white); mapping of the reference camera block with the five unique and available depth values (in this case z_4 is undetermined, z_5 and z_7 are duplicate depths); refined depth map window.

This final post-processing refinement is used only if a second video-camera is available. Otherwise, an alternative refinement can be done by simply applying a JBF to the super-resolved depth. This single-camera post processing is called *bilateral refinement*. Concluding, the final result is a high-resolution depth map, with smooth flat areas due to the filtering, and a good fitting between depth edges and image edges as a result of the color image usage in the super-resolution algorithm. Figure 8.9 shows the 1920 × 1080 super-resolved depth for the *pyramid* scene.

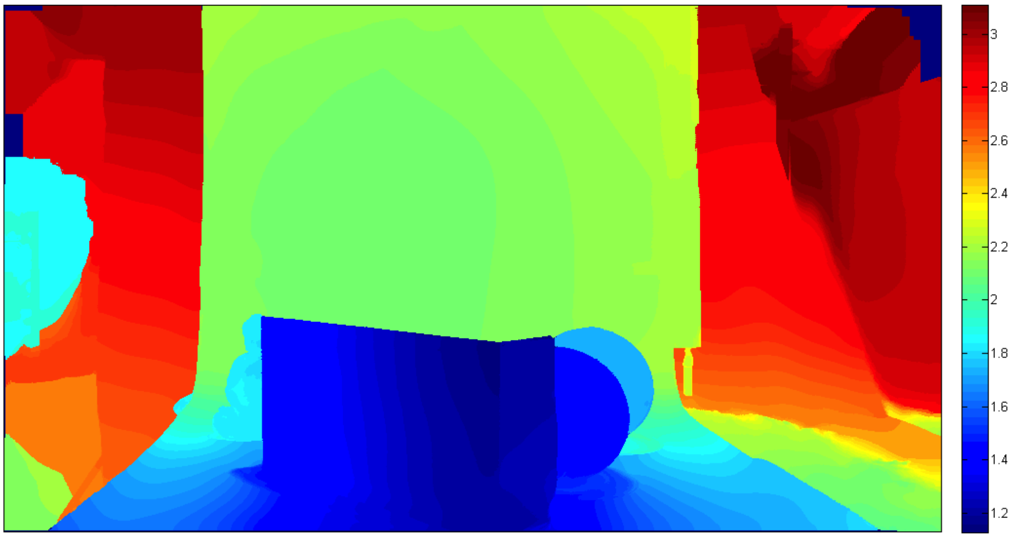


Figure 8.9: *Pyramid* scene super-resolved depth image after the stereo refinement (scale in meters).

8.6 Experimental results

In this Section some results are presented. Different scenes have been acquired with the camera rig of Figure 7.2. Some scenes have a black background, whereas others have a white background. The super-resolution is done using the WJBF. For each scene are provided:

- The 200 × 200 raw ToF depth image.
- The 1920 × 1080 super-resolved depth image.
- The 1920 × 1080 reference video-camera color image.

8.6.1 Pyramid

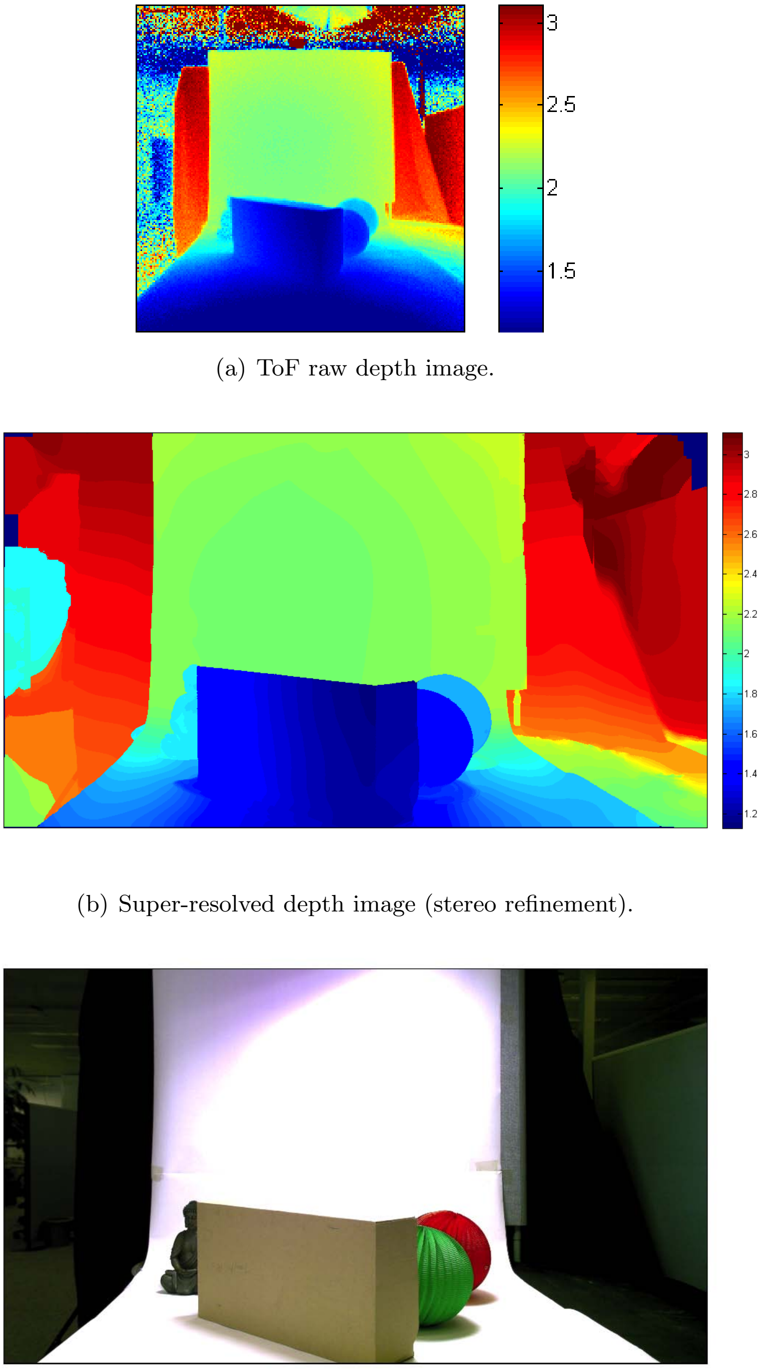
The super-resolution of this image was already showed during the description of the algorithm, in particular for the case of WJBF. In this scene it is possible to see that the reconstruction of all the objects is good, the depth discontinuities are well defined and without flying pixels. One problem of the super-resolution algorithm can be noticed looking to the top of the brown box. In fact, because of the additional illumination, there is a color saturation of this part of the box, which results white as the background. Therefore the filter assigns the background depth values to this area. Nevertheless, looking at the ToF raw data (Figure 8.10(a)) it can be seen that the top part of the box has a width of only 3 pixels, which are all flying pixels. Therefore the ToF information of this area is not reliable and further information are necessary. Figure 8.10 shows the result of the WJBF super-resolution presented before.

8.6.2 Elk

The *Elk* scene shows a box with an elk toy model on its top. The result of the super-resolution is accurate, especially on the elk's horns and scarf. Like for the other acquisitions all the area around the background is out of range for the ToF camera. Therefore, this part should not be considered for the evaluation. Only a small error is visible, between the elk's legs. This is due to the stereo refinement. In fact, after the super-resolution the background area between the two legs is undetermined, but the pixels in the blocks used by the stereo refinement have only foreground values. Therefore the area is filled with these values. Figure 8.11 shows the result of the WJBF super-resolution.

8.6.3 Objects

The *Objects* scene represents some objects with a black background. With this configuration it is possible to see that the head of the Buddha is not perfectly reconstructed. This because of the dark colors of the Buddha, which are quite similar to the black background. Anyhow, looking at the raw ToF data, it can be seen that the Buddha's head is composed by only few pixels, and after the flying pixels removal almost all the depth information in that area is lost. As for the other scenes, Figure 8.12 shows the result of the WJBF super-resolution.

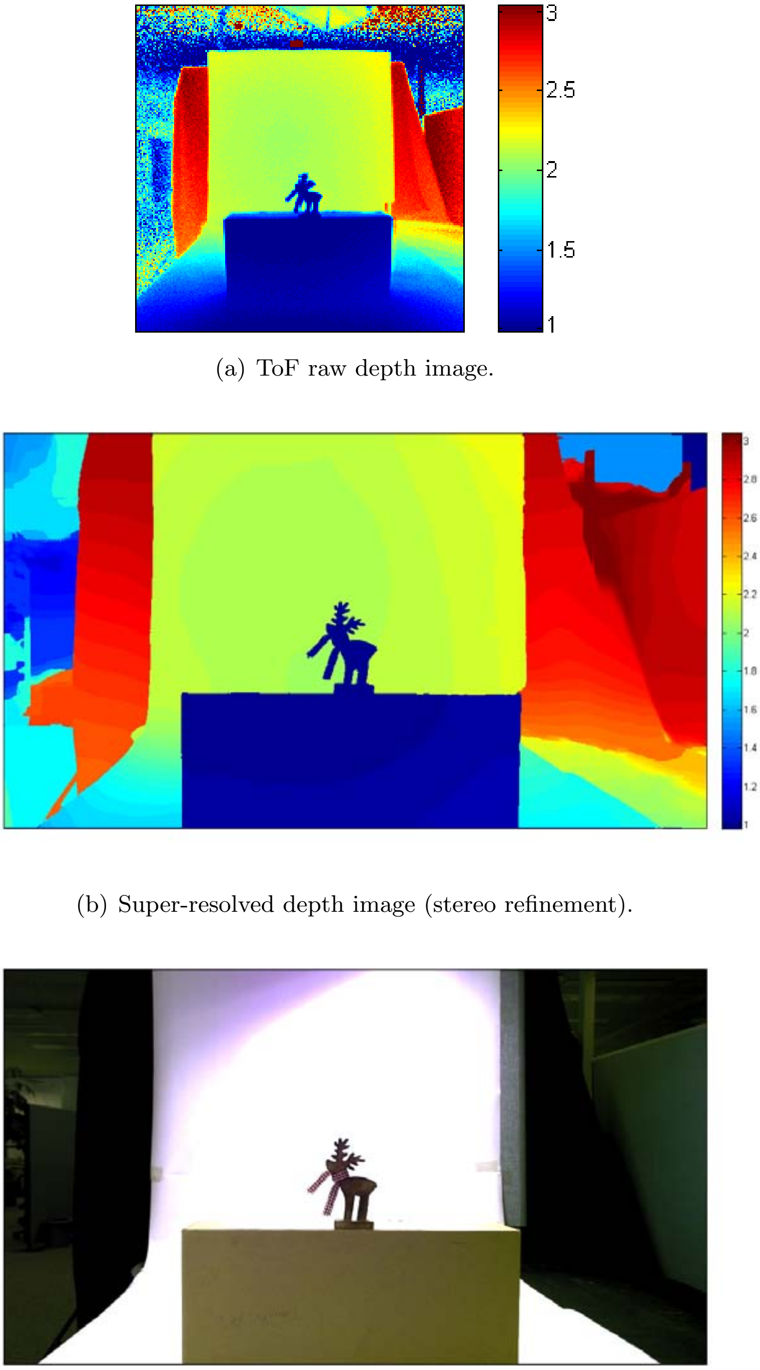


(a) ToF raw depth image.

(b) Super-resolved depth image (stereo refinement).

(c) Color image.

Figure 8.10: Super-resolution of the *Pyramid* scene (scale in meters).

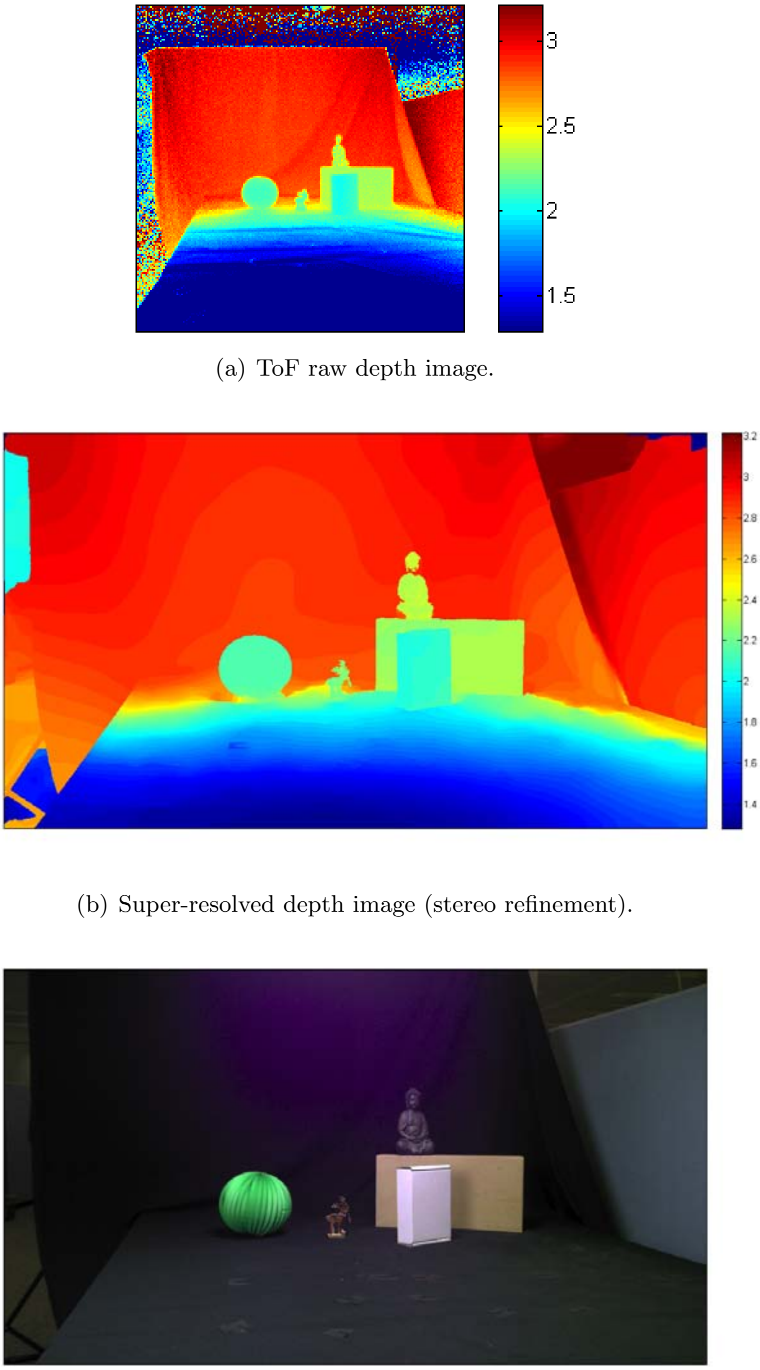


(a) ToF raw depth image.

(b) Super-resolved depth image (stereo refinement).

(c) Color image.

Figure 8.11: Super-resolution of the *Elk* scene (scale in meters).



(a) ToF raw depth image.

(b) Super-resolved depth image (stereo refinement).

(c) Color image.

Figure 8.12: Super-resolution of the *Objects* scene (scale in meters).

8.7 Applications

The presented super-resolution algorithm allows to obtain a high-resolution depth image starting from a low-resolution ToF depth. One possible application is to use this high-resolution depth for *scene segmentation*. Scene segmentation is the problem of identifying all the different elements in a scene. The main drawback of image segmentation is that the information carried by a single image may not suffice to completely understand the scene structure. Therefore, with the fusion of depth and color image it is possible to overcome this problem and enhance the existing segmentation techniques. An example of depth image usage for segmentation purposes can be found in [33].

The super-resolved depth image can be further improved by using a stereo disparity approach. As explained in Section 7.2, the standard deviation of the ToF camera noise is about $1[cm]$. Therefore, the depth resolution of the acquired image is about few centimeters. Moreover, with the bilateral approach the final depth resolution can be even worse. On the other hand, as previously explained in Section 4.3 the depth resolution of stereo vision systems depend on the distance, and for closer objects is better than ToF. Therefore, it can be possible to identify, for each distance, which system has the most accurate depth resolution. For areas where stereo is more precise than ToF, stereo matching should be used to refine the disparity. Then, for background areas where ToF has a better depth resolution, the disparity value from the super-resolved depth should be considered. The textures level is also an important indicator, since for textureless areas a stereo system cannot give a reliable disparity estimation, and the information from ToF should be used.

Chapter 9

Conclusion

The aim of this thesis was to combine the information from a ToF camera with a stereo vision system in order to obtain high-resolution depth images. Two super-resolution approaches have been developed: the first based on compressive sensing, the second based on joint bilateral filtering. The two approaches have been tested in an ideal scenario, with a perfect mapping between ToF and video-camera, and a Gaussian noise for the ToF camera measurement. The performed experiments showed that the bilateral filtering approach was the most suitable for the super-resolution purposes. Therefore this method was chosen for the real ToF depth image up-sampling. The entire super-resolution framework was presented. The first part is a camera rig calibration to obtain the intrinsic and extrinsic parameters. Then a ToF depth image pre-processing procedure based on some camera noise measurements is performed, in order to reduce the noise of the acquired depth map and to detect the flying pixels. Finally, it is possible to increase the ToF resolution up to 1920 × 1080 pixel by exploiting the correlation between the depth and the color image from one video-camera. The final result is a high-resolution depth image, with a strong reduction of the noise thanks to the filtering procedure, and sharp edges thanks to the additional information coming from the color image.

Bibliography

[1] M. A. Albota, R. M. Heinrichs, D. G. Kocher, D. G. Fouche, B. E. Player, M. E. O'Brien, B. F. Aull, J. J. Zayhowski, J. Mooney, B. C. Willard, and R. R. Carlson. **Three-dimensional imaging laser radar with a photon-counting avalanche photodiode array and microchip laser**. *Appl. Opt.*, **41**(36):7671–7678, December 2002. 4

[2] J.-Y. Bouguet. **Camera Calibration Toolbox for Matlab** [online]. Available from: `http://www.vision.caltech.edu/bouguetj/calib_doc/`. 27, 84, 85

[3] B. Büttgen, T. Oggier, M. Lehmann, R. Kaufmann, and F. Lustenberger. **CCD/CMOS lock-in pixel for range imaging: Challenges, limitations and state-of-the-art**. In *Proceedings of the First Range Imaging Research Day*, ETH Zurich, 2005. 4, 6, 9, 13, 14

[4] B. Büttgen and P. Seitz. **Robust Optical Time-of-Flight Range Imaging Based on Smart Pixel Structures**. *IEEE Transactions on Circuits and Systems*, **55**(6):1512–1525, 2008. 11, 13

[5] J. Canny. **A Computational Approach to Edge Detection**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **PAMI-8**(6):679–698, 1986. 48

[6] **Dense Disparity Maps from Sparse Disparity Measurements original Matlab code** [online]. Available from: `http://www.gol.ei.tum.de/index.php?id=25`. 50

[7] S. Foix, G. Alenya, and C. Torras. **Lock-in Time-of-Flight (ToF) Cameras: A Survey**. *Sensors Journal, IEEE*, **11**(9):1917–1926, 2011. 15, 16, 17

[8] A. Fusiello, V. Roberto, and E. Trucco. **Symmetric stereo with multiple windowing**. *International Journal of Pattern Recognition and Artificial Intelligence*, **14**:1053–1066, 2000. 35

[9] A. Fusiello, E. Trucco, and A. Verri. **A compact algorithm for rectification of stereo pairs**. *Machine Vision and Applications*, **12**(1):16–22, July 2000. 32

[10] L. Guan, J.-S. Franco, and M. Pollefeys. **3D Object Reconstruction with Heterogeneous Sensor Data**. In *International Symposium on 3D Data Processing, Visualization and Transmission*, 2008. 41

[11] S. A. Gudmundsson, H. Aanaes, and R. Larsen. **Fusion of stereo vision and Time-of-Flight imaging for improved 3D estimation**. *Int. J. Intell. Syst. Technol. Appl.*, **5**(3/4):425–433, 2008. 41

[12] R. Gvili, A. Kaplan, E. Ofek, and G. Yahav. **Depth keying**. *Proc. SPIE*, pages 564–574, 2003. 4

[13] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, second edition, 2004. 25, 27, 29, 30

[14] S. Hawe, M. Kleinsteuber, and K. Diepold. **Dense Disparity Maps from Sparse Disparity Measurements**. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2126–2133, Washington, DC, USA, 2011. IEEE Computer Society. 42, 45, 46, 47, 48

[15] C. Hernandez, G. Vogiatzis, and Y. Furukawa. **3D Shape Reconstruction from Photographs: A Multi-View Stereo Approach**. San Francisco, June 2010. Available from: `http://cvl.umiacs.umd.edu/conferences/cvpr2010/tutorials/`. 78

[16] H. Hirschmüller. **Stereo Processing by Semiglobal Matching and Mutual Information**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **30**(2):328–341, 2008. 36

[17] H. Hirschmüller and D. Scharstein. **Evaluation of Cost Functions for Stereo Matching**. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, **0**:1–8, 2007. 2, 42, 53, 65, 68, 70

[18] B. Huhle, P. Jenke, and W. Straer. **On-the-Fly scene acquisition with a handy multisensory-system**. *International Journal of Intelligent Systems Technologies and Applications*, **5**(3/4):255–263, 2008. 17

[19] T. Kahlmann and H. Ingensand. **Calibration and development for increased accuracy of 3D range imaging cameras**. *Journal of Applied Geodesy*, **2**:1–11, 2008. 11, 39

[20] T. Kahlmann, F. Remondino, and H. Ingensand. **Calibration for increased accuracy of the range imaging camera Swissranger**. In Isprs, editor, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences,,* **Vol. XXXVI**, pages 136–141, Dresden, Germany, September 2006. 11

[21] T. Kanade and M. Okutomi. **A stereo matching algorithm with an adaptive window: theory and experiment**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(9):920–932, 1994. 35

[22] C. Kim, H. Yu, and G. Yang. **Depth Super Resolution Using Bilateral Filter**. In *Image and Signal Processing (CISP), 2011 4th International Congress on,* **2**, pages 1067–1071, 2011. 63, 64, 65, 70

[23] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. **Joint Bilateral Upsampling**. *ACM Trans. Graph.*, **26**(3), July 2007. 42, 63, 70

[24] R. Lange. *3D Time-of-flight distance measurement with custom solid-state sensors in CMOS/CCD-technology.* PhD thesis, University of Siegen (Germany), 2000. 6, 12

[25] R. Lange, P. Seitz, A. Biber, and R. Schwarte. **Time-of-flight range imaging with a custom solid state image sensor**. *Proc. SPIE*, pages 180–191, 1999. 5

[26] **Leica Geosystems** [online]. Available from: `http://www.leica-geosystems.com/`. 81

[27] M. Lindner, M. Lambers, and A. Kolb. **Sub-Pixel Data Fusion and Edge-Enhanced Distance Refinement for 2D/3D Images**. *Int. J. Intell. Syst. Technol. Appl.*, **5**(3/4):344–354, November 2008. 17, 41

[28] **Lux Media Plan** [online]. Available from: `http://luxmediaplan.de/`. 79

[29] **Mesa Imaging** [online]. Available from: `http://www.mesa-imaging.ch/`. 3

[30] **Microsoft®** [online]. Available from: `http://www.microsoft.com/`. 3

[31] P. Monasse, J. M. Morel, and Z. Tang. **Three-step image rectification**. In *Proceedings of the British Machine Vision Conference*, pages 89.1–89.10. BMVA Press, 2010. 32

[32] F. Mufti and R. Mahony. **Statistical analysis of measurement processes for time-of-flight cameras**. In *Proceedings of SPIE the International Society for Optical Engineering*, 2009. 13

[33] C. Dal Mutto. *Acquisition and Processing of ToF and Stereo Data.* PhD thesis, University of Padua (Italy), 2013. 4, 7, 8, 9, 11, 12, 15, 20, 23, 34, 36, 39, 41, 79, 96

[34] T. Oggier, B. Buttgen, F. Lustenberger, G. Becker, B. Ruegg, and A Hodac. **Swissranger sr3000 and first experiences based on miniaturized 3D-ToF cameras.** In *Proceedings of the First Range Imaging Research Day*, ETH Zurich, 2005. 8

[35] S. Paris, P. Kornprobst, and J. Tumblin. *Bilateral Filtering.* Now Publishers Inc., Hanover, MA, USA, 2009. 62, 63

[36] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. **Digital photography with flash and no-flash image pairs**. In *ACM SIGGRAPH 2004 Papers*, pages 664–672, New York, NY, USA, 2004. ACM. 62

[37] D. Piatti. *Time-of-Flight cameras: tests, calibration and multi-frame registration for automatic 3D object reconstruction.* PhD thesis, Polytechnic University of Turin (Italy), 2010. 4, 16, 18

[38] D. Piatti and F. Rinaudo. **SR-4000 and CamCube3.0 Time of Flight (ToF) Cameras: Tests and Comparison**. *Remote Sensing*, **4**(4):1069–1089, 2012. 15, 79

[39] **PMD Technologies** [online]. Available from: `http://www.pmdtec.com/`. 3, 20

[40] L.C. Potter, E. Ertin, J.T. Parker, and M. Cetin. **Sparsity and Compressed Sensing in Radar Imaging**. *Proceedings of the IEEE*, **98**(6):1006–1020, 2010. 46

[41] T. Prasad, K. Hartmann, W. Weihs, S. Ghobadi, and A. Sluiter. **First steps in enhancing 3D vision technique using 2D/3D sensors**. In *11th Computer Vision Winter Workshop*, pages 82–86. V. Chum, O. Franc Ed., 2006. 40

[42] F. Pukelsheim. **The Three Sigma Rule**. *The American Statistician*, **48**(2):88–91, 1994. 67

[43] A. Rochas, M. Gosch, A. Serov, P. A. Besse, R.S. Popovic, T. Lasser, and R. Rigler. **First fully integrated 2-D array of single-photon detectors in standard CMOS technology**. *IEEE Photonics Technology Letters*, **15**(7):963–965, 2003. 4

[44] S. Roy and I. J. Cox. **A Maximum-Flow Formulation of the N-Camera Stereo Correspondence Problem**. In *Proceedings of the Sixth International Conference on Computer Vision*, pages 492–499, 1998. 36

[45] D. Scharstein and R. Szeliski. **A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms**. *Int. J. Comput. Vision*, **47**(1-3):7–42, April 2002. 36

[46] M. Schmidt. *Analysis, Modeling and Dynamic Optimization of 3D Time-of-Flight Imaging Systems.* PhD thesis, University of Heidelberg (Germany), 2011. 6, 8, 20

[47] M. Schmidt and B. Jähne. **A Physical Model of Time-of-Flight 3D Imaging Systems, Including Suppression of Ambient**. In *Dynamic 3D Imaging*, pages 1–15. Springer, 2009. 4

[48] **SoftKinetic** [online]. Available from: `http://www.softkinetic.com/`. 3

[49] R. Streubel. *Simultaneous Time-of-Flight and Stereo Camera Calibration*. Master's thesis, University of Stuttgart (Germany), 2012. 24, 28, 29, 30, 31, 78, 79

[50] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer, New York, 2010. 32

[51] C. Tomasi and R. Manduchi. **Bilateral Filtering for Gray and Color Images**. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 839–, Washington, DC, USA, 1998. IEEE Computer Society. 42

[52] C. Tomasi and R. Manduchi. **Bilateral Filtering for Gray and Color Images**. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 839–, Washington, DC, USA, 1998. IEEE Computer Society. 62

[53] C. Uriarte, B. Scholz-Reiter, S. Ramanandan, and D. Kraus. **Modeling Distance Nonlinearity in ToF Cameras and Correction Based on Integration Time Offsets**. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 214–222. Springer Berlin Heidelberg, 2011. 11

[54] C. A. Weyer, K. H. Bae, K. Lim, and D. D. Lichti. **Extensive metric performance evaluation of a 3D range camera**. In *Proceedings ISPRS Conf.*, **37**, pages 939–944, Beijing, July 2008. 16

[55] **Wikipedia** [online]. Available from: `http://en.wikipedia.org/wiki/Image_rectification`. 32

[56] **Wikipedia** [online]. Available from: `http://en.wikipedia.org/wiki/68-95-99.7_rule`. 67

[57] Q. Yang, R. Yang, J. Davis, and D. Nistr. **Spatial-Depth Super Resolution for Range Images**. In *2007 IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA. IEEE Computer Society, 2007. 41