PhD-FSTC-2012-09
The Faculty of Sciences, Technology and Communication

DISSERTATION

Defense held on 21/03/2012 in Luxembourg

to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG

EN INFORMATIQUE

by

Frederic GARCIA BECERRO
Born on 2 October 1982 in Palafrugell (Girona-Spain)

SENSOR FUSION COMBINING 3-D AND 2-D FOR DEPTH DATA ENHANCEMENT

Dissertation defense committee

Dr. Thomas ENGEL, Chairman
Professor, University of Luxembourg

Dr. David FOFI, Vice Chairman
Professor, University of Burgundy

Dr. Björn OTTERSTEN, dissertation supervisor
Professor, University of Luxembourg

Dr. Bruno MIRBACH, dissertation supervisor
Senior Computer Vision Engineer, IEE S.A.

Dr. Erhardt BARTH
Professor, University of Lübeck

Dr. Djamila Aouada
Research Associate, University of Luxembourg

Abstract

Time-of-Flight (ToF) cameras are known to be cost-efficient 3-D sensing systems capable of providing full scene depth information at a high frame rate. Among many other advantages, ToF cameras are able to provide distance information regardless of the illumination conditions and with no texture dependency, which makes them very suitable for computer vision and robotic applications where reliable distance measurements are required. However, the resolution of the given depth maps is far below the resolution given by standard 2-D video cameras which, indeed, restricts the use of ToF cameras in real applications such as those for safety and surveillance. In this thesis, we therefore investigate how to enhance the resolution of ToF data and how to reduce the noise level within distance measurements. To that end, we propose to combine 2-D and ToF data using a low-level data fusion approach that enhances the low-resolution depth maps up to the same resolution as their corresponding 2-D images.

Low-level data fusion requires the data to be fused to be accurately aligned. Therefore, the first part of this thesis proposes a real-time mapping procedure for data matching. The challenge addressed thereby is to cope with the distance-dependent disparity in an efficient way. To that end, a set of look-up tables for an array of disparities is pre-computed. Then, the mapping is performed through an iterative algorithm that selects pixel by pixel the look-up table that corresponds to the distance measurement of the pixel to be mapped. The experimental results of this part show that in addition to being straightforward and easy to compute, our proposed data matching approach is highly accurate.

The second part of this thesis presents a unified multi-lateral filter for real-time low-resolution depth map enhancement. We propose a unified multi-lateral filter that in addition to adaptively considering 2-D grayscale images

and depth data as guidance information, accounts for the inaccuracy of the position of depth edges due to the low-resolution of ToF depth maps. Consequently, unwanted artefacts such as texture copying and edge blurring are almost entirely eliminated. Moreover, the proposed filter is configurable to behave as most of the alternative depth enhancement methods based upon a bilateral filter. Using a convolution-based formulation and data quantization and downsampling, the proposed filter has been effectively and efficiently implemented for dynamic scenes in real-time applications. The results show a significant qualitative improvement on our own recorded sequences as well as on the Middlebury dataset, outperforming alternative depth enhancement solutions.

Finally, we propose two extensions to improve the quality of the enhanced depth maps. Edge blurring increases when considering grayscale images instead of the original coloured ones. Although the generalization of our filter to consider 3-colour channels is straightforward, the processing time and memory demands prevent it from performing in real-time. We therefore propose a new 1-D colour model whose representation is equivalent to, but more compact than, the 3-D HCL conical representation. It consists in gathering all the hue, chroma and luminance information in one component, namely, the cumulative spiral angle, where the spirals in question are defined as a sampling of the solid HCL cone. The results show that, in addition to preserving the perceptual properties of the HCL colour representation, using the proposed colour model leads to a solution that is more accurate than when using grayscale images. The second extension focuses on enhancing the frame rate of the hybrid ToF multi-camera rig up to the frame rate of the coupled 2-D camera. To that end, we predict new low-resolution depth maps using the flow information estimated from each pair of 2-D frames. Then, we enhance such predicted depth maps by using our proposed multilateral filter. In the end, we provide video frame rate depth maps that present more accurate depth measurements and a significant reduction of the global noise level. Furthermore, we note that the concepts presented herein are not only intended to enhance the depth information given by ToF cameras, as they also apply to other 3-D sensing modalities.

To my mum

M^a José Becerro

And

To my dad

Federico García

Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisor Prof. Dr. Björn Ottersten for giving me the opportunity to be part of his research team as a Ph.D. candidate. I would like to thank him for the continuous support during my Ph.D study and research, for his patience, motivation, enthusiasm, and immense knowledge.

My sincere thanks also goes to my industrial advisor Dr. Bruno Mirbach who was abundantly helpful and offered invaluable assistance, support and guidance. I could not have imagined having a better fellow traveller for my Ph.D study.

Besides my academic and industrial advisor, I thank Dr. Djamila Aouada for her support, guidance, and encouragement in all the time of research and writing of this thesis. I will never forget the sleepless nights we were working together before deadlines.

Deepest gratitude are also due to Prof. Dr. Thomas Engel, the Chairman and member of the supervisory committee without whose assistance this study would not have been successful. I would also show my gratitude to Prof. Dr. David Fofi and Prof. Dr.-Ing. Erhardt Barth for being part of my reviewing committee.

My sincere thanks also goes to my colleagues from the Advanced Engineering Department at IEE S.A. and to my fellow labmates at SnT, for the stimulating discussions and for all the fun we have had in the last three years and a half.

I would also like to convey thanks for supporting this Ph.D project to the AFR Grant Scheme (Aides à la Formation-Recherche), managed by the National Research Fund of Luxembourg (FNR).

I am tempted to individually thank all of my friends which have joined me in the discovery of what is life about and how to make the best of it. However, because the list might be too long and by fear of leaving someone out, I will simply say THANK YOU VERY MUCH TO YOU ALL. Some of you are, anyway, quite lucky: thank you Gerard, Jordi, Lluís, Pascu, Dani, and Ester, as well as Carlos, Sandra, Cesar, David, Lourdes, Miriam, Fernando, and Joëlle for being some of my very best friends.

I cannot finish without saying how grateful I am with my family: grandparents, uncles, aunts, cousins, and girlfriend. All have given me a loving environment where to develop. Particular thanks, of course, to my brother and best mate Josep M^{a} Garcia and to my partner in this life Laetitia Connesson. Lastly, and most importantly, I wish to thank my parents, M^{a} José Becerro and Federico García. They have always supported and encouraged me to do my best in all matters of life. To them I dedicate this thesis, their achievement. **OS QUIERO!**

The master in the art of living makes little distinction between his work and his play, his labour and his leisure, his mind and his body, his information and his recreation, his love and his religion. He hardly knows which is which. He simply pursues his vision of excellence at whatever he does, leaving others to decide whether he is working or playing. To him he's always doing both.

James A. Michener

Contents

Notation

In this thesis, matrices are denoted by boldface, uppercase letters, $\mathbf{M}$, and vectors are denoted by boldface, lowercase letters, $\mathbf{v}$. Scalars are denoted by italic letters, *e.g.,* x, K, α. The following mathematical notation will be used:

$\mathbf{M}^{-1}$	the inverse of a matrix $\mathbf{M}$
$\overline{\mathbf{M}}$	the mean of a matrix $\mathbf{M}$
$\nabla\mathbf{M}$	the gradient of a matrix $\mathbf{M}$
$\mathbf{M}_{\downarrow}$	the downsampling of matrix $\mathbf{M}$
$\mathbf{I}_K$	the identity matrix of dimension K by K
$\mathbf{v}^{\mathrm{T}}$	the transpose of a vector $\mathbf{v}$
$\vert\mathbf{v}\vert$	the Euclidean norm of a vector $\mathbf{v}$
$\Vert\mathbf{v}\Vert_p$	the $p-$norm of a vector $\mathbf{v}$
$x \to \infty$	means x tends to infinity
$\lfloor x \rfloor$	the largest previous integer of x
$\overrightarrow{\mathbf{ab}}$	the vector from $\mathbf{a}$ to $\mathbf{b}$
$a \otimes b$	the cross correlation of functions a and b
$a \equiv b$	means a is equivalent to b
$a \cong b$	means a is congruent to b
$a << b$	means a is much smaller than b
$\arg\min$	the minimizing argument

Abbreviations

3-D (2-D)	3 (2) dimensional
ALOI	Amsterdam Library of Object Images
ARTTS	Action Recognition and Tracking based on Time-of-flight Sensors
CA	Cumulative Angle
CCD	Charge-Coupled Device
CIE	International Commission on Illumination
CMOS	Complementary MetalOxideSemiconductor
FOV	Field of View
HCL	Hue Chroma Luminance
HSL	Hue Saturation Luminance
HSV	Hue Saturation Value
IEE	International Electronics and Engineering S.A.
JBU	Joint Bilateral Upsampling
LIDAR	Light detection and ranging
LUT	Lookup-Table
MLI	Modulated Light Intensity
MRF	Markov Random Fields
NAFDU	Noise-Aware Filter for Depth Upsampling
NIR	Near Infrared
PMD	Photonic Mixer Device
PSNR	Peak Signal-to-Noise Ratio
PWAS	Pixel Weighted Average Strategy
RGB	Red Green Blue
RMSE	Root Mean Square Error
SnT	Interdisciplinary Centre for Security, Reliability and Trust

SPADs	Single-Photon Avalanche Diodes
SPSD	Single-Photon Synchronous Detection
SSIM	Structural SIMilarity index
ToF	Time-of-Flight
UML	Unified Multi-Lateral filter
VGA	Video Graphics Array

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

The discipline of computer vision has undergone a thorough revolution in the last decades, making possible the realisation of intelligent automated applications in a vast range of areas, such as industry automation, surveillance and security, medical imaging, gaming, automotive safety or robotics. The goal of many researchers is to build a system that is able to autonomously operate and interact with the real world, being able to recognise objects, identify targets, or take accurate decisions in real-time. In order to address such a challenge, a fundamental step concerns scene understanding in which computer vision plays a big role. Images are the raw material of computer vision processes for such scene understanding, and can take many forms, such as views from multiple cameras as well as multi-dimensional data from the combination of different vision technologies. Indeed, the use of multiple cameras or multi-view systems that share at least part of their field of view, allows for depth estimation [SAB+07]. Depth information is highly valuable as it enables the perception of the world in three dimensions (3-D), facilitating image processing processes such as the recognition of an object within the scene [OLK+04]. In general, depth information is derived by stereopsis, an analogy to human's perception of depth where scene features are projected onto two slightly displaced cameras to obtain depth from triangulation [FL04, SAB+07]. Unfortunately, depth estimation through triangulation methods requires to solve the well-known but still challenging correspondence problem [HZ03, SS02].

With the ongoing progress in technology, new emerging depth sensing devices based on the Time-of-Flight (ToF) principle [LS01] (Section 1.2.1) are becoming available [FAT11, KBKL09]. While first ToF-based devices for 3-D measuring [HK92], such as light detection and ranging (LIDAR) scanners, were expensive and bulky, the development of the novel so-called demodulation lock-in pixels [SSVH95] allowed to build a new ToF-based device, the ToF camera. In contrast to stereo vision systems, ToF cameras simultaneously provide intensity and depth information for every pixel at a high frame rate. Indeed, such ToF cameras promise to be an alternative to other 3-D sensing systems such as stereo vision systems, laser scanners or structured light systems [FAT10]. Nevertheless, the downside of this promising technology is the low resolution of these cameras which is much lower than the resolution given by alternative 3-D sensing systems. Besides, the acquired depth measurements are highly contaminated by noise [FB07].

The aim of this thesis is to enhance the quality of the data acquired by ToF cameras; namely, to increase their resolution as well as to reduce the noise within depth measurements. To that end, we propose to combine the depth data with the data recorded by a standard 2-D video camera coupled to the ToF camera in a hybrid ToF multi-camera rig. This sensor fusion will exploit the advantages of both 2-D and ToF cameras while avoiding their individual drawbacks. The objective is to improve the quality of depth data by considering industrial requirements for real-world safety and security applications, specifically, robustness to noise, accuracy, and reduced memory and time consumptions.

In what follows, we describe different techniques for depth measurement highlighting their advantages as well as their drawbacks. Then, we introduce the ToF camera which provides full-scene depth distance based on the ToF principle. We also introduce the concept of sensor fusion to address the drawbacks of ToF cameras. Finally, we present the objectives and challenges of this thesis as well as their outline and contributions.

1.1 Depth measurement techniques

In general, computer vision applications are based on the optical sensing of the world in order to recognise, classify, and identify objects or people or take decisions based on their behaviour or activities in a delimited area [YMH06], *i.e.,* the main reception of a building, the checking area of an airport or the subway, to name a few. Many applications require a high accuracy and performance when taking decisions and thus, the data to be processed must be consistent, precise and accurate, *i.e.,* it has to be of good quality.

Depth information is a quantifiable measure that enables 3-D perception of the observed scene, yielding to more robust computer vision applications. Consequently, a vast number of 2-D camera-based approaches for depth sensing, with their own benefits, drawbacks and costs, have been proposed over the past years [Bla04, SAB$^+$07], *e.g.,* depth-from-focus/defocus [NN94, RCM04], depth-from-motion [DW93], depth-from-shading [Wan08], stereo imaging [SK98] or structured light [SFPL10].

Depth measurement techniques can be divided into two main groups depending on the technology they use, namely contact and non-contact techniques. Contact techniques are intended to reconstruct a 3-D model of the scanned object with very high accuracy. However, these scanners are out of the scope of our work as they require physical contact with the object being scanned; which is not feasible for many applications and impractical to survey a delimited area. Therefore, most 2-D camera-based applications belong to non-contact techniques, where the most important concepts fall under active or passive triangulation methods [FAT11]. In general, triangulation methods involve two sensors with at least one being a 2-D camera. We talk about passive triangulation when the depth sensing system is composed of two 2-D cameras as we detail in Section 1.1.1. In this case, depth measurements result from solving the correspondence problem between the reflected ambient radiation within the scene. In contrast, active triangulation refers to systems where one of the sensors is replaced by an emitter that emits some kind of radiation, *e.g.,* light (projector, laser) [TV98] (see Section 1.1.2). In this case, depth measurements result from detecting the reflection of the emitted light. In what follows, we describe the two depth sensing approaches, passive and active.

1.1.1 Passive sensing

Depth measurement techniques based on triangulation, estimate the distance at which a point $\mathbf{P}$ is located in the scene from its projections, $\mathbf{p}_l$ and $\mathbf{p}_r$, on each of the camera reference frames, as shown in Figure 1.1. The classic implementation of passive triangulation is the approach of stereopsis or stereo vision [VT86], which reproduces the human stereo vision by using a camera rig of two standard 2-D video cameras. The two reference frames of the individual cameras are not co-centric, *i.e.,* the two cameras (left and right) are displaced with respect to each other by a distance between the

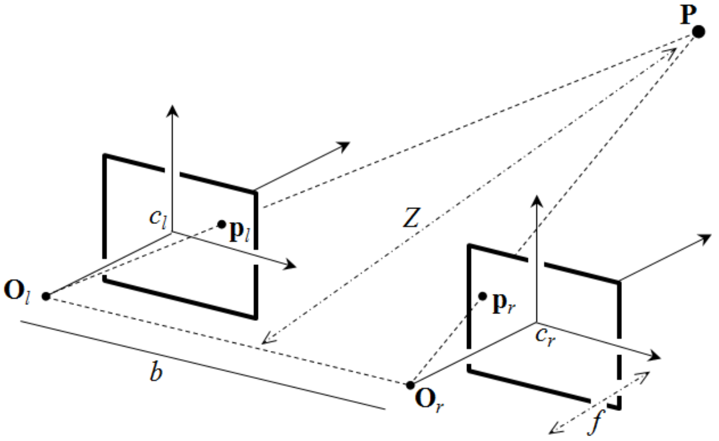


Figure 1.1: Passive sensing using a stereo system setup. The location of a point $\mathbf{P}$ in space is estimated from its projections $\mathbf{p}_l$ and $\mathbf{p}_r$ on the lenses of the left and right cameras, respectively [HZ03].

centres of projection, $\mathbf{O}_l$ and $\mathbf{O}_r$, respectively. This distance is known as the *baseline* b of the stereo system and limits the working depth range. The wider the baseline, the deeper the working depth range. The distance Z at which the point $\mathbf{P}$ is located with respect to the baseline b is obtained from the similar triangles $\mathbf{p}_l\mathbf{P}\mathbf{p}_r$ and $\mathbf{O}_l\mathbf{P}\mathbf{O}_r$ such that

$$\frac{b + x_l - x_r}{Z - f} = \frac{b}{Z}, \tag{1.1}$$

where x_l and x_r are the coordinates of the projections $\mathbf{p}_l$ and $\mathbf{p}_r$ with respect to the principal points c_l and c_r, and f is the common focal length. Solving (1.1) for Z, we obtain

$$Z = f\frac{b}{\rho}, \tag{1.2}$$

where $\rho = x_r - x_l$, the *binocular disparity*, measures the difference in retinal position between the corresponding points in the two images. In stereo systems, the disparity leads to the estimation of the distance Z. However, this requires the detection of the projections $\mathbf{p}_l$ and $\mathbf{p}_r$ which relates to the well-known *correspondence problem* [SS02], which is typically performed by feature matching or correlation analysis and thus, numerically demanding and suffering from shadow effects or texture patterns. In contrast, we tackle the opposite case by the use of the ToF camera as it provides the distance at which each point is located within the given depth maps. This allows us to estimate the disparity for each of the ToF camera pixels, which simplifies the mapping by avoiding demanding operations such as feature matching and image correlation (see Section 3.2).

1.1.2 Active sensing

In contrast to passive triangulation approaches, active systems based on laser or structured light techniques [SFPL10] reduce the dependency on texture to deal with feature-correspondence pairs. In this case, one of the cameras in the setup of Figure 1.1 is replaced by an emitter that projects a pattern to the scene. By doing so, the viewing camera is able to distinguish the projected pattern from the rest of elements, regardless of their texture. Thus, the projected pattern generates a group of features that may be detected in the recorded intensity image. However, and despite the efforts in re-designing the illumination patterns [GAVN11], disadvantages arise when the projected pattern is too weak compared to the background light, *e.g.,* sun light, which happens either if the object is too far away from the sensor, or if the background light is too intensive.

Regardless whether the sensing system is passive or active, triangulation methods can be quite time consuming as they have to cope either with the correspondence problem or to process several encoded illumination patterns, respectively. Besides, the working depth range in triangulation techniques is linked to the baseline between the two cameras or camera plus light source which may introduce non desired effects such as occlusion or shadowing in wide baseline systems.

Conversely, ToF cameras cope with these issues as they allow for depth perception based on the *Time-of-Flight* principle [LS01]. In a nutshell, the ToF principle consists of measuring the time the light emitted by the active sensor needs to reach the surface being scanned and receiving its reflection. Since the velocity of the propagation of light is known, the distance between the sensor and the surface can be estimated from the travelling time. In the following, we present the ToF camera and its working principle.

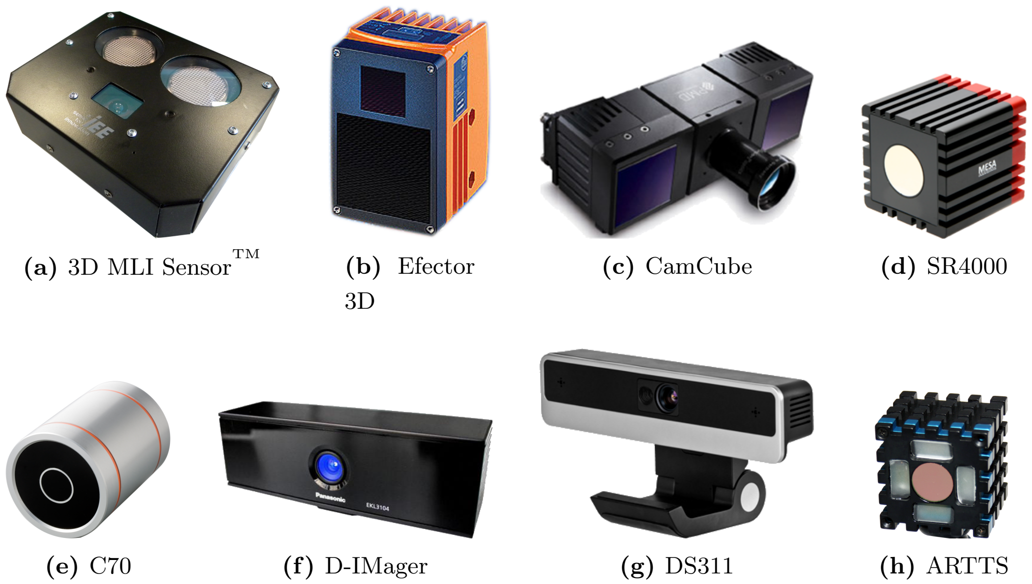
1.2 Time-of-Flight cameras

ToF cameras are capable to provide full-scene depth information at a high frame rate. Two different techniques allow the measurement of the time of flight; pulse detection, where distance directly amounts from the time of flight of a discrete pulse; and amplitude-modulated continuous-wave, where distance is given by the shift in phase between an emitted modulated signal and its reflection [HK92]. Although there exist ToF camera prototypes based on pulse detection techniques, such as the ZCam by 3DV Systems (assets sold to Microsoft in 2009) or the ToF camera line developed by the Fraunhofer Institute of Microelectronic Circuits and Systems [Fra11] and TriDiCam [Tri11], most of the ToF cameras in the market (see Figure 1.2) are based on amplitude-modulated continuous-wave techniques. This is mainly due to the high detection accuracy required to determine the exact time delay of the discrete pulse for pulse detection in ToF devices. In contrast, continuous modulated systems are technically less demanding, *i.e.,* they require a lower power of light source and they avoid photodetectors with a fast electronic shutter. However, due to the periodicity of the modulated signal, the range of the measurements is limited. Lange et al. [LSBL00] and Oggier et al. [OLK$^+$04] describe in detail the physical limitations of depth measurement devices based on such a continuous modulated wave.

We focus on ToF cameras based on an array of demodulation pixels, concretely demodulation lock-in pixels [SSVH95]. In that case, the full scene is illuminated by a modulated signal. Then, each pixel demodulates the reflected light by the scene and recovers the original wave. The difference in phase between the emitted and the received signal is proportional to the distance between the ToF camera and the object being scanned. Therefore, ToF cameras built on these sensors are not only compact and cost-efficient, but also capable of estimating full scene range data in a fast way. Unlike classical techniques for depth sensing, ToF cameras do not rely on mechanical setups, like laser scanners or expensive computations, as in stereo vision, making them very attractive and compact for interactive or real-time applications [FAT10]. In the following, we present the working principle of continuous modulation based ToF cameras and briefly discuss common ToF camera drawbacks in order to motivate our work.

1.2.1 Working principle

As illustrated in Figure 1.3, ToF cameras based on demodulation lock-in pixels [SSVH95] provide distance measurements from the difference in phase between emitted and received modulated near-infrared (NIR) signals. The amplitude and



(a) 3D MLI Sensor$^{\text{TM}}$ **(b)** Efector 3D **(c)** CamCube **(d)** SR4000

(e) C70 **(f)** D-IMager **(g)** DS311 **(h)** ARTTS

Figure 1.2: Active ToF camera brands (as of 2011). (a) 3D MLI Sensor$^{\text{TM}}$by IEE S.A. (56 pixels × 61 pixels). (b) Efector 3D image sensor by ifm electronic (64 pixels × 48 pixels). (c) PMD[vision]®CamCube by PMDTechnologies (200 pixels × 200 pixels). (d) SwissRanger$^{\text{TM}}$SR4000 by MESA Imaging (176 pixels × 144 pixels). (e) Fotonic C70 by Fotonic (160 pixels × 120 pixels)). (f) D-IMager by Panasonic (160 pixels × 120 pixels)). (g) DepthSense$^{\text{TM}}$DS311 by Softkinect (160 pixels × 120 pixels). (h) ARTTS camera prototype.

phase of the incoming modulated signal can be retrieved by synchronously demodulating the investigated signal within the detector [LS01]. To that end, the cross correlation between the received modulated signal $r(t)$ of amplitude a and phase ϕ, and the emitted modulated signal $s(t)$ is performed. The phase of the received modulated signal can be determined by taking the measurement of the cross correlation function at selectively chosen temporal positions or phases. Although other periodic functions can be considered, we assume a sinusoidal formulation for the signals $s(t)$ and $r(t)$ as in [HK92, LS01, OLK^{+}04], *i.e.,*

$$s(t) = 1 + \ \cos(\omega t), \tag{1.3}$$

and

$$r(t) = h + a\ \cos(\omega t - \phi), \tag{1.4}$$

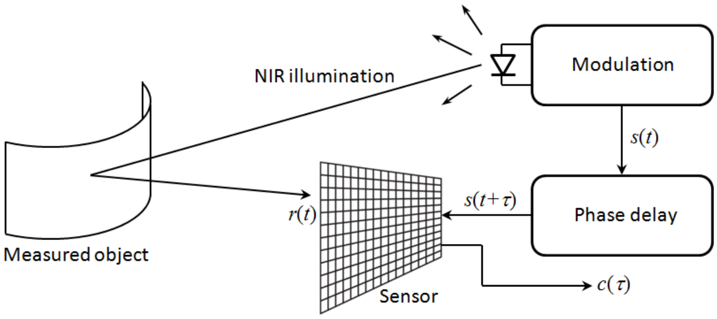


Figure 1.3: The principle of continuous modulation based ToF cameras [Lin10].

with $\omega = 2\pi f_m$ the angular modulation frequency with f_m the modulation frequency. h is the background light plus the non-modulated part of the incident signal, illustrated in Figure 1.4. We calculate the cross correlation $c(\tau)$ function as follows

$$c(\tau) = r(\tau) \otimes s(\tau) = \frac{1}{T} \int\limits_{t=0}^{T} r(t) \cdot s(t+\tau)\,\mathrm{d}t, \tag{1.5}$$

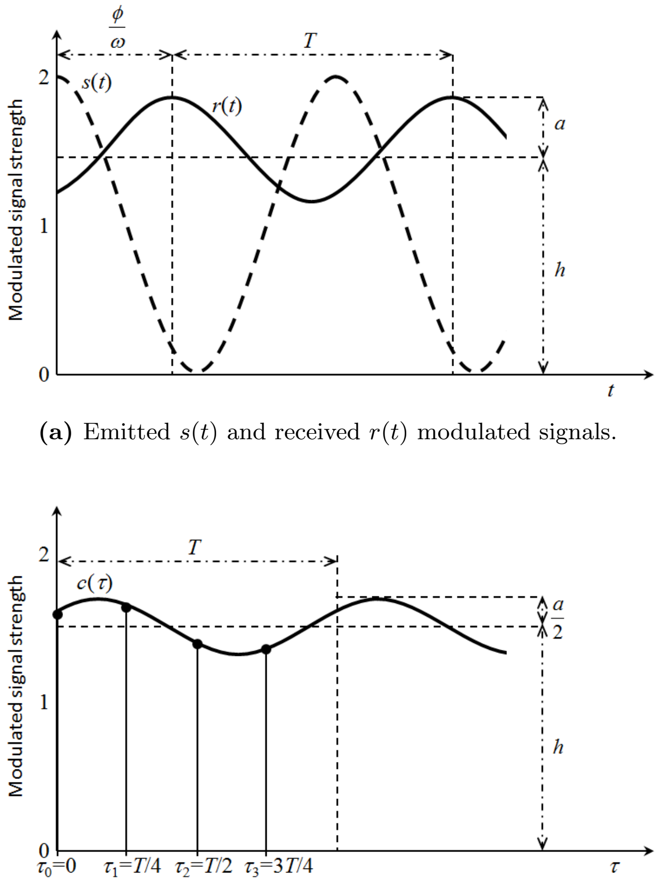
where $\otimes$ denotes the cross correlation. By doing so, the cross correlation sample $c(\tau)$ in (1.5) amounts to

$$c(\tau) = h + \frac{a}{2} cos(\omega\tau + \phi). \tag{1.6}$$

From (1.6), three or more samples per modulated period T are needed in order to unambiguously determine the phase ϕ and the amplitude a of the incident signal [Cre88], as well as its offset h. To that end, we use the so-called four-taps technique in which four samples $c(\tau_k), k = 0, ..., 3$, are taken at four subsequent time intervals $\tau_k = k \cdot T/4 = k/4f_m$ within a modulated period T, as illustrated in Figure 1.4b. As a result,

$$\begin{aligned} \phi &= \arctan\left(\frac{c(\tau_3) - c(\tau_1)}{c(\tau_0) - c(\tau_2)}\right), & (1.7)\\ a &= \frac{1}{2}\sqrt{\big(c(\tau_3) - c(\tau_1)\big)^2 + \big(c(\tau_0) - c(\tau_2)\big)^2}, & (1.8)\\ h &= \frac{c(\tau_0) + c(\tau_1) + c(\tau_2) + c(\tau_3)}{4}. & (1.9) \end{aligned}$$

The reasons to choose four samples instead of three are to improve robustness against noise, to enable a highly symmetric design of the sensor, to ensure that the phase is insensitive to quadratic non-linearities in detection, and to simplify the formulae for the



(a) Emitted $s(t)$ and received $r(t)$ modulated signals.

(b) Cross correlation $c(\tau)$ between $r(t)$ and $s(t)$.

Figure 1.4: ToF measurement principle. Four selectively chosen samples $(c(\tau_k), k = 0, ..., 3)$ are taken within a modulated period T in order to determine the phase ϕ, amplitude a, and offset h of the received modulated signal $r(t)$ [SSVH95].

phase ϕ, the amplitude a, and the offset h. The distance measurements d are obtained from

$$d = \frac{L}{2\pi} \cdot \phi, \tag{1.10}$$

with $\mathsf{c} \equiv 3 \cdot 10^8$ m/s the speed of light and L the working range or non-ambiguity distance range of the ToF camera, given by [LS01]

$$L = \frac{\mathsf{c}}{2f_m}. \tag{1.11}$$

The factor 1/2 in (1.10) and in (1.11) is due to the fact that light travels twice the distance between the camera and the sensed object, as depicted in Figure 1.3. The ToF camera is actually an image sensor whose size corresponds to the camera resolution $(m \times n)$, as illustrated in Figure 1.3. Hence, each single pixel constituting the image sensor is identified by the pixel position (i, j), where i indicates the row and j indicates the column. Each pixel measures a distance d_{ij} obtained using (1.10). As a result, the ToF camera provides a distance image or depth map $\mathbf{D}$ defined as $\mathbf{D} = [d_{ij}]_{m\times n}$, the matrix of all the elements d_{ij}. In the same way, an amplitude image $\mathbf{A}$ defined as $\mathbf{A} = [a_{ij}]_{m\times n}$ is obtained using (1.8). Indeed, this amplitude image allows us to calibrate the ToF camera, as we detail in Section 2.4.

1.2.2 Drawbacks of ToF cameras

In addition to simultaneously provide full-scene depth information at a high frame rate, the recent advances in industrializing and producing economic, compact, robust to illumination changes and light ToF cameras are starting to have an impact on commercial applications [FAT10, KBKL09]. However, ToF cameras and specially the industrial ones (see Figures 1.2a and 1.2b), cannot yet attain the resolution and precision of alternative 3-D sensing systems, such as laser scanners or stereo systems. Indeed, two main drawbacks are currently restricting the use of ToF cameras in a wide range of computer vision and robotics applications; namely, the noise within depth measurements and the low resolution of the given depth maps. In the following we give more details of these drawbacks.

1.2.2.1 Noise within depth measurements

In Section 1.2.1, we have presented the theory related to the working principle of continuous modulated ToF cameras. However, in practise, in order to reduce the effect of noise, the cross correlation $c(\tau)$ in (1.5) is integrated over nT. As a result, one obtains four samples $\tilde{c}(\tau_k) = n \cdot c(\tau_k)$, which are proportional to the number of periods integrated over. In the following we assume that these samples are expressed in units of number of photoelectrons. When considering $\tilde{c}(\tau_k)$ in (1.7) instead of $c(\tau_k)$, the measured phase ϕ remains the same, *i.e.,*

$$\phi \quad = \quad \arctan\left(\frac{\tilde{c}(\tau_3) - \tilde{c}(\tau_1)}{\tilde{c}(\tau_0) - \tilde{c}(\tau_2)}\right), \tag{1.12}$$

while the amplitude a in (1.8) and the offset h in (1.9) become proportional to n as follows

$$\begin{aligned} \tilde{a} &= \frac{1}{2}\sqrt{\left(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\right)^2+\left(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\right)^2} = n\cdot a, && (1.13)\\ \tilde{h} &= \frac{\tilde{c}(\tau_0)+\tilde{c}(\tau_1)+\tilde{c}(\tau_2)+\tilde{c}(\tau_3)}{4} = n\cdot h, && (1.14) \end{aligned}$$

and expressed in units of number of photoelectrons.

The main sources of noise within the distance measurements d (see (1.10)) in continuous modulated ToF cameras are electronic noise n_e, dark noise n_t and photon shot noise n_p [LS01]. Electronic noise is a random fluctuation which is characteristic of all electronic circuits such as analog to digital converters. Dark noise summarizes additional photodetector noise sources such as thermal noise, *i.e.,* random fluctuations due to changes of temperature. Photon shot noise is due to the photon character of light. The generation of a given number of photoelectrons in a fixed interval of time occurs randomly with a known average rate and independently of time. Therefore, the photon shot noise n_p can be modelled by a Poisson distribution where the number of observed photoelectrons fluctuates about its mean with a standard deviation $\sigma_{\tilde{c}(\tau_k)} = \sqrt{\tilde{c}(\tau_k)}, k = 0, ..., 3$. We note that the number of observed photoelectrons is large which leads to the approximation of the Poisson distribution by an additive Gaussian distribution of mean zero and with the same standard deviation $\sigma_{\tilde{c}(\tau_k)}$, *i.e.,* $n_p \rightarrow \mathcal{N}\big(0, \sigma^2_{\tilde{c}(\tau_k)}\big)$. Similarly, the electronic noise n_e may also generate or vary the number of photoelectrons while converting from analog to digital. In the same way, the thermal noise n_t also varies the number of photoelectrons by the excited photoelectrons due to the variations in temperature. They both can be modelled as additive Gaussian noises with mean zero, and variances σ_e^2 and σ_t^2, respectively.
The noisy cross correlation samples $\tilde{c}_n(\tau_k)$ may thus be modelled as follows

$$\tilde{c}_n(\tau_k) = \tilde{c}(\tau_k) + n_{total}, \quad \text{with } n_{total} = n_p + n_e + n_t. \tag{1.15}$$

Since both electronic noise and thermal noise are also independent and uncorrelated, the total variance of the noise equals to the sum of the variances of the photon shot noise, the electronic noise, and the thermal noise, *i.e.,* $\sigma^2_{total} = \sigma^2_p + \sigma^2_e + \sigma^2_t$, respectively.

In [LSBL00], Lange applies the rules of error propagation to (1.12) in order to determine the error on the phase ϕ, also considered to be the standard deviation σ_ϕ

of the noise on ϕ. Following the same rules, we determine the standard deviation σ_ϕ taking into account all sources of noise, as detailed in Appendix B. We find

$$\sigma_\phi = \sqrt{\sum_{k=0}^{3} \left(\frac{\partial \phi}{\partial \tilde{c}(\tau_k)} \right)^2 \cdot \sigma^2_{total}} \ . \tag{1.16}$$

Considering (1.16) for the four samples $\tilde{c}(\tau_k), k = 0, ..., 3$, taken at four subsequent time intervals $\tau_k = k \cdot T/4 = k/4f_m$ within a modulated period T, we find the uncertainty σ_d on the measured distance $d_e = d \pm \sigma_d$. Below is the final simplified expression of σ_d,

$$\sigma_d = \frac{L}{\sqrt{2}\pi} \cdot \frac{\sqrt{\tilde{h} + \sigma_e^2 + \sigma_t^2}}{2\tilde{a}}, \tag{1.17}$$

with $\tilde{h} = \tilde{a} + \tilde{b}$, the number of photoelectrons from the emitted active light (amplitude) plus the background light, respectively. We note that Lange refers to σ_e^2 as a number of pseudo electrons, that he defines as an increment or decrement of the number of electrons due to the rounding when converting from analog to digital. We observe in (1.17) that the fluctuation within distance measurements is inversely proportional to the number of photoelectrons within the amplitude of the signal in the case where the number of photoelectrons from the background light $\tilde{b}$ and from the error due to electronic noise and thermal noise, are smaller compared to the number of photoelectrons within the amplitude $\tilde{a}$. That is, σ_d is proportional to $1/\sqrt{\tilde{a}}$. In addition, the amplitude $\tilde{a}$ is proportional to the power density of the light in the scene [BW99]. Thus, according to the inverse square law, the power density of the light in the scene decays at a rate proportional to $1/d^2$ due to the active illumination.

We also remark that in the case of low amplitudes $\tilde{a}$ where background light $\tilde{b}$ is not present (no sun light influence), the electronic noise becomes dominant. In this case, ToF camera prototypes based on digital phase demodulation using single-photon synchronous detection (SPSD) are expected to perform better [NFK$^+$08, SPS$^+$07] than current ToF cameras based on amplitude-modulated continuous-wave. An SPSD image sensor is based on single-photon avalanche diodes (SPADs) [NRBC05] rather than the CCD/CMOS photogates used by lock-in pixels [SSVH95]. In this case, the analog accumulating diffusion used by lock-in pixels is replaced by a digital counter. As a result, SPSD ToF cameras are considered to be virtually free of electronic noise at signal detection and demodulation as they do not use any analog processing or analog-to-digital conversion.

Furthermore, we see in (1.17) that σ_d is directly linked to the non-ambiguity range of distance L. Thus, in order to obtain accurate and robust depth measurements, the modulation frequency f_m has to be chosen as high as possible. However, the current limitations of technology allow to modulate a signal around 20 MHz. Therefore, if we set $f_m = 20$ MHz in (1.11), the non-ambiguity range of distance is $L = 7.5$ m, which corresponds to the working range of most of the continuous modulated ToF cameras. In the case where the application requires a higher range distance, the reference signal $s(t)$ can be modulated at a smaller frequency; however, the influence of noise will be higher according to (1.17).

1.2.2.2 Resolution of depth maps

In addition to the noise influence within the depth measurements, the given depth maps suffer from a low resolution compared to the data given by alternative 3-D sensing systems. Moreover, this resolution problem is even more prominent in industrial ToF cameras as a compromise for their higher robustness to ambient conditions, *e.g.,* larger working temperature range and higher reliability under sun lighting. We note that the noticeable difference between the resolution of a 2-D camera and the resolution of a ToF camera is directly linked to the physical difference in the dimensions of the imager. Indeed, the ToF imager used to be up to 10 times bigger than the imager of a 2-D camera. This is due to all the electronics that surround a ToF pixel in order to cope with the distance measurement, *i.e.,* wiring to read the four samples or electronics to demodulate the incident signal (see Section 1.2.1). We note that there exist some “high-resolution” ToF cameras such as those intended for research purposes, *e.g.,* the PMD[vision]®CamCube, the SwissRanger™SR4000 or the ARTTS prototype (Figures 1.2c,1.2d,1.2h, respectively), indoor applications in constrained environments, *e.g.,* the Fotonic C70, the D-IMager or the DepthSense™DS311 (Figures 1.2e,1.2f,1.2g, respectively) or 3-D sensing devices intended for gaming applications such as Microsoft’s Kinect camera [LMW$^+$11]. However, in contrast to all these devices, ToF cameras that are used for automotive applications or applications in industrial automation have resolutions lower than (64 pixels × 64 pixels). Therefore, in applications where the limited resolution of a ToF camera is critical, a very promising strategy is sensor fusion [GBQ$^+$08, ZWY$^+$10]. To summarize, the concept is to combine ToF data with data provided by other sensors, usually 2-D cameras [FBK10]. Indeed, some first attempts in ToF and 2-D data fusion have shown promising dense depth maps, outperforming, in some cases, alternative 3-D sensing systems [CBTT08, YYDN07].

1.3 Sensor fusion

Sensor fusion is a strategy that combines the data provided by similar or disparate sensors such that the resulting fused data is in general better, *i.e.,* more accurate, less noisy and more precise than the data acquired by a single sensor. As discussed in Section 1.2.2, the resolution of ToF cameras is far below the resolution of standard 2-D video cameras. Therefore, in applications where the ToF camera resolution is critical, we can resort to sensor fusion approaches and complement the ToF camera with a 2-D camera.

The goal of this thesis is to combine the raw data provided by each of the cameras that constitute a hybrid ToF multi-camera rig in order to enhance the quality of the low-resolution ToF data. In computer vision, fusion processes are often categorized as low, intermediate, or high level fusion, depending on the processing stage at which fusion takes place.

Within this thesis, we talk about low-level fusion, also called data fusion, in contrast to higher fusion levels in which the fusion deals with post-processed data (feature or decision fusion). Over the last years, there have been some attempts for ToF data enhancement by means of data fusion. The application of Markov Random Fields (MRFs) to cope with the problem of enhancing ToF data by considering both ToF and 2-D data was first presented by Diebel et al. [DT05]. In contrast to MRFs-based approaches, data fusion based upon a bilateral filter, an edge-preserving image filter [TM98], enables a real-time data fusion. Indeed, recent contributions [GAM$^+$11b, GAM$^+$11a] have proven to outperform triangulation-based techniques (Section 1.1). Thus, we focus on depth enhancement methods that couple a single 2-D camera with a single ToF camera in contrast to other approaches that combine several 2-D and ToF cameras [KH10, KTD$^+$09, KS06, STDT08] for dense 3-D reconstruction.

Intermediate-level fusion, also known as feature-level fusion, combines different features such as edges, corners, lines or texture parameters, determined from several raw data sources. In [GBQ$^+$08], Gloud et al. combine 2-D features extracted from data recorded by a high resolution video camera and 3-D features from data acquired by a laser range scanner in order to deal with the problem of object detection and recognition for a robotic system. Another example of feature fusion is given by Natroshvili et al. in [NSS$^+$08] where they profit of the high frame rate of a ToF camera to determine coarse features that serve a CMOS camera as an input to realize a finer object detection, segmentation and classification for real-time pedestrian detection.

High-level fusion, also known as decision-level fusion, performs as a function of the confidence resulting from different processes. In this sense, high-level fusion approaches cope with advanced intelligent systems for path-planning and obstacle avoidance, handling robot position uncertainty or other related problems [LS99]. Another field where the application of those methods demonstrate a sequence of significant advantages is in automotive applications, *i.e.,* assistance systems for driving, autonomous vehicles or object recognition in road environments [FPA$^+$07, GAVA08].

1.4 Objectives and challenges

The present thesis is intended to overcome the limitations of ToF cameras and especially industrialized ToF cameras such as the 3D MLI Sensor$^{\text{TM}}$(see Figure 1.2a). To that end, we propose to attach a supplemental imaging sensor to the ToF camera. As a result, we obtain raw images from different modalities which allows for low-level data fusion.

Within this thesis, we first couple an industrialized ToF camera with a standard 2-D video camera in a hybrid ToF multi-camera rig. Next, we combine the raw data given by each of the cameras in a low-level data fusion approach where the 2-D images help to enhance the ToF data. By doing so, we exploit the advantages of each of the cameras that constitute the multi-camera rig while avoiding their individual drawbacks. As a result, the quality of the data given by the ToF camera is significantly improved. The aim of this thesis is to enable the use of ToF cameras in computer vision or robotics applications beyond their current limitations.

The first challenge relates to the alignment of the data given by each of the cameras in our hybrid ToF multi-camera rig prototype (see Figure 1.2a). Any low-level data fusion approach requires the data to be fused to match pixel to pixel, which is far from a trivial task for most real-world data and scenarios. Indeed, most of the approaches for depth enhancement address the data matching process within an off-line pre-processing step where the data is undistorted and rectified to be pixel aligned. This, in turn, restricts the use of such a hybrid ToF multi-camera rig for dynamic scenes, where real-time is a requirement. We note that mapping the distance measurements from the ToF camera onto the 2-D camera is a straightforward procedure which would result in the assignment of a colour value to each of the (low-resolution) ToF pixels. To that end, one can resort to commonly used 3-D warping techniques [LH10a, IMN$^+$10, ZWY$^+$10] as discussed in Section 3.3. However, within this thesis we want to tackle just the opposite case. We want to assign to each of the high-resolution 2-D pixels an accurate distance value. In this case, we need to map each 2-D pixel onto the corresponding ToF pixel, which is not straightforward if one has to take into account the distance dependency of the disparity. Furthermore, such dependency on the distance requires to recompute the whole mapping procedure for each recorded frame and thus, it makes the real-time implementation quite challenging.

Once the data from each camera matches pixel to pixel, we focus on the data fusion approach to enhance the quality of the low-resolution depth maps. There exists a number of data fusion approaches [CBTT08, DT05, YYDN07] that yield to satisfactory dense depth maps. However, two main artefacts in low-level data fusion motivate our work; namely, texture copying and edge blurring within the enhanced depth maps. The coarse combination of depth and intensity data may lead to erroneously copy 2-D texture into actually smooth depth geometries within the depth map [CBTT08]. Edge blurring is the second artefact which results from the misalignment between 2-D and depth edges, mainly driven by the huge difference of resolutions, or because depth edges have no corresponding edges in the 2-D guidance image, *i.e.,* in situations where objects on either side of a depth discontinuity have a similar colour [CBTT08]. Hence, our second challenge is to overcome these artefacts by developing appropriate filtering techniques.

In addition, industrial requirements for real-world applications are addressed, which implies an easy and transparent adaptability of the methods and an implementation capable to perform in real-time.

1.5 Outline and contributions

This section gives the outline of the thesis, highlights the contributions, and provides references to the articles where the results were (or will be) presented. The main body of the thesis is separated into three parts as detailed below:

- **Part I: Data Alignment**

The first part of this thesis deals with the alignment of data for further low-level data fusion purposes. *Chapter 2* introduces the perspective or pinhole camera model, *i.e.,* the basis to formulate the relationship between the real world and the camera. This formulation involves the knowledge of the camera within the world, *i.e.,* its orientation and position with respect to a known world reference (extrinsic camera parameters) as well as the knowledge of the camera itself, *i.e.,* its own characteristics (intrinsic camera parameters). Another intrinsic parameter to consider is the distortion produced by the lens of the camera. After defining the relationship between reality and images, we present the hybrid ToF multi-camera rig and its components. We conclude Chapter 2 by proposing a practical calibration method that in addition to estimating the multi-camera rig parameters with high accuracy, is feasible for a hybrid ToF multi-camera rig calibration in a mass production line. *Chapter 3* focuses on the alignment of the data delivered by each of the cameras in the test rig. After fulfilling the calibration process, the data is undistorted and ready to be aligned. In this chapter, we propose a novel alignment approach that assigns to each 2-D pixel its corresponding depth value. In addition, the method considers the distance-dependent disparity due to the displacement between the two cameras in the test rig. Our contribution achieves a real-time data acquisition and alignment, facilitating further fusion steps. The results of this part have previously been published in (or submitted as) the following articles:

∗ F. Garcia, D. Aouada, B. Mirbach, and B. Ottersten. Distance-Dependent Mapping for Hybrid ToF Multi-Camera Rig. *IEEE Journal of Selected Topics in Signal Processing.* Accepted for publication with mandatory minor revisions.

∗ F. Garcia, D. Aouada, B. Mirbach, T. Solignac, and B. Ottersten. Real-time Hybrid ToF multi-camera Rig Fusion System for Depth Map Enhancement. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1-8. June 2011.

- ∗ F. Garcia and B. Mirbach. Range Image Pixel Matching Method. *PCT-patent application under priority of following Patent application in Luxembourg.* Ref. P-IEE-292/LU. Application No. 91 745 of 15 October 2010.

- **Part II: Data Fusion**

The second part tackles the core of this thesis, the fusion of the raw data acquired by each of the cameras in the hybrid ToF multi-camera rig in order to enhance the quality of the ToF data. *Chapter 4* covers state-of-the-art low-level data fusion approaches for depth enhancement. This chapter describes the different strategies to enhance low-resolution depth maps by considering additional 2-D information. *Chapter 5* presents the proposed approaches to enhance the quality of low-resolution ToF data. Our main contribution is a filter that accounts for the reliability within the depth measurements while considering 2-D edges. Consequently, unwanted but common artefacts in state-of-the-art filtering techniques such as texture copying and edge blurring get almost entirely eliminated. The filter is extended with a new factor that increases the accuracy of the depth measurements within smooth regions in the scene. Furthermore, taking into account the industry requirements, we propose an effectively and efficiently algorithm for real-time applications. *Chapter 6* quantifies the proposed approaches against state-of-the-art methods and shows the experimental results. The results of this part have previously been published in (or submitted as) the following articles:

- ∗ F. Garcia, D. Aouada, B. Mirbach, and B. Ottersten. Unified Multi-Lateral Filter for Real-Time Depth Enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).* Submitted.

- ∗ F. Garcia, D. Aouada, B. Mirbach, T. Solignac, and B. Ottersten. A New Multi-lateral Filter for Real-Time Depth Enhancement. *IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS).* September 2011.

- ∗ F. Garcia, B. Mirbach, B. Ottersten, F. Grandidier, and A. Cuesta. Pixel Weighted Average Strategy for Depth Sensor Data Fusion. *IEEE International Conference on Image Processing (ICIP)*, 2805-2808. September 2010.

- **Part III: Extensions**

The last part proposes two different extensions for our fusion approach. *Chapter 7* copes with the edge blurring artefact. Although edge blurring is almost eliminated when using the proposed data fusion approach, there are some cases where it still appears. Such cases are due to the fact that there are depth edges without a corresponding 2-D edge. Thus, depth edges cannot be accordingly adjusted. This can occur in situations where foreground and background objects in the scene have a similar colour. In that case, the segmentation of these objects in 2-D becomes intricate as no edges help to distinguish the objects. However, edge blurring intensifies as soon as we consider grayscale images when fusing, instead of the original coloured ones. Many image processing algorithms as well as the one we propose in this thesis, consider grayscale images as input data. By doing so, system requirements such as low processing time and memory constraints can be overcome. The downside is that the probabilities of having the same intensity value in both foreground and background objects increases while transforming from colour to grayscale, as many colours get collapsed to the same intensity value. Therefore, in this chapter we propose to reduce the complexity of processing three channels by compactly storing the same information in only one channel. Thus, we show that much better results can be obtained by replacing grayscale images by images transformed into our new 1-D colour space in which the same information as in the non-transformed image is preserved without losses. *Chapter 8* presents an extension of the filter that increases the frame rate of the ToF camera. Until now, the goal was the enhancement of the quality of a given low-resolution depth map. However, there are security and safety applications that in addition to this data quality enhancement also require a high frame rate. ToF cameras are known to be fast but still slow compared to standard 2-D video cameras. Therefore, we propose an extension that estimates the motion between the 2-D camera frames, compensates the motion on the low-resolution depth maps and enhances their quality using the proposed data fusion filter. The results of this part have previously been published in (or submitted as) the following articles:

∗ F. Garcia, D. Aouada, B. Mirbach, and B. Ottersten. Spatio-Temporal ToF Data Enhancement by Fusion. *IEEE International Conference on Image Processing (ICIP).* September 2012.

- ∗ F. Garcia, D. Aouada, B. Mirbach, and B. Ottersten. A new 1-D colour model and its application to image filtering. *IEEE International Symposium on Image and Signal Processing and Analysis (ISPA).* September 2011. Best Student Paper Award.

- ∗ F. Garcia, D. Aouada, B. Mirbach, and B. Ottersten. Spiral colour model: Reduction from 3-D to 2-D. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1305-1308. May 2011.

Chapter 9 concludes the thesis, and elaborates on possible lines for future research.

Part I

Data Alignment

Chapter 2

System model

In this chapter we introduce the basis to understand the relationship between the scene and the data given by each of the cameras in the hybrid ToF multi-camera rig. Although the geometry of a camera as well as the parameters that relate the coordinates of a point in the scene to a single or multiple viewing cameras are well known by the computer vision community, we have considered appropriate to describe them for completeness and consistency when introducing our system model. We refer the reader to textbooks such as *Multiple View Geometry in Computer Vision* from Hartley et al. [HZ03] or *Introductory Techniques for 3-D Computer Vision* from Trucco et al. [TV98] for further details. Then, we detail the system model we have considered to evaluate our concepts, which is composed of a 2-D video camera and an industrialized ToF camera, *i.e.,* a hybrid ToF multi-camera rig. Finally, we propose a practical full-system calibration approach in order to determine the system parameters.

2.1 Camera geometry and single view geometry

This section presents the geometry of a single camera and describes the projection from the scene space onto the image frame of the viewing camera.

2.1.1 Perspective camera model

The simplest model of a camera in computer vision and computer graphics is the *perspective* or *pinhole* camera model in which all optical distortions are neglected. It is based on the principle of collinearity, where each point in the scene is projected on

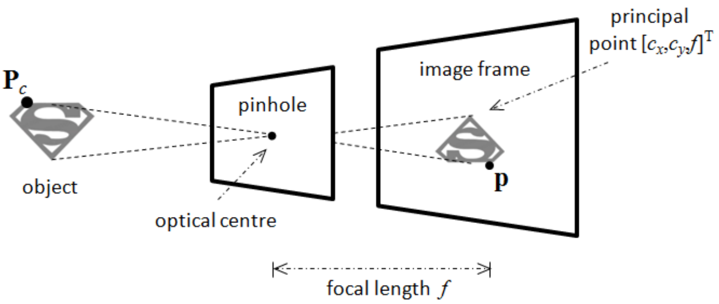


Figure 2.1: The perspective or pinhole camera model.

the image frame by a straight line passing through the optical centre [TV98], as shown in Figure 2.1. The distance between the optical centre and the image frame is usually referred to as the *focal length* f of the pinhole camera while the point where the optical axis of the camera intersects the image frame corresponds to the *principal point* whose coordinates are $[c_x, c_y, f]^{\mathrm{T}}$ (we define $[\cdot]^{\mathrm{T}}$ as the transpose of a matrix or a vector). If we consider a point $\mathbf{P} = [X, Y, Z]^{\mathrm{T}}$ in the scene, its projection on the camera image frame $\mathbf{p} = [x, y, z]^{\mathrm{T}}$ are expressed as

$$x = f\frac{X}{Z}, \tag{2.1}$$

$$y = f\frac{Y}{Z}, \tag{2.2}$$

and

$$z = f. \tag{2.3}$$

Nevertheless, these perspective projections require some knowledge of the geometry of the camera which is given by the intrinsic camera parameters, usually determined within the system calibration process.

2.1.2 Intrinsic camera parameters

In general, the data recorded by a camera is related to its own reference frame, usually called the camera reference frame. We refer to a point in the scene $\mathbf{P}$ whose coordinates are related to the camera reference frame as $\mathbf{P}_c = [X_c, Y_c, Z_c]^{\mathrm{T}}$ (see Figure 2.1). Images, however, are generally specified as pixel arrays with their origin in the upper-left corner. Thus, the point coordinates need to be transformed or mapped from the camera frame to the image frame. This in turn, requires the knowledge of the intrinsic camera parameters that characterize the optical, geometric and digital characteristics of the viewing camera. These parameters are the effective size of the pixel or pixel pitch (in μm) (δ_x, δ_y), the pixel coordinates of the principal point (c_x, c_y) relative to the upper-left corner, and the focal length f [HZ03, TV98]. By considering these parameters, the expressions in (2.1) and (2.2) can be generalized to

$$\mathbf{p} = \mathbf{K} \cdot \mathbf{P}_c \Leftrightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{K} \cdot \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}, \tag{2.4}$$

with $\mathbf{K}$ the matrix of intrinsic parameters defined as follows

$$\mathbf{K} = \mathbf{K_s K_f} = \begin{bmatrix} \delta_x^{-1} & 0 & c_x \\ 0 & \delta_y^{-1} & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \delta_x^{-1} f & 0 & c_x \\ 0 & \delta_x^{-1} f & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.5}$$

where f is the focal length, δ_x and δ_y are the effective horizontal, and respectively vertical pixel size, and (c_x, c_y) is the position of the optical axis or principal point in the image (all units are in millimetres). Thus, from (2.4), the x, y, and z coordinates of $\mathbf{p}$ are

$$\begin{aligned} x &= \delta_x^{-1} f X_c + c_x Z_c, & (2.6) \\ y &= \delta_y^{-1} f Y_c + c_y Z_c, & (2.7) \\ z &= Z_c. & (2.8) \end{aligned}$$

Expressions in (2.1) and (2.2) result from dividing (2.6) by (2.8) and (2.7) by (2.8), respectively, *i.e.,* homogeneous coordinates where the z coordinate is equal to 1. We note that in the pinhole camera the principal point (c_x, c_y) is assumed to be the centre of the sensor, *i.e.,* $(0, 0)$.

Nevertheless, the pinhole camera model is only an approximation of the real camera projection that simplifies the mathematical formulation of the relationship between objects in the scene and their image coordinates. In practise, real cameras differ from the pinhole camera model insofar as they use lenses that introduce geometrical distortion to the image coordinates. For most applications, therefore, the pinhole model is a basis that is extended with some corrections for the systematically distorted image coordinates.

2.1.3 Image distortion

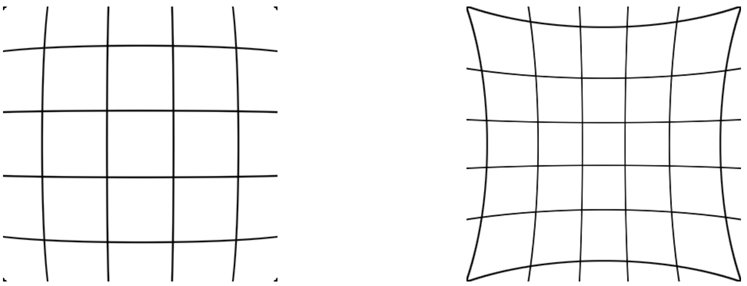
In most computer vision applications, there are mainly two types of distortions that are modelled in order to be corrected; namely, the radial distortion and the tangential distortion. The former distortion is characterized by a symmetric displacement along the radial direction from the principal point. A negative displacement decreases the image magnification resulting in a visual effect similar to mapping the image around a sphere (or barrel), *i.e.,* barrel distortion (Figure 2.2a). A positive displacement increases the image magnification resulting in a visible effect where lines that do not go through the centre of the image are bent inwards, towards the centre of the image, *i.e.,* pincushion distortion (Figure2.2b). In contrast, the tangential distortion is generally caused by improper lens alignment due to inaccuracies during the camera assembling [HS97, WCH92]. In this case, the visual effect is a decentering of the image. Although both types of distortions can be modelled and corrected, the tangential distortion is usually neglected when camera-lenses are assembled by the camera manufacturer. The radial distortion can be approximated by

$$x = \tilde{x}(1 + k_1 r^2 + k_2 r^4 + ...) \tag{2.9}$$

and

$$y = \tilde{y}(1 + k_1 r^2 + k_2 r^4 + ...), \tag{2.10}$$

where k_i, $i = 1, 2, ...$, are the coefficients for the radial distortion [HS97]. Thus, the expressions in (2.1) and (2.2) are replaced by (2.9) and (2.10), respectively. In this case, the coordinates x and y are the non-observable, distortion-free image coordinates. The



(a) Simulation of a barrel distortion.

(b) Simulation of a pincushion distortion.

Figure 2.2: Effect of radial distortions [Lin10, WCH92].

coordinates $\tilde{x}$ and $\tilde{y}$ are the corresponding distorted coordinates, and $r^2 = \tilde{x}^2 + \tilde{y}^2$. In the case of standard lenses, one or two coefficients (k_1, k_2) are enough to compensate for the radial distortion. However, a higher order is needed in case of wide angle lenses, *e.g.,* 90°, where radial distortion can be 30% of the image radius. The expression of the tangential distortion is often written in the following form

$$x = 2p_1\tilde{x}\tilde{y} + p_2(r^2 + 2\tilde{x}^2) \tag{2.11}$$

and

$$y = p_1(r^2 + 2\tilde{y}^2) + 2p_2\tilde{x}\tilde{y}, \tag{2.12}$$

where p_1 and p_2 are coefficients for the tangential distortion [HS97]. We recall that the tangential distortion is due to improper lens alignment which is in general neglected. Indeed, its contribution is much lower than the contribution of the radial distortion.

2.1.4 Extrinsic camera parameters

The relationship between the coordinates of a point $\mathbf{P}_c$ in the camera reference frame and its coordinates in the image frame is given by (2.4). However, in most computer vision applications, the point coordinates must be related to a known reference frame, the so-called world reference frame, to which we refer with the subscript w. In general, the camera reference frame does not coincide with the world reference frame. Therefore, it is usually necessary to first map the 3-D points related to the world reference frame, onto the camera reference frame. To that end, a typical choice for describing the transformation between the camera reference frame and the world reference frame is to use a 3-D translation vector $\mathbf{t} = [t_x, t_y, t_z]^{\mathrm{T}}$, which describes the relative positions between the origins of both reference frames, and a 3×3 rotation matrix $\mathbf{R}$, which defines the orientation between the two reference frames [HZ03, TV98]. Thus, the relationship between the coordinates of a point $\mathbf{P}$ from the world frame to the camera frame, $\mathbf{P}_w$ and $\mathbf{P}_c$ respectively, is

$$\mathbf{P}_c = \mathbf{R}\Big[\mathbf{P}_w - \mathbf{t}\Big], \tag{2.13}$$

with

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}. \tag{2.14}$$

Definition 1 The camera extrinsic parameters are the elements of the translation vector $\mathbf{t}$, and the elements of the rotation matrix $\mathbf{R}$, which specify the transformation between the camera reference frame and the world reference frame [TV98].

Once the extrinsic and the intrinsic camera parameters are determined, the relationship of a point $\mathbf{P}$ from the world reference frame $\mathbf{P}_w$ to the image reference frame $\mathbf{p}$ is given, using (2.4) and (2.13), by

$$\mathbf{p} = \mathbf{K} \cdot \mathbf{R}\Big[\mathbf{P}_w - \mathbf{t}\Big], \tag{2.15}$$

with $\mathbf{p} = [x, y, z]^{\mathrm{T}}$. The image coordinates in pixels, *i.e.,* $\mathbf{p}' = [u, v, 1]^{\mathrm{T}}$ are given by:

$$\begin{cases} u = x/z \quad \Rightarrow \quad u = x/Z_c, \\ v = y/z \quad \Rightarrow \quad v = y/Z_c, \end{cases} \tag{2.16}$$

as $z = Z_c$ from (2.8). We note that $\mathbf{p} = Z_c \cdot \mathbf{p}'$. From (2.4) and (2.16), and considering (2.5), the image coordinates (u, v) are defined as

$$\begin{cases} u = \delta_x^{-1} f \frac{X_c}{Z_c} + c_x, \\ v = \delta_y^{-1} f \frac{Y_C}{Z_c} + c_y. \end{cases} \tag{2.17}$$

2.2 Two-view geometry

This section covers the geometry of two perspective views. In Section 2.1, we have presented the model of a single viewing camera and the relationship between the coordinates of a point in the scene and its image coordinates. In the following, we present the relationship between the image coordinates of a point in two viewing cameras, *i.e.,* the relative extrinsic parameters. We recall that our two-view geometry is composed of a 2-D video camera and an industrialized ToF camera, *i.e.,* a hybrid ToF multi-camera rig.

2.2.1 Relative extrinsic parameters

Whereas the extrinsic parameters introduced in Section 2.1.4 describe the position and orientation of the viewing camera with respect to the world reference frame, the relative extrinsic parameters describe the location and orientation between the cameras in the two-view geometry. Similarly to the extrinsic parameters, the relative extrinsic parameters are a translation vector $\mathbf{t}'$, that describes the relative positions between the origins of both camera frames, and a (3 × 3) rotation matrix $\mathbf{R}'$, that defines the orientation between the two camera frames [HZ03].

The relative extrinsic parameters relate the projections of a point $\mathbf{P}_w$ in space on the two image frames. We refer to the 2-D camera reference frame as $\mathcal{A}$ and to the ToF camera reference frame as $\mathcal{B}$. Thus, following (2.15), the projection of a point in the scene $\mathbf{P}_w$ on the 2-D camera image frame is

$$\mathbf{p}_{\mathcal{A}} = \mathbf{K}_{\mathcal{A}} \cdot \mathbf{R}_{\mathcal{A}} \cdot (\mathbf{P}_w - \mathbf{t}_{\mathcal{A}}). \tag{2.18}$$

Similarly, the projection of $\mathbf{P}_w$ on the ToF camera image frame is

$$\mathbf{p}_{\mathcal{B}} = \mathbf{K}_{\mathcal{B}} \cdot \mathbf{R}_{\mathcal{B}} \cdot (\mathbf{P}_w - \mathbf{t}_{\mathcal{B}}). \tag{2.19}$$

The subscripts $\mathcal{A}$ and $\mathcal{B}$ indicate that the subscripted parameter relates to the 2-D and ToF camera, respectively. Solving (2.18) and (2.19) for $\mathbf{P}_w$, we end with the following relationship

$$\mathbf{K}_{\mathcal{A}}^{\text{-1}} \cdot \mathbf{p}_{\mathcal{A}} = \mathbf{R}_{\mathcal{A}} \cdot \mathbf{R}_{\mathcal{B}}^{\text{-1}} \cdot (\mathbf{K}_{\mathcal{B}}^{\text{-1}} \cdot \mathbf{p}_{\mathcal{B}} + \mathbf{R}_{\mathcal{B}} \cdot \mathbf{t}_{\mathcal{B}} - \mathbf{R}_{\mathcal{B}} \cdot \mathbf{t}_{\mathcal{A}}). \tag{2.20}$$

If the intrinsic parameters of both cameras $\mathbf{K}_{\mathcal{A}}$ and $\mathbf{K}_{\mathcal{B}}$ are known, (2.20) amounts to

$$\mathbf{P}_{\mathcal{A}} = \mathbf{R}_{\mathcal{A}} \cdot \mathbf{R}_{\mathcal{B}}^{\text{-1}} \cdot (\mathbf{P}_{\mathcal{B}} + \mathbf{R}_{\mathcal{B}} \cdot \mathbf{t}_{\mathcal{B}} - \mathbf{R}_{\mathcal{B}} \cdot \mathbf{t}_{\mathcal{A}}) \tag{2.21}$$

using (2.4). Hence, the relative extrinsic parameters that relate $\mathbf{P}_{\mathcal{A}}$ and $\mathbf{P}_{\mathcal{B}}$ are

$$\mathbf{R}' = \mathbf{R}_{\mathcal{A}} \cdot \mathbf{R}_{\mathcal{B}}^{\text{-1}}, \tag{2.22}$$

and

$$\mathbf{t}' = \mathbf{R}_{\mathcal{B}} \cdot \mathbf{t}_{\mathcal{B}} - \mathbf{R}_{\mathcal{B}} \cdot \mathbf{t}_{\mathcal{A}} = \mathbf{R}_{\mathcal{B}} \cdot (\mathbf{t}_{\mathcal{B}} - \mathbf{t}_{\mathcal{A}}). \tag{2.23}$$

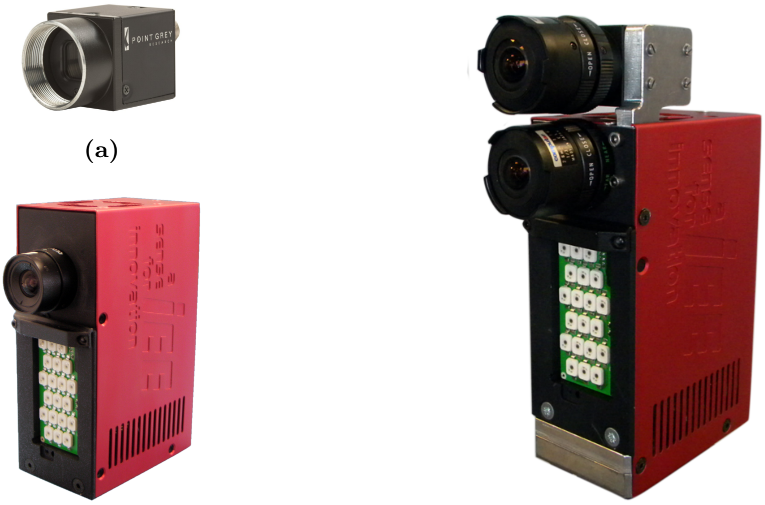
Finally, by replacing (2.22) and (2.23) in (2.21) we obtain

$$\mathbf{P}_{\mathcal{A}} = \mathbf{R}' \cdot (\mathbf{P}_{\mathcal{B}} + \mathbf{t}'). \tag{2.24}$$

In Section 2.4.2 we detail how to estimate $\mathbf{R}'$ and $\mathbf{t}'$.

2.3 Hybrid ToF multi-camera rig

In the following, we introduce the two hybrid ToF multi-camera rig prototypes that have been used to evaluate our concepts. The major part of the experiments was obtained using a first test rig prototype that integrates a 3D MLI Sensor$^{\text{TM}}$prototype from IEE S.A. [IEE11] and a Flea®2 CCD Camera from Point Grey Research, Inc. [Poi11], shown in Figure 2.3c. This first prototype couples the two cameras with a narrow baseline of 36 mm, which corresponds to the minimum baseline allowed by the dimensions of the cameras. We decided to fill the lens mount of the Flea®2 camera by 5 mm in order to convert it from C-mount to a CS-mount lens[1]. This enabled us to use the same lens in both cameras and thus, to share the same intrinsic parameters (Section 2.1.2), which are easier to determine from the 2-D camera due to its higher resolution.



(a)

(b)

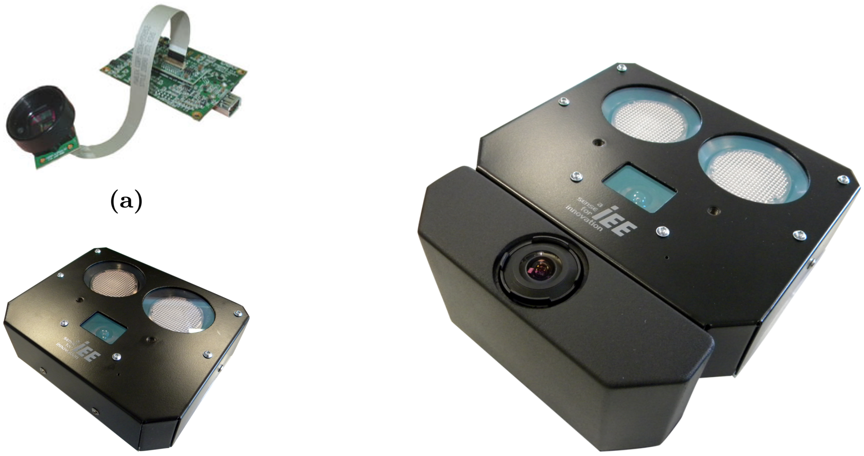
(c) Hybrid ToF multi-camera rig prototype

Figure 2.3: Former hybrid ToF multi-camera rig prototype and its components: (a) Flea®2 CCD Camera from Point Grey Research, Inc. [Poi11]. (b) 3D MLI Sensor$^{\text{TM}}$prototype from IEE S.A. [IEE11].

Once first results on depth enhancement were validated, a second ToF multi-camera rig prototype was built. This second test rig prototype integrates an IEE S.A. industrialized ToF camera, the 3D MLI Sensor$^{\text{TM}}$, and a Dragonfly®2 CCD Camera from Point Grey Research, Inc. The 2-D camera was changed because the Dragonfly®2 has

[1] A C-mount is a type of lens mount that has a flange focal distance of 25.4 mm (1 inch) diameter, and is otherwise identical to the CS-mount (12.50 mm, 0.492 inches).

a remote head that facilitates its integration within the 3D MLI Sensor$^{\text{TM}}$housing, as shown in Figure 2.4c. However, the larger dimensions of the 3D MLI Sensor$^{\text{TM}}$housing restrict the minimum baseline to 65 mm, which makes the handling of the distance-dependent disparity more challenging (Section 3.1). In contrast to its prototype, the industrialized 3D MLI Sensor$^{\text{TM}}$does not allow to change the lens. Thus, it is possible to use the intrinsic camera parameters accurately determined during the serial production by IEE S.A. The specifications of each camera are reported in Appendix A.



(a)

(b)

(c) Hybrid ToF multi-camera rig prototype

Figure 2.4: Second hybrid ToF multi-camera rig prototype and its components: (a) Dragonfly®2 CCD Camera from Point Grey Research, Inc. (b) 3D MLI Sensor$^{\text{TM}}$from IEE S.A.

2.4 Proposed system calibration

In order to complete the data alignment process, both intrinsic camera parameters as well as the relative extrinsic parameters that relate the camera position and orientation to each other have to be determined. To that end, a first calibration step of the hybrid ToF multi-camera rig must be done. This classical first step in computer vision allows to correct or rectify the raw distorted images [TV98] which will enable data matching.

In order to determine the 2-D camera parameters, one can resort to classical calibration tools such as Bouguet's toolbox for Matlab [Bou09] or image processing tools such as those included in Intel's computer vision library *OpenCV* [BK08]. Although new insights have been proposed in [LKR08], the research on ToF camera calibration is not yet extensive, and the same 2-D calibration approaches are applied on ToF cameras. These calibration approaches are suitable when calibrating “high-resolution” ToF cameras such as those intended for research purposes (see Figure 1.2c and Figure 1.2d), but the same approaches are not suitable for lower resolution images such as those provided by industrialized ToF cameras (see Figure 1.2a and Figure 1.2b). Moreover, these calibration methods are time consuming.

In the following, we propose an alternative approach to standard calibration methods [HS97, Tsa87, Zha00] that allows to estimate the system parameters, *i.e.,* intrinsic and relative extrinsic camera parameters, under industrial time and accuracy requirements.

2.4.1 Estimation of the intrinsic camera parameters

The current literature in ToF camera calibration does not tackle the lateral calibration, *i.e.,* the estimation of the intrinsic and extrinsic camera parameters. Instead, research focuses on the depth calibration, which consists of improving the accuracy and reducing the noise level on the given depth measurements [FH08, KI07, Lin10]. The limitations, *i.e.,* low resolution and high noise level within depth measurements, of our industrialized ToF camera as well as the industrial requirements motivated us to investigate a practical calibration approach to estimate the intrinsic ToF camera parameters. We note that the same approach also applies to the 2-D camera. According to Tsai [Tsa87], a coplanar set of control points is sufficient to determine the intrinsic camera parameters. We therefore assume a planar calibration pattern with a known orientation. We refer to this calibration method over multi-view calibration techniques [Bou09, Zha00] which search for a global optimum of both extrinsic and intrinsic parameters. Since the low resolution of ToF cameras only allows for the detection of a few features or control points per acquisition, by fixing the extrinsic parameters we make the solution for the intrinsics more stable, which is crucial for our system calibration. If the plane that contains the control points is parallel to the image frame, the Z coordinate is equal for all control points and known. The X and Y coordinates are also assumed to be known but up to an offset λ_x and λ_y with respect to the unknown principal point (c_x, c_y). Hence, (2.9) and (2.10) take the following form, respectively,

$$X = \lambda_x + c_0\tilde{x} + c_1\tilde{x}r^2 + c_2\tilde{x}r^4 + \dots \tag{2.25}$$

and

$$Y = \lambda_y + c_0\tilde{y} + c_1\tilde{y}r^2 + c_2\tilde{y}r^4 + \dots, \tag{2.26}$$

with

$$c_0 = \frac{Z}{f}, \quad c_i = \frac{Z}{f}k_i, \quad i = 1, 2, \dots. \tag{2.27}$$

Equations (2.25) and (2.26) are two overdetermined linear equations that can be solved for unknown parameters $c_0, c_1, c_2, \dots$ by least square regression. This can be achieved by solving either of the two equations. Thereby, the following points should be considered.

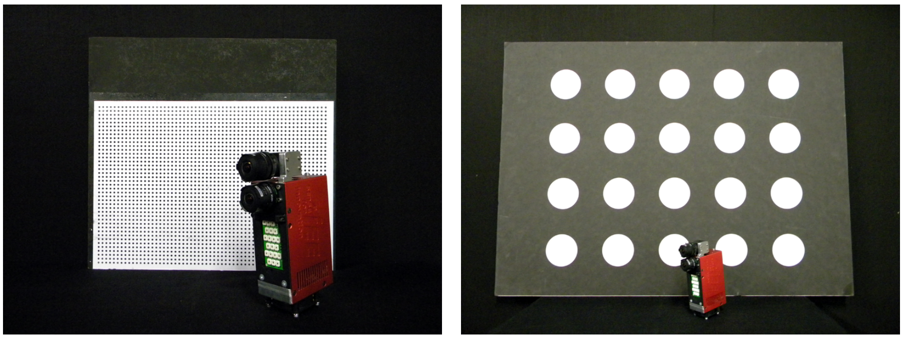
1. Solving (2.25) or (2.26) requires the knowledge of the principal point (c_x, c_y). We propose to use, in turn, the least square regression for the parameters to determine also the principal point. This can be simply achieved by varying (c_x, c_y) till the residual of the regression is minimal.

2. The proposed method assumes a calibration board perfectly parallel to the image plane. In order to verify the robustness of the calibration against a non-perfect alignment of the calibration board, a simulation of a tilted board has been performed at various angles. The result for an assumed 90° optics with 30% distortion showed that at a tilt of 5°, the distortion is still correctly determined with a sufficient accuracy of 0.3%, while the determined principal point has been shifted by 1% of the imager size due to the tilt. We note that a non-accurate principle point is, for our purpose, not critical as these deviations will be corrected in the subsequent calibration steps, *i.e.,* the estimation of the relative extrinsic parameters proposed in Section 2.4.2.

3. Equation (2.27) allows to determine the focal length whenever the distance Z is known. As the precision with which Z is known is limited, one can maximise the accuracy by taking measurements at a very large distance compared to the focal length (and the board accordingly large) or taking measurements at several distances.

We found that for calibrating the 2-D camera, two sets of co-linear control points (*e.g.,* along the x and y axes) are sufficient to determine the distortion parameters and also, an accurate principal point.

We point out that our calibration procedure does not require special tools. The ToF calibration pattern must contain circular targets, large enough to be distinguished



(a) 2-D camera calibration pattern. **(b)** ToF camera calibration pattern.

Figure 2.5: Calibration patterns used to estimate the intrinsic and relative extrinsic camera parameters.

in the low-resolution amplitude image. We remind the reader that in addition to the depth map $\mathbf{D}$, ToF cameras also provide an amplitude image $\mathbf{A}$ that results from (1.8) and which can be considered as a grayscale intensity image for calibration together with the 2-D image $\mathbf{I}$ given by the 2-D camera. Figure 2.5 shows the calibration patterns to estimate each camera's intrinsic parameters.

2.4.2 Estimation of the relative extrinsic parameters

From [RZFM92], the determination of the relative extrinsic parameters introduced in Section 2.2.1, *i.e.,* the rotation matrix $\mathbf{R}'$ and the translation vector $\mathbf{t}'$ that relate each camera to each other, requires four correspondence points with no three points collinear on either plane. The more correspondence points we consider, the more accurate will be the determination of the relative extrinsic parameters since inaccuracies due to the detection of the correspondence points will be compensate. To that end, we use the same ToF calibration pattern shown in Figure 2.5b as it allows to estimate up to 20 correspondence points. The correspondence points correspond to the centroid of each dot in the image, which are determined with sub-pixel accuracy, only limited by the image resolution as shown in Figure 2.6b and Figure 2.6d. Nevertheless, our case differs from common stereo vision calibration approaches due to the knowledge of the $Z_{\mathcal{B}}$ coordinate of the projections of the correspondence points on the ToF camera frame. According to (2.17), the image coordinates $\mathbf{p}_{\mathcal{A}}$ and $\mathbf{p}_{\mathcal{B}}$ of the correspondence points on the 2-D $\mathbf{I}$ and ToF amplitude $\mathbf{A}$ images, respectively, are determined up to a scale factor, *i.e.,*

$$\mathbf{p}_{\mathcal{A}} = \begin{bmatrix} u_{\mathcal{A}} \cdot Z_{\mathcal{A}} \\ v_{\mathcal{A}} \cdot Z_{\mathcal{A}} \\ Z_{\mathcal{A}} \end{bmatrix} = Z_{\mathcal{A}} \cdot \begin{bmatrix} u_{\mathcal{A}} \\ v_{\mathcal{A}} \\ 1 \end{bmatrix}, \tag{2.28}$$

and

$$\mathbf{p}_{\mathcal{B}} = \begin{bmatrix} u_{\mathcal{B}} \cdot Z_{\mathcal{B}} \\ v_{\mathcal{B}} \cdot Z_{\mathcal{B}} \\ Z_{\mathcal{B}} \end{bmatrix} = Z_{\mathcal{B}} \cdot \begin{bmatrix} u_{\mathcal{B}} \\ v_{\mathcal{B}} \\ 1 \end{bmatrix}, \tag{2.29}$$

where $Z_{\mathcal{B}}$ is known as it is given by the ToF depth map $\mathbf{D}$. Let denote $\mathbf{p}'_{\mathcal{A}} = [u_{\mathcal{A}}, v_{\mathcal{A}}, 1]^{\mathrm{T}}$ and $\mathbf{p}'_{\mathcal{B}} = [u_{\mathcal{B}}, v_{\mathcal{B}}, 1]^{\mathrm{T}}$ the image coordinates of the correspondence points on the 2-D $\mathbf{I}$ and ToF amplitude $\mathbf{A}$ images, respectively. From (2.24) and replacing $\mathbf{P}_{\mathcal{A}}$ and $\mathbf{P}_{\mathcal{B}}$ by their definition in (2.4), we obtain

$$\mathbf{K}_{\mathcal{A}}^{-1}\mathbf{p}_{\mathcal{A}} = \mathbf{R}'(\mathbf{K}_{\mathcal{B}}^{-1}\mathbf{p}_{\mathcal{B}} + \mathbf{t}') \quad \Rightarrow \quad \mathbf{K}_{\mathcal{A}}^{-1} Z_{\mathcal{A}}\mathbf{p}'_{\mathcal{A}} = \mathbf{R}'(\mathbf{K}_{\mathcal{B}}^{-1} Z_{\mathcal{B}}\mathbf{p}'_{\mathcal{B}} + \mathbf{t}'). \tag{2.30}$$

Finding the value for $\mathbf{p}'_{\mathcal{A}}$ in (2.30), we obtain

$$\mathbf{p}'_{\mathcal{A}} = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{A}}}\mathbf{K}_{\mathcal{A}}\mathbf{R}'\mathbf{K}_{\mathcal{B}}^{-1}\Big[\mathbf{p}'_{\mathcal{B}} + \frac{\mathbf{K}_{\mathcal{B}}}{Z_{\mathcal{B}}}\mathbf{t}'\Big], \tag{2.31}$$

where $(\mathbf{K}_{\mathcal{B}}/Z_{\mathcal{B}})\mathbf{t}'$ corresponds to the disparity ρ correction applied to the correspondence point coordinates $\mathbf{p}'_{\mathcal{B}}$. $\mathbf{t}' = [t_x, t_y, t_z]^{\mathrm{T}}$ is the vectorial baseline b between the cameras. We note that (2.31) is the generalization of the known disparity expression in (1.2) obtained by assuming that both cameras have the same intrinsic parameters $K_{\mathcal{A}} = K_{\mathcal{B}}$ and setting $\mathbf{R}'$ to identity, *i.e.,* assuming that the two cameras are equally oriented.

We remark that the two cameras in the camera rig are coplanar, thus $t_z = 0$. Let denote $\mathbf{p}''_{\mathcal{B}}$ as the correspondence point coordinates corrected by the disparity shift, thus

$$\mathbf{p}''_{\mathcal{B}} = \begin{bmatrix} u_{\mathcal{B}} \\ v_{\mathcal{B}} \\ 1 \end{bmatrix} + \frac{1}{Z_{\mathcal{B}}}\begin{bmatrix} \delta_{x,\mathcal{B}}^{-1} f_{\mathcal{B}} & 0 & c_{x,\mathcal{B}} \\ 0 & \delta_{y,\mathcal{B}}^{-1} f_{\mathcal{B}} & c_{y,\mathcal{B}} \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} t_x \\ t_y \\ 0 \end{bmatrix} = \begin{bmatrix} u_{\mathcal{B}} \\ v_{\mathcal{B}} \\ 1 \end{bmatrix} + \begin{bmatrix} \delta_{x,\mathcal{B}}^{-1} f_{\mathcal{B}} \cdot t_x/Z_{\mathcal{B}} \\ \delta_{y,\mathcal{B}}^{-1} f_{\mathcal{B}} \cdot t_y/Z_{\mathcal{B}} \\ 0 \end{bmatrix}. \tag{2.32}$$

By replacing (2.32) in (2.31) we obtain

$$\mathbf{p}'_{\mathcal{A}} = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{A}}}\mathbf{K}_{\mathcal{A}}\mathbf{R}'\mathbf{K}_{\mathcal{B}}^{-1}\mathbf{p}''_{\mathcal{B}} = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{A}}}\mathbf{H}\mathbf{p}''_{\mathcal{B}}, \tag{2.33}$$

with $\mathbf{H}$ being the so-called homography that relates the image coordinates $\mathbf{p}'_{\mathcal{A}}$ with the ones corrected by the disparity shift $\mathbf{p}''_{\mathcal{B}}$, defined in (2.32). We note that in our case, the homography $\mathbf{H}$ corresponds to an affine transformation since the z coordinate of $\mathbf{p}'_{\mathcal{A}}$ and $\mathbf{p}''_{\mathcal{B}}$ is 1, *i.e.,*

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.34}$$

Thus, only 3 correspondence points are needed to determine $\mathbf{H}$ in contrast to the 4 correspondence points needed to estimate a full projective transformation with 9 unknowns. Also, we compensate possible inaccuracies due to the intrinsic camera parameters determined in Section 2.4.1 when determining the affine transformation $\mathbf{H}$. We assume the ratio $Z_{\mathcal{B}}/Z_{\mathcal{A}}$ to be constant as $Z_{\mathcal{A}}$ and $Z_{\mathcal{B}}$ are the same for all correspondence points on each camera image frame $\mathcal{A}$ and $\mathcal{B}$, respectively. The possible error when determining the focal length of the ToF camera $f_{\mathcal{B}}$ can be neglected as $f_{\mathcal{B}} << Z_{\mathcal{B}}$ when correcting the disparity in (2.32). We also note that in the ideal case, *i.e.,* where the two cameras are equally oriented and only shifted by the baseline, the affine transformation $\mathbf{H}$ would correspond to a (3×3) identity matrix $\mathbf{I}_3$.

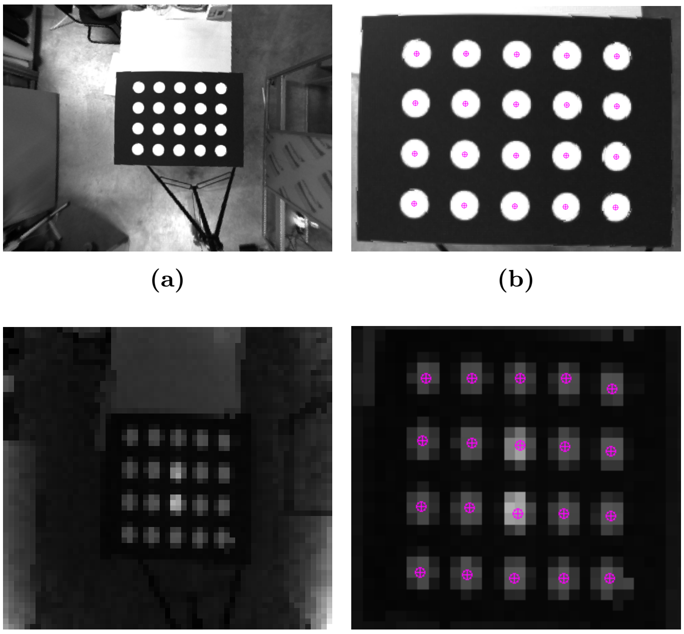


Figure 2.6: ToF calibration pattern images recorded by the 2-D and the ToF camera, (a) and (c) respectively, to estimate the relative extrinsic parameters. The centroid operator detects with sub-pixel accuracy the centroid of each target, shown in red in (b) and (d).

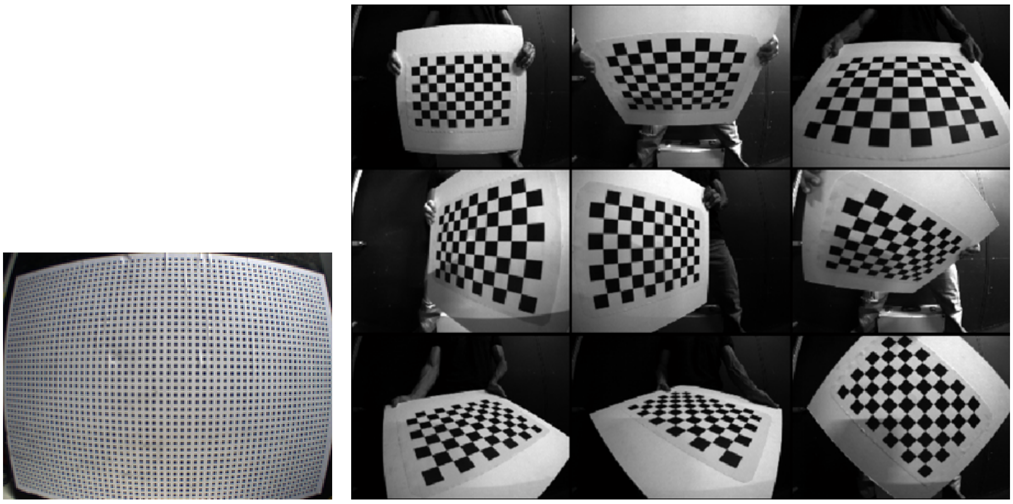
To summarize, our calibration approach determines the intrinsic camera parameters of each camera in the rig by using a single acquisition of the calibration patterns shown in Figure 2.5. We assume the calibration patterns to be located parallel to each camera image frame and at a known distance along the optical axis. In order to determine the relative extrinsic parameters, we just need three correspondence points to determine the homography that relates pixel coordinates from one camera to the other. The same pattern to determine the intrinsic camera parameters of the ToF camera can be used. In the following, we present some experimental results that quantify and qualify our practical calibration approach.

2.4.3 Experimental results

We note that the experiments have been performed using the data recorded by the second hybrid ToF multi-camera rig prototype introduced in Section 2.3. The specifications of the cameras that constitute the second hybrid ToF multi-camera rig such as pixel size δ or pixel resolution can be found in the Appendix A. In the following, we focus on the practicability and accuracy of our concept and we compare our results with the ones obtained by the commonly used Bouguet's calibration toolbox [Bou09].

Our calibration approach requires first the calibration pattern to be installed parallel to the camera image frame and second to fix the distance at which the pattern is located with respect to the camera. By accomplishing these constraints, the intrinsic camera parameters are directly determined from one input image as shown in Figure 2.7a. In contrast, Bouguet's technique searches for a global optimum of both intrinsic and extrinsic camera parameters, which requires a minimum of two input images with different orientations of the calibration pattern, as shown in Figure 2.7b, and is thus time consuming. As shown in Figure 2.8, in addition to the intrinsic camera parameters, Bouguet's technique also determines the external camera parameters that relate each acquisition of the calibration pattern with the viewing camera. The figure shows how the multiple acquisitions in Figure 2.7b are oriented and located with respect to the viewing camera reference frame O_c (depicted by a red pyramid). In contrast, our concept can be automated for a mass calibration process as only one image acquisition with a known position and orientation is required.

We next compare the results obtained with our calibration concept with the results obtained using Bouguet's toolbox. Figure 2.9 shows the 2-D calibration pattern in Fig-



(a) **(b)**

Figure 2.7: Input image(s) to estimate the 2-D intrinsic camera parameters. (a) Single image used in our calibration approach. (b) Multiple acquisitions required for Bouguet's [Bou09] calibration.

ure 2.7a undistorted using our approach (Figure 2.9a) and undistorted using Bouguet's calibration toolbox (Figure 2.9b). We determine the coordinates of the centroid of each dot and we overlap them together with the ground truth grid of centroids, depicted in black crosses (+) in Figure 2.10. We evaluate the accuracy of each technique by measuring how much off the undistorted centroids are from the ground truth. In our case, the maximum distance between a centroid and its ground truth coordinates is 6.92 pixels while using Bouguet's calibration toolbox is 10.75 pixels. The mean distance between all pairs of centroids and ground truth coordinates is 2.10 pixels while using Bouguet's toolbox is 2.38 pixels. We have seen that our calibration approach is able to correct the distortion introduced by our optical lenses. It thus, enables to estimate the relationship between both cameras. In addition and for this concrete setup, our estimated parameters are more accurate than the ones estimated with Bouguet's calibration toolbox.

With regards to the relative extrinsic parameters, their accuracy is linked to the accuracy with which the centroid of each control point from the relative calibration pattern has been estimated (see Figure 2.6b and Figure 2.6d). In the case where a control point appears as only one pixel, the centroid will be the image coordinates of this pixel

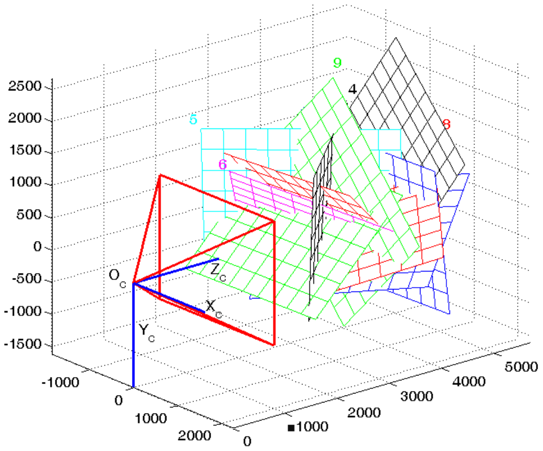


Figure 2.8: Orientation and location (extrinsic parameters) of the multiple calibration pattern acquisitions in Figure 2.7b estimated with Bouguet's toolbox for Matlab [Bou09].

and therefore will induce a discretization error in the interval $[-\delta/2, \delta/2]$. Assuming that the discretization error is statistically equally distributed over that interval, one can easily calculate the Root Mean Square Error (RMSE) to be $\Delta = \delta/\sqrt{12}$. When a dot appears as a blob of N pixels, one obtains a RMSE of

$$\Delta = \frac{1}{\sqrt{12N}}\delta, \tag{2.35}$$

which is then more accurate than when using edges, *i.e.*, $\Delta = \delta/2$. The relative calibration pattern is located at a distance of 1530 mm from the sensing system and roughly positioned in the centre of the FOV (see Figure 2.6c). In addition, we consider the 20 control points in order to obtain a maximum accuracy. Thereby, we take as reference the positions detected in the 2-D image **I**. In the ToF amplitude image **A**, the average size of the detected dots is 7.7 pixels, yielding, according to (2.35), a sub-pixel accuracy of the centroid of $\Delta_x = 7.1$ μm and $\Delta_y = 5.1$ μm. We note that the pixel size of the ToF camera is $\delta_x = 68$ μm and $\delta_y = 49$ μm. The RMSE of the centroid coordinates after relating the centroid coordinates in **A** with the centroid coordinates in **I** is 5.4 μm in the x direction and 7.9 μm in the y direction.

We can confirm that our centroid operator achieves an accuracy of the same order as the one given by (2.35), which is clearly better than the low resolution of the ToF camera and close to 1 pixel of the 2-D camera resolution, which is (7.4 μm × 7.4 μm).

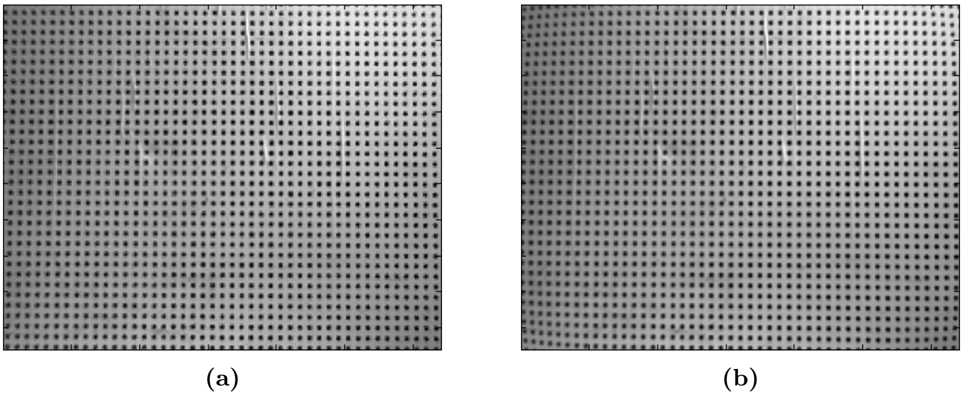


Figure 2.9: Undistorted 2-D calibration pattern in Figure 2.7a using (a) our approach and (b) Bouguet's calibration toolbox.

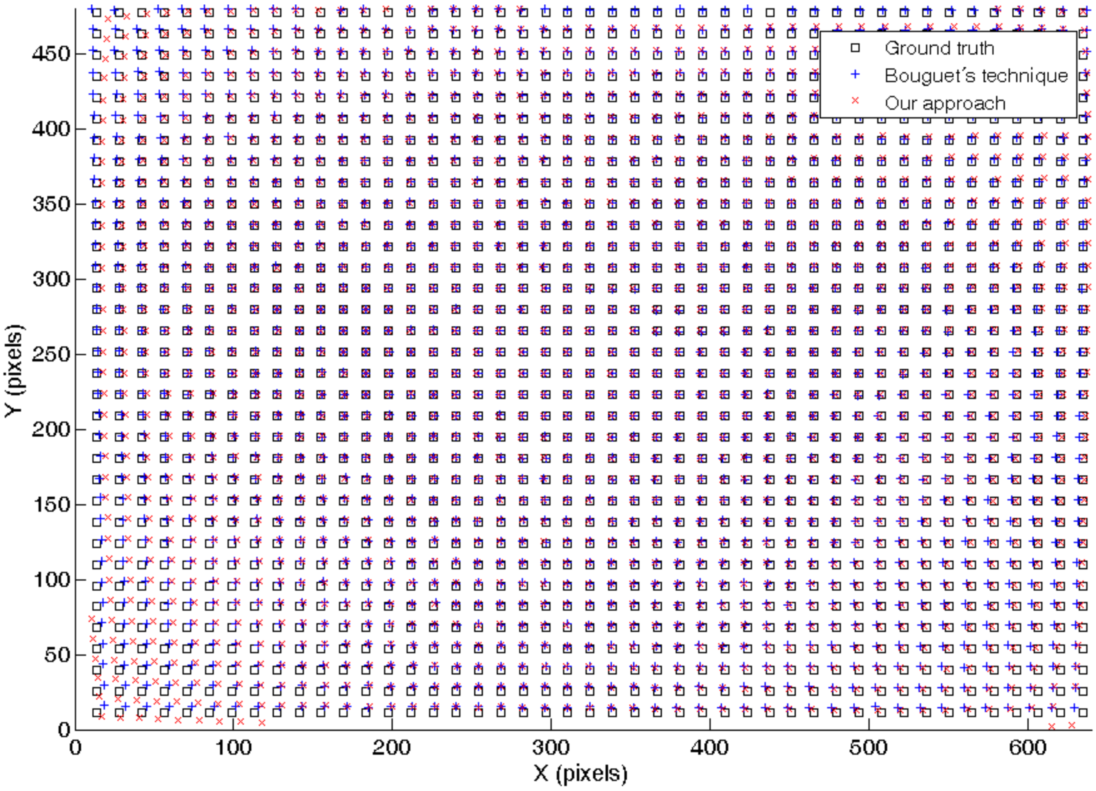


Figure 2.10: Overlapping of the dot centroids after undistorting using our approach (*) and Bouguet's approach (x) onto the ground truth centroids (+).

Chapter 3

Data matching

In this chapter, we propose an original framework to align the data recorded by each of the cameras that constitute the hybrid ToF multi-camera rig. We first introduce the problem of disparity due to the displacement between the camera centres. Then, we present a unified reference frame where the recorded data by each of the cameras will be mapped in order to be pixel aligned. We propose a real-time implementation by an iterative algorithm that considers associative arrays, *i.e.,* look-up tables, that relates non-mapped and mapped image coordinates. Finally, we present some experimental results to quantify the accuracy between mapped image coordinates. We note that our method is not only intended to map the data from low-resolution ToF cameras but conceptually applies also to other 3-D sensing modalities such as the recently emerging laser scanners, *i.e.,* the ibeo LUX [Ibe11] or the Eco Scan FX8 [Nip11] whose resolutions are also far below the resolutions of standard 2-D cameras.

3.1 Distance-dependent disparity

In general, the two reference frames of each individual camera constituting a hybrid ToF multi-camera rig are not co-centric, *i.e.,* the centres of projection of each camera are displaced by a baseline b, similarly to the stereo vision system presented in Figure 1.1. Indeed, in this case one of the two 2-D cameras is replaced by a ToF camera. Thus, the projections of a point $\mathbf{P}$ in space onto each camera image frame and with respect to each camera's principal point, also differs by the *binocular disparity* introduced in Section 1.1 and generalized to our setup in Section 2.4.2. We remark that we referred to each camera's reference frame as $\mathcal{A}$ for the 2-D camera and $\mathcal{B}$ for the ToF camera in Section 2.2.1. In stereo vision systems, the disparity leads to the estimation of the distance Z at which the point $\mathbf{P}$ is located in the scene (see Figure 1.1) [TV98]. However, this requires finding the feature-correspondence pairs that in general result from the solution of the correspondence problem [HZ03, TV98]. In contrast, in our case the problem is reversed. The ToF camera measures the distance at which the point $\mathbf{P}$ is located with respect to its reference frame $\mathcal{B}$, *i.e.,* $Z_{\mathcal{B}}$ and thus, allows to estimate the disparity $\rho(Z_{\mathcal{B}})$ for each of the ToF camera pixels, as discussed in Section 2.4.2.

We note that the relationship between the $Z_{\mathcal{B}}$ measurements and the disparity $\rho(Z_{\mathcal{B}})$ causes a dependency on the scene. Therefore, it has to be recalculated whenever the scene changes, which is typically the case for every frame of data acquisition, and for each ToF camera pixel as it is not constant for all pixel locations. By differentiating disparity $\rho(Z_{\mathcal{B}})$ in (1.2) with respect to the distance $Z_{\mathcal{B}}$, we define the absolute disparity variation $\Delta\rho(Z_{\mathcal{B}})$ as a function of the absolute depth variation $\Delta Z_{\mathcal{B}}$, and obtain

$$\Delta\rho(Z_{\mathcal{B}}) = f_{\mathcal{B}} b \frac{\Delta Z_{\mathcal{B}}}{Z_{\mathcal{B}}^2}, \tag{3.1}$$

where $f_{\mathcal{B}}$ is the focal length of the ToF camera and b the baseline between the camera centres. We note that only in situations where the depth variation of the object in the scene $\Delta Z_{\mathcal{B}}$ is small enough compared to the squared distance $Z_{\mathcal{B}}^2$ from the object to the system, the disparity $\rho(Z_{\mathcal{B}})$ can be assumed as constant and thus, included in a simple projective transformation for all recorded frames. Actually, this scenario is commonly used in research efforts that integrate non-industrial ToF cameras such as the SwissRanger$^{\text{TM}}$ToF camera, in their ToF multi-camera rig (Figure 1.2d) [CBTT08, KCTT08, KTD$^+$09]. In this case, the rather small field of view provided by the SwissRanger$^{\text{TM}}$camera, *i.e.,* 47.5° × 39.6°, forces such systems to be installed at a relatively large distance from the object. As a consequence, these systems can still function while neglecting the distance-dependent disparity, which is not the case for the majority of ToF cameras, which require the variation of disparity to be taken into account. In what follows we propose to solve this problem by defining a new matching procedure that exploits the distance-dependent disparity. As a result, any ToF camera available on the market can be integrated in a hybrid ToF multi-camera rig intended for low-level data fusion regardless of its specifications.

3.2 Unified reference frame

A hybrid ToF multi-camera rig provides multi-modal data. Thus, a 2-D image **I** related to the reference frame $\mathcal{A}$ and a pair of depth **D** and amplitude **A** images related to the reference frame $\mathcal{B}$ are delivered. We denote the image coordinates of a point in image **I** as $(u^{\mathbf{I}}_{\mathcal{A}}, v^{\mathbf{I}}_{\mathcal{A}})$. Accordingly, the image coordinates of a point in images **D** or **A** are denoted as $(u^{\mathbf{D}}_{\mathcal{B}}, v^{\mathbf{D}}_{\mathcal{B}})$. We remark that these image coordinates have been distortion corrected from (2.9) and (2.10) by using the intrinsic camera parameters estimated during the calibration process (see Section 2.4). To achieve the low-level data matching required for data fusion, we proceed by transforming these image coordinates to a unified reference frame $\mathcal{C}$, which is the basis for the data matching (or warping) described in Section 3.3. This transformation will allow to establish a mapping of the data recorded by each camera to a unique coordinate grid on $\mathcal{C}$, where the mapped images are pixel aligned, and ready to be fused.

3.2.1 Choice of the unified reference frame

The image coordinates $\mathbf{p}'_{\mathcal{A}} = [u^{\mathbf{I}}_{\mathcal{A}}, v^{\mathbf{I}}_{\mathcal{A}}, 1]^{\mathrm{T}}$ of a point $\mathbf{P}_{\mathcal{A}} = [X_{\mathcal{A}}, Y_{\mathcal{A}}, Z_{\mathcal{A}}]^{\mathrm{T}}$ related to the 2-D camera reference frame $\mathcal{A}$ are transformed to the unified reference frame $\mathcal{C}$ using (2.31), *i.e.,*

$$\mathbf{p}'_{\mathcal{C}} = \frac{Z_{\mathcal{A}}}{Z_{\mathcal{C}}}\mathbf{K}_{\mathcal{C}}\mathbf{R}_{\mathcal{A}\mathcal{C}}\mathbf{K}^{-1}_{\mathcal{A}}\Big[\mathbf{p}'_{\mathcal{A}} + \frac{\mathbf{K}_{\mathcal{A}}}{Z_{\mathcal{A}}}\mathbf{t}_{\mathcal{A}\mathcal{C}}\Big], \tag{3.2}$$

with $\mathbf{R}_{\mathcal{A}\mathcal{C}}$ and $\mathbf{t}_{\mathcal{A}\mathcal{C}}$ the rotation matrix and translation vector from the reference frame $\mathcal{A}$ to the reference frame $\mathcal{C}$, respectively. Since the image transformation in (3.2) requires the knowledge of the coordinate $Z_{\mathcal{A}}$, we choose the unified reference frame $\mathcal{C}$ to be co-centric to the 2-D camera reference frame $\mathcal{A}$, *i.e.,* $\mathbf{t}_{\mathcal{A}\mathcal{C}} = [0, 0, 0]^{\mathrm{T}}$. Hence, (3.2) amounts to

$$\mathbf{p}'_{\mathcal{C}} = \mathbf{K}_{\mathcal{C}}\mathbf{R}_{\mathcal{A}\mathcal{C}}\mathbf{K}^{-1}_{\mathcal{A}}\mathbf{p}'_{\mathcal{A}} =: \mathbf{H}_{\mathcal{A}\mathcal{C}} \cdot \mathbf{p}'_{\mathcal{A}}, \tag{3.3}$$

with $\mathbf{H}_{\mathcal{A}\mathcal{C}}$ a plane-to-plane transformation or projective transformation from reference frame $\mathcal{A}$ to reference frame $\mathcal{C}$. Similarly, the transformation of the image coordinates of a point $\mathbf{p}'_{\mathcal{B}}$, related to the ToF camera reference frame $\mathcal{B}$, to the unified reference frame $\mathcal{C}$ is analogous. Using (2.31), we find

$$\mathbf{p}'_{\mathcal{C}} = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{C}}}\mathbf{K}_{\mathcal{C}}\mathbf{R}_{\mathcal{B}\mathcal{C}}\mathbf{K}^{-1}_{\mathcal{B}}\Big[\mathbf{p}'_{\mathcal{B}} + \frac{\mathbf{K}_{\mathcal{B}}}{Z_{\mathcal{B}}}\mathbf{t}_{\mathcal{B}\mathcal{C}}\Big] = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{C}}}\mathbf{H}_{\mathcal{B}\mathcal{C}}\Big[\mathbf{p}'_{\mathcal{B}} + \frac{\mathbf{K}_{\mathcal{B}}}{Z_{\mathcal{B}}}\mathbf{t}_{\mathcal{B}\mathcal{C}}\Big], \tag{3.4}$$

where $\mathbf{R}_{\mathcal{B}\mathcal{C}}$ and $\mathbf{t}_{\mathcal{B}\mathcal{C}}$ are the rotation matrix and the translation vector from the reference frame $\mathcal{B}$ to the reference frame $\mathcal{C}$, respectively. $\mathbf{H}_{\mathcal{B}\mathcal{C}}$ is the projective transformation from reference frame $\mathcal{B}$ to reference frame $\mathcal{C}$. We note that in this case the distance $Z_{\mathcal{B}}$ is known as it results from

$$Z_{\mathcal{B}} = \mathbf{D}(u_{\mathcal{B}}^{\mathbf{D}}, v_{\mathcal{B}}^{\mathbf{D}}) \cdot \frac{f_{\mathcal{B}}}{d(u_{\mathcal{B}}^{\mathbf{D}}, v_{\mathcal{B}}^{\mathbf{D}})}, \tag{3.5}$$

with

$$d(u_{\mathcal{B}}^{\mathbf{D}}, v_{\mathcal{B}}^{\mathbf{D}}) = \sqrt{f_{\mathcal{B}}^2 + \big(\delta_{x,\mathcal{B}}(u_{\mathcal{B}}^{\mathbf{D}} - c_{x,\mathcal{B}})\big)^2 + \big(\delta_{y,\mathcal{B}}(v_{\mathcal{B}}^{\mathbf{D}} - c_{y,\mathcal{B}})\big)^2}. \tag{3.6}$$

Since each pixel in $\mathbf{D}$ corresponds to a radial measurement, the conversion in (3.5) is therefore necessary to obtain the distance $Z_{\mathcal{B}}$ that relates to each pixel $(u_{\mathcal{B}}^{\mathbf{D}}, v_{\mathcal{B}}^{\mathbf{D}})$ in $\mathbf{D}$. This in turn allows the transformation of the image coordinates from the reference frame $\mathcal{B}$ to the reference frame $\mathcal{C}$.

3.2.2 Distance-dependent disparity shift

The transformation of the image coordinates in (3.4) consists of two steps. The first step concerns the binocular disparity shift, *i.e.,*

$$\mathbf{p}''_{\mathcal{B}} = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{B}} - t_z}\Big[\mathbf{p}'_{\mathcal{B}} + \frac{\mathbf{K}_{\mathcal{B}}}{Z_{\mathcal{B}}}\mathbf{t}_{\mathcal{B}\mathcal{C}}\Big], \tag{3.7}$$

followed by the the projective transformation $\mathbf{p}'_{\mathcal{C}} = Z_{\mathcal{B}}/Z_{\mathcal{C}} \cdot \mathbf{H}_{\mathcal{B}\mathcal{C}}\mathbf{p}''_{\mathcal{B}}$. The factor $Z_{\mathcal{B}}/(Z_{\mathcal{B}} - t_z)$ (t_z is the third component of the vector $\mathbf{t}_{\mathcal{B}\mathcal{C}} = [t_x, t_y, t_z]^{\mathrm{T}}$) in (3.7) makes $\mathbf{p}''_{\mathcal{B}}$ to be in homogeneous coordinates, *i.e.,* $\mathbf{p}''_{\mathcal{B}} = [u'^{\mathbf{D}}_{\mathcal{B}}, v'^{\mathbf{D}}_{\mathcal{B}}, 1]^{\mathrm{T}}$. For our setup, we may neglect t_z, *i.e.,* $t_z \approx 0$ since the two cameras in the hybrid ToF multi-camera rig are chosen to be co-planar, *i.e.,* the rotation matrix $\mathbf{R}_{\mathcal{B}\mathcal{C}}$ is a rotation in two dimensions and $Z_{\mathcal{B}} = Z_{\mathcal{C}}$, that is, $\mathbf{H}_{\mathcal{B}\mathcal{C}}$ can be approximated by an affine transformation. As a result, (3.7) simplifies to

$$\begin{aligned}
\mathbf{p}''_{\mathcal{B}} = \begin{bmatrix} u'^{\mathbf{D}}_{\mathcal{B}} \\ v'^{\mathbf{D}}_{\mathcal{B}} \\ 1 \end{bmatrix} &= \begin{bmatrix} u^{\mathbf{D}}_{\mathcal{B}} \\ v^{\mathbf{D}}_{\mathcal{B}} \\ 1 \end{bmatrix} + \frac{1}{Z_{\mathcal{B}}} \begin{bmatrix} \delta_{x,\mathcal{B}}^{-1} f_{\mathcal{B}} & 0 & c_{x,\mathcal{B}} \\ 0 & \delta_{y,\mathcal{B}}^{-1} f_{\mathcal{B}} & c_{y,\mathcal{B}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} u^{\mathbf{D}}_{\mathcal{B}} \\ v^{\mathbf{D}}_{\mathcal{B}} \\ 1 \end{bmatrix} + \begin{bmatrix} \delta_{x,\mathcal{B}}^{-1} f_{\mathcal{B}} \cdot t_x/Z_{\mathcal{B}} \\ \delta_{y,\mathcal{B}}^{-1} f_{\mathcal{B}} \cdot t_y/Z_{\mathcal{B}} \\ 0 \end{bmatrix} = \begin{bmatrix} u^{\mathbf{D}}_{\mathcal{B}} \\ v^{\mathbf{D}}_{\mathcal{B}} \\ 1 \end{bmatrix} + \frac{f_{\mathcal{B}}}{Z_{\mathcal{B}}} \begin{bmatrix} t_x/\delta_{x,\mathcal{B}} \\ t_y/\delta_{y,\mathcal{B}} \\ 0 \end{bmatrix} \\
&=: \begin{bmatrix} u^{\mathbf{D}}_{\mathcal{B}} \\ v^{\mathbf{D}}_{\mathcal{B}} \\ 1 \end{bmatrix} + \begin{bmatrix} \rho_x(Z_{\mathcal{B}}) \\ \rho_y(Z_{\mathcal{B}}) \\ 0 \end{bmatrix} = \begin{bmatrix} u^{\mathbf{D}}_{\mathcal{B}} \\ v^{\mathbf{D}}_{\mathcal{B}} \\ 1 \end{bmatrix} + \frac{f_{\mathcal{B}}}{Z_{\mathcal{B}}} \begin{bmatrix} b_{x,\mathcal{B}} \\ b_{y,\mathcal{B}} \\ 0 \end{bmatrix},
\end{aligned} \tag{3.8}$$

which corresponds to $\mathbf{p}'_{\mathcal{B}}$ plus the binocular disparity introduced in (1.2). The possible error when determining the focal length $f_{\mathcal{B}}$ of the ToF camera can be neglected as $f_{\mathcal{B}} << Z_{\mathcal{B}}$ when correcting the disparity in (3.8). The binocular disparity in (3.8) is decomposed into two components as $\rho(Z_{\mathcal{B}}) = \rho_x(Z_{\mathcal{B}}) \cdot \vec{e}_x + \rho_y(Z_{\mathcal{B}}) \cdot \vec{e}_y$, where $\vec{e}_x$ and $\vec{e}_y$ are respectively the unit vectors along the x and y axes of the ToF reference frame $\mathcal{B}$.

We note that the order of the two previous steps can be exchanged by multiplying in (3.4) the transformation $\mathbf{H}_{\mathcal{B}\mathcal{C}}$ inside the disparity shift, *i.e.,*

$$\mathbf{p}'_{\mathcal{C}} = \frac{Z_{\mathcal{B}}}{Z_{\mathcal{C}}} \mathbf{H}_{\mathcal{B}\mathcal{C}} \mathbf{p}'_{\mathcal{B}} + \frac{\mathbf{K}_{\mathcal{C}} \mathbf{R}_{\mathcal{B}\mathcal{C}}}{Z_{\mathcal{C}}} \mathbf{t}_{\mathcal{B}\mathcal{C}} =: \mathbf{p}''_{\mathcal{C}} + \frac{\mathbf{K}_{\mathcal{C}}}{Z_{\mathcal{C}}} \mathbf{t}'_{\mathcal{B}\mathcal{C}}, \tag{3.9}$$

with the baseline $\mathbf{t}'_{\mathcal{B}\mathcal{C}} = [t'_x, t'_y, t'_z]^{\mathrm{T}}$ measured from the reference frame $\mathcal{C}$ and $\mathbf{p}'_{\mathcal{B}}$ being transformed to $\mathbf{p}''_{\mathcal{C}}$ by $\mathbf{H}_{\mathcal{B}\mathcal{C}}$. Analogously to (3.7), (3.9) simplifies to

$$\begin{aligned}
\mathbf{p}'_{\mathcal{C}} = \begin{bmatrix} u^{\mathbf{D}}_{\mathcal{C}} \\ v^{\mathbf{D}}_{\mathcal{C}} \\ 1 \end{bmatrix} &= \begin{bmatrix} u'^{\mathbf{D}}_{\mathcal{C}} \\ v'^{\mathbf{D}}_{\mathcal{C}} \\ 1 \end{bmatrix} + \frac{1}{Z_{\mathcal{B}}} \begin{bmatrix} \delta^{-1}_{x,\mathcal{C}} f_{\mathcal{C}} & 0 & c_{x,\mathcal{C}} \\ 0 & \delta^{-1}_{y,\mathcal{C}} f_{\mathcal{C}} & c_{y,\mathcal{C}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} t'_x \\ t'_y \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} u'^{\mathbf{D}}_{\mathcal{C}} \\ v'^{\mathbf{D}}_{\mathcal{C}} \\ 1 \end{bmatrix} + \begin{bmatrix} \delta^{-1}_{x,\mathcal{C}} f_{\mathcal{C}} \cdot t'_x / Z_{\mathcal{B}} \\ \delta^{-1}_{y,\mathcal{C}} f_{\mathcal{C}} \cdot t'_y / Z_{\mathcal{B}} \\ 0 \end{bmatrix} =: \begin{bmatrix} u'^{\mathbf{D}}_{\mathcal{C}} \\ v'^{\mathbf{D}}_{\mathcal{C}} \\ 1 \end{bmatrix} + \begin{bmatrix} \rho_x(Z_{\mathcal{B}}) \\ \rho_y(Z_{\mathcal{B}}) \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} u'^{\mathbf{D}}_{\mathcal{C}} \\ v'^{\mathbf{D}}_{\mathcal{C}} \\ 1 \end{bmatrix} + \frac{f_{\mathcal{C}}}{Z_{\mathcal{B}}} \begin{bmatrix} b_{x,\mathcal{C}} \\ b_{y,\mathcal{C}} \\ 0 \end{bmatrix}.
\end{aligned} \tag{3.10}$$

We see from (3.10) that image coordinates $\mathbf{p}'_{\mathcal{B}}$ are first transformed to $\mathbf{p}''_{\mathcal{C}}$ and then the disparity is computed using the intrinsic parameters in $\mathcal{C}$ and the distance $Z_{\mathcal{B}}$ given by the ToF camera. The values of the depth map $\mathbf{D}$ are, however, not invariant under this disparity shift, but may be recomputed according to (see equations (3.5) and (3.6))

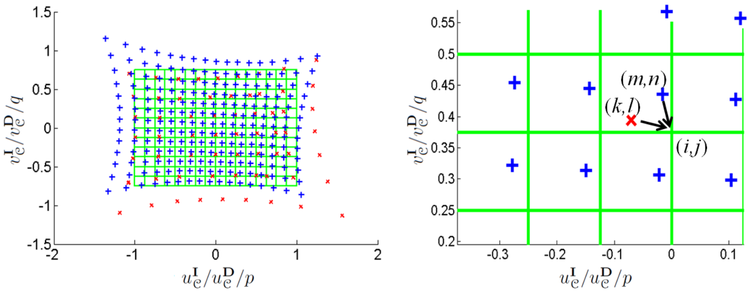
$$\mathbf{D}'(u'^{\mathbf{D}}_{\mathcal{B}}, v'^{\mathbf{D}}_{\mathcal{B}}) = Z_{\mathcal{B}} \cdot \frac{d(u'^{\mathbf{D}}_{\mathcal{B}}, v'^{\mathbf{D}}_{\mathcal{B}})}{f_{\mathcal{B}}}, \tag{3.11}$$

where $(u'^{\mathbf{D}}_{\mathcal{B}}, v'^{\mathbf{D}}_{\mathcal{B}})$ are the image coordinates shifted by the disparity, according to (3.8).

3.3 Mapping procedure for data matching

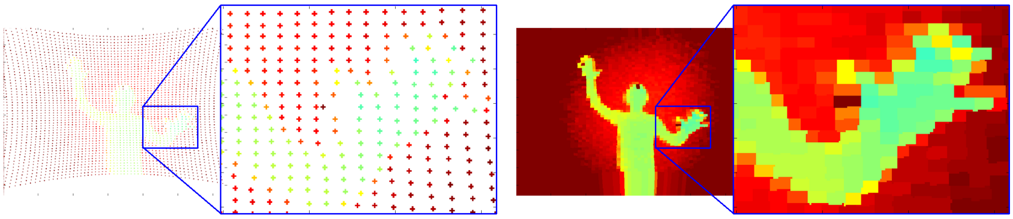
Data matching results from mapping the images $\mathbf{I}$ and $\mathbf{D}'$ on a common grid of pixels related to the reference frame $\mathcal{C}$, where the mapped images will be pixel aligned.

Let us consider $\mathbf{I}$ to be the 2-D image of (M pixels × N pixels) with image coordinates $\{(u^{\mathbf{I}}_{\mathcal{A},mn}, v^{\mathbf{I}}_{\mathcal{A},mn}),\ m = 1,\ldots,M;\ n = 1,\ldots,N\}$. Similarly, we consider $\mathbf{D}'$ to be the disparity shifted depth map of (K pixels × L pixels) with image coordinates $\{(u'^{\mathbf{D}}_{\mathcal{B},kl}, v'^{\mathbf{D}}_{\mathcal{B},kl}),\ k = 1,\ldots,K;\ l = 1,\ldots,L\}$. Due to the transformation to the common grid, these image coordinates become $\{(u^{\mathbf{I}}_{\mathcal{C},mn}, v^{\mathbf{I}}_{\mathcal{C},mn}),\ m = 1,\ldots,M;\ n = 1,\ldots,N\}$ and $\{(u^{\mathbf{D}}_{\mathcal{C},kl}, v^{\mathbf{D}}_{\mathcal{C},kl}),\ k = 1,\ldots,K;\ l = 1,\ldots,L\}$, respectively. We define such a common mesh grid as $\mathbf{\Psi} = \{(p_{ij}, q_{ij}),\ i = 1,\ldots,M;\ j = 1,\ldots,N\}$, where the pair (p_{ij}, q_{ij}) represents the location of the image pixel corresponding to the row index i and the column index j. We set the grid $\mathbf{\Psi}$ to be of the same resolution ($M \times N$) as the 2-D camera. There is, however, no restriction regarding the resolution of the resulting mapped images. Our choice of M and N in this paper is motivated by the low-level data fusion, which is intended for enhancing the ToF depth map up to the same 2-D camera resolution. In general, state-of-the-art approaches that deal with the mapping of images to a common grid intended for data matching are based on forward warping [DNN$^+$11, LH10b]. Thus, each mapped image coordinate from $\mathbf{I}$ and $\mathbf{D}$ are assigned to the nearest pixel of the common grid. However, in most of the cases, the resolution of the depth map $\mathbf{D}$ is far below the resolution of the 2-D image $\mathbf{I}$, *i.e.,* $K << M$ and $L << M$, as illustrated in Figure 3.1a. As a result, the warping of such a depth map $\mathbf{D}$ onto the common grid presents a large number of missing depth pixels. In other words, forward warping generates a sparse number of warped depth pixels, as shown in Figure 3.2a. In contrast, we propose a back warping approach in which we determine for each pixel (p_{ij}, q_{ij}) on the common grid, the nearest pixel $(u^{\mathbf{I}}_{\mathcal{C},mn}, v^{\mathbf{I}}_{\mathcal{C},mn})$ on the image $\mathbf{I}$ after being transformed onto $\mathcal{C}$, as illustrated in Figure3.1b. Similarly, we determine for each pixel (p_{ij}, q_{ij}) the nearest pixel $(u^{\mathbf{D}}_{\mathcal{C},kl}, v^{\mathbf{D}}_{\mathcal{C},kl})$. As a result, our mapped images, $\mathbf{I}_{\mathcal{C}}$ and $\mathbf{D}_{\mathcal{C}}$ are perfectly aligned with a major advantage of $\mathbf{D}_{\mathcal{C}}$ being a dense depth map. Indeed, we show in Figure 3.2b a comparison of the deth maps obtained using a forward mapping and our proposed counterpart; that could be referred to as backward warping. The two techniques are overall equivalent. Our proposed approach has however one clear advantage. It provides a dense depth map while the forward warping provides a very sparse depth map. As a result if there is a requirement for depth map downsampling, which is common for a real-time implementation, the downsampled sparse depth map becomes unusable. We claim therefore that our proposed backward warping is more appropriate for real-time applications.



(a) **(b)**

Figure 3.1: Image coordinate transformation. (a) Shown are the transformed 2-D image coordinates $(u^{\mathbf{I}}_{\mathcal{C}}, v^{\mathbf{I}}_{\mathcal{C}})$ depicted as '+', the transformed ToF image coordinates $(u'^{\mathbf{D}}_{\mathcal{C}}, v'^{\mathbf{D}}_{\mathcal{C}})$ depicted as '×', and the mesh grid Ψ coordinates (p, q). (b) Detail of the mapping procedure. It is apparent that a certain ToF pixel (k, l) will be mapped to several mesh grid pixels (i, j). Reference frames $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ are depicted in blue, red, and green, respectively.



(a) Forward warping **(b)** Proposed backward warping

Figure 3.2: Comparison of the sparse depth map points obtained by forward warping (a) and the dense depth map obtained using our method, *i.e.*, backward warping (a). We refer the reader to the electronic version of the paper in order to better appreciate the differences between the forward and backward warping result.

3.3.1 2-D camera LUT

The relationship between the raw images and the mapped ones can be represented by an array that associates each pixel coordinates in the unified reference frame $\mathcal{C}$ to a unique pixel in $\mathcal{A}$ and $\mathcal{B}$, as illustrated in Figure3.3. This associative array or look-up table (LUT) can be computed off-line in order to reduce the complexity of the mapping procedure to a single indexing operation and leading to real-time implementation.

We define the mapping $(i, j) \mapsto (m, n) = \mathbf{L}_{\mathcal{AC}}(i, j)$, as $\mathbf{L}_{\mathcal{AC}}(i, j) = \arg\min_{(m,n)} \|(p_{ij}, q_{ij}) - (u^{\mathbf{I}}_{\mathcal{C},mn}, v^{\mathbf{I}}_{\mathcal{C},mn})\|_2$. The stored LUT $\mathbf{L}_{\mathcal{AC}}$ allows to generate the new mapped image as follows $\mathbf{I}_{\mathcal{C}}(i, j) = \mathbf{I}\big(\mathbf{L}_{\mathcal{AC}}(i, j)\big)$, for all i, j.

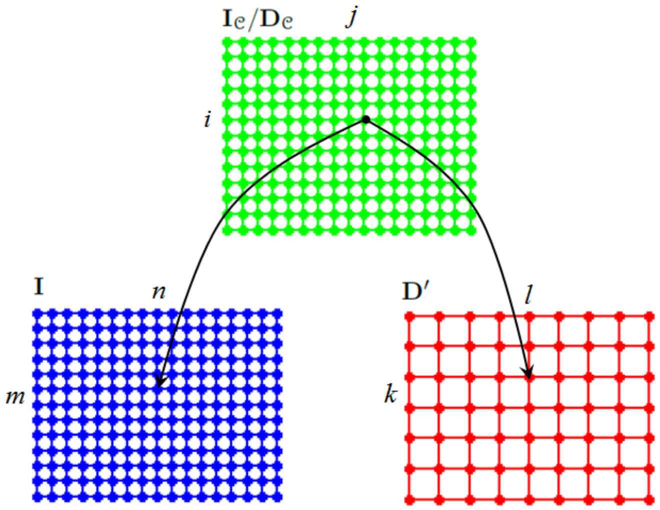


Figure 3.3: The look-up tables $\mathbf{L}_{\mathcal{A}\mathcal{C}}$ and $\mathbf{L}_{\mathcal{B}\mathcal{C}}$ associate each pixel coordinates in $\mathcal{C}$ to a unique pixel in $\mathcal{A}$ and $\mathcal{B}$, respectively.

3.3.2 ToF camera LUT

The same procedure as the one presented for determining the 2-D camera LUT applies for the ToF camera LUT that we refer to as $\mathbf{L}_{\mathcal{B}\mathcal{C}}$. Thus, we place the same mesh grid $\mathbf{\Psi}$ onto the disparity corrected and transformed image coordinates $(u_{\mathcal{C}}^{\mathbf{D}}, v_{\mathcal{C}}^{\mathbf{D}})$ and we perform a nearest neighbour search to determine the pixel (k, l) from $\mathbf{D}'$ with the position $(u_{\mathcal{C}}^{\mathbf{D}}, v_{\mathcal{C}}^{\mathbf{D}})$ nearest to (p_{ij}, q_{ij}). The mapped depth map $\mathbf{D}_{\mathcal{C}}$ results from $\mathbf{D}_{\mathcal{C}}(i,j) = \mathbf{D}\big(\mathbf{L}_{\mathcal{B}\mathcal{C}}(i,j)\big)$, for all (i,j). We note that the mapping described by this mesh grid also upsamples the mapped image coordinates to the 2-D camera resolution $(M \times N)$. We did not consider other interpolation techniques such as linear or bilinear interpolation because they may generate unwanted artefacts when applied on ToF data due to their characteristics such as incorrect measurements at large distances. These pixel values must not be considered in an interpolation, but require a special treatment. Also, real distances within the edges in the scene should not be interpolated. At the end of the mapping process, both resulting images $\mathbf{I}_{\mathcal{C}}$ and $\mathbf{D}_{\mathcal{C}}$ generated from their respective $\mathbf{L}_{\mathcal{A}\mathcal{C}}$ and $\mathbf{L}_{\mathcal{B}\mathcal{C}}$ LUTs are pixel aligned. Nevertheless, $\mathbf{L}_{\mathcal{B}\mathcal{C}}$ that generates $\mathbf{D}_{\mathcal{C}}$ is distance-dependent. Due to the disparity shift presented in Section 3.2.2, the resulting $\mathbf{L}_{\mathcal{B}\mathcal{C}}$ LUT depends on the depth map information and thus on the scene configuration. The easiest way to deal with this dependence would be computing the $\mathbf{L}_{\mathcal{B}\mathcal{C}}$ LUT for each recorded ToF frame; however, this implies a high computational time, and consequently, it will not be viable if real-time performance is required. Indeed, the off-line computation of a single $\mathbf{L}_{\mathcal{BC}}$ is close to 15 minutes using Matlab for Windows on the system we have used to run our experimental results.

3.4 Real-time implementation

In order to achieve real-time performance on dynamic scenes, we propose to consider an array $\{\mathbf{L}_{\mathcal{BC},k}\},\ k = 0, \ldots, K-1$, of LUTs where each LUT $\mathbf{L}_{\mathcal{BC},k}$ tackles a different disparity $\rho_k(Z_{\mathcal{B}})$, corresponding to a plane at a fixed distance $Z_k = f_{\mathcal{C}} \cdot |\mathbf{b}|/k$ to the system. We choose the discrete disparities as multiples of the pixel size in the mapped depth map $\mathbf{D}_{\mathcal{C}}$, *i.e.,* $\rho_k = s_b k, k = 0, \ldots, K-1$ where $s_b = \mathbf{b}/|\mathbf{b}|$ is the unit vector of the baseline shift. Dividing the Z range of the ToF camera into K intervals $[\zeta_{k+1}, \zeta_k]$ around Z_k with

$$\begin{aligned} \zeta_0 &= \infty \\ \zeta_k &= f \cdot \frac{|\mathbf{b}|}{(k - \frac{1}{2})}, \ k = 1, \ldots, K, \end{aligned} \tag{3.12}$$

one finds that for each pixel of the ToF camera with a Z value in the interval $[\zeta_{k+1}, \zeta_k]$, the disparity equals $\rho_k(Z_{\mathcal{B}})$, with an error less than $\delta/2$, *i.e.,* half the size of a pixel in the mapped depth map $\mathbf{D}_{\mathcal{C}}$, as shown in Figure 3.4. The maximum binocular disparity is given by the minimum $Z-$measurement range of the ToF camera, Z_{min} (the minimum Z value in the setup). The number K of different disparities to be considered is given by $K \geq f \cdot \frac{|\mathbf{b}|}{Z_{min}} + \frac{1}{2}$. The mapping is then performed by the iterative Algorithm 3.1, where $\mathbf{Z}$ denotes the image of $Z_{\mathcal{B}}$ values calculated from the depth map $\mathbf{D}$ using (3.5). This mapping procedure allows the low-resolution depth map $\mathbf{D}$ to be mapped in real-time to a depth map $\mathbf{D}_{\mathcal{C}}$, where each pixel matches a pixel in the already mapped $\mathbf{I}_{\mathcal{C}}$ image. In the occlusion handling block, we check if the condition $Z \in [\zeta_{k+1}, \zeta_k]$ is fulfilled. If not, the selected pixel is labelled as occluded.

Although we achieve a high performance within the mapping procedure, the memory required to store the K LUTs is considerable, being a problem to deal with in case of real embedded applications. To that end, we propose a procedure to reduce the memory requirements intended for hybrid ToF multi-camera systems with their cameras almost co-planar. In this case, we proceed by considering the transformation $\mathbf{H}_{\mathcal{BC}}$ inside the

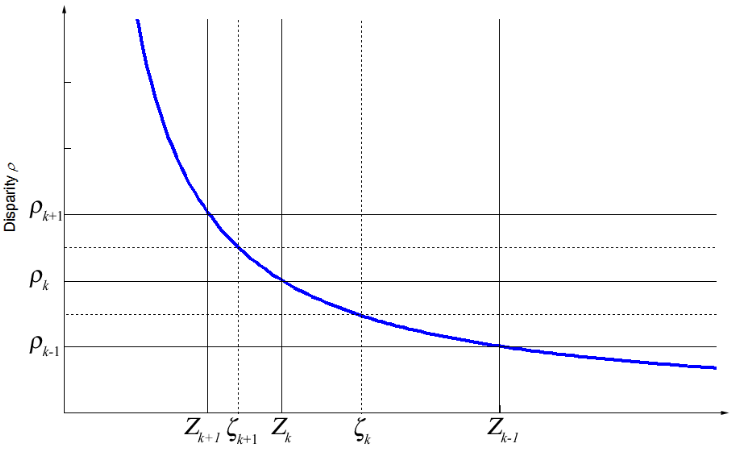


Figure 3.4: Z range of the ToF camera divided into K intervals $[Z_{k+1}, Z_k]$ defined by equidistant disparity values $\rho_k(Z_{\mathcal{B}}) = k \times \rho$. Within each interval, the disparity ρ varies less than 1 pixel size δ.

disparity shift, as discussed in Section 3.2.2 (see (3.10)). In our case, the x axes of the camera reference frames are chosen to be parallel to the baseline between the cameras, *i.e.,* $\mathbf{b} = [b_x, 0, 0]^{\mathrm{T}}$, and thus the disparity shift extends in the x direction of the image frame. The disparity differs by exactly one pixel in x direction when calculated at two different distances Z_k and Z_{k+1}. The corresponding two LUTs are then related via $L_{\mathcal{BC},k+1}(i,j) = L_{\mathcal{BC},k}(i,j-s)$ with $s = sign(b) = \pm 1$ being the sign of the baseline shift with respect to the x axis, *i.e.,* indicating on which side of the ToF camera the 2-D camera is positioned with respect to the x axis of the unified reference frame. Consequently, it is sufficient to store a single LUT $L_{\mathcal{BC},0}$ calculated on an extended mesh grid $\mathbf{\Psi}$ of size $M \times (N+k)$, which defines all K LUTs via $L_{\mathcal{BC},k}(i,j) = L_{\mathcal{BC},0}(i,j-sk)$ with $i = 1,\ldots,M$, $j = 1\ldots,N$, and $k = 0,\ldots,K-1$. Unlike the distance image $\mathbf{D}$, the $\mathbf{Z}$ image needs to be recalculated by the same projective transformation resulting in a new $\mathbf{Z}'$ image (see (3.5)). We proceed by using the Algorithm 3.2, where $\mathbf{Z}_{\mathcal{C}}$ is the resulting matrix of $Z_{\mathcal{B}}$ coordinates on the common coordinate grid in the unified reference frame. The latter allows to calculate a radial distance image $\mathbf{D}_{\mathcal{C}}$ using (3.11) for the coordinates of the common coordinate grid.

Algorithm 3.1 Mapping algorithm

```
for $i = 1$ to N do
  for $j = 1$ to M do
    $k = K$
    $\{$Search $Z_k$ interval$\}$
    while $(k > 0)$ and $(\mathbf{Z}(\mathbf{L}_{\mathcal{BC},k}(i,j)) > \zeta_k)$ do
      $k \leftarrow k - 1$
    end while
    $\{$Occlusion handling$\}$
    if $(k < K)$ and $(\mathbf{Z}(\mathbf{L}_{\mathcal{BC},k}(i,j)) < \zeta_{k+1})$ then
      $k \leftarrow k + 1$
    end if
    $\{$Mapping$\}$
    $\mathbf{D}_{\mathcal{C}}(i,j) = \mathbf{D}'(\mathbf{L}_{\mathcal{BC},k}(i,j))$
  end for
end for
```

Algorithm 3.2 Optimized mapping algorithm

```
for $i = 1$ to N do
  for $j = 1$ to M do
    $k = K$
    $\{$Search $Z_k$ interval$\}$
    while $(k > 0)$ and $(\mathbf{Z}'(\mathbf{L}_{\mathcal{BC},0}(i,j-sk)) > \zeta_k)$ do
      $k \leftarrow k - 1$
    end while
    $\{$Occlusion handling$\}$
    if $(k < K)$ and $(\mathbf{Z}'(\mathbf{L}_{\mathcal{BC},0}(i,j-sk)) < \zeta_{k+1})$ then
      $k \leftarrow k + 1$
    end if
    $\{$Mapping$\}$
    $\mathbf{Z}_{\mathcal{C}}(i,j) = \mathbf{Z}'(\mathbf{L}_{\mathcal{BC},0}(i,j-sk))$
  end for
end for
```

3.5 Experimental results

In order to analyse the data mapping step, we have considered six different test cases in which we recorded the calibration pattern displaced around the FOV of the sensing system, and at different depths and orientations (see Figure 3.5). We first quantify our proposed approach against to a common mapping using a simple projective transformation, *i.e.,* a plane-to-plane transformation or 2-D homography. To that end, we focus on the four first test cases where the recorded pattern is always located parallel to the sensing system. In Table 3.1, the two first rows report the RMSE of the centroids of the mapped control points using a 2-D homography. As expected, the use of a 2-D homography performs better if the distance at which it has been computed coincides with the distance at which the control points are located (see the first four test cases in the second row of Table 3.1). However, if we use a unique homography for these test cases, the matching error increases as soon as we vary the depth at which the pattern is located (see the first four test cases in the first row of Table 3.1). In general cases where the pattern is arbitrary located and oriented in front of the sensing system (see test cases 5 and 6 in Figure 3.5 and the last two columns of Table 3.1), the use of a plane-to-plane transformation reports an error much bigger than using the proposed approach. Indeed, the proposed data mapping approach presents an accuracy up to one 2-D pixel, which is caused by the approximation, given in (3.12), of Z_k by the interval $[\zeta_{k+1}, \zeta_k]$. We note that the errors reported in Table 3.1 also include the inaccuracies introduced by the centroid estimation and the calibration step, which correspond to 1 pixel according to the 2-D camera pixel size (see Section 2.4.3). Thus, the evaluation results for our mapping method show a consistent error of about 2 mapped image pixels, or less if we take into account the error due to the centroid operator. This observation confirms that the proposed method accurately adapts to the distance-dependent disparity explained in Section 3.1. The last row of Table 3.1 reports the error when considering the most common or general approach for data mapping, *i.e.,* using a full 3-D projection (with no approximations). By using this 3-D projection, the mapped centroids are matching with more accuracy than by using the proposed approach. However, the loss in accuracy is worth a significant gain in speed. We note that the mean in seconds for computing this 3-D projection for a single frame is 782.67, while using

Table 3.1: Data matching error for the six test cases. The table compares the RMSE (in pixels) over 20 control points, separately computed for x and y pixel coordinates, between our mapping procedure and the mapping using first a simple projective transformation (two first rows) and a 3-D transformation without approximations (last row).

Test cases		1	2	3	4	5	6
RMSE using a unique	x	7.52	1.67	3.66	1.33	2.59	3.75
proj. transf. ($z = 1.5\ m$)	y	1.45	1.26	1.23	1.88	1.42	1.57
RMSE using a computed proj.	x	1.29	1.31	1.87	1.33	3.52	3.90
transf. for each test case	y	1.48	1.26	1.22	1.88	1.42	1.69
RMSE using the proposed	x	2.14	1.45	1.69	1.56	1.47	2.04
mapping procedure	y	1.40	1.27	1.37	1.84	1.43	1.72
RMSE using a 3-D	x	1.58	1.37	1.51	1.42	1.48	2.00
projection	y	1.43	1.25	1.21	1.79	1.35	1.76

the proposed approach only 0.54 seconds are required for the whole mapping procedure (in Matlab).

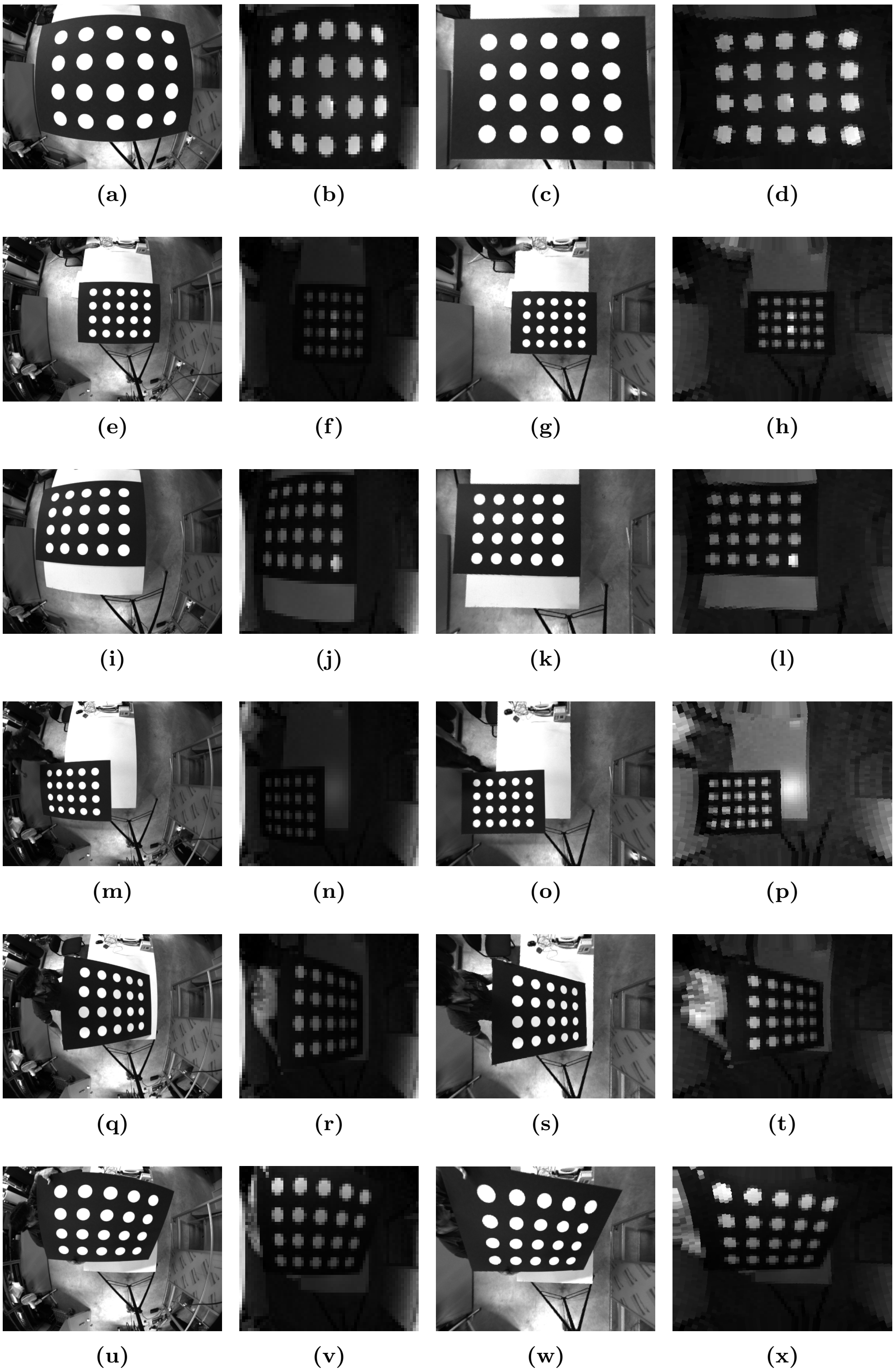


Figure 3.5: Test cases for data matching. 1^{st} row: test case 1, $z = 0.7\ m$. 2^{nd} row: test case 2, $z = 1.5\ m$. 3^{rd} row: test case 3, $z = 1.0\ m$. 4^{th} row: test case 4, $z = 1.5\ m$. 5^{th} row: test case 5, $z \in [0.9, 1.5]\ m$. 6^{th} row: test case 6, $z \in [0.8, 1.5]\ m$. 1^{st} column: 2-D acquisitions. 2^{nd} column: ToF acquisitions. 3^{rd} column: 2-D mapped. 4^{th} column: ToF mapped.

Part II

Data Fusion

Chapter 4

Depth enhancement by filtering

This chapter is an overview of the filtering techniques intended for the enhancement of low-resolution ToF depth maps by means of data fusion with high-resolution 2-D images. Different filtering techniques that combine low-resolution depth maps with accurately aligned high-resolution 2-D images have been proposed during the last decade. Among the early results for low-resolution data fusion, the application of Markov random fields (MRFs) to the fusion of ToF and 2-D data was proposed by Diebel et al. [DT05], and extended by Gloud et al. [GBQ$^+$08]. Despite their promising results, the evaluation of depth enhancement methods based upon an MRF is in general computationally intensive and thus not suitable if real-time processing is a requirement. Yang et al. [YYDN07] presented an alternative depth enhancement method based upon a cost volume in which the final depth map was estimated through a different refinement module. Another approach is used in methods based upon a bilateral filter [Ela02, TM98]. These approaches achieve similar results to those based upon an MRF or iterative methods with a major advantage of a faster computation time. This motivates us to focus on the bilateral filtering techniques as real applications usually require fast performance. Therefore, we first introduce the bilateral filter and later, we present the most relevant bilateral filter based techniques for low-resolution depth enhancement.

4.1 Background: Bilateral filtering

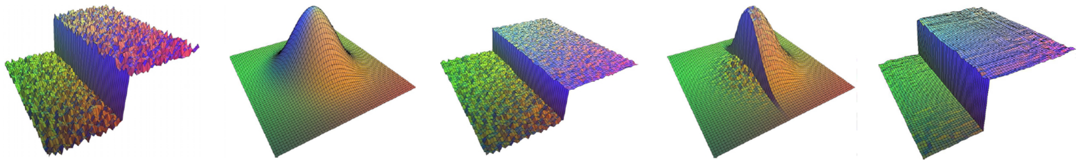
The bilateral filter was first introduced by Tomasi et al. [TM98] as an alternative to iterative approaches for image noise removal such as anisotropic diffusion, weighted least squares, and robust estimation [Ela02]. This non-iterative filter formulation is a weighted average of the local neighbourhood samples, where the weights are computed based on spatial and radiometric distances between the centre of the considered sample and the neighbouring samples. Thus, its kernel is decomposed into a spatial weighting term $f_{\mathbf{S}}(\cdot)$ that applies to the pixel position $\mathbf{p}$, and a range weighting term $f_{\mathbf{I}}(\cdot)$ that applies to the pixel value $\mathbf{I}(\mathbf{q})$. The filtering process locally adapts the kernel as follows

$$\mathbf{J_1}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q}))\mathbf{I}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q}))}, \tag{4.1}$$

where $N(\mathbf{p})$ is the neighbourhood at the pixel indexed by the position vector $\mathbf{p} = (i,j)^T$, with i and j indicating the row, respectively column corresponding to the pixel position. The weighting functions $f_{\mathbf{S}}(\cdot)$ and $f_{\mathbf{I}}(\cdot)$ are generally chosen to be Gaussian functions with standard deviations $\sigma_{\mathbf{S}}$ and $\sigma_{\mathbf{I}}$, respectively. The resulting filtered image $\mathbf{J_1}$ is a smoothed version of $\mathbf{I}$, that presents less discontinuities and a significantly reduced noise level, *i.e.,* $\mathbf{I}$ is smoothed while its edges are preserved, as illustrated in Figure 4.1. Thus, the bilateral filter is a non-linear filter that adapts its kernel to the data to be filtered (see Figure 4.1e) and consequently makes real-time processing quite challenging. However, recent implementation techniques for bilateral filtering based on data downsampling [PD09], data quantization [Por08, YTA09] or, adapting the block size to the data to be filtered [WFH$^+$10], have shown that real-time performance on high-resolution 2-D images is feasible.

4.2 State-of-the-art depth enhancement filters

As presented in Section 4.1, the bilateral filter combines a spatial weighting term $f_{\mathbf{S}}(\cdot)$ based on the pixel position $\mathbf{p}$ with its corresponding range weighting term $f_{\mathbf{I}}(\cdot)$ based



(a) Input image $\mathbf{I}$ **(b)** Spatial weighting $f_{\mathbf{S}}(\cdot)$ **(c)** Range weighting $f_{\mathbf{I}}(\cdot)$ **(d)** Weight $f_{\mathbf{S}}(\cdot)f_{\mathbf{I}}(\cdot)$ **(e)** Output image $\mathbf{J_1}$

Figure 4.1: Bilateral filtering. The kernel is applied on the central pixel [DD02].

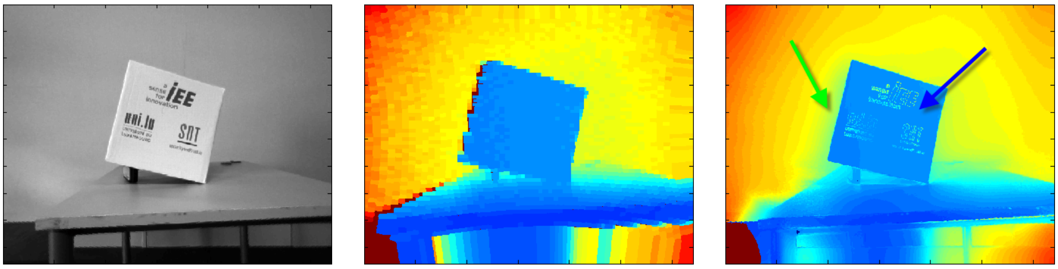
on the pixel value $\mathbf{I}(\mathbf{q})$. As a result, the filtered image $\mathbf{J_1}$ preserves much more detail given by the range weighting term, *e.g.,* edges from the input image $\mathbf{I}$. Based on this working principle, different filtering techniques have been proposed for different data enhancement purposes, such as image denoising by combining flash/no-flash image pairs [PAH$^+$04], *i.e.,* the range weighting term applies to a flashed image while filtering the no-flashed image pair, or depth enhancement by combining ToF and 2-D data. We remark that in cases where the filter considers different data sources, the data to be filtered has to be correctly aligned and thus every pixel matching to and from each image pair. To that end, we refer the reader to Chapter 3 where we detail how to align a low-resolution depth map with its corresponding high-resolution 2-D image. From now on and for the sake of simplicity, we will refer to the aligned data $\mathbf{I}_{\mathcal{C}}$ and $\mathbf{D}_{\mathcal{C}}$ as $\mathbf{I}$ and $\mathbf{D}$, respectively.

4.2.1 Joint Bilateral Upsampling

Kopf et al. presented in [KCLU07] the Joint Bilateral Upsampling (JBU) filter, a modification of the bilateral filter expression in (4.1) that considers two different data sources within the kernel of the filter. This way, it becomes possible to compute a solution for image analysis and enhancement tasks, such as tone mapping or colourization through a downsampled version of the data. This idea was also applied for depth map enhancement in the context of real-time matting as presented by Crabb et al. [CTPD08]. The JBU filter enhances an aligned depth map $\mathbf{D}$ to the higher resolution of its correspondence 2-D guidance image $\mathbf{I}$, as follows

$$\mathbf{J_2}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)}. \tag{4.2}$$

As in (4.1), the resulting depth map $\mathbf{J_2}$ is an enhanced version of $\mathbf{D}$ with the same resolution as the 2-D guidance image, as shown in Figure 4.2. Nevertheless, according to the bilateral filter principle, the fundamental heuristic assumptions about the relationship between depth and intensity data, *i.e.,* a difference between intensity values indicates a jump in depth, may lead to erroneous copying of 2-D texture into actually smooth geometries within the depth map (see blue arrow in Figure 4.2c). Figure 4.2c also shows a second unwanted artefact known as edge blurring (see green arrow in Figure 4.2c) that appears along depth edges that have no corresponding edges in the 2-D



(a) 2-D guidance image **I**

(b) Depth map **D**

(c) Enhanced depth map using JBU $\mathbf{J_2}$

Figure 4.2: Depth map enhancement using the JBU filter. The arrows in (c) indicate unwanted artefacts, *i.e.,* blue and green arrows indicate texture copying and edge blurring, respectively.

image, *i.e.,* in situations where objects on either side of a depth discontinuity have a similar colour. In most of the cases, this is due to the transformation from the original coloured image to its grayscale version (see Chapter 7). Edge blurring also occurs due to the misalignment between the data to be filtered, *i.e.,* data does not perfectly match. Despite the undesired artefacts obtained from the direct application of the JBU filter for low-level data fusion, this filter has been used as a basis for the next multi-lateral filters for depth enhancement as developed below.

4.2.2 Colour and depth joint bilateral filter

Kim et al. [KCKA10] presented a straightforward extension of the JBU filter to slightly reduce the JBU's texture copying and edge blurring artefacts. In addition to the range weighting term $f_{\mathbf{I}}(\cdot)$ that applies to the 2-D guidance image, they propose to use an additional range weighting factor $f_{\mathbf{D}}(\cdot)$ that applies to the depth measurements as follows

$$\mathbf{J_3}(\mathbf{p}) = \frac{\sum\limits_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big) f_{\mathbf{D}}\big(\mathbf{D}(\mathbf{p}),\mathbf{D}(\mathbf{q})\big)\mathbf{D}(\mathbf{q})}{\sum\limits_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big) f_{\mathbf{D}}\big(\mathbf{D}(\mathbf{p}),\mathbf{D}(\mathbf{q})\big)}. \tag{4.3}$$

This way, in case of depth discontinuities, the texture copying artefact is reduced whereas the erroneous depth values along depth edges are not corrected.

4.2.3 Noise-Aware Filter for Depth Upsampling

Chan et al. proposed in [CBTT08] an improved version of the JBU filter that preserves the benefits of using the JBU filter and prevents artefacts in those areas where JBU is likely to cause erroneous texture copying. This filter is referred to as Noise-Aware Filter for Depth Upsampling (NAFDU). The NAFDU strategy also relies on depth information. In contrast to the previous filter, where the depth information was directly taken into account within the filter kernel, the NAFDU filter splits each data source contribution as follows

$$\begin{aligned}\mathbf{J_4}(\mathbf{p}) \quad = \quad & \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\big[\alpha\big(\Delta\Omega(\mathbf{p})\big) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)+}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\big[\alpha\big(\Delta\Omega(\mathbf{p})\big) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)+} \\ & \frac{\Big(1-\alpha\big(\Delta\Omega(\mathbf{p})\big)\Big) f_{\mathbf{D}}(\mathbf{D}(\mathbf{p}),\mathbf{D}(\mathbf{q}))\big]\mathbf{D}(\mathbf{q})}{\Big(1-\alpha\big(\Delta\Omega(\mathbf{p})\big)\Big) f_{\mathbf{D}}(\mathbf{D}(\mathbf{p}),\mathbf{D}(\mathbf{q}))\big]}, \qquad (4.4)\end{aligned}$$

where $\alpha(\cdot)$ is the blending function that decides how each data source contribution, from the 2-D image and depth data, must be considered. A high weight α makes the filter behave like the original JBU filter whereas a low weight α makes it behave like the standard bilateral filter, *i.e.,* both spatial and range weighting terms are applied to the same data source $\mathbf{D}$ without considering the 2-D image $\mathbf{I}$. Intuitively, NAFDU tries to preserve the benefits of JBU except in the areas that are geometrically smooth but heavily contaminated with random noise within the distance measurements. The blending function is defined as $\alpha\big(\Omega(\mathbf{p})\big) = 1/\big(1 + e^{\epsilon\cdot(\Omega(\mathbf{p})-\tau)}\big)$, with $\Omega(\mathbf{p})$ the difference between the maximum and minimum measured depth value in the pixel neighbourhood $N(\mathbf{p})$. Parameters ϵ and τ control at what min-max difference the blending interval shall be centred. The downside of this method is that those values must be manually tuned. Besides, the NAFDU expression corresponds to a weighted average of two non normalized kernels, which makes the contribution of each of the kernels inconsistent and inaccurate. In addition, it leads to a more complex real-time implementation.

Chapter 5

Proposed depth enhancement techniques

In this chapter we propose two adaptive multi-lateral filters to overcome the drawbacks of the alternative depth enhancement techniques presented in Section 4.2. Our proposed techniques are based on the JBU filter and are extended by a new factor that considers the low reliability of depth measurements along the low-resolution depth map edges. Our former contribution enhances low-resolution depth maps relying on 2-D data as guidance information. Consequently, whereas edge blurring is almost eliminated, texture copying may still appear within geometrically smooth regions, *i.e.,* within non-abrupt changes on surfaces. In order to entirely remove texture copying, we propose a second technique that in addition to adaptively considering 2-D information, also considers depth data as guidance information. Moreover, this latter contribution can be configured in order to behave as most of the existing multi-lateral filters for depth enhancement based upon a bilateral filter. Furthermore, both of the proposed filters may be effectively and efficiently implemented for dynamic scenes and thus, for real-time applications.

5.1 Pixel Weighted Average Strategy

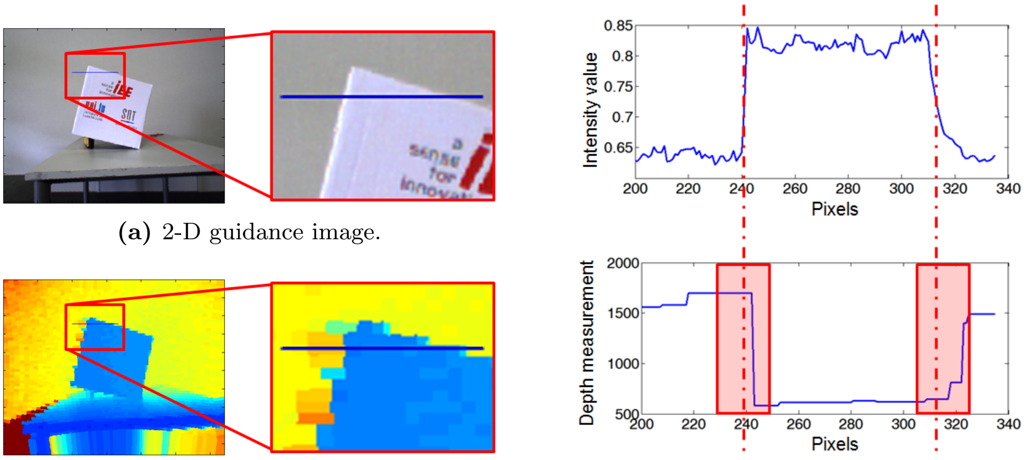
Adjusting the right distance measurement along real depth edges without mismatching with texture is quite challenging. We proposed in [GMO^{+}10] an alternative improvement of the JBU filter, that we refer to as the Pixel Weighted Average Strategy (PWAS) filter. This filter copes well with inaccurate edge values. In contrast to most of depth enhancement methods proposed in the literature, the PWAS filter contains an additional factor $\mathbf{Q}(\cdot)$ to the kernel in (4.2), named credibility map. It assigns a reliability weight to each depth map value as a function of the scene's geometry. By so doing, depth measurements that are considered to be unreliable are replaced by reliable values in their neighbourhood and adjusted to the 2-D guidance image. The PWAS filter takes the following form

$$\mathbf{J_5}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})}. \tag{5.1}$$

Similarly to the filters presented in Section 4.2, the weighting functions $f_{\mathbf{S}}(\cdot)$ and $f_{\mathbf{I}}(\cdot)$ are taken to be Gaussian functions with standard deviations $\sigma_{\mathbf{S}}$ and $\sigma_{\mathbf{I}}$, respectively.

5.1.1 Credibility map Q

Due to the low spatial resolution provided by ToF cameras, a measured pixel can cover both foreground and background from the scene at the same time, resulting in low accuracy depth measurements along depth edges. In addition, the position of an edge within a depth map is defined with the accuracy of this low spatial resolution. Therefore, in most of the cases it does not coincide with the position of its corresponding 2-D edge, as shown in Figure. 5.1. Consequently, this misalignment introduces edge blurring artefacts as described in Section 4.2.1. The introduction of the new factor $\mathbf{Q}(\cdot)$ allows us to explicitly account for the unreliability of the depth measurements along the edges. This credibility map $\mathbf{Q}(\cdot)$ is computed directly from the real data and requires no manual parameter tuning. Indeed, $\mathbf{Q}(\cdot)$ is defined as a Gaussian kernel applied on the low-resolution depth map such that $\mathbf{Q} = f_{\mathbf{Q}}(-|\nabla\mathbf{D}|)$, $f_{\mathbf{Q}}(\cdot)$ being the Gaussian function with standard deviation $\sigma_{\mathbf{Q}}$. A low credibility map weight indicates an unreliable depth measurement whereas a high credibility map weight indicates a reliable depth measurement. In summary, the credibility map boundaries define in which areas the depth measurements are unreliable and are thus adjusted according to the 2-D guidance image. Figure 5.2 shows an example of the credibility map considering the depth map in Figure 5.1b. This term enables the reduction of texture copying and edge blurring since range values along depth discontinuities are given less weight by the credibility map as shown in Figure 5.3. However, the edge blurring effect may still appear when a real depth edge has no corresponding edge in the guidance image.



(a) 2-D guidance image.

(b) Depth map.

(c) Plot of the selected pixels (marked with a blue line) from (a) and (b).

Figure 5.1: Inaccuracy of depth measurements within edge pixels. Due to the difference in resolution, edges in (a) and (b) may not match each other.

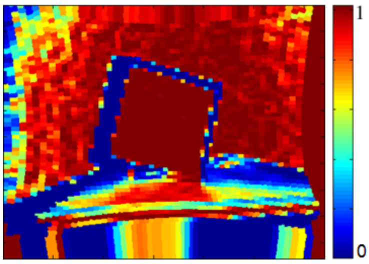
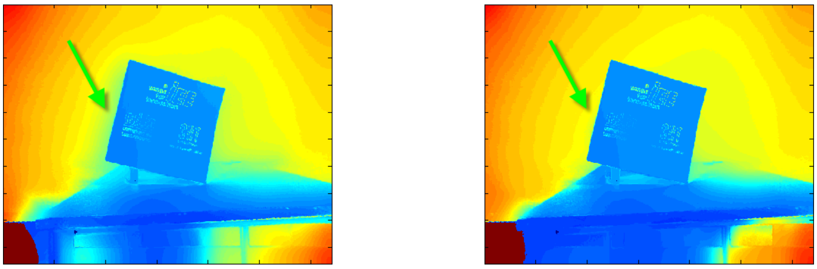


Figure 5.2: Credibility map of the raw depth map in Figure 5.1b where a weight of 1 indicates a reliable depth measurement. Depth discontinuities are set to zero.

5.2 Unified Multi-Lateral filter

As presented in Section 5.1, our PWAS filter overcomes the edge blurring artefact due to the misalignment between 2-D and depth edges by using the credibility map (see Section 5.1.1). Thus, we correctly addressed the depth values along depth edges outperforming the alternative depth enhancement techniques presented in Section 4.2, as shown in Chapter 6. However, the range weighting term $f_{\mathbf{D}}(\cdot)$ within the PWAS kernel only applies to the 2-D information. As a result, this may cause texture copying in regions that actually are geometrically smooth with, in general, reliable depth measurements (see Figure 5.4a). Instead, we propose to define two separate normalized



(a) Depth enhancement using JBU $\mathbf{J_2}$

(b) Depth enhancement using PWAS $\mathbf{J_5}$

Figure 5.3: Comparison between JBU and PWAS filtering. The green arrow indicates edge blurring, which is almost entirely removed in (b) by using the credibility map in Figure 5.2.

kernels with each one considering a different data source, 2-D and depth information, respectively. The decision on which kernel the filter has to consider is directly given by the reliability weight of the pixel to be filtered. We therefore propose the Unified Multi-Lateral (UML) filter whose main benefit is the increase of the accuracy of the depth measurements within smooth regions, as shown in Figure 5.4b. The UML filter takes the form of

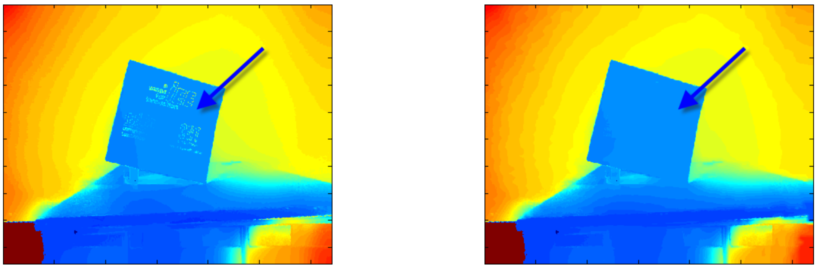
$$\mathbf{J_7}(\mathbf{p}) = \big(1 - \beta(\mathbf{p})\big) \cdot \mathbf{J_5}(\mathbf{p}) + \beta(\mathbf{p}) \cdot \mathbf{J_6}(\mathbf{p}), \tag{5.2}$$

where $\beta = \mathbf{Q}$, the blending function to weight the contribution of the pixel to be filtered from each individual data source. $\mathbf{J_6}(\mathbf{p})$ is the filtered range value at pixel $\mathbf{p}$ given by a modified PWAS filter with a range weighting term that applies to the depth information $\mathbf{D}$, *i.e.*,

$$\mathbf{J_6}(\mathbf{p}) = \frac{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) f_{\mathbf{D}}\big(\mathbf{D}(\mathbf{p}), \mathbf{D}(\mathbf{q})\big) \mathbf{Q}(\mathbf{q}) \mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) f_{\mathbf{D}}\big(\mathbf{D}(\mathbf{p}), \mathbf{D}(\mathbf{q})\big) \mathbf{Q}(\mathbf{q})}. \tag{5.3}$$

5.2.1 Filter parametrization

We chose the weighting functions $f_{\mathbf{S}}(\cdot)$, $f_{\mathbf{I}}(\cdot)$, $f_{\mathbf{D}}(\cdot)$, and $f_{\mathbf{Q}}(\cdot)$ to be Gaussian functions with standard deviations $\sigma_{\mathbf{S}}$, $\sigma_{\mathbf{I}}$, $\sigma_{\mathbf{D}}$, and $\sigma_{\mathbf{Q}}$, respectively. The reason is mainly because Gaussian functions can be computed at constant time [Der93]. We notice that these standard deviations are data-dependent and thus cannot be fixed to a unique



(a) Depth enhancement using PWAS $\mathbf{J_5}$

(b) Depth enhancement using UML $\mathbf{J_7}$

Figure 5.4: Comparison between PWAS and UML filtering. The blue arrow indicates texture copying, which is almost entirely removed by using the UML filter.

value. However, we herein define how to automatically set standard deviations to adequate values for each weighting function. The standard deviation $\sigma_{\mathbf{S}}$ must be at least as large as the depth edge resolution which is, in fact, the width of the credibility map boundaries. This value usually coincides with the scale factor between the low-resolution depth map $\mathbf{D}$ and the high-resolution 2-D guidance image $\mathbf{I}$. We set the values of $\sigma_{\mathbf{I}}$ and $\sigma_{\mathbf{D}}$ to the mean of the 2-D image gradient $\overline{\nabla \mathbf{I}}$ and to the mean of the depth map gradient $\overline{\nabla \mathbf{D}}$, respectively. The value of $\sigma_{\mathbf{Q}}$ is directly related to the noise level within the depth data discussed in Section 1.2.2.1.

5.2.2 Limit cases

The filter expression presented in (5.2) allows different filter configurations in order to make it behave as other multi-lateral filters for depth enhancement from the literature. To that end, the blending function $\beta(\cdot)$ has to be considered as a data source flag that can be enabled or disabled in order to consider either the depth map or the 2-D image as a guidance information, respectively. Another parameter to be configurable is the standard deviation of the credibility map $\sigma_{\mathbf{Q}}$. By making it tend to infinity $\sigma_{\mathbf{Q}} \to \infty$, the credibility map becomes constant and equal to one for all pixel values. In that case, there is no credibility map contribution. Then, our multi-lateral filter can be configured to behave like a:

- bilateral filter. We may set the data source flag $\beta(\cdot) = 1$ and $\sigma_{\mathbf{Q}} \to \infty$ to neglect the credibility map contribution.

- JBU fitler. We may set the data source flag $\beta(\cdot) = 0$ and $\sigma_{\mathbf{Q}} \rightarrow \infty$.
- PWAS filter. We may set the data source flag $\beta(\cdot) = 0$. The remaining part in (5.2) coincides with the PWAS filter.

In contrast to the NAFDU filter, our proposed multi-lateral filter is a weighted average of two normalized kernels. Thus, each kernel in (5.2) provides a consistent contribution. Making our filter behave like the NAFDU filter implies a normalization factor that is too complex and thus out of the scope of this thesis. With regards to the new joint bilateral filter, it is clear that there is no possible configuration of our multi-lateral filter that derives the same filter expression. Nevertheless, we already discussed, in Section 4.2.2, the limitations of applying the depth measurements in such a straightforward way.

5.3 Real-time implementation

In order to ensure that the UML filter maintains a high computational efficiency for real-time applications, we adopted the bilateral filter implementation proposed by Yang et al. [YTA09]. They presented a fast bilateral filter implementation that enables the real-time computation of the filter in (4.1). They showed that their fast implementation outperforms state-of-the-art methods for accuracy, speed and memory consumption [PD09, Por08]. In what follows, we adapted Yang et al.'s implementation to our proposed UML filter.

5.3.1 Range data quantization

Similarly to [YTA09], we quantify the range of the 2-D intensity values and depth measurements, *i.e.,* $I_k = s_{\mathbf{I}} \cdot k$, and $D_l = s_{\mathbf{D}} \cdot l$, with $k = 0, ..., K$ and $l = 0, ..., L$. $s_{\mathbf{I}}$ and $s_{\mathbf{D}}$ are the 2-D and depth quantization factors; thus $(s_{\mathbf{I}} \times K)$ and $(s_{\mathbf{D}} \times L)$ are equal or larger than the maximum 2-D intensity values and depth measurements, respectively. Then, inserting in (5.1) and (5.3) the quantized levels I_k and D_l for $\mathbf{I}(\mathbf{p})$, respectively $\mathbf{D}(\mathbf{p})$, one obtains for each level a filtered range image

$$\mathbf{J_5}(\mathbf{p}, I_k) = \frac{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) f_{\mathbf{I}}\big(I_k, \mathbf{I}(\mathbf{q})\big) \mathbf{Q}(\mathbf{q}) \mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) f_{\mathbf{I}}\big(I_k, \mathbf{I}(\mathbf{q})\big) \mathbf{Q}(\mathbf{q})}, \tag{5.4}$$

and

$$\mathbf{J_6}(\mathbf{p}, D_l) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})}. \tag{5.5}$$

We define four mappings, *i.e.,* $E^{I_k}(\cdot)$ and $F^{I_k}(\cdot)$, for a quantized intensity value at the pixel position $\mathbf{p}$ such that:

$$E^{I_k}: \quad \mathbf{q} \longmapsto f_{\mathbf{I}}\big(I_k, \mathbf{I}(\mathbf{q})\big)\cdot \mathbf{Q}(\mathbf{q})\cdot \mathbf{D}(\mathbf{q}), \tag{5.6}$$

$$F^{I_k}: \quad \mathbf{q} \longmapsto f_{\mathbf{I}}\big(I_k, \mathbf{I}(\mathbf{q})\big)\cdot \mathbf{Q}(\mathbf{q}) \tag{5.7}$$

and $G^{D_l}(\cdot)$ and $H^{D_l}(\cdot)$ for a quantized depth measurement at the pixel position $\mathbf{p}$, such that:

$$G^{D_l}: \quad \mathbf{q} \longmapsto f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big)\cdot \mathbf{Q}(\mathbf{q})\cdot \mathbf{D}(\mathbf{q}), \tag{5.8}$$

$$H^{D_l}: \quad \mathbf{q} \longmapsto f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big)\cdot \mathbf{Q}(\mathbf{q}). \tag{5.9}$$

We may then rewrite (5.4) and (5.5) as follows:

$$\mathbf{J_5}(\mathbf{p}, I_k) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\cdot E^{I_k}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\cdot F^{I_k}(\mathbf{q})}, \tag{5.10}$$

and

$$\mathbf{J_6}(\mathbf{p}, D_l) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\cdot G^{D_l}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\cdot H^{D_l}(\mathbf{q})}. \tag{5.11}$$

We note that $f_{\mathbf{S}}(\mathbf{p},\mathbf{q})$ is a function of the difference $(\mathbf{p}-\mathbf{q})$. Hence we may write (5.10) and (5.11) as:

$$\mathbf{J_5}(\mathbf{p}, I_k) = \frac{\big(f_{\mathbf{S}} \otimes E^{I_k}\big)(\mathbf{p})}{\big(f_{\mathbf{S}} \otimes F^{I_k}\big)(\mathbf{p})}, \tag{5.12}$$

and

$$\mathbf{J_6}(\mathbf{p}, D_l) = \frac{\big(f_{\mathbf{S}} \otimes G^{D_l}\big)(\mathbf{p})}{\big(f_{\mathbf{S}} \otimes H^{D_l}\big)(\mathbf{p})}, \tag{5.13}$$

where $\otimes$ denotes the convolution between functions. The filtered value $\mathbf{J_5}\big(\mathbf{p},\mathbf{I}(\mathbf{p})\big)$ results from a linear interpolation of the filtered range images $\mathbf{J_5}(\mathbf{p},\cdot)$ obtained for the different levels at position $\mathbf{p}$ and intensity value $\mathbf{I}(\mathbf{p})$ between I_k and I_{k+1}, *i.e.,*

$$\begin{aligned}\mathbf{J_5}\big(\mathbf{p},\mathbf{I}(\mathbf{p})\big) =& \text{interpolate}\big(\mathbf{J_5}(\mathbf{p},\cdot),\mathbf{I}(\mathbf{p})\big)\\ =& \frac{1}{s_{\mathbf{I}}}\Big(\big(I_{k+1}-\mathbf{I}(\mathbf{p})\big)\mathbf{J_5}(\mathbf{p},I_{k+1})+\\ & \big(\mathbf{I}(\mathbf{p})-I_k\big)\mathbf{J_5}(\mathbf{p},I_k)\Big).\end{aligned} \tag{5.14}$$

The same applies to $\mathbf{J_6}(\mathbf{p}, D_l)$; thus from a linear interpolation between D_l and D_{l+1}:

$$\begin{aligned}\mathbf{J_6}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big) =&\text{interpolate}\big(\mathbf{J_6}(\mathbf{p},\cdot),\mathbf{D}(\mathbf{p})\big)\\ =&\frac{1}{s_{\mathbf{D}}}\Big(\big(D_{l+1}-\mathbf{D}(\mathbf{p})\big)\mathbf{J_6}(\mathbf{p},D_{l+1})+\\ &\big(\mathbf{D}(\mathbf{p})-D_l\big)\mathbf{J_6}(\mathbf{p},D_l)\Big). \end{aligned} \tag{5.15}$$

Finally, the enhanced depth map $\mathbf{J_7}$ results from (5.2) considering (5.14) and (5.15).

5.3.2 Data downsampling

In addition to the range quantization presented in Section 5.3.1, one can ensure a good memory and speed performance by downsampling the data to be filtered. According to the study that Paris et al. conducted in [PD09], the sampling of the input data does not introduce significant errors. The same strategy applies to the UML filter presented in Section 5.2. To that end, we downsample the input data, *i.e.,* $\mathbf{I}_{\downarrow}$ = downsample$(\mathbf{I}, \lambda)$ and $\mathbf{D}_{\downarrow}$ = downsample$(\mathbf{D}, \lambda)$, with λ being the scale factor. The downsampled credibility map $\mathbf{Q}_{\downarrow}$ is then computed over a downsampled depth map $\mathbf{D}_{\downarrow}$, *i.e.,* $\mathbf{Q}_{\downarrow} = f_{\mathbf{Q}}(-|\nabla\mathbf{D}_{\downarrow}|)$. We apply equations (5.4)-(5.13) using $\mathbf{I}_{\downarrow}$ and $\mathbf{D}_{\downarrow}$, resulting in low-resolution filtered images $\mathbf{J_{5\downarrow}}$ and $\mathbf{J_{6\downarrow}}$. Formally, the values $\mathbf{J_5}\big(\mathbf{p},\mathbf{I}(\mathbf{p})\big)$ and $\mathbf{J_6}\big(\mathbf{p},\mathbf{I}(\mathbf{p})\big)$ of the high-resolution filtered depth maps can be obtained by spatially interpolating the low-resolution filtered images, *i.e.,*

$$\mathbf{J_5}\big(\mathbf{p},\mathbf{I}(\mathbf{p})\big) = \text{interpolate}\Big(\mathbf{J_{5\downarrow}}\big(\cdot,\mathbf{I}(\mathbf{p})\big),\mathbf{p}/\lambda\Big) \tag{5.16}$$

and

$$\mathbf{J_6}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big) = \text{interpolate}\Big(\mathbf{J_{6\downarrow}}\big(\cdot,\mathbf{D}(\mathbf{p})\big),\mathbf{p}/\lambda\Big). \tag{5.17}$$

Notice that for this bi-linear (*i.e.,* four point) interpolation, the low resolution filtered images $\mathbf{J_{5\downarrow}}$ and $\mathbf{J_{6\downarrow}}$ would have to be computed for each value $\mathbf{I}(\mathbf{p})$ and $\mathbf{D}(\mathbf{p})$ of the high resolution input images. At this point, we combine both the linear range interpolation and the bi-linear spatial interpolation to a tri-linear (*i.e.,* eight point) interpolation as follows:

$$\mathbf{J_5}\big(\mathbf{p},\mathbf{I}(\mathbf{p})\big) = \text{interpolate}\big(\mathbf{J_{5\downarrow}}(\cdot,\cdot),\mathbf{p}/\lambda,\mathbf{I}(\mathbf{p})\big), \tag{5.18}$$

and

$$\mathbf{J_6}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big) = \text{interpolate}\big(\mathbf{J_{6\downarrow}}(\cdot,\cdot),\mathbf{p}/\lambda,\mathbf{D}(\mathbf{p})\big). \tag{5.19}$$

Thereby, $\mathbf{J_{5\downarrow}}(\cdot,\cdot)$ and $\mathbf{J_{6\downarrow}}(\cdot,\cdot)$ is the set of low resolution filtered images calculated for the different levels I_k and D_l, respectively. The final output of the UML filter is then obtained according to (5.2) by superposing the two filter outputs in (5.18), (5.19) using the credibility map $\mathbf{Q}$ that defines a pixel-dependent weight for each of the two contributions.

We address a further optimization of the proposed real-time implementation by weighting the superposition on a low-resolution level before the interpolation. Thus, the tri-linear interpolation in (5.19) of $\mathbf{J_6}$ is approximated by a bi-linear spatial interpolation of a single low resolution filtered image $\mathbf{J_{6\downarrow}} = \mathbf{J_{6\downarrow}}(\cdot, \mathbf{D}_\downarrow(\cdot))$. This is possible in the case where the resolution of the original depth map is smaller than the resolution of the downsampled depth map $\mathbf{D}_\downarrow$. Then, the values of $\mathbf{D}(\mathbf{p})$ and $\mathbf{Q}(\mathbf{p})$ may be approximated by the nearest pixel in the low versions of the maps. This interpolation formula for $\mathbf{J_7}$ takes the following form:

$$\begin{aligned}\mathbf{J_7}(\mathbf{p}) =&\text{interpolate}\big(\mathbf{Q}_\downarrow(\cdot)\mathbf{J_{5\downarrow}}(\cdot,\cdot), \mathbf{p}/\lambda, \mathbf{I}(\mathbf{p})\big)+\\ &\text{interpolate}\Big(\big(1-\mathbf{Q}_\downarrow(\cdot)\big)\mathbf{J_{6\downarrow}}\big(\cdot,\mathbf{D}_\downarrow(\mathbf{p})\big), \mathbf{p}/\lambda\Big).\end{aligned} \tag{5.20}$$

The main benefit of this implementation is, apart from some run-time optimization, the fact that no high resolution image except the 2-D image $\mathbf{I}$ has to be kept in memory.

In order to avoid filtering artefacts due to the data quantization and sampling introduced above, the standard deviations $\sigma_\mathbf{I}$, $\sigma_\mathbf{D}$, and $\sigma_\mathbf{S}$ may be chosen greater than $s_\mathbf{I}$, $s_\mathbf{D}$, and $s_\mathbf{S}$, respectively. Otherwise, the approximation may be poor, *i.e.,* numerically unstable. According to the above mappings (see equations (5.7) and (5.9)), the noise due to quantization only affects the range mapping functions, *i.e.,* F^{I_k} and H^{D_l}, and both the intensity values of the 2-D image $\mathbf{I}(\mathbf{q})$ as well as the depth measurements of the depth map $\mathbf{D}(\mathbf{q})$ are preserved.

5.3.3 Special treatment of background pixels

Background pixels are those pixels in the imager that have not been able to estimate a distance measurement. These pixels are identified during the generation of the provided depth map and set to a defined value. In the case of IEE's ToF camera (see Section 2.3), background pixels are equal to the maximum reachable distance, *i.e.,* 7500 mm. However, this constant value will differ depending on the camera manufacturer.

Background pixels must be identified and treated separately during the filtering process in order to avoid considering their default value as a real measurement. Otherwise, non valid distance measurements would appear within the enhanced depth map. To that end, we compute a relative background weight $\mathbf{W_{bg}}$ for each pixel $\mathbf{p}$ within the enhanced depth map by integrating the spatial kernel over all background pixels, respectively over all pixels as follows

$$\mathbf{W_{bg}}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N_{bg}(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\mathbf{Q}(\mathbf{q})\mathbf{B}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q})\mathbf{Q}(\mathbf{q})}, \tag{5.21}$$

where $N_{bg}(\mathbf{p})$ is the neighbourhood of background pixels and $\mathbf{B}$ is a mask of the same resolution as the depth map $\mathbf{D}$ to be filtered where only those pixels that correspond to background pixels in $\mathbf{D}$ are set to 1. Non-background pixels are set to 0. Thus, the resulting value within the enhanced depth map for the selected pixel $\mathbf{p}$ will be directly set to the defined background pixel value in the case where $\mathbf{W_{bg}}(\mathbf{p}) \geq 0.5$. Instead, the filtered value is computed according to (5.1) and (5.3), taking however, only the foreground pixels into account.

Chapter 6

Experimental results

This chapter analyses four main aspects of our UML filter. We first quantify the improvement achieved on the final depth maps resulting from the low-level data fusion process as compared to the original raw depth maps delivered by the ToF camera alone. To that end, we evaluate the dimensions of a box under different setup configurations, *i.e.,* we set the box at different locations within the field of view of the system and we repeat the experiments at different depths. Then, we quantify the UML filter against state-of-the-art low-level filtering solutions. In that case, we consider our own recorded sequences as well as various scenes from the Middelbury dataset [Mid11]. Then, we check the filter response against noise, and we end with a runtime analysis using the filter implementation proposed in Section 5.3.

6.1 Quantification of depth map enhancement

We start the assessment of our method with a quantitative comparison between the raw depth map acquired by the ToF camera and the enhanced depth map resulting from the low-level data fusion process proposed in Chapter 5. To that end, we have used the camera rig described in Section 2.3 previously calibrated using the proposed calibration approach in Section 2.4 and frame-synchronised. We first recorded a box with known dimensions (see Figure 6.1) and from 8 different setup configurations, *i.e.,* we displaced the box along the x, y, and z axes with respect to the hybrid ToF multi-camera rig. Each sequence contains a total of 20 frames. For the given setup, the pixel size is roughly (16 mm × 25 mm) for the 3D MLI Sensor$^{\text{TM}}$and (2.8 mm × 2.8 mm)

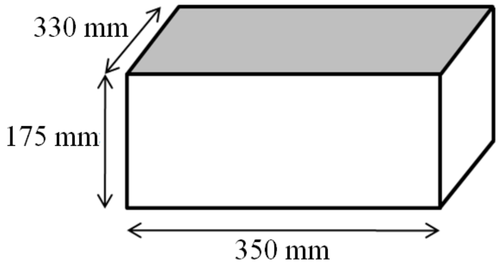


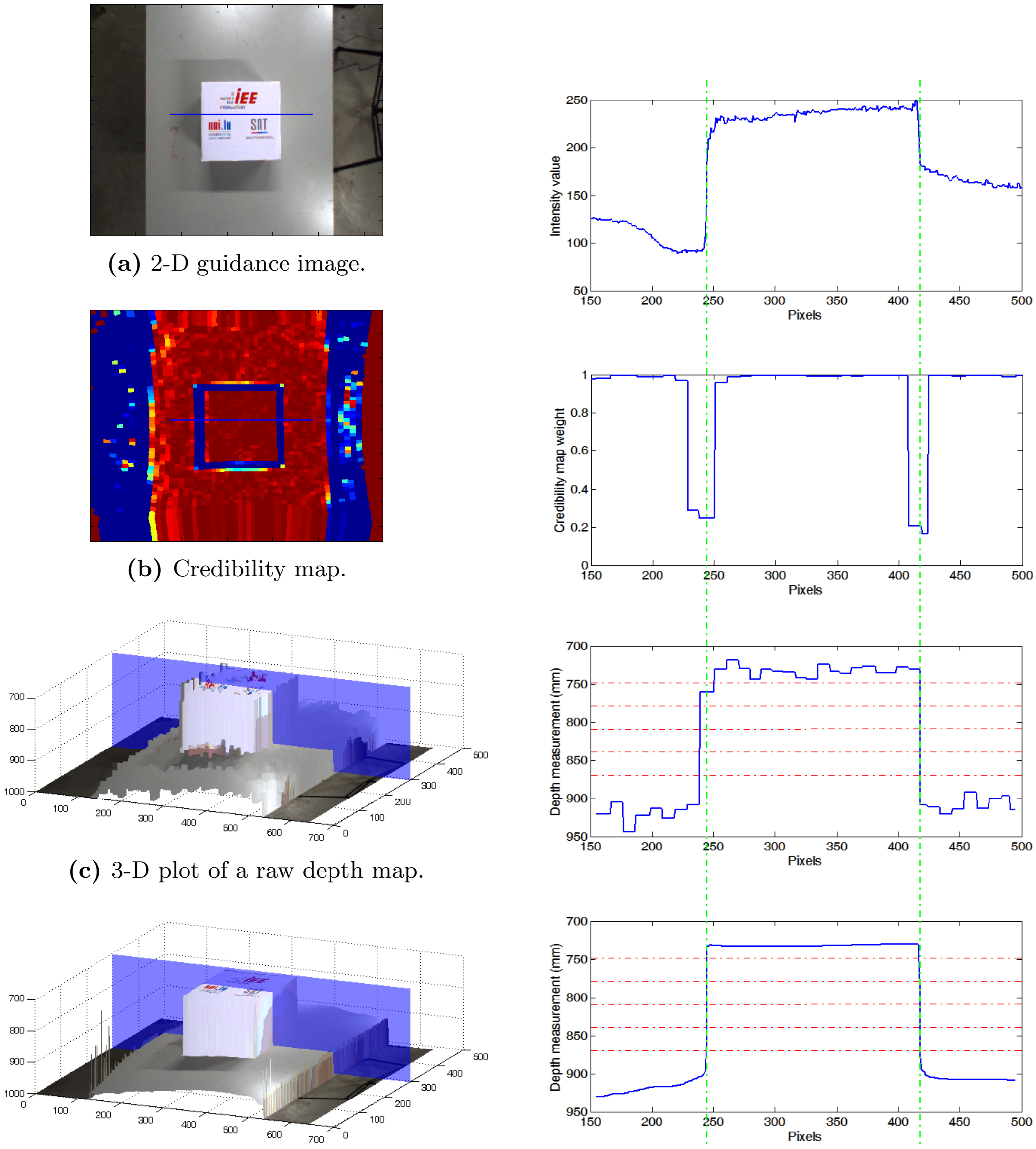
Figure 6.1: Dimensions of the selected box for the experimental test; width = 350 mm, height = 175 mm, and depth = 330 mm.

for the Flea®2 camera. We obtain the box dimensions by fitting a rectangle to the box area that has been previously segmented using a depth threshold. From Table 6.1 we notice that the measured box dimensions are much more accurate while considering the enhanced depth maps. Indeed, the accuracy for the lateral dimensions, *i.e.,* length and width, are on average 1.3 mm when considering the enhanced depth maps, which corresponds to half the pixel resolution on the Flea®2 camera. In contrast, the accuracy when considering the raw depth maps is only 12 mm. The filling ratio of the fitted box has increased accordingly. This demonstrates that depth edges have been accurately adjusted according to the guidance image resolution. Regarding the accuracy of the height measurement of the box, which is not related to the pixel resolution but to the noise within the distance measurements, we observe that has increased by a factor of 3. Indeed, the error due to the noise within distance measurements (see Section 1.2.2.1) is compensate when filtering, thanks to the nature of the bilateral filtering in which our

Table 6.1: Quantitative comparison of the box dimensions measured from the raw and the enhanced depth maps (units are in millimetres). Shown are the mean of the measured dimensions and accuracy taken over the 8 box configurations.

	Box dimensions	Raw depth map		Enhanced depth map	
		Mean Measure	Mean Accuracy	Mean Measure	Mean Accuracy
Width	350	360	11.7	349	1.4
Height	175	187	12.2	177	4.2
Depth	330	324	6.3	330	1.2
Filling ratio	100%	98.50%	1.50%	99.72%	0.28%

filters are based (see Section 4.1).



(a) 2-D guidance image.

(b) Credibility map.

(c) 3-D plot of a raw depth map.

(d) 3-D plot of an enhanced depth map.

(e) Plot of the slice cut

Figure 6.2: Comparison between the raw and the enhanced depth maps. The dotted red lines in the right correspond to the selected depth threshold values.

In order to compute the dimensions of the box, we have considered the best depth threshold that segments the surface of the box. However, the selection of the best depth threshold value is far from a trivial task. Indeed, a slight variation on the depth threshold value may significantly affect the computed box dimensions. Note the difference along the left depth edge in the plot of the section in Figure 6.2c when considering a depth threshold value of 750 mm or 810 mm. In contrast and from the credibility map contribution (see Figure 6.2b), the misaligned depth edges from the raw depth maps are accurately adjusted resulting in enhanced depth maps that allow a larger tolerance while selecting the depth threshold value (see Figure 6.2d). Table 6.2 reports the dimensions of the test box when considering different depth threshold values.

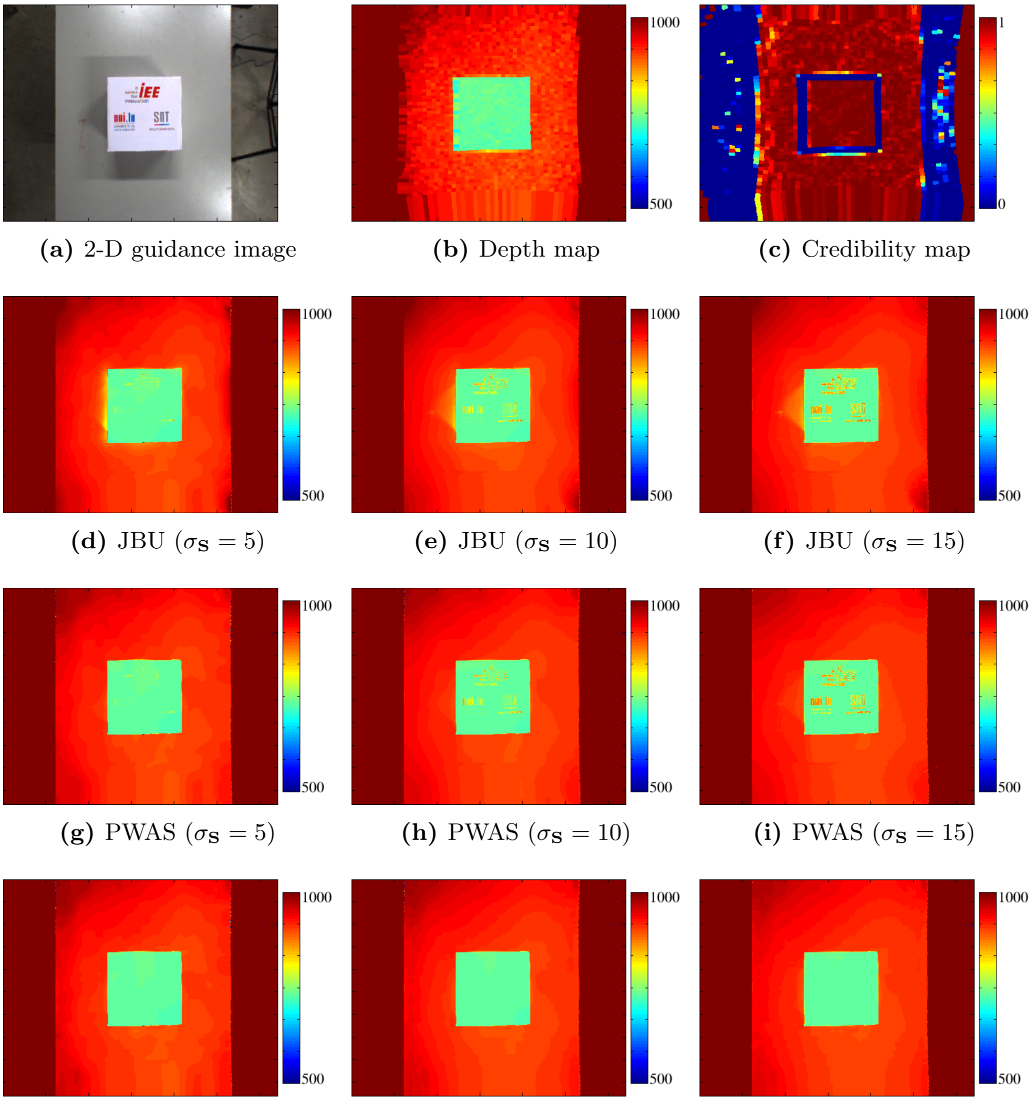
Table 6.2: Robustness against depth threshold selection (units are in millimetres). Shown are the mean values of the measured dimensions taken over the 8 box configurations and their variation (std) with the threshold value.

		Depth threshold					Std
		750	780	810	840	870	
Enhanced depth map	Width	345	346	346	347	347	0.8
	Height	178	178	178	178	177	3.2
	Depth	327	328	328	329	329	1.0
	Filling ratio	99.90%	99.68%	99.73%	99.61%	99.65%	0.04%
Raw depth map	Width	356	363	363	362	362	3.0
	Height	189	188	188	187	185	1.4
	Depth	323	324	324	327	334	4.5
	Filling ratio	97.76%	99.31%	99.31%	98.89%	98.73%	0.63%

6.2 Comparison to alternative filters

6.2.1 Comparison using recorded data

We perform a qualitative comparison of the proposed UML filter against the JBU and the PWAS filters employing real data. Thereby, we have also varied the sigma spatial $\sigma_{\mathbf{S}}$ to demonstrate the influence of these filter settings. From Figure 6.3, we clearly see the contribution of each filter, *i.e.,* the JBU, the PWAS and the UML filter. Regarding the JBU filter, we notice that the higher the sigma spatial $\sigma_{\mathbf{S}}$, the better the depth edges are adjusted. However, a large $\sigma_{\mathbf{S}}$ makes texture copying appear, as shown in Figure 6.3f. The texture copying artefact can be almost eliminated by setting a smaller $\sigma_{\mathbf{S}}$ value, as shown in Figure 6.3d. However, the edge blurring artefact appears due to the 2-D



(a) 2-D guidance image **(b)** Depth map **(c)** Credibility map

(d) JBU ($\sigma_\text{S} = 5$) **(e)** JBU ($\sigma_\text{S} = 10$) **(f)** JBU ($\sigma_\text{S} = 15$)

(g) PWAS ($\sigma_\text{S} = 5$) **(h)** PWAS ($\sigma_\text{S} = 10$) **(i)** PWAS ($\sigma_\text{S} = 15$)

(j) UML ($\sigma_\text{S} = 5$) **(k)** UML ($\sigma_\text{S} = 10$) **(l)** UML ($\sigma_\text{S} = 15$)

Figure 6.3: Depth enhancement filtering comparison based on the sigma spatial σ_S value.

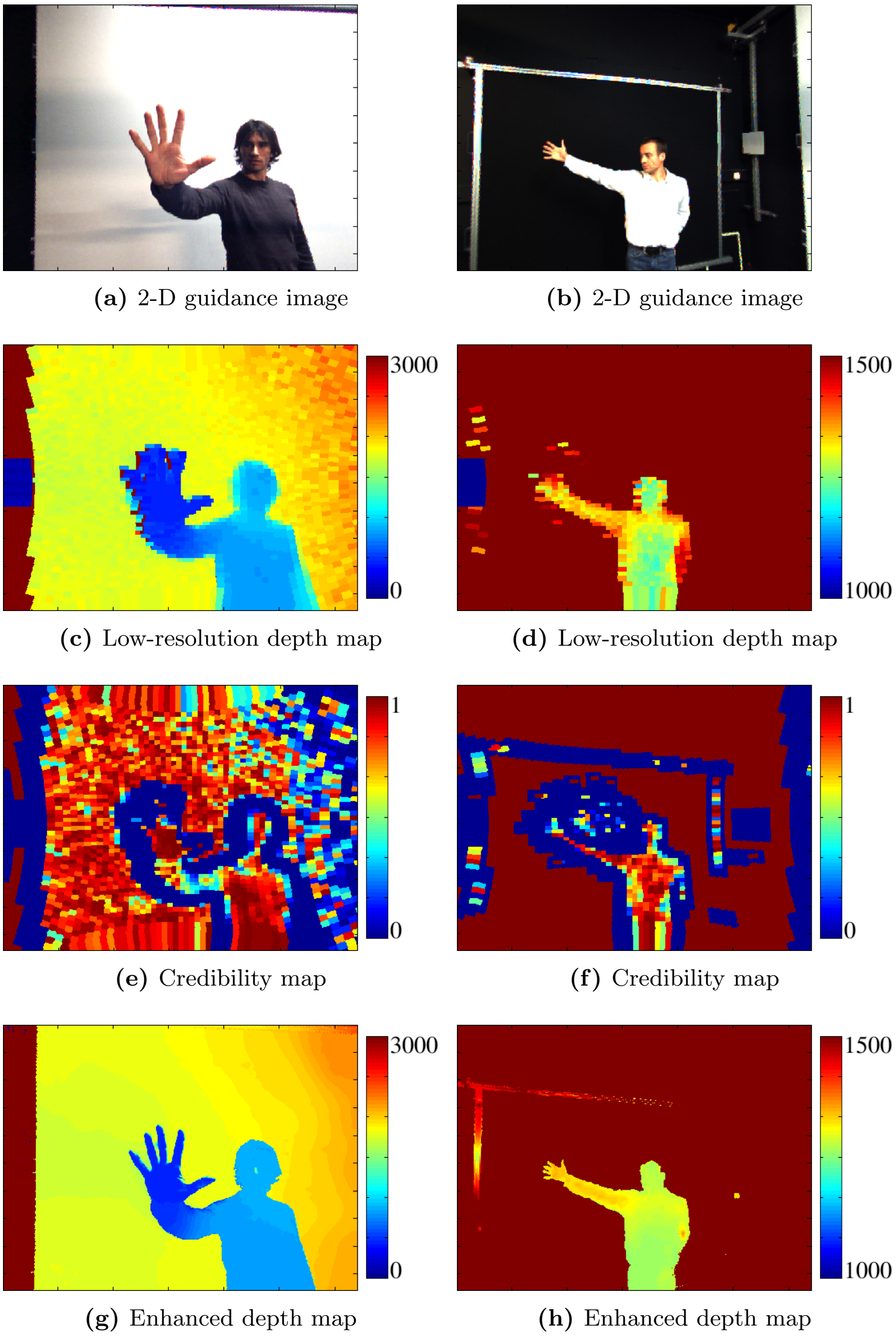
and depth edge misalignment problem discussed in Figure 5.1. Thus, it is necessary to tune the σ_S value to minimize both edge blurring and texture copying artefacts. The PWAS filter tackles the edge blurring artefact and significantly reduces the texture copying artefact (see Figure 6.3g, Figure 6.3h, and Figure 6.3i). However, texture copying can remain since σ_S has to be chosen large enough to cover the credibility map boundaries (see Figure 6.3c). Instead, the UML filter (see Figure 6.3j, Figure 6.3k, and

Figure 6.3l) perfectly copes with texture copying and edge blurring artefacts providing accurate enhanced depth maps.

Figure 6.4 shows the final depth map for two real scenarios in addition to the input data to be filtered, *i.e.,* the high-resolution 2-D image, the low-resolution depth map and the credibility map. First, one recognizes that our adaptive filter enhances the low-resolution depth maps from (56 pixels × 61 pixels) to the VGA-resolution of the coupled 2-D image. Also, the noise level has been greatly reduced. From the credibility map, depth edges weighted with a lower value, *i.e.,* closer to 0, are accurately adjusted to the ones in the guidance image. Hence, resolving details like the fingers of the person in Figure 6.4g and Figure 6.4h, that are not resolved in the raw depth map. Figure 6.5 compares a detailed region of our enhanced depth maps with the ones given by the JBU and the PWAS filters. In the first example one recognizes the edge blurring within the contour of the hand when filtering with JBU, which is drastically reduced for both the PWAS and the UML filter. Although PWAS performance is not improved when adjusting depth edges, depth accuracy for pixels with a high credibility weight is increased by maintaining smooth regions. Also, Figure 6.5 shows an example where the black belt of the person has the same (black) colour as the background. Contrary to the JBU and PWAS responses, our adaptive filter correctly addresses that situation, as shown in Figure 6.5f.

6.2.2 Comparison using the Middelbury dataset

In order to quantify the accuracy of our method against the alternative filtering solutions, we employ the *Teddy*, *Art*, *Books*, and *Moebius* scenes from the Middlebury dataset [Mid11]. Each scene contains an intensity image and its corresponding disparity map, from which we have generated a depth map as a ground truth using the also provided system specifications. We simulate the low-resolution depth map to be enhanced by downsampling (at different sampling rates) the ground truth depth map. Figure 6.6 shows an example of the *Teddy* and the *Art* scenes where the ground truth depth maps were downsampled by a factor of nine. As also occurs in the real data examples, the UML filter enhances the downsampled depth map to the intensity image resolution. Figure 6.7 shows a zoomed area where we can observe the same differences between the different filters applied on the real data examples shown in Figure 6.5. The JBU filter shows a strong edge blurring where the grey image contrast is low,



(a) 2-D guidance image

(b) 2-D guidance image

(c) Low-resolution depth map

(d) Low-resolution depth map

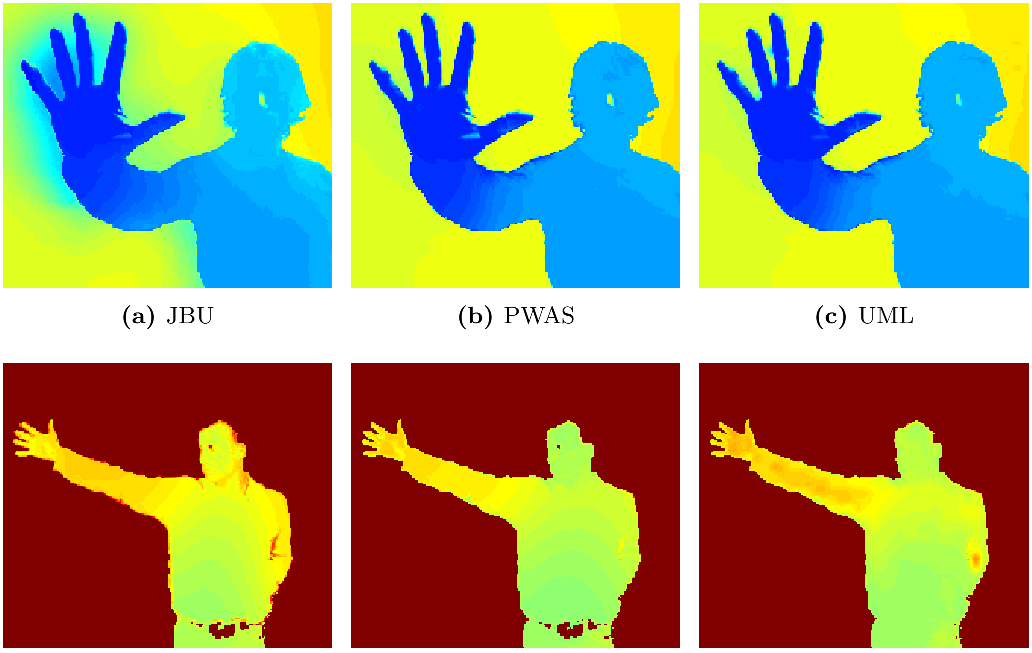
(e) Credibility map

(f) Credibility map

(g) Enhanced depth map

(h) Enhanced depth map

Figure 6.4: Depth map enhancement applying the UML filter onto our own recorded sequences.



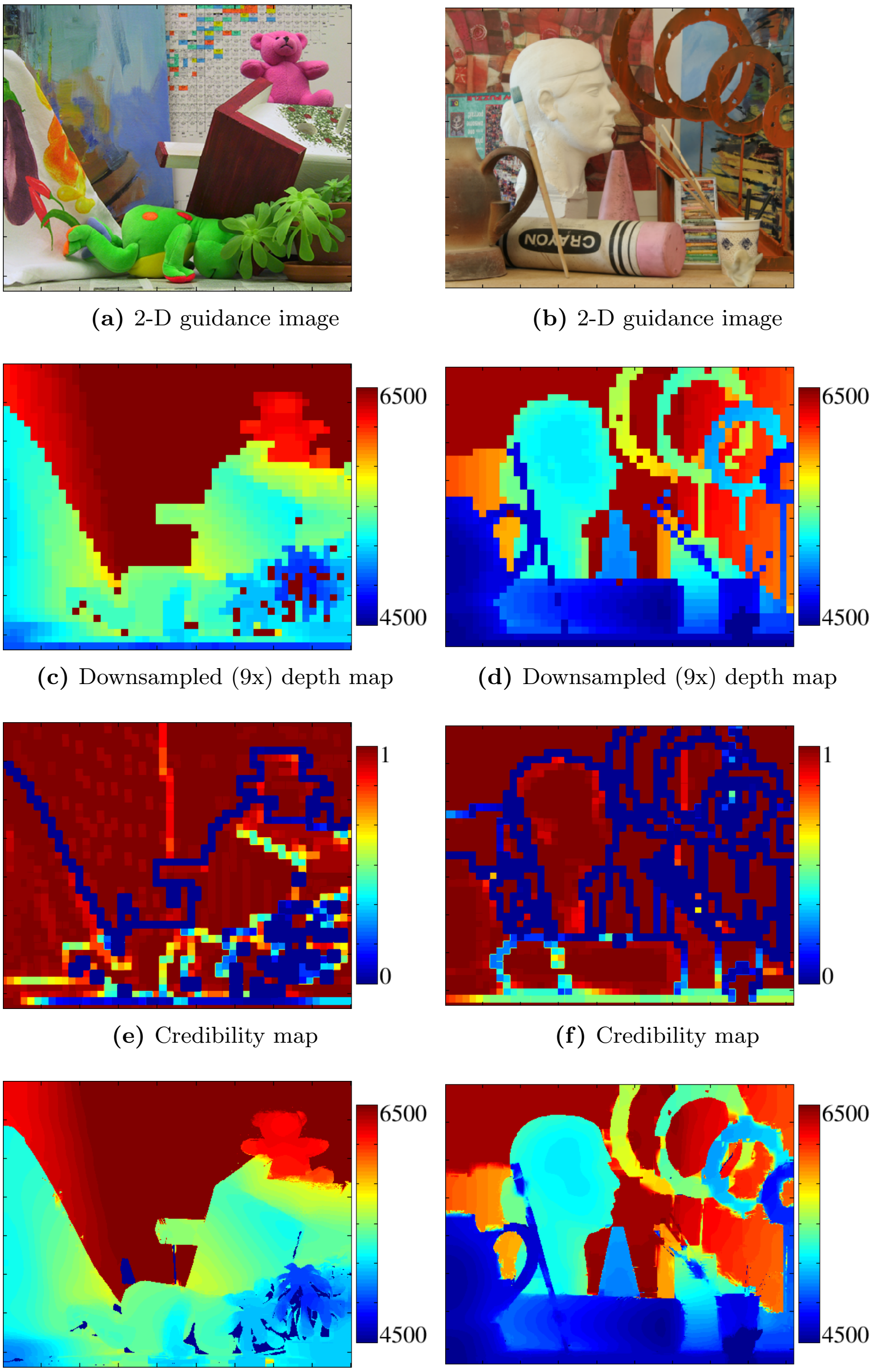
(a) JBU (b) PWAS (c) UML

(d) JBU (e) PWAS (f) UML

Figure 6.5: Visual comparison of enhanced depth maps using different depth enhancement filters.

e.g., around the teddy's ears, marked as red rectangles in Figure 6.7c. In addition to strongly reducing this artefact, our adaptive filter also removes the texture copying effect inside the teddy's head (see the green marked rectangle in Figure 6.7c), which remains in both JBU and PWAS final depth maps. Figure 6.8 shows an example of the limitations of the UML filter. This scene contains really small objects (in the regions indicated by red rectangles in Figure 6.8c) that are tackled as outliers. This occurs because the credibility map gives a low weight to these objects and consequently their value is replaced by the neighbourhood pixel values. Exactly the same occurs when filtering using PWAS. However, on the larger surfaces in the scene (see areas inside the green rectangles in Figure 6.8c), the resulting depth values of the UML filter are much more accurate than those of JBU and thus, on average, a better performance can be expected.

Although the root mean square error (RMSE) is a frequently-used measure to quantify the visibility of errors between a treated image and a reference image, we use an alternative complementary framework for quality assessment based on the degradation



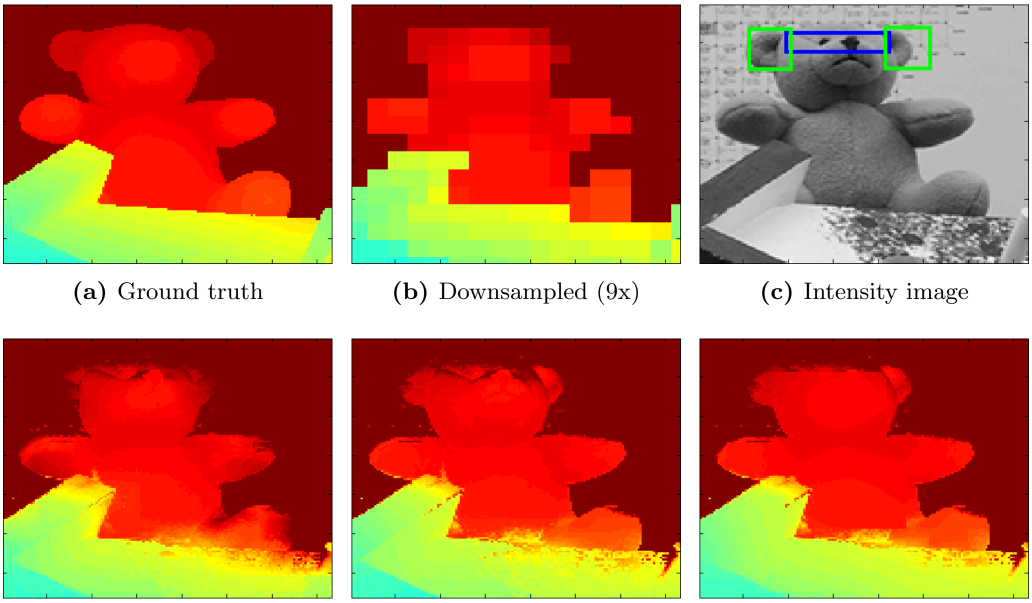
(a) 2-D guidance image **(b)** 2-D guidance image

(c) Downsampled (9x) depth map **(d)** Downsampled (9x) depth map

(e) Credibility map **(f)** Credibility map

(g) Enhanced depth map **(h)** Enhanced depth map

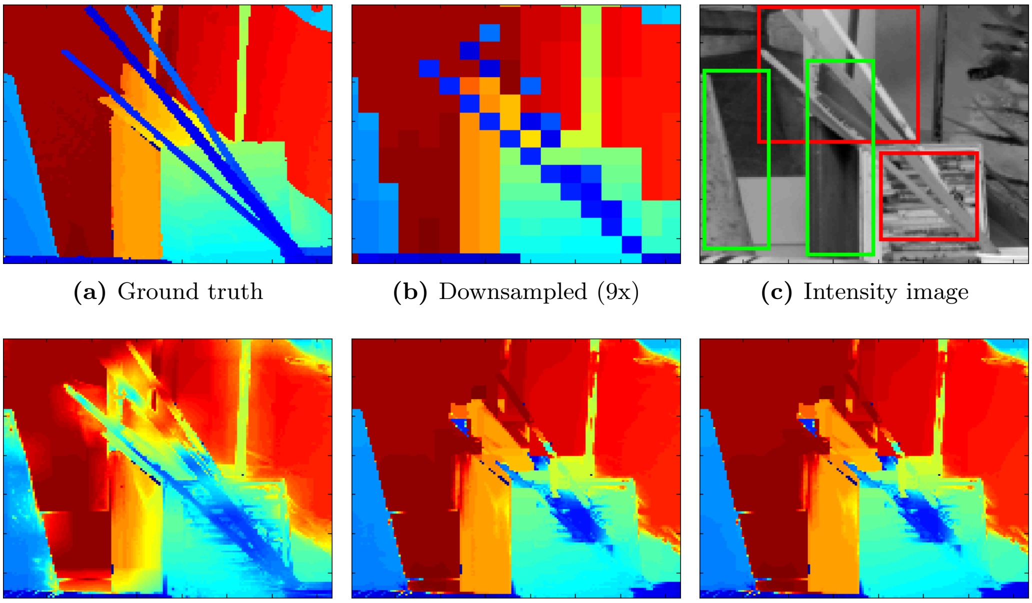
Figure 6.6: Depth map enhancement employing the *Teddy* and the *Art* scenes, 1^{st} and 2^{nd} rows respectively.



(a) Ground truth **(b)** Downsampled (9x) **(c)** Intensity image

(d) JBU output, SSIM: 62.62 **(e)** PWAS output, SSIM: 69.14 **(f)** UML output, SSIM: 69.95

Figure 6.7: Visual filtering output comparison employing the *Teddy* scene.



(a) Ground truth **(b)** Downsampled (9x) **(c)** Intensity image

(d) JBU output, SSIM: 44.01 **(e)** PWAS output, SSIM: 49.95 **(f)** UML output, SSIM: 50.13

Figure 6.8: Visual filtering output comparison employing the *Art* scene.

of structural information, the Structural SIMilarity (SSIM) Index [ZBSS04]. Table 6.3 reports the SSIM measure that quantifies our method against alternative depth enhancement solutions. We can observe that under a global error measure, the UML filter performs at least as well as the PWAS filter. The only case where the UML filter does not outperform the JBU is in the *Art* scene with a downsampling rate of 3. This occurs due to the suppressed small details in the scene as discussed above. For higher downsampling rates, the performance is, however, superior to JBU.

Table 6.3: Quantitative comparison using the SSIM measure (100 corresponds to a perfect matching with the ground truth).

	Downsampled	JBU	PWAS	UML
Teddy	3x	97.65	97.71	97.81
	5x	96.29	96.80	96.90
	9x	93.47	94.57	94.79
Moebius	3x	96.57	96.65	96.71
	5x	94.67	94.68	94.75
	9x	90.75	90.96	91.45
Books	3x	96.89	97.44	97.46
	5x	95.59	96.11	96.13
	9x	92.51	93.01	93.59
Art	3x	92.96	91.52	91.59
	5x	88.42	88.07	88.21
	9x	81.09	83.28	83.42

6.3 Robustness to noise

The main sources of noise that affect to the given distance measurements, as described in Section 1.2.2.1, generally provoke random variations within the provided depth maps, as shown in Figure 6.4c and Figure 6.4d. We thus want to quantify how the UML filter behaves against different noise levels. Due to the active illumination of ToF cameras, the noise level increases according to the measured distance, as discussed in Section 1.2.2.1. Therefore we simulate this behaviour by adding Gaussian noise with a standard deviation linearly dependent on the distance measurement [LS01]. We used the *Teddy* scene downsampled by a factor of 5 and with a noise of ±100 mm at the maximum distance (8976 mm). The results in the graph from Figure 6.9 were obtained by a Monte Carlo simulation over 100 times, which gave us an accuracy of $\pm 1.2 \times 10^{-3}$, $\pm 2.2 \times 10^{-4}$, and $\pm 2.2 \times 10^{-4}$ for the JBU, the PWAS, and the UML filter, respectively. Within individual executions only the last digit varies. Then, from Figure 6.9 we note that the UML filter is more robust to noise than both JBU and PWAS filters independently of the added noise level.

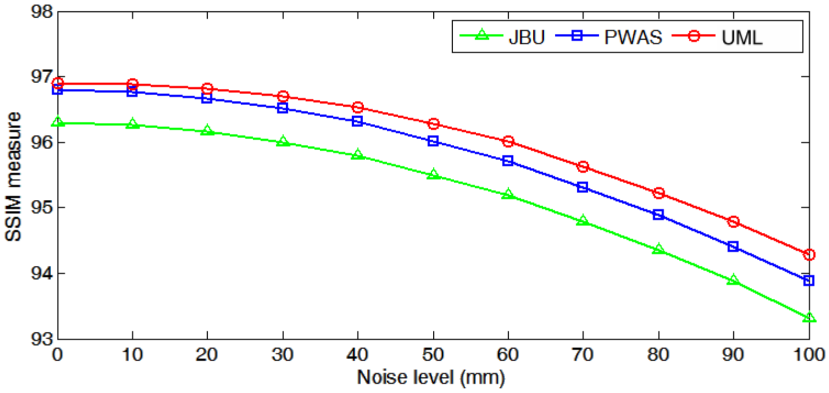


Figure 6.9: JBU, PWAS, and UML filter response against Gaussian noise of ±100 mm at the maximum distance (8976 mm).

6.4 Runtime analysis

We next present a runtime analysis to validate that the implementation proposed in Section 5.3 enables real-time applications. We ran the tests to estimate the time consumption on an Intel Core 2 Solo processor SU3500 (1.4 GHz, 800 MHz FSB) with an integrated graphic card Intel GMA 4500MHD. The filter was implemented in C language and the tests have been performed on our own recorded scenes, enhancing from (56 pixels × 61 pixels) to VGA-resolution. Table 6.4 reports the seconds per filtered frame calculated over 1000 iterations. Also, we have sampled the input data by a factor of 3x, 5x, 9x, and 17x. With the latter sampling factor, the filtering process only takes 0.08 seconds per frame. In addition, we have quantified the corresponding induced error to each sampling rate. Table 6.5 reports the SSIM measure considering the non downsampled case as a reference, and the final depth maps for each sampling rate. We notice that a sampling factor of 9x or 17x drastically reduces the time consumption without inducing a significant error in the final depth map. As a consequence, data

Table 6.4: Run-time analysis for the tested input data sampling rates (units are in seconds; average over 1000 iterations).

Sampling	JBU	PWAS	UML
1x	1.88	1.89	13.59
3x	0.49	0.50	3.17
5x	0.13	0.13	0.65
9x	0.06	0.06	0.18
17x	0.05	0.05	0.08

sampling enables a real-time depth enhancement despite being restricted by the ToF camera frame rate of 10 fps.

Table 6.5: SSIM measure depending on the input data sampling.

Sampling	JBU	PWAS	UML
3x	95.78	99.71	99.85
5x	95.46	99.51	99.65
9x	94.89	98.80	98.86
17x	92.25	95.11	95.17

Part III

Extensions

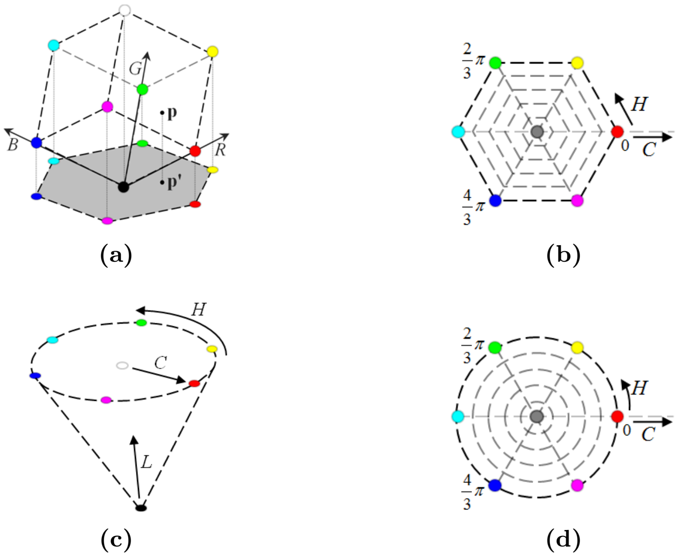
Chapter 7

Colour representation for edge blurring avoidance

Unlike the alternative techniques for depth enhancement presented in Section 4.2, our proposed filtering techniques in Chapter 5 cope very well with the edge blurring artefact due to the misalignment between 2-D and depth edges. However, edge blurring can still appear when depth edges have no corresponding 2-D edge. This occurs when background and foreground objects have a similar colour. Thus, depth edges cannot be accordingly adjusted to any reference edge yielding to edge blurring. This situation can occur in many scenarios but we notice that in general it is due to the transformation from the original coloured images to their grayscale version. Although the results can be more accurate when considering the full colour information, most systems are restricted to use a grayscale converted image to accomplish with the processing time and memory constraints, mainly if real-time is a requirement. A grayscale image is defined as a linear combination of the red, green, and blue channels in the RGB space. This combination leads to a non-unique representation of the true colours, which may cause objects having a different colour to be represented with the same grayscale value. As a result, a more accurate processing of images requires the use of their true colours, and using three components. Indeed, most filtering techniques have their definitions extended to three channels. Paris et al. tested several alternatives on a colour image [PD09]. They first filtered an RGB image as three independent channels. Despite a correlation between the three channels, they showed that edges may be smoothed in one channel while they are preserved in another channel. This consequently induces incoherent results between channels. They then tested reducing these inconsistencies, that resulted in bleeding effect, by processing the R, G, and B channels altogether. The downside of this approach was, however, a longer computational time required for processing. The same authors tested filtering images in the CIE-Lab space [KA00], which is known to be perceptually meaningful. Indeed, this solved the colour-bleeding problem but not the demanding computation time. In this chapter, we propose to reduce the complexity of processing 3 channels by compactly storing the same information in only one channel. To that end, we exploit the geometrical structure of 3-D conical colour spaces and show how to accurately define one parameter to represent the solid HCL conical colour space [SM05]. We equip this representation with an associated colour similarity measure inspired from the cylindrical distance used for cylindrical and conic colour spaces such as HSV/HSL [SM05, GW02, ST97]. In addition, the proposed colour model represents a novel colour ordering that might be useful in the context of colour morphology [Ang07]. Indeed, morphological colour operators, *i.e.,* morphological filters such as opening and closing or morphological centre, can be adapted to the proposed colour ordering for further image processing such as image denoising. We note that our work is not only related to data compression from 3-D to 1-D [VD10] but deals also with a colour codification for an efficient subsequent processing.

7.1 Background: Transformation from RGB to HCL

The objective of this work is to define a colour model that is almost as reduced as the grayscale representation, but preserving all the colour information contained in 3-D spaces. In other words, we want to define a model that is in one dimension, and that is still reversible from and to the RGB colour cube (see Figure 7.1a). In addition, our colour model should also bring in a perceptual meaning. This last property will be important when computing the distance between colours for pattern recognition purposes. To that end, we base our work on the conic HCL model shown in Figure 7.1(c) [SM05]. We define in what follows the HCL model and relate it to the RGB cube as it will be a transition step in converting our proposed model to RGB space and vice versa. The projection of the RGB cube onto a regular hexagon Ψ in the *chromaticiy* plane defines the chroma C and hue H related to R, G, and B (Figure 7.1b). Let $\mathbf{p}'$ be the projection of a point $\mathbf{p}$ in the RGB cube on Ψ and $\mathbf{o}$ be the origin of Ψ. Geometrically, the



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

Figure 7.1: (a) Cubic RGB model projected onto a regular hexagon in the chromaticity plane. From (b) to (d), warping from hexagons into circles. (c) Conic HCL model.

chroma component c along C, of $\mathbf{p}$, is the length of $\overrightarrow{\mathbf{op}'}$ relative to the maximal radius of Ψ passing through $\mathbf{p}'$. The hue component h corresponds to the angle formed by $\overrightarrow{\mathbf{op}'}$ and $\overrightarrow{\mathbf{or}'}$, where $\mathbf{r}'$ is the projection of the red colour $\mathbf{r} = (1, 0, 0)$ on Ψ. The luminance component l is equal to $||\overrightarrow{\mathbf{pp}'}||$. This is equivalent to $c = m_1 - m_2$ and $l = \frac{1}{2}(m_1 + m_2)$, where $m_1 = \max(r, g, b)$, and $m_2 = \min(r, g, b)$, and

$$h = \begin{cases} \text{undefined} & \text{if} \quad c = 0, \\ \frac{\pi}{3}(\frac{g-b}{c} \mod 6) & \text{if} \quad m_1 = r, \\ \frac{\pi}{3}(\frac{b-r}{c} + 2) & \text{if} \quad m_1 = g, \\ \frac{\pi}{3}(\frac{r-g}{c} + 4) & \text{if} \quad m_1 = b. \end{cases} \tag{7.1}$$

In what follows, we propose an approximation of the HCL space that only requires two parameters for colour description. There are alternative methods such as the proposed by Vahdat et al. [VD10], that describe the colour information by two or even one parameter. However, these methods are mainly related to data compression or codification and thus it is necessary to uncompress or decode the data to treat it.

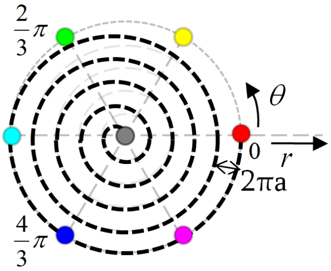


Figure 7.2: Chromaticity disk approximation with a spiral.

7.2 Proposed spiral model

We propose to describe the colour information contained in the HCL (hue, chroma, luminance) conic space by approximating the cone using two parameters, θ and l, instead of using the three coordinates (h, c, l). We thus keep the luminance value l, and rewrite c and h as functions of a new variable θ. Our key idea is to approximate the chromaticity disk with a spiral, as shown in Figure 7.2. Specifically, we choose to use an *Archimedean* spiral [Loc67] whose radial distance is defined as

$$r(\theta) := \mathsf{a} \cdot \theta, \tag{7.2}$$

where $\mathsf{a} = \frac{1}{2\pi}$ is a constant defining the distance between successive turns, and θ is the polar angle of the spiral, such that $\theta \in [0, 2\pi\mathsf{K}]$, K being the total number of turns. We approximate the chromaticity disk by fitting the spiral to it, such that

$$h(\theta) = \theta + 2\pi k, \tag{7.3}$$

where $k \in \{0, 1, \cdots, \mathsf{K}\}$, and the $C-$axis is uniformly sampled into $(\mathsf{K}+1)$ values c_k, with a step equal to a. We note that c_k is dependent on the hue h, or equivalently of the angle θ. Thus we define $c_k(\theta)$ as

$$c_k(\theta) = r\big(h(\theta)\big) + \mathsf{a} \cdot k. \tag{7.4}$$

We then save the luminance value l, and rewrite h and c as functions of a new variable θ such that

$$\theta = h - 2\pi \operatorname{round}\left(\mathsf{K} \cdot c - \frac{h}{2\pi}\right). \tag{7.5}$$

By setting the spiral extremities as the starting point $(c_0, h_0) = (0, 0)$, and ending point $(c_{\max}, h_{\max}) = (1, 0)$, and by replacing these values in (7.2) and (7.3), we find $\mathsf{a} = 1/\mathsf{K}$. In the continuous case, *i.e.,* $\mathsf{K} \to \infty$, we may write

$$c = r(\theta) = \frac{\theta}{2\pi\mathsf{K}} \quad \Rightarrow \quad \theta = 2\pi\mathsf{K} \cdot c. \tag{7.6}$$

Replacing (7.6) in (7.3), we find

$$k = \text{round}\left(\mathsf{K} \cdot c - \frac{h}{2\pi}\right), \tag{7.7}$$

with round(·) being a rounding function that assigns the nearest integer value to k. We may now define the transformation from (c, h) to θ as follows:

$$\theta = h - 2\pi\,\text{round}\left(\mathsf{K} \cdot c - \frac{h}{2\pi}\right). \tag{7.8}$$

The inverse transformation from θ to (c, h) is fully defined by (7.2), (7.3), and (7.10), with $k = \text{round}\left(\frac{\theta - \theta \mod (2\pi)}{2\pi}\right)$. Next step is the conversion from the recovered c and h values to the initial (r, g, b). We compute an intermediate value $x = c(1 - |(\frac{3}{\pi}h)$ mod $2 - 1|)$ to be applied to the following system of equations

$$(r', g', b') = \begin{cases} (0,0,0) & \text{if} \quad \text{h} \quad \text{is undefined,} \\ (c,x,0) & \text{if} \quad 0 \le h < \frac{\pi}{3}, \\ (x,c,0) & \text{if} \quad \frac{\pi}{3} \le h < \frac{2\pi}{3}, \\ (0,c,x) & \text{if} \quad \frac{2\pi}{3} \le h < \pi, \\ (0,x,c) & \text{if} \quad \pi \le h < \frac{4\pi}{3}, \\ (x,0,c) & \text{if} \quad \frac{4\pi}{3} \le h < \frac{5\pi}{3}, \\ (c,0,x) & \text{if} \quad \frac{5\pi}{3} \le h < 2\pi. \end{cases} \tag{7.9}$$

To obtain the point $\mathbf{p} = (r, g, b)$ from $\mathbf{q} = (r', g', b')$, we translate $\mathbf{q}$ in the R, G, and B directions by the minimal distance m_2 defined in Section 7.1, *i.e.,* $(r, g, b) = (r' + m_2, g' + m_2, b' + m_2)$.

We note that the values of m_2 may be stored when extracting c from (r, g, b), or equivalently from $m_2 = (l - \frac{1}{2}c)$ (see Section 7.1).

7.3 Proposed 1-D colour model

In order to include the luminance parameter in the definition of the spiral model in (7.8), we propose to uniformly sample the luminance axis into $(\mathsf{K_L} + 1)$ values l_n. We thus have

$$l_n = \frac{n}{\mathsf{K_L}}, \tag{7.10}$$

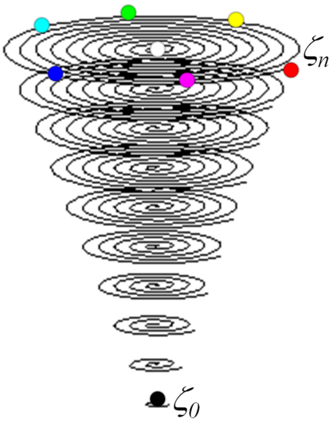


Figure 7.3: Approximation of the HCL cone by a set of spirals.

where $n \in \{0, 1, \cdots, \mathsf{K_L}\}$. At each luminance level l_n, we define a spiral of radius $r(\theta_n) = \mathsf{a} \cdot \theta_n$, with $\theta_n \in [0, 2\pi n]$. In other words, for larger sections of the cone, we impose a larger number of spiral turns, as shown in Figure 7.3. In order to keep a single parametrization of all the $\mathsf{K_L}$ spirals, we need to relate all of them to the same parameter. To that end, for a point on the spiral at the level l_n, we introduce the *cumulative angle* (CA) ζ as

$$\zeta = \zeta_{n-1} + \theta, \tag{7.11}$$

with

$$\zeta_{n-1} = \sum_{i=0}^{n-1} 2\pi \cdot i = n(n-1) \cdot \pi. \tag{7.12}$$

We note that ζ_{n-1} is the CA of the spirals at level l_{n-1}. The colour model proposed in (7.11), that we call CA model, reduces the HCL space to a 1-D representation by a single parameter ζ. We also notice that the CA model is reversible from and to the original HCL space and consequently to all colour spaces that can be converted from and to HCL, such as RGB. To do so, one simply needs to follow two steps; first to retrieve l or its approximation l_n, and second find the pair (h, c) at the corresponding level. By definition, we find that

$$\zeta_{n-1} \leq \zeta < \zeta_n \implies l \approx l_n \text{ and } \theta = \zeta - \zeta_{n-1}. \tag{7.13}$$

By solving the two inequalities in (7.13), we find an analytic expression for n avoiding a search among the intervals $[\zeta_{n-1}, \zeta_n[$. As an intermediary step, we get:

$$\left(\frac{-1+\sqrt{1+\frac{4}{\pi}\zeta}}{2}\right) < n \leq \left(\frac{1+\sqrt{1+\frac{4}{\pi}\zeta}}{2}\right). \tag{7.14}$$

Given that $n \in \mathbb{N}$, we find

$$n = \left\lfloor \frac{1}{2}\sqrt{1+\frac{4}{\pi}\zeta} + \frac{1}{2} \right\rfloor, \tag{7.15}$$

and from (7.10) we obtain l_n. $\lfloor x \rfloor$ indicates the floor function that maps a real number x to the largest integer not greater than x. To obtain the pair (h, c), we simply need to follow the same steps presented in Section 7.2 using the θ angle from (7.11). We finally get the following result

$$\begin{cases} h \cong \zeta \mod 2\pi, \\ c = \frac{1}{2\pi \mathsf{K}} \left\lfloor \frac{1}{2}\sqrt{1+\frac{4}{\pi}\zeta} + \frac{1}{2} \right\rfloor \left\lfloor \frac{1}{2}\sqrt{1+\frac{4}{\pi}\zeta} - \frac{1}{2} \right\rfloor, \\ l = \frac{1}{\mathsf{K_L}} \cdot \left\lfloor \frac{1}{2}\sqrt{1+\frac{4}{\pi}\zeta} + \frac{1}{2} \right\rfloor. \end{cases} \tag{7.16}$$

With equations (7.16) at hand, we fully defined a bijective transformation from (h, c, l) to ζ. This means that the CA representation encodes in one channel all the information contained in three channels with an easy way to back-transform. Such a model gives the possibility to apply the same algorithms used with grayscale images on full color information, but without extending the algorithms to 3 channels. We illustrate this by using the CA model for depth map filtering.

7.4 Application to depth map filtering

We consider as an application example the PWAS filter presented in Section 5.1. We recall that the term $f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}), \mathbf{I}(\mathbf{q})\big)$ in (5.1) corresponds to a Gaussian function with standard deviations $\sigma_{\mathbf{I}}$ which in fact is the distance between two image intensity values, $\mathbf{I}(\mathbf{p})$ and $\mathbf{I}(\mathbf{q})$, *i.e.,* $G_{\mathbf{I}}\big(d(\mathbf{I}(\mathbf{p}), \mathbf{I}(\mathbf{q}))\big)$. In the standard case of grayscale images, this distance $d(\cdot,\cdot)$ is Euclidean. In order to avoid edge blurring due to the grayscale conversion, we propose to represent the 2-D image $\mathbf{I}$ using the proposed CA model instead. We hence need to replace d with a new distance d_{CA} between two values

$\zeta_{\mathbf{p}} = \mathbf{I}(\mathbf{p})$ and $\zeta_{\mathbf{q}} = \mathbf{I}(\mathbf{q})$. We define d_{CA} as an approximation of the cylindrical distance d_{cyl} commonly used on the HCL space and defined as [SM05]

$$d_{cyl}(\zeta_{\mathbf{p}}, \zeta_{\mathbf{q}}) = \sqrt{(\Delta l)^2 + (\Delta c)^2 + 4 \cdot c_{\mathbf{p}} \cdot c_{\mathbf{q}} \cdot \sin^2\left(\frac{\Delta h}{2}\right)}, \tag{7.17}$$

where $c_{\mathbf{p}}$ and $c_{\mathbf{q}}$ are chrominance values corresponding to $\zeta_{\mathbf{p}}$ and $\zeta_{\mathbf{q}}$, respectively. We simplify this distance for our model by considering a normalized value $\Delta\zeta$ instead of the first term Δl. The normalization factor a_1 is such that we achieve a total distance of one between the two reference colours black and white, where all the other terms become zero. Thus, we find $\mathsf{a}_1 = \frac{1}{\mathsf{K_L}(\mathsf{K_L}+1)(\pi+1)}$. In addition, we consider the L_1 norm, and define d_{CA} as:

$$d_{CA}(\zeta_{\mathbf{p}}, \zeta_{\mathbf{q}}) = \mathsf{a}_1 \left|\Delta\zeta\right| + \left|\Delta c\right| + 2 \cdot \sqrt{c_{\mathbf{p}} \cdot c_{\mathbf{q}}} \left|\sin\left(\frac{\Delta\zeta}{2}\right)\right| .$$

Although the above expression is relatively complex due to computing chrominance values from (7.16), it is a first step towards defining a better distance d_{CA} in terms of performance. Indeed, the evaluation of d_{CA} is currently restricting the use of the CA model for our depth enhancement purposes.

7.5 Experimental results

We start by a global evaluation of the CA model by testing 100 different coloured images of objects from the Amsterdam Library of Object Images (ALOI) [Alo11]. These images are in the RGB space. We transform them to the proposed CA colour model by following the steps presented in Section 7.2 and Section 7.3. Figure 7.4 plots the root mean square error (RMSE) between the original RGB images and the recovered ones for K and $\mathsf{K_L}$ varying from 0 to 255. We see that the error drops whenever $\mathsf{K_L}$ is less than K, which means that a very sparse sampling of the luminance component can be sufficient for an accurate representation. Moreover, as soon as K reaches approximately 100, the error approaches zero. While this number may vary depending on the nature of the images, it clearly does not need to be set greater than 255, as the intensity of digital images falls between 0 and 255. We proceed by evaluating the performance of the PWAS filter presented in Section 5.1, when filtering considering grayscale or CA encoded images. We use data from the Middlebury stereo dataset [Mid11]. Each selected scene is represented by a 2-D RGB image and the corresponding depth map.

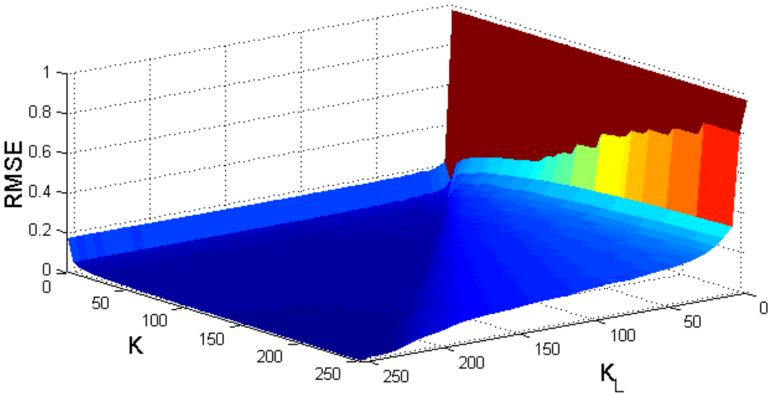
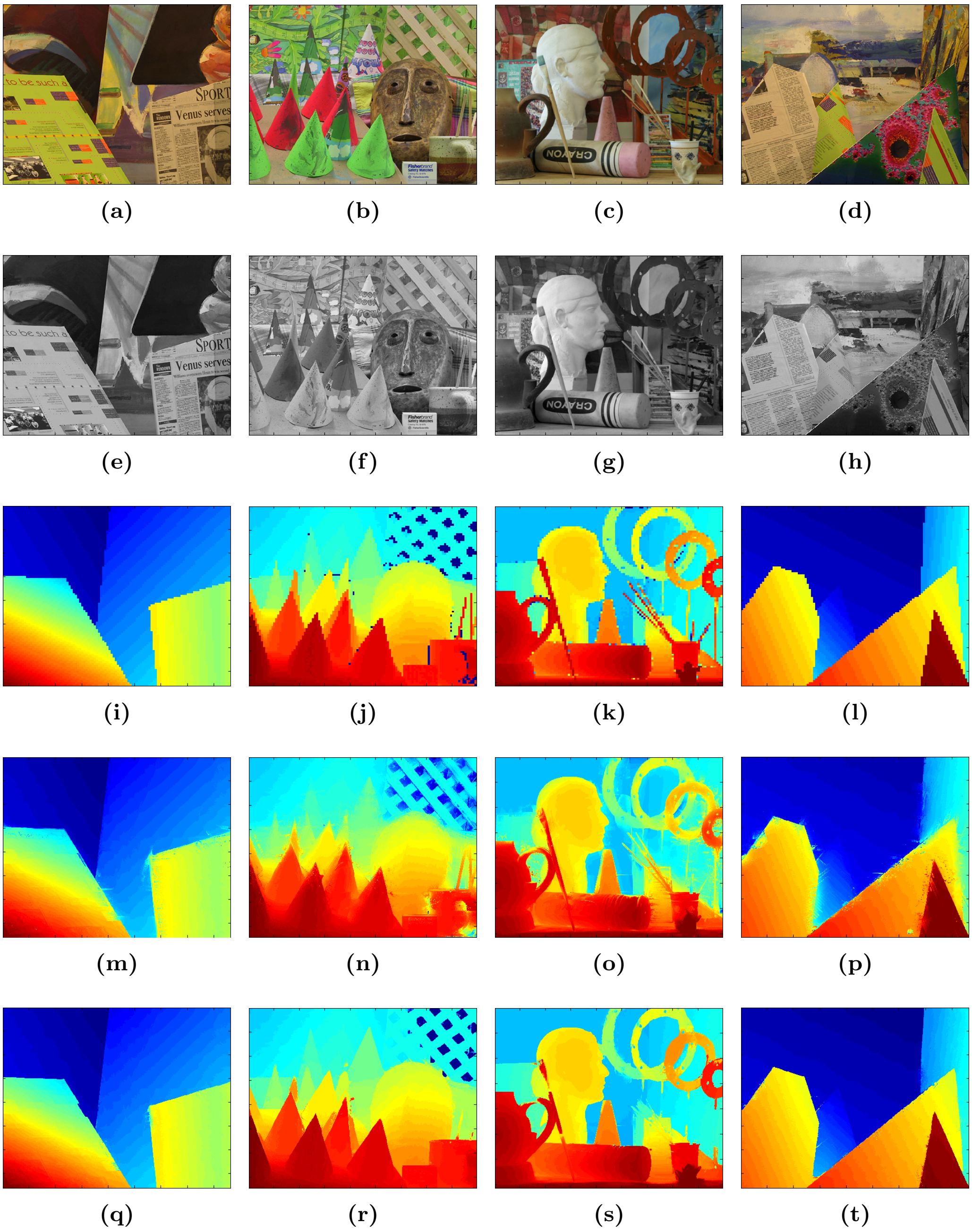


Figure 7.4: RMSE between 100 images from the ALOI database and their CA transformed versions.

We downsample the original depth maps by a factor of 8 in order to use them as low-resolution depth maps inputs (**D** in (5.1)) (see Figure 7.5). After the filtering process, we compare the resulting enhanced depth maps with the original ones by using the structural similarity index (SSIM) [WBSS04]. Table 7.1 reports the computed SSIM values, where 1 means that the enhanced depth map perfectly coincides with the original one. Note that the PWAS filter always performs better when considering CA images. This significant improvement is well illustrated in Figure 7.6 where we zoomed on a region from the Teddy scene. Figure 7.6 also illustrates the enhanced depth map using the 2-D guidance image with different colour representations, similarly to experiments in [PD09]. Edge blurring and texture copying are clearly visible when considering a grayscale image (Figure 7.6d). These artefacts are significantly reduced when filtering using RGB images, but colour bleeding is another artefact that remains due to filtering the 3 channels independently (Figure 7.6e). If one filters all channels together (Figure 7.6f), then some bleeding still occurs. Instead, filtering using an HCL image achieves satisfactory results (Figure 7.6g), which are similar to those obtained from filtering using the proposed CA image (Figure 7.6h).

Table 7.1: SSIM comparison for the four scenes shown in Figure 7.5 (1 corresponds to a perfect matching).

	Venus	Cones	Art	Barn
SSIM for Grayscale	0.974	0.835	0.837	0.948
SSIM for CA model	0.989	0.888	0.873	0.974



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

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

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

(m) (n) (o) (p)

(q) (r) (s) (t)

Figure 7.5: Comparison between PWAS filtering considering grayscale and CA images. 1^{st} row: RGB images. 2^{nd} row: Grayscale images. 3^{th} row: Downsampled input depth maps. 4^{th} row.: Enhanced depth maps using grayscale images ($\sigma_\mathsf{s} = 10, \sigma_\mathsf{d} = 0.02$). 5^{th} row: Enhanced depth maps using CA images ($\sigma_\mathsf{s} = 10, \sigma_\mathsf{d} = 0.1$). 1^{st} col.: Venus scene. 2^{nd} col.: Cones scene. 3^{rd} cool.: Art scene. 4^{th} col.: Barn scene.

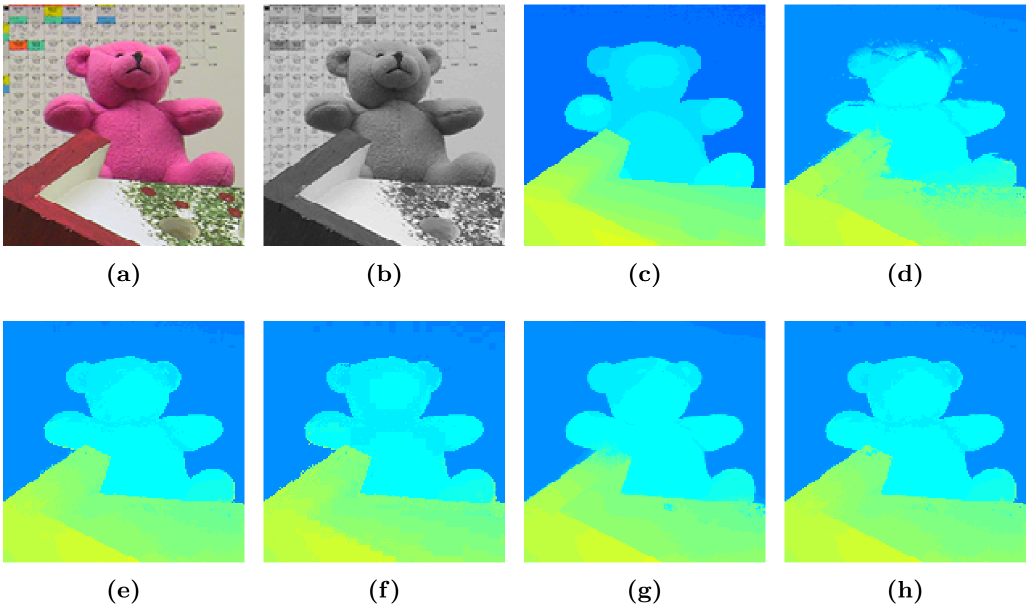


Figure 7.6: Detail of a region from the Teddy scene. (a) RGB image. (b) Grayscale image. (c) Ground truth depth map. (d) PWAS output using the grayscale image (b). (e) PWAS output using "per-channel RGB image" (a). (f) PWAS output using the RGB image (a). (g) PWAS output using the HCL image. (h) PWAS output using the CA image.

Chapter 8

Depth map enhancement over time

This chapter proposes an extension of the filtering techniques proposed in Chapter 5 in order to increase the frame rate of the hybrid ToF multi-camera rig, *i.e.,* to increase its resolution in time. ToF cameras are known by their capability to provide depth information at a high frame rate. However, this frame rate is usually lower than the frame rate of standard 2-D video cameras, which is even more prominent in industrialized ToF cameras. As a result, computer vision applications such as the identification of a moving object (or multiple objects) over time, become intricate or even impossible. In Chapter 5 we enhance the spatial resolution of ToF cameras by combining the ToF data with the 2-D data given by a coupled 2-D camera into a hybrid ToF multi-camera rig. In this chapter, we want to take advantage of the same setup in order to enhance the depth information over time. To that end, we propose to estimate the motion between each pair of 2-D camera frames and use it to compensate the motion in the low-resolution depth maps. As a result, we predict new low-resolution depth maps corresponding in time to the considered 2-D frames. The final enhanced depth video results from the fusion between the predicted depth maps and their corresponding 2-D frames by using one of the proposed filters in Chapter 5. In the following, we briefly describe how to estimate the motion between consecutive 2-D frames. Then, we present our concept to generate enhanced depth maps at the highest available frame rate of the hybrid ToF multi-camera rig, *i.e.,* the 2-D camera frame rate.

8.1 Background, related work, and problem statement

Motion estimation is still a key problem in computer vision that involves the relationship of correspondences between video frames along time. To cope with this problem, a wide number of strategies can be found in the literature, starting from the first approaches proposed by Horn and Schunck [HS81] as well as Lucas and Kanade [LK81], to more recent concepts overcoming drawbacks of previous approaches, such as robust statistics [BA91, BA96], coarse-to-fine strategies [Ana89, MP98], non-linearised models [AWS00, NE86], or spatio-temporal approaches [Nag90, BA91, WS01], among others. Within this thesis we do not propose a new solution to the problem of motion estimation since current strategies based on optical flow [BB95] can be used for our purpose with promising results. The reason we consider motion estimation techniques based on optical flow is because we need to estimate a dense motion field between a pair of two consecutive 2-D frames. Therefore, the motion field obtained from motion estimation techniques based on feature tracking [ST94] that consider the trajectory of salient image points (features) over frame series is not sufficient.

The literature in depth enhancement over time is not yet extensive. Choi et al. [CMHS09] proposed to use a slightly modified NAFDU filter [CBTT08] to tackle the spatial resolution problem for a given low-resolution depth map. Regarding the temporal resolution problem, they proposed to interpolate depth maps according to their corresponding 2-D frames, as they assume the frame rate of 2-D cameras to be higher than that of the ToF camera. To that end, they used the motion given by a Full-search Block Matching Algorithm (FBMA) between the previous and the next 2-D frames. The final enhanced depth video is the result of filtering the interpolated depth maps and their corresponding 2-D frames. The same authors proposed in [CMS10, CMKS10] to reduce the temporal fluctuation problem by filtering where they start by simultaneously filtering several depth and 2-D image pairs in order to preserve depth consistency within static regions in the scene. Kim et al. [KCKA10] proposed to enhance the spatial resolution of a given depth map by minimizing the unmatched boundary problem between depth and 2-D image pairs using joint bilateral upsampling (JBU) [KCLU07], in addition to a boundary refinement to reduce the edge blurring artefact by using linear interpolation with a color segment set. In addition, they minimized temporal depth flickering artefacts on stationary objects, *i.e.,* they preserved depth consistency using the motion between two consecutive frames.

We proposed in [GAMO12] to extend our previous work on spatial domain Time-of-Flight (ToF) data enhancement to the temporal domain. Similarly to the aforementioned approaches, we also assume the frame rate of the 2-D camera to be higher than that of the ToF camera. Our aim is to predict the missing low-resolution depth maps by using the motion between consecutive 2-D frames. The resulting enhanced depth video will result from filtering such predicted depth maps and their corresponding 2-D frames using the UML filter (see Section 5.2).

For our purpose and to estimate the motion between video frames, we have considered the work proposed by Brox et al. [BBPW04] which combines several of the aforementioned motion estimation approaches with a consistent numerical approximation yielding to an excellent performance. In [BBPW04], Brox et al. propose a high accuracy optical flow estimation based on an energy formulation. For two given consecutive frames $\mathbf{I}_i$ and $\mathbf{I}_{i-1}$, taken at times i and $i-1$, respectively, the grey value at a pixel position $\mathbf{p}_i = (u, v)^{\mathrm{T}}$ is assumed invariant to displacement, *i.e.,*

$$\mathbf{I}_i(\mathbf{p}_i) = \mathbf{I}_{i-1}(\mathbf{p}_i - \mathbf{w}_i), \tag{8.1}$$

with $\mathbf{w}_i = (\tilde{u}, \tilde{v})^{\mathrm{T}}$ being the investigated displacement vector of the pixel $\mathbf{p}_i$ between the frames $\mathbf{I}_i$ and $\mathbf{I}_{i-1}$. In order to overcome the high sensitivity to slight changes in brightness from this first assumption, the gradient of a grey value image is considered to be invariant to displacement, *i.e.,*

$$\nabla\mathbf{I}_i(\mathbf{p}_i) = \nabla\mathbf{I}_{i-1}(\mathbf{p}_i - \mathbf{w}_i), \tag{8.2}$$

where $\nabla = (\partial_u, \partial_v)^{\mathrm{T}}$ denotes the spatial gradient. The global deviations are minimized due to the grey and gradient constancy assumptions, and are measured by the following energy function

$$E_{data}(\mathbf{w}_i) = \int_\Omega \Gamma\big(|\mathbf{I}_{i-1}(\mathbf{p}_i - \mathbf{w}_i) - \mathbf{I}_i(\mathbf{p}_i)|^2 + \gamma|\nabla\mathbf{I}_{i-1}(\mathbf{p}_i - \mathbf{w}_i) - \nabla\mathbf{I}_i(\mathbf{p}_i)|^2\big)\mathbf{dp}_i, \tag{8.3}$$

with $\Omega \subset \mathbb{R}^2$ being the image space and γ being a weight between both assumptions. $\Gamma(\mathbf{s}^2) = \sqrt{\mathbf{s}^2 + \epsilon^2}$, $\epsilon = 0.001$ is the robust norm to reduce the influence of outliers. However, these two assumptions operate locally without considering neighbouring pixels.

Therefore, the smoothness of the flow field is introduced as

$$E_{smooth}(\mathbf{w}_i) = \int_{\Omega} \Gamma\big(|\nabla \tilde{u}|^2 + |\nabla \tilde{v}|^2\big)\mathbf{dp}_i. \tag{8.4}$$

The total energy is the weighted sum between (8.3) and (8.4),

$$E(\mathbf{w}_i) = E_{data}(\mathbf{w}_i) + \alpha E_{smooth}(\mathbf{w}_i), \tag{8.5}$$

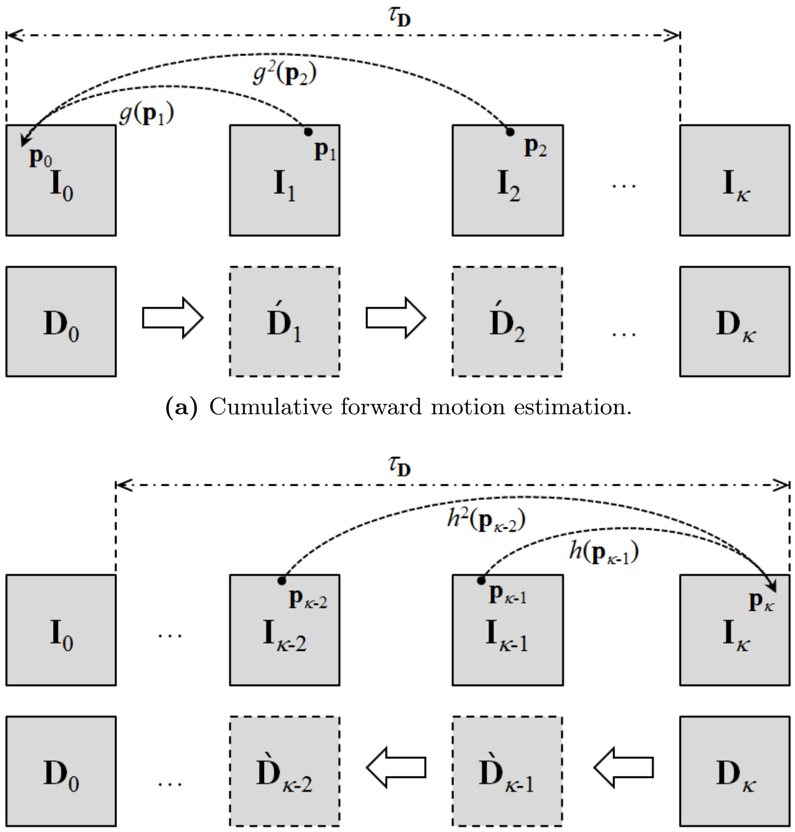
with $\alpha > 0$ being a regularisation parameter. Finally, in the case of large pixel displacements between video frames, multi-scale ideas were considered; starting from a coarse, smoothed version of the problem and finishing with a multi-resolution strategy. In [BBPW04], Brox et al. proposed the estimation of a high accuracy optical flow by minimizing the non-linear energy function defined in (8.5). The resulting motion vector $\mathbf{w}_i$ accomplishes that $\mathbf{I}_i(\mathbf{p}_i) = \mathbf{I}_{i-1}(\mathbf{p}_{i-1})$ with

$$\mathbf{p}_{i-1} = \mathbf{p}_i - \mathbf{w}_i = g(\mathbf{p}_i), \tag{8.6}$$

and assuming a translational motion. We note that the function $g(\cdot)$ gives the flow between any pair of two consecutive frames. Then, no subscript is needed as it only depends on its argument. We also note that the subscript i in a pixel position $\mathbf{p}_i$ or motion vector $\mathbf{w}_i$ is to relate the frame $\mathbf{I}_i$ and not their pixel position within the image. In [ST06], Sand et al. combined the minimization of Brox et al. with the regularization of the estimated flow proposed by Xiao et al. in [XCS$^+$06]. In what follows, we have considered Sand et al's motion estimation algorithm and used its Matlab implementation provided by Chari [Vis11]. We note that the better the motion estimation is, the more accurate will be the enhanced depth map. In the following, we present how to compensate the estimated dense optical flow in the given low-resolution depth maps in order to generate enhanced depth maps at the 2-D camera frame rate.

8.2 Proposed motion cumulation

We now investigate the problem of depth resolution enhancement over time. That is, we are in the case of a sequence of 2-D frames $\mathbf{I}_i$ taken at a frame rate $1/\tau_{\mathbf{I}}$, where the subscript $i \in \mathbb{N}$, indicates the i^{th} frame taken at time $(i \times \tau_{\mathbf{I}})$. We consider the corresponding sequence of ToF frames $\mathbf{D}_{n\kappa}$, $n \in \mathbb{N}$, taken at a frame rate of $1/\tau_{\mathbf{D}}$, such that the period $\tau_{\mathbf{D}}$ is multiple of $\tau_{\mathbf{I}}$, *i.e.,* $\tau_{\mathbf{D}} = \kappa \cdot \tau_{\mathbf{I}}$. Indeed, during a time period $\tau_{\mathbf{D}}$, the 2-D camera provides κ frames while the ToF camera provides a single one. We refer to the depth maps $\mathbf{D}_{n\kappa}$ as ToF keyframes and to their corresponding frame-synchronised 2-D images $\mathbf{I}_{n\kappa}$ as 2-D keyframes. We recall that our objective is to increase the hybrid ToF multi-camera rig resolution over time. To that end, we first estimate the motion vectors $\mathbf{w}_{n\kappa+i}$ between every consecutive 2-D frames $\mathbf{I}_{n\kappa+i}$ and $\mathbf{I}_{n\kappa+i+1}$, $0 \leq i < \kappa$ using the optical flow based approach presented in Section 8.1. Then, we use the estimated motion vectors to predict the missing ToF frames between every consecutive ToF keyframes $\mathbf{D}_{n\kappa}$ and $\mathbf{D}_{(n+1)\kappa}$. For the sake of simplicity, we formulate our concept for the first period $\tau_{\mathbf{D}}$, *i.e.*, $n = 0$.



(a) Cumulative forward motion estimation.

(b) Cumulative backward motion estimation.

Figure 8.1: Proposed cumulative motion estimation techniques.

In (8.6), we have introduced the function $g(\cdot)$ that relates the pixel positions between two consecutive frames. However, in general we want to relate pixel positions between non-consecutive frames. Indeed, we want to relate the pixel position of $\mathbf{p}_i$ on the current image frame $\mathbf{I}_i$, $0 < i < \kappa$ with its corresponding pixel position on the keyframe $\mathbf{I}_0$. We therefore propose a *cumulative forward motion estimation* approach and define it as the cumulation of the estimated motion between each pair of 2-D frames starting from the current 2-D frame $\mathbf{I}_i$ until the 2-D keyframe $\mathbf{I}_0$, as illustrated in Figure 8.1a. In Appendix C.1, we show and prove by induction that

$$\mathbf{p}_0 = g^i(\mathbf{p}_i), \text{ where } g^i = \underbrace{g \circ ... \circ g}_{i \text{ times}}, \tag{8.7}$$

where $\circ$ is the combination of functions, and $i \in \mathbb{N}^*$ being the number of frames between the current frame $\mathbf{I}_i$ and the keyframe $\mathbf{I}_0$. The predicted depth map $\acute{\mathbf{D}}_i$, where '$\acute{}$' denotes forward-predicted frame, results from using the estimated cumulative forward motion between the current frame $\mathbf{I}_i$ and the keyframe $\mathbf{I}_0$, on the ToF keyframe $\mathbf{D}_0$ as follows

$$\acute{\mathbf{D}}_i(\mathbf{p}_i) = \mathbf{D}_0\big(g^i(\mathbf{p}_i)\big), \tag{8.8}$$

for all pixel positions $\mathbf{p}_i$. The final enhanced depth video results from the fusion between the predicted depth frames $\acute{\mathbf{D}}_i$ and their corresponding 2-D frames $\mathbf{I}_i$ by using one of the filtering techniques proposed in Chapter 5. We thus end up with a depth map $\acute{\mathbf{J}}_i$, enhanced both in time and space.

Nevertheless, we realize that the edge blurring artefact (see Section 4.2.1) appears within the enhanced depth maps $\acute{\mathbf{J}}_i$ that are closer in time to their next ToF keyframe $\mathbf{D}_\kappa$ than to their precedent ToF keyframe $\mathbf{D}_0$, from which they have been predicted (compare Figure 8.2n and Figure 8.2f). The reason is due to the large displacement in both time and space between the frame $\mathbf{I}_i$ and its preceding keyframe $\mathbf{I}_0$. We therefore propose a *cumulative backward motion estimation* in which in contrast to the cumulative forward motion estimation approach, the predicted depth maps result from the next ToF keyframe $\mathbf{D}_\kappa$, as illustrated in Figure 8.1b. Thus, in this case, the estimated motion vector $\mathbf{w}_i$ accomplishes that $\mathbf{I}_i(\mathbf{p}_i) = \mathbf{I}_{i+1}(\mathbf{p}_{i+1})$ with

$$\mathbf{p}_{i+1} = \mathbf{p}_i + \mathbf{w}_i = h(\mathbf{p}_i). \tag{8.9}$$

In Appendix C.2 we also prove by induction that $\mathbf{p}_\kappa = h^i(\mathbf{p}_{\kappa-i})$. Thus, the predicted depth map $\grave{\mathbf{D}}_i$, where '$\grave{}$' denotes backward-predicted frame, follows as

$$\grave{\mathbf{D}}_i(\mathbf{p}_i) = \mathbf{D}_\kappa\big(h^i(\mathbf{p}_{\kappa-i})\big). \tag{8.10}$$

Enhanced depth maps that result from considering forward-predicted depth maps $\acute{\mathbf{D}}_i$ are more accurate the closer they are to the precedent ToF keyframe $\mathbf{D}_0$. Instead, enhanced depth maps that result from considering backward-predicted depth maps $\grave{\mathbf{D}}_i$ are more accurate the closer they are to the next ToF keyframe $\mathbf{D}_\kappa$ (compare Figure 8.2o with Figure 8.2g). We therefore propose to linearly combine the forward-predicted and backward-predicted low-resolution depth frames, and define a *bidirectional motion estimation*, as follows

$$\hat{\mathbf{D}}_i = \frac{\kappa - i}{\kappa} \cdot \acute{\mathbf{D}}_i + \frac{i}{\kappa} \cdot \grave{\mathbf{D}}_i, \tag{8.11}$$

where ‘ˆ’ denotes bidirectionally predicted frame. Enhanced depth maps that result from considering bidirectional motion estimation are expected to present a major advantage of reducing the noise within depth measurements between consecutive ToF frames $\mathbf{D}_0$ and $\mathbf{D}_\kappa$ [CMKS10, FZY10, KCKA10]; hence, preserving depth consistency and reducing the temporal fluctuation problem. In addition, enhanced depth maps from such a combination are more accurate and less noisy than when considering depth maps resulting from a single directional motion estimation. It is, however, important to note that both backward and bidirectional approaches require the next ToF keyframe and thus impose a higher latency. Next, we quantify the final enhanced depth maps when considering forward motion estimation, backward motion estimation, and bidirectional motion estimation.

8.3 Experimental results

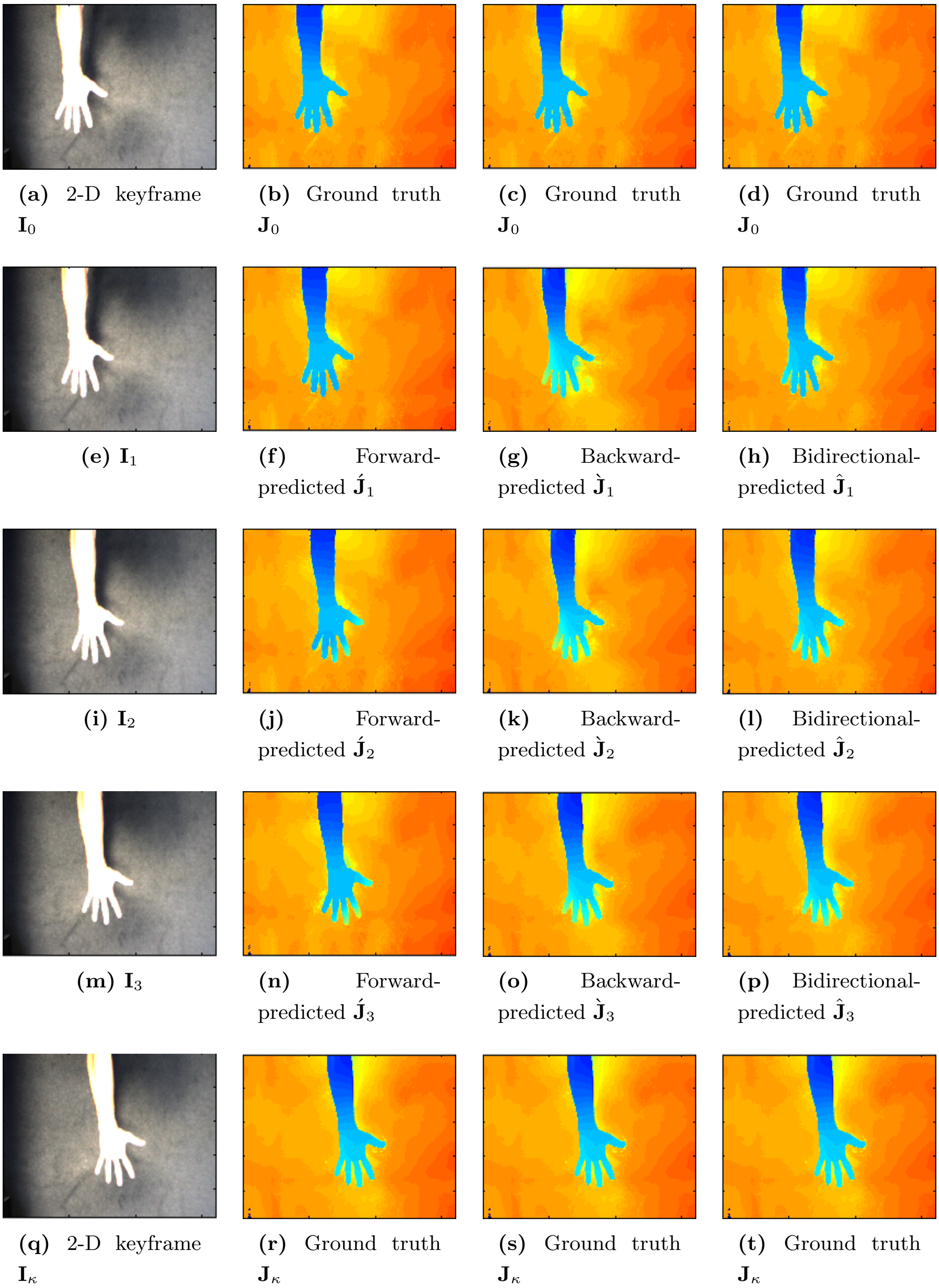
In the following, we present some experimental results computed on a real sequence of a hand moving through the scene. The sequence has been recorded using the second hybrid ToF multi-camera rig presented in Section 2.3 and has the same frame rate of the ToF camera. In order to quantify our concept for depth video enhancement, we assume the frame rate of the 2-D camera to be four times higher than the frame rate of the ToF camera, *i.e.,* $\kappa = 4$. That is, three low-resolution depth maps are replaced every four 2-D frames by the predicted low-resolution depth maps. In order to quantify the performance of our proposed method we compute the peak signal-to-noise ratio (PSNR) as well as the structural similarity (SSIM) index between the enhanced depth maps resulting from filtering using the predicted depth maps and the enhanced depth maps resulting from filtering using the neglected depth maps, *i.e.,* the ground truth.

Figure 8.2 shows an experiment where enhanced depth maps using the UML filter have been predicted from forward motion estimation (1^{st} column), backward motion estimation (2^{nd} column), and bidirectional motion estimation (3^{rd} column). From now on and for the sake of simplicity, we refer to the output of the UML filter given by (5.2) as $\mathbf{J}$ instead of $\mathbf{J_7}$. It can be observed that forward-predicted depth maps are visually better the closer they are to $\mathbf{J}_0$, the enhanced depth map using the precedent ToF keyframe $\mathbf{D}_0$. Instead, the backward-predicted depth maps are better the closer they are to $\mathbf{J}_\kappa$, the enhanced depth map that results form the next ToF keyframe $\mathbf{D}_\kappa$. Thus, the combination of both strategies gives better results as reported in Table 8.1. Indeed, Table 8.1 quantifies the predicted enhanced depth maps with their corresponding ground truth, *i.e.,* the enhanced depth maps that result from filtering the pair of $\mathbf{I}_{n\kappa}$ and $\mathbf{D}_{n\kappa}$ given by the camera rig. From the table, we can observe that the predicted depth enhancement frames from bidirectional motion are more similar to the ground truth than considering either forward or backward motion.

Table 8.1: Quantification of forward-predicted, backward-predicted and bidirectional-predicted enhanced depth maps.

	Forward		Backward		Bidirectional	
Frame	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
$\hat{\mathbf{J}}_1$	53.72	0.98	46.69	0.97	54.97	0.99
$\hat{\mathbf{J}}_2$	53.41	0.98	48.88	0.97	54.26	0.98
$\hat{\mathbf{J}}_3$	49.17	0.98	51.50	0.98	52.93	0.99

We next quantify the robustness to the noise within distance measurements discussed in Section 1.2.2.1. Thus, we add Gaussian noise with a standard deviation linearly dependent on the distance measurement. In Figure 8.3 and Figure 8.4 we present the PSNR and the SSIM index responses, respectively, of the proposed depth maps prediction strategies obtained over 50 Monte Carlo runs. The graphs confirm that the forward strategy performs better when predicting frames closer to the precedent keyframe while the backward strategy performs better the closer the predicted frame is to the next keyframe. In contrast, the bidirectional strategy outperforms any predicted frame. However, the runtime is doubled as both forward and backward motion estimation have to be computed and combined.



(a) 2-D keyframe $\mathbf{I}_0$

(b) Ground truth $\mathbf{J}_0$

(c) Ground truth $\mathbf{J}_0$

(d) Ground truth $\mathbf{J}_0$

(e) $\mathbf{I}_1$

(f) Forward-predicted $\acute{\mathbf{J}}_1$

(g) Backward-predicted $\grave{\mathbf{J}}_1$

(h) Bidirectional-predicted $\hat{\mathbf{J}}_1$

(i) $\mathbf{I}_2$

(j) Forward-predicted $\acute{\mathbf{J}}_2$

(k) Backward-predicted $\grave{\mathbf{J}}_2$

(l) Bidirectional-predicted $\hat{\mathbf{J}}_2$

(m) $\mathbf{I}_3$

(n) Forward-predicted $\acute{\mathbf{J}}_3$

(o) Backward-predicted $\grave{\mathbf{J}}_3$

(p) Bidirectional-predicted $\hat{\mathbf{J}}_3$

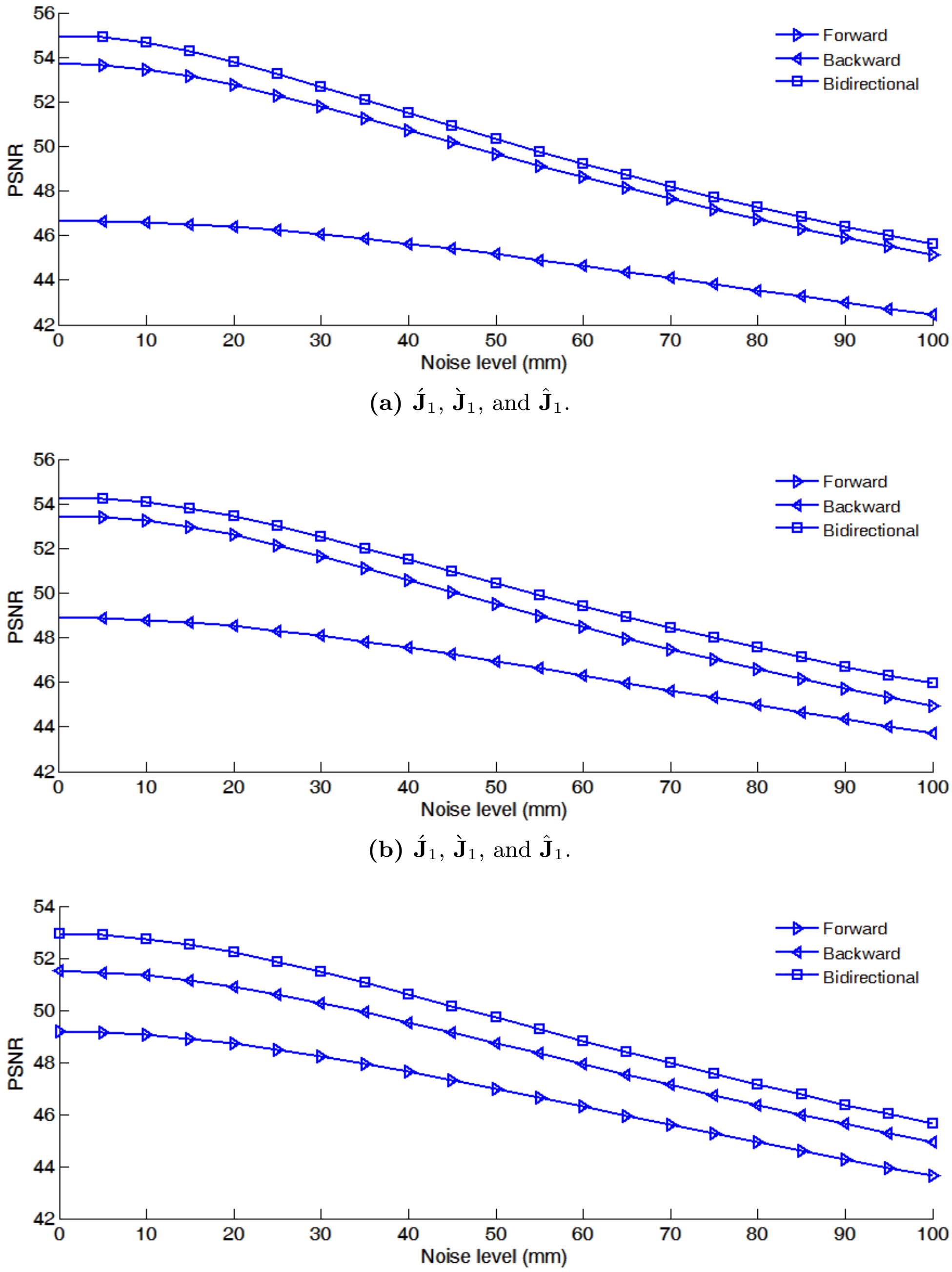
(q) 2-D keyframe $\mathbf{I}_\kappa$

(r) Ground truth $\mathbf{J}_\kappa$

(s) Ground truth $\mathbf{J}_\kappa$

(t) Ground truth $\mathbf{J}_\kappa$

Figure 8.2: Predicted enhanced depth maps. 1^{st} column: 2-D frames. 2^{nd} column: Forward-predicted enhanced depth maps $\acute{\mathbf{J}}$. 3^{rd} column: Backward-predicted enhanced depth maps $\grave{\mathbf{J}}$. 4^{th} column: Bidirectional-predicted enhanced depth maps $\hat{\mathbf{J}}$.

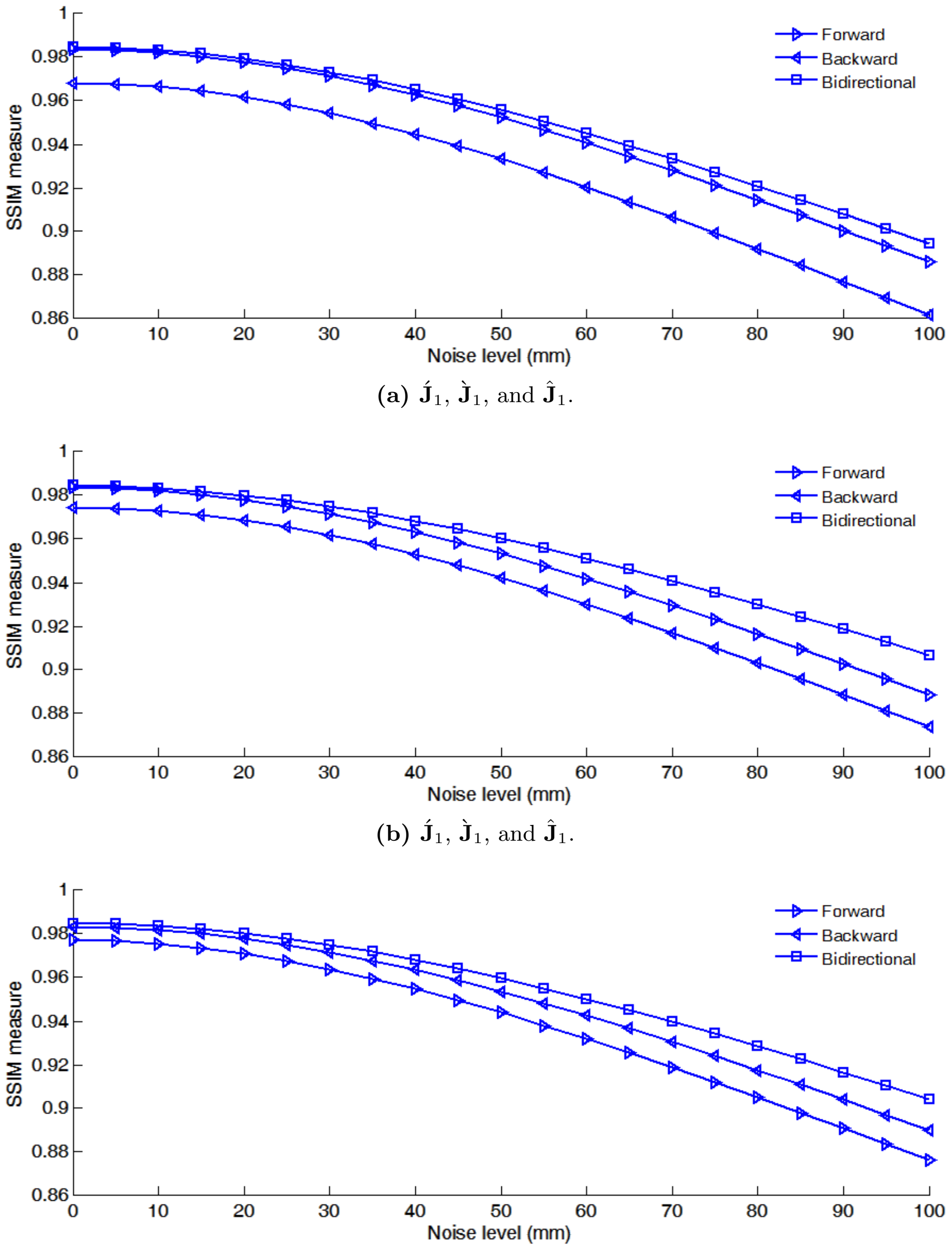


(a) $\acute{\mathbf{J}}_1$, $\grave{\mathbf{J}}_1$, and $\hat{\mathbf{J}}_1$.

(b) $\acute{\mathbf{J}}_1$, $\grave{\mathbf{J}}_1$, and $\hat{\mathbf{J}}_1$.

(c) $\acute{\mathbf{J}}_1$, $\grave{\mathbf{J}}_1$, and $\hat{\mathbf{J}}_1$.

Figure 8.3: PSNR responses against Gaussian noise using cumulative forward (▷), backward (◁), and bidirectional (□) motion estimation approaches.



(a) $\acute{\mathbf{J}}_1$, $\grave{\mathbf{J}}_1$, and $\hat{\mathbf{J}}_1$.

(b) $\acute{\mathbf{J}}_1$, $\grave{\mathbf{J}}_1$, and $\hat{\mathbf{J}}_1$.

(c) $\acute{\mathbf{J}}_1$, $\grave{\mathbf{J}}_1$, and $\hat{\mathbf{J}}_1$.

Figure 8.4: SSIM index responses against Gaussian noise using cumulative forward (▷), backward (◁), and bidirectional (□) motion estimation approaches.

Chapter 9

Conclusions

This thesis deals with the enhancement of depth data by means of low-level 3-D and 2-D data fusion. ToF cameras are cost-efficient and compact devices capable to provide depth maps with distance information of the observed scene in a single shot. In addition, due to the nature of their working principle, ToF cameras are able to provide distance information regardless of illumination conditions and independently of the texture in the scene, which makes them very attractive for many computer vision and robotic applications. However, the resolution of the given depth maps is still far below the resolution given by alternative 3-D sensing systems with an additional disadvantage of being highly influenced by noise. Thus, we chose to combine an industrialized ToF camera with a standard 2-D video camera in a hybrid ToF multi-camera rig in order to enhance the ToF data and overcome the limitations of ToF cameras that currently restrict their use in real applications such as those for safety and security. We presented a low-level data fusion approach that combines the data given by each of the cameras in the multi-camera rig and provides enhanced depth maps at the highest available frame rate, *i.e.,* the frame rate of the 2-D camera. The enhanced depth maps have the same resolution as the images given by the 2-D camera and the distance measurements are more accurate, *i.e.,* the global noise level has been significantly reduced. As a result, the work presented in this thesis facilitates computer vision processes to recognise, segment or classify an object within the provided enhanced depth maps. In other words, our work allows the use of such a hybrid ToF multi-camera rig for computer vision or robotic applications where the use of industrialized ToF cameras was restricted due to their limitations. We note that our concepts for ToF data enhancement consider industrial requirements for real-world applications, specifically, robustness to noise, accuracy, and reduced memory and time consumptions. In addition, the concepts for depth enhancement presented in this thesis are applicable to alternative 3-D sensing systems such stereo vision or structured light systems, or laser scanners in combination with a 2-D video camera. Thus, our methods apply either to recently emerging laser scanners such as the ibeo LUX [Ibe11] or the Eco Scan FX8 [Nip11] or, to new gaming devices such as Microsoft's Kinect camera.

In the first part of this thesis we tackled the calibration of the hybrid ToF multi-camera rig and the alignment of the recorded data to be further fused. We proposed a practical calibration approach to estimate the intrinsic camera parameters of each of the cameras that constitute the multi-camera rig as well as their relative extrinsic ones. In addition to determining these parameters as accurate as the commonly used Bouguet's calibration toolbox [Bou09], our calibration approach can be automated for a mass calibration process as only one image acquisition with a known position and orientation is required. With the system parameters accurately determined, we proceeded with a dedicated mapping procedure intended for data matching. This mapping is based on projecting the image coordinates from each camera reference frame to a unified reference frame where the projected data is pixel aligned. To that end, we had to deal with the binocular disparity which is due to the baseline between the cameras in the camera rig. We overcame the disparity problem by using the depth information acquired by the ToF camera. However, since the disparity is distance dependent, the mapping process had to be recomputed for each acquired depth map, making the real-time mapping intricate. We addressed this challenge by accounting for disparity variations in the mapping model. To that end, we precomputed a set of look-up tables for an array of disparities. As a result, real-time is feasible by an iterative algorithm that selects pixel by pixel the look-up table that corresponds to the distance measurement of the pixel to be mapped. By using the optimal implementation discussed in Section 3.4 in which only one look-up table must be precomputed, the alignment of a given depth map with its corresponding 2-D guidance image performs in only 2 milliseconds using our experimental setup. Therefore, the proposed method is feasible for real-time applications under industrial constraints. In addition, we showed that this proposed mapping is suitable for all kinds of ToF cameras even with large fields of view and low resolutions. The final experimental results of this part showed an accurate pixel alignment that assists further low-level data fusion techniques.

The second part of this thesis presented a new multi-lateral filter for low-level data fusion in real-time, the so-called UML filter. The proposed filter overcomes many of the drawbacks of existing fusion techniques, *i.e.,* texture copying and edge blurring in the enhanced depth maps. This is achieved by adding a new term in the kernel of the filter, the so-called credibility map. The credibility map accounts for the misalignment of edges between the data to be filtered by setting a low weight to the distance measurements that belong to depth edge pixels. Even with the good accuracy of the previous mapping process, this misalignment may appear due to the difference between the 2-D and depth map resolutions. Hence, the UML filter enhances a low-resolution depth map given by a ToF camera up to the image resolution given by the coupled 2-D video camera in the hybrid ToF multi-camera rig. The generated dense depth maps presented more accurate measurements where the depth discontinuities were well defined and adjusted to the 2-D guidance image. In addition, we increased the depth accuracy in such areas that were geometrically smooth adjusting the right weights during the filtering process. Regarding the global noise level, it was significantly reduced thanks to the nature of the bilateral filter on which the UML filter is based. The experimental results of this part were conducted on our own recorded sequences as well as on different scenes from the Middelbury dataset. We showed that our filter outperforms previous fusion techniques, delivering better results even in the case where depth edges have no corresponding 2-D edge in the guidance image. In addition, we proposed a fast implementation inspired by the work of Yang et al. [YTA09] and following the recommendations of Paris et al. [PD09] that enables real-time applications. Thus, in a similar way than in [YTA09], we proposed to quantify the range of the data to be filtered. As a result of this quantization our non-linear filter became a linear filter where the convolution between the spatial and range weighting terms could be applied. Considering the work of Paris et al. in [PD09], we also proposed to downsample the data before filtering. As a result, we ensured a good memory and speed performance without introducing significant errors. Finally, the final enhanced depth map results from combining the linear range interpolation and the bi-linear spatial interpolation to a tri-linear (*i.e.,* eight point) interpolation.

The last part of this thesis proposed two extensions of the filtering techniques presented in Part II. Due to restrictions in processing time and memory constraints, many filtering techniques in computer vision applications consider grayscale images instead of the original coloured ones. Although the generalization of our filter to colour images is straightforward, the memory and computation time demands that result from the processing of the 3-colour channels of a colour image, *e.g.,* red, green, and blue channels in the case of an RGB image, prevent from a real-time implementation. However, we noticed that when filtering using colour images, the edge blurring artefact is significantly reduced. The reason is that different colours are not collapsed to the same intensity value suppressing 2-D edges between objects in the scene. Hence, we have proposed a new 1-D colour model, *i.e.,* the cumulative angle model, that reduces the dimensionality of the 3-D HCL representation to a unique dimension while preserving original perceptual properties. We derived the cumulative angle model by sampling the HCL cone in two dimensions using spirals. By using this new colour model, the edge blurring artefact may only appear in the case where objects at different distances share exactly the same colour. The second extension tackled the enhancement of the hybrid ToF multi-camera rig resolution over time. We proposed to estimate the motion between each pair of 2-D frames and to compensate it in the low-resolution depth maps. The predicted low-resolution depth maps were then fused with their corresponding 2-D frames by using the depth enhancement techniques presented in Chapter 5. As a result, the hybrid ToF multi-camera rig is capable to provide enhanced depth video where computer vision processes can be applied to improve the robustness of real applications.

9.1 Future research directions

Due to the time constraints that cover the working plan of this project, there are a list of points related to the work that can be further investigated. In the following, we address some issues.

- **Credibility map.** Since we know that ToF cameras provide inaccurate depth measurements within depth edges, we defined the credibility map as a weight related to the gradient of the low-resolution depth maps such that a low credibility weight indicates an unreliable depth measurement whereas a high credibility map weight indicates a reliable depth measurement. However, more sophisticated ways to estimate the credibility map, such as the work proposed by Reynolds et al. [RDP+11], can be considered although the computational complexity will be increased.

- **Cumulative angle colour model.** Although we have proposed a new colour model for filtering purposes, we have realised that the two sampling rates K and K_L are important parameters that need to be further investigated in order to evaluate the extent of the colour data compression rate. On the other side, we will work towards a simple and discriminative distance for the proposed model as it is another open question important for real-time colour filtering.

- **3-D optical flow.** Our last contribution relates the enhancement of the depth information over time, *i.e.,* increasing the frame rate of the enhanced depth data delivered by the hybrid ToF multi-camera rig. To do so, we estimate the motion between each pair of 2-D frames and we use it to predict new low-resolution depth maps. The enhanced depth video results from the fusion between predicted depth maps and their corresponding 2-D frames. However, we assume that the motion in the scene is always parallel to the sensing system. As a future work, we would like to consider an extra dimensionality and thus generalise our concept to any possible motion within the scene. Therefore, an appropriate concept for 3-D optical flow has to be investigated.

- The application of our depth enhancement techniques to other 3-D sensing modalities.

  - **Stereo vision systems.** Stereo vision systems reproduce the observed scene from the triangulation of feature-correspondence pairs. As a result, there are areas in the resulting depth map without distance information. Thus, a registration process that interpolates between the estimated 3-D points is required in order to obtain a dense depth map. Instead, we propose to use our concepts and fuse the estimated 3-D points, without registration, with one of the 2-D images acquired by the camera rig. To that end, we need to investigate how to extend our concept to cope with unreliable areas instead of only edges.

- **Structured light systems.** The same idea of enhancing the 3-D information proposed for stereo vision system applies to structured light systems. In that case, the 3-D information may be combined with the 2-D image given by the 2-D camera. A straightforward application would be the fusion between the depth maps given by Microsoft's Kinect camera and the 2-D images given by its VGA camera. This solution would enhance the final depth maps with more accurate depth edges and a significant reduction of shadowing and occlusion artefacts.
- **ToF scanner.** New emerging ToF scanners, *e.g.,* the ibeo LUX [Ibe11] or the Eco Scan FX8 [Nip11] are able to generate low-resolution depth maps with more precise distance measurements than current ToF cameras. However, these depth maps are generated successively point by point which yields a known time delay between them. As a result, some motion artefacts can be observed in the given depth map. We propose to investigate the extension and adaptation of our temporal enhancement technique to reduce these motion artefacts. Then, the generated depth maps can be enhanced by using our fusion filtering techniques.

Part IV

Appendix

Appendix A

Hybrid ToF multi-camera rig devices

This appendix reports the specifications of each of the cameras considered within our hybrid ToF multi-camera rig prototypes.

A.1 3D MLI Sensor™Prototype

The 3D Modulated Light Intensity (MLI) Sensor™prototype (Figure A.1) is a compact ToF camera prototype fully manufactured by IEE S.A. [IEE11]. It is able to generate 3-D imaging without requiring additional cameras or specific processing. Due to the ToF principle in which it is based on (Section 1.2.1), lighting conditions as well as temperature do not influence to the generated depth measurements. Table A.1 presents the main hardware specifications of the 3D MLI Sensor™prototype.

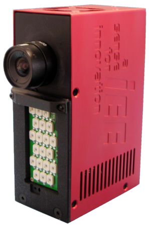


Figure A.1: 3D MLI Sensor™prototype from IEE S.A.

Table A.1: Hardware specifications of the 3D MLI Sensor™prototype.

Imager technology	Time-of-Flight (ToF)
Silicon process	CMOS with CCD
Pixel resolution	61 pixels × 56 pixels
Pixel size δ	68 μm × 49 μm
Field of view	(130° × 100°) or CS-mount lenses
Lens mount	CS-mount lenses
Frame rate	Up to 10 Hz
Illumination type	LED array
Ambient light	0 to full sunlight
Non ambiguity	7.5 m at 20 MHz modulation frequency
Distance accuracy	±2 cm at 1.5 m at 20 MHz modulation frequency
Operating temperature	−20°C to +50°C full operation, storage up to 110°C
Housing dimensions (L×W×H)	104 mm ×54 mm×144 mm
Supply voltage	90 V-220 V to 12 V, 50 Hz to 60 Hz
Digital interface	USB 2.0 full speed

A.2 Flea®2 CCD Camera

The Flea®2 CCD Camera (Figure A.2) is an ultra-compact, cost effective, and versatile 2-D video camera for demanding imaging applications in industrial machine vision. It is manufactured by Point Grey Research, Inc. [Poi11]. The main reason why we chose this 2-D camera was first because it is commonly used by the research community in computer vision and second because of its dimensions, that facilitated the attachment with the 3D MLI Sensor™prototype from IEE S.A. (Figure A.1). We selected the FL2-03S2C model, since its resolution is sufficient (about ten times higher than the IEE's ToF camera) to evaluate our depth enhancement approaches. Table A.2 presents the main hardware specifications of the Flea®2 camera.

Figure A.2: Flea®2 CCD Camera from Point Grey Research, Inc.

Table A.2: Hardware specifications of the Flea®2 CCD Camera.

Specification	FL2-03S2C
Image sensor model	Sony progressive scan interline transfer CCD's with square pixels and global shutter, color
Maximum resolution	648 pixels × 488 pixels
Pixel size δ	7.4 μm × 7.4μm
Lens mount	C-mount lenses
Maximum frame rate	80 frames per second (fps)
Operating temperature	0°C to +45°C
Housing dimensions (L×W×H)	29 mm ×29 mm×30 mm
Digital interface	Bilingual 9-pin IEEE-1394b for camera control, video data transmission, and power

A.3 3D MLI Sensor™

In the same way as its prototype (Section A.1), the 3D Modulated Light Intensity (MLI) Sensor™(Figure A.3) is a sensing system that collects real-time distance images of objects by means of infrared reflection. The main differences between the prototype and the serialized cameras are the housing dimensions, digital interface and other features such as water proof or web interface. Table A.3 presents the main hardware specifications of the 3D MLI Sensor™.

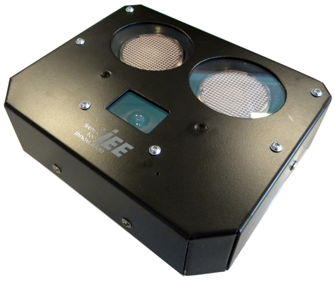


Figure A.3: 3D MLI Sensor™prototype from IEE S.A.

Table A.3: Hardware specifications of the 3D MLI Sensor™.

Imager technology	Time-of-Flight (ToF)
Silicon process	CMOS with CCD
Pixel resolution	61 pixels × 56 pixels
Pixel size δ	68 μm × 49 μm
Field of view	130° × 100°
Frame rate	Up to 10 Hz
Illumination type	LED with optimized diffuser
Ambient light	0 to full sunlight
Non ambiguity	7.5 m at 20 MHz modulation frequency
Distance accuracy	±2 cm at 1.5 m at 20 MHz modulation frequency
Operating temperature	-20°C to $+50$°C full operation, storage up to 110°C
Housing dimensions (L×W×H)	150 mm ×180 mm×108mm
Supply voltage	24 V DC ±15%
Digital interface	Ethernet

A.4 Dragonfly®2 CCD Camera

The Dragonfly®2 CCD Camera (Figure A.4) presents similar features as the Flea®2 CCD Camera (Section A.2). However, its remote head as well as the possibility of buying an OEM style board facilitated its integration within the 3D MLI Sensor™ housing. Table A.4 presents the main hardware specifications of the Dragonfly®2 camera.

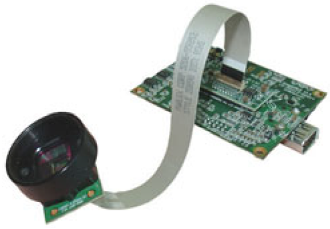


Figure A.4: Dragonfly®2 CCD Camera from Point Grey Research, Inc.

Table A.4: Hardware specifications of the Dragonfly®2 CCD Camera.

Specification	DR2-COL-XX
Image sensor model	Sony progressive scan interline transfer CCD's with square pixels and global shutter, color
Maximum resolution	648 pixels × 488 pixels
Pixel size δ	7.4 μm × 7.4μm
Lens mount	CS-mount lenses
Maximum frame rate	60 frames per second (fps)
Operating temperature	0°C to +45°C
Dimensions	64 mm ×51 mm (bare board without case or lens holder)
Digital interface	6-pin IEEE-1394 for camera control and video data transmission

Appendix B

Uncertainty of the measured distance d

In this appendix we derive the expression of the uncertainty of a measured distance σ_d, presented in (1.17). The uncertainty of a measured distance σ_d is proportional to the uncertainty of the determined phase ϕ by a factor $L/2\pi$, from (1.10) and (1.16). Then,

$$\begin{aligned}\sigma_d = \frac{L}{2\pi} \cdot \sigma_\phi = \frac{L}{2\pi} \cdot \sqrt{\sum_{k=0}^{3} \left(\frac{\partial \phi}{\partial \tilde{c}(\tau_k)} \right)^2 \cdot \sigma^2_{\tilde{c}(\tau_k),total}} \\ = \frac{L}{2\pi} \cdot \sqrt{\sum_{k=0}^{3} \left(\frac{\partial \phi}{\partial \tilde{c}(\tau_k)} \right)^2 \cdot \left(\tilde{c}(\tau_k) + \sigma_e^2 + \sigma_t^2 \right)} \qquad \text{(B.1)}\end{aligned}$$

with

$$\begin{aligned}
&\sum_{k=0}^{3}\left(\frac{\partial\phi}{\partial\tilde{c}(\tau_k)}\right)^2\cdot\big(\tilde{c}(\tau_k)+\sigma_e^2+\sigma_t^2\big)\\
&=\left(\frac{-\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)}{\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2}\right)^2\cdot\big(\tilde{c}(\tau_0)+\sigma_e^2+\sigma_t^2\big)\\
&+\left(\frac{-\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)}{\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2}\right)^2\cdot\big(\tilde{c}(\tau_1)+\sigma_e^2+\sigma_t^2\big)\\
&+\left(\frac{\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)}{\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2}\right)^2\cdot\big(\tilde{c}(\tau_2)+\sigma_e^2+\sigma_t^2\big)\\
&+\left(\frac{\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)}{\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2}\right)^2\cdot\big(\tilde{c}(\tau_3)+\sigma_e^2+\sigma_t^2\big)\\
&=\frac{\big(\tilde{c}(\tau_1)+\tilde{c}(\tau_3)+2(\sigma_e^2+\sigma_t^2)\big)\cdot\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2}{\Big(\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2\Big)^2}\\
&+\frac{\big(\tilde{c}(\tau_0)+\tilde{c}(\tau_2)+2(\sigma_e^2+\sigma_t^2)\big)\cdot\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2}{\Big(\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2\Big)^2}.
\end{aligned}\tag{B.2}$$

From (1.6),

$$\begin{aligned}
\tilde{c}(\tau_0)&=\tilde{h}+\frac{\tilde{a}}{2}cos(\omega\tau_0-\phi)=\tilde{h}+\frac{\tilde{a}}{2}cos(\phi),\quad(\omega\tau_0=0),\\
\tilde{c}(\tau_1)&=\tilde{h}+\frac{\tilde{a}}{2}cos(\omega\tau_1-\phi)=\tilde{h}+\frac{\tilde{a}}{2}sin(\phi),\quad(\omega\tau_1=\pi/2),\\
\tilde{c}(\tau_2)&=\tilde{h}+\frac{\tilde{a}}{2}cos(\omega\tau_2-\phi)=\tilde{h}-\frac{\tilde{a}}{2}cos(\phi),\quad(\omega\tau_2=\pi),\\
\tilde{c}(\tau_3)&=\tilde{h}+\frac{\tilde{a}}{2}cos(\omega\tau_3-\phi)=\tilde{h}-\frac{\tilde{a}}{2}sin(\phi),\quad(\omega\tau_3=3\pi/4).
\end{aligned}\tag{B.3}$$

with $\tau_k=k\cdot T/4$. From (B.3) we see that $\tilde{c}(\tau_0)+\tilde{c}(\tau_2)=\tilde{c}(\tau_1)+\tilde{c}(\tau_3)$, thus

$$\begin{aligned}
&\sum_{k=0}^{3}\left(\frac{\partial\phi}{\partial\tilde{c}(\tau_k)}\right)^2\cdot\big(\tilde{c}(\tau_k)+\sigma_e^2+\sigma_t^2\big)\\
&=\frac{\big(\tilde{c}(\tau_1)+\tilde{c}(\tau_3)+2(\sigma_e^2+\sigma_t^2)\big)\cdot\Big(\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2\Big)}{\Big(\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2\Big)^2}\\
&=\frac{\big(\tilde{c}(\tau_1)+\tilde{c}(\tau_3)+2(\sigma_e^2+\sigma_t^2)\big)}{\big(\tilde{c}(\tau_0)-\tilde{c}(\tau_2)\big)^2+\big(\tilde{c}(\tau_3)-\tilde{c}(\tau_1)\big)^2}.
\end{aligned}\tag{B.4}$$

From (1.14) and assuming that $\tilde{c}(\tau_0) + \tilde{c}(\tau_2) = \tilde{c}(\tau_1) + \tilde{c}(\tau_3)$,

$$\tilde{h} = \frac{\tilde{c}(\tau_0) + \tilde{c}(\tau_1) + \tilde{c}(\tau_2) + \tilde{c}(\tau_3)}{4} = \frac{\tilde{c}(\tau_0) + \tilde{c}(\tau_2)}{2}. \tag{B.5}$$

Finally, we determine σ_d by substituting (1.13), (B.4), and (B.5) in (B.1), *i.e.,*

$$\sigma_d = \frac{\sqrt{2} \cdot L}{2\pi} \cdot \frac{\sqrt{\tilde{h} + \sigma_e^2 + \sigma_t^2}}{2\tilde{a}} = \frac{L}{\sqrt{2}\pi} \cdot \frac{\sqrt{\tilde{h} + \sigma_e^2 + \sigma_t^2}}{2\tilde{a}}. \tag{B.6}$$

We note that the expression of σ_d in (B.6) differs from the one presented by Lange [LSBL00] by a constant factor called the demodulation contrast, which depends on the sensor characteristics, *i.e.,* the way the demodulation is practically implemented.

Appendix C

Proof of the proposed motion cumulation

In this appendix we show and prove by induction the proposed cumulative forward and backward motion estimation approaches.

C.1 Cumulative forward motion estimation

In (8.6), we have introduced the function $g(\mathbf{p}_{i+1})$ that relates the position of a pixel $\mathbf{p}_i$ on $\mathbf{I}_i$ with its corresponding position $\mathbf{p}_{i+1}$ on the consecutive frame $\mathbf{I}_{i+1}$, as follows

$$\mathbf{p}_i = g(\mathbf{p}_{i+1}). \tag{C.1}$$

In the following, we prove by induction that

$$\mathbf{p}_i = g^n(\mathbf{p}_{i+n}), \text{ where } g^n = \underbrace{g \circ ... \circ g}_{n \text{ times}}, \tag{C.2}$$

and $n \in \mathbb{N}^*$. From (C.1), we check that the case of $n = 1$ in (C.2) is true by definition. We then assume that (C.2) is correct and we show that

$$\mathbf{p}_i = g^{n+1}(\mathbf{p}_{i+n+1}). \tag{C.3}$$

If we replace i by $(i + n)$ in (C.1), we obtain

$$\mathbf{p}_{i+n} = g(\mathbf{p}_{i+n+1}), \tag{C.4}$$

and by replacing (C.4) in (C.2), we find

$$\mathbf{p}_i = g^n\big(g(\mathbf{p}_{i+n+1})\big) = (g^n \circ g)(\mathbf{p}_{i+n+1}) = g^{n+1}(\mathbf{p}_{i+n+1}). \tag{C.5}$$

Hence, we have demonstrated that our assumption is correct. In the cumulative forward motion estimation approach (Section 8.2), we refer to the index of the current frame n as i, and to the frame with subscript i as the keyframe 0. Then, by replacing n by i and i by 0 in (C.2), we obtain

$$\mathbf{p}_0 = g^i(\mathbf{p}_i). \tag{C.6}$$

C.2 Cumulative backward motion estimation

In (8.9), we have introduced the function $h(\mathbf{p}_{i-1})$ that relates the position of a pixel $\mathbf{p}_i$ on $\mathbf{I}_i$ with its corresponding position $\mathbf{p}_{i-1}$ on the immediately preceding frame $\mathbf{I}_{i-1}$, as follows

$$\mathbf{p}_i = h(\mathbf{p}_{i-1}). \tag{C.7}$$

Following the same ideas as in Appendix C.1, we prove by induction that

$$\mathbf{p}_i = h^n(\mathbf{p}_{i-n}), \text{ where } h^n = \underbrace{h \circ ... \circ h}_{n \text{ times}}, \tag{C.8}$$

and $n \in \mathbb{N}^*$.

Bibliography

[Alo11] Amsterdam library of object images. http://staff.science.uva.nl/aloi, May 2011. 96

[Ana89] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310, January 1989. 102

[Ang07] J. Angulo. Morphological colour operators in totally ordered lattices based on distances: Application to image filtering, enhancement and analysis. *in Computer Vision and Image Understanding*, 107(2–3):56–73, 2007. 90

[AWS00] Luis Alvarez, Joachim Weickert, and Javier Sánchez. Reliable Estimation of Dense Optical Flow Fields with Large Displacements. *International Journal of Computer Vision*, 39:41–56, August 2000. 102

[BA91] M.J. Black and P. Anandan. Robust dynamic motion estimation over time. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 296–302, June 1991. 102

[BA96] Michael J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63:75–104, January 1996. 102

[BB95] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Comput. Surv.*, 27:433–466, September 1995. 102

[BBPW04] Thomas Brox, Andres Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In

In Proceedings of European Conference on Computer Vision (ECCV), volume 4, pages 25–36, May 2004. 103, 104

[BK08] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library.* O'Reilly Media, 1st edition, 2008. 31

[Bla04] Franois Blais. Review of 20 years of range sensor development. *Journal of Electronic Imaging*, 13(1):231–243, January 2004. 3

[Bou09] Jean-Yves Bouguet. Camera calibration toolbox for matlab. http://vision.caltech.edu/bouguetj/calib, November 2009. 31, 32, 37, 38, 39, 114

[BW99] Max Born and Emil Wolf. *Principles of Optics.* Cambridge University Press, 7th edition, 1999. 12

[CBTT08] Derek Chan, Hylkea Buisman, Christian Theobalt, and Sebastian Thrun. A noise-aware filter for real-time depth upsampling. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (ECCVW)*, 2008. 14, 16, 42, 61, 102

[CMHS09] Jinwook Choi, Dongbo Min, Bumsub Ham, and Kwanghoon Sohn. Spatial and temporal up-conversion technique for depth video. pages 3525–3528, November 2009. 102

[CMKS10] Jinwook Choi, Dongbo Min, Donghyun Kim, and Kwanghoon Sohn. 3D JBU based depth video filtering for temporal fluctuation reduction. In *IEEE International Conference on Image Processing (ICIP)*, pages 2777–2780, September 2010. 102, 107

[CMS10] Jinwook Choi, Dongbo Min, and Kwanghoon Sohn. 2D-plus-depth based resolution and frame-rate up-conversion technique for depth video. *IEEE Transactions on Consumer Electronics*, 56(4):2489–2497, November 2010. 102

[Cre88] K Creath. Phase measurement interferometric techniques. 11:349–353, 1988. 8

[CTPD08] Ryan Crabb, Colin Tracey, Akshaya Puranik, and James Davis. Real-time foreground segmentation via range and color imaging. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–5, 2008. 59

[DD02] Frédo Durand and Julie Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. *Proceedings of ACM Transactions on Graphics (TOG)*, 21:257–266, July 2002. 58

[Der93] R. Deriche. Recursively Implementing the Gaussian and its Derivatives. Technical report, Institut National de Recherche en Informatique et en Automatique, Unite de Recherche Inria-Sophia Antipolis, 1993. 66

[DNN$^+$11] M.N. Do, Q.H. Nguyen, H.T. Nguyen, D. Kubacki, and S.J. Patel. Immersive Visual Communication. *IEEE Transactions on Signal Processing Magazine, IEEE*, 28(1):58–66, January 2011. 46

[DT05] James Diebel and Sebastian Thrun. An application of markov random fields to range sensing. In *NIPS*, pages 291–298. MIT Press, 2005. 14, 16, 57

[DW93] R. Dutta and C.C. Weems. Parallel dense depth-from-motion on the image understanding architecture. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 154–159, June 1993. 3

[Ela02] M. Elad. On the origin of the bilateral filter and ways to improve it. *IEEE Transactions on Image Processing*, 11(10):1141–1151, 2002. 57, 58

[FAT10] S. Foix, G. Aleny, and C. Torras. Exploitation of time-of-flight (ToF) cameras. Technical Report IRI-TR-10-07, Institut de Robtica i Informtica Industrial, CSIC-UPC, 2010. 2, 6, 10

[FAT11] S. Foix, G. Alenya, and C. Torras. Lock-in Time-of-Flight (ToF) Cameras: A Survey. *IEEE Sensors Journal*, 11(9):1917–1926, September 2011. 1, 3

[FB07] D. Falie and V. Buzuloiu. Noise Characteristics of 3D Time-of-Flight Cameras. In *International Symposium on Signals, Circuits and Systems (ISSCS)*, volume 1, pages 1–4, July 2007. 2

[FBK10] A. Frick, B. Bartczack, and R. Koch. 3D-TV LDV content generation with a hybrid ToF-multicamera RIG. In *3DTV-CON*, pages 1–4, 2010. 13

[FH08] Stefan Fuchs and Gerd Hirzinger. Extrinsic and Depth Calibration of ToF-cameras. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2008. 32

[FL04] Olivier Faugeras and Quang-Tuan Luong. *The Geometry of Multiple Images*. MIT Press, 2004. 1

[FPA$^+$07] N. Floudas, A. Polychronopoulos, O. Aycard, J. Burlet, and M. Ahrholdt. High Level Sensor Data Fusion Approaches For Object Recognition In Road Environment. In *IEEE Intelligent Vehicles Symposium*, pages 136–141, June 2007. 15

[Fra11] Fraunhofer Institute of Microelectronic Circuits and Systems. http://www.fraunhofer.de, August 2011. 6

[FZY10] Deliang Fu, Yin Zhao, and Lu Yu. Temporal consistency enhancement on depth sequences. In *Picture Coding Symposium (PCS)*, pages 342–345, December 2010. 107

[GAM$^+$11a] Frederic Garcia, Djamila Aouada, Bruno Mirbach, Thomas Solignac, and Björn Ottersten. A New Multi-lateral Filter for Real-Time Depth Enhancement. In *Advanced Video and Signal-Based Surveillance (AVSS)*, 2011. 14

[GAM$^+$11b] Frederic Garcia, Djamila Aouada, Bruno Mirbach, Thomas Solignac, and Björn Ottersten. Real-time Hybrid ToF multi-camera Rig Fusion System for Depth Map Enhancement. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–8, June 2011. 14

[GAMO12] Frederic Garcia, Djamila Aouada, Bruno Mirbach, and Björn Ottersten. Spatio-Temporal ToF Data Enhancement by Fusion. In *International Conference on Image Processing (ICIP)*, 2012. 103

[GAVA08] R. Garcia, O. Aycard, Trung-Dung Vu, and M. Ahrholdt. High level sensor data fusion for automotive applications using occupancy grids. In *International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 530–535, December 2008. 15

[GAVN11] Mohit Gupta, Amit Agrawal, Ashok Veeraraghavan, and Srinivasa G. Narasimhan. Structured light 3d scanning in the presence of global illumination. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 713–720, June 2011. 5

[GBQ$^+$08] Stephen Gloud, Paul Baumstarck, Morgan Quigley, Y. Ng Andrew, and Koller Daphne. Integrating visual and range data for robotic object detection. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (ECCVW)*, 2008. 13, 14, 57

[GMO$^+$10] Frederic Garcia, Bruno Mirbach, Björn Ottersten, Frederic Grandidier, and Angel Cuesta. Pixel Weighted Average Strategy for Depth Sensor Data Fusion. In *International Conference on Image Processing (ICIP)*, pages 2805–2808, September 2010. 63

[GW02] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, 2nd edition, 2002. 90

[HK92] Martial Hebert and Eric Krotkov. 3D measurements from imaging laser radars: how good are they? *Image and Vision Computing*, 10:170–178, April 1992. 1, 6, 7

[HS81] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981. 102

[HS97] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1106–1112, June 1997. 26, 27, 32

[HZ03] Richard Hartley and Andew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2003. 1, 4, 23, 25, 27, 29, 42

[Ibe11] Ibeo atutomotive systems. http://www.ibeo-as.com, August 2011. 41, 114, 118

[IEE11] IEE S.A. http://www.iee.lu, August 2011. 30, 122

[IMN$^+$10] K.N. Iyer, K. Maiti, B. Navathe, H. Kannan, and A. Sharma. Multiview video coding using depth based 3d warping. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1108–1113, July 2010. 16

[KA00] N. Plataniotis Konstantinos and N. Venetsanopoulos Anastasios. *Color Image Processing and Applications*. Springer, 1st edition, 2000. 90

[KBKL09] Andreas Kolb, Erhardt Barth, Reinhard Koch, and Rasmus Larsen. Time-of-Flight Sensors in Computer Graphics. In *Eurographics - State of the Art Reports*, pages 119–134, 2009. 1, 10

[KCKA10] Sung-Yeol Kim, Ji-Ho Cho, A. Koschan, and M.A. Abidi. Spatial and Temporal Enhancement of Depth Images Captured by a Time-of-Flight Depth Sensor. In *International Conference on Pattern Recognition (ICPR)*, pages 2358–2361, August 2010. 60, 102, 107

[KCLU07] Johannes Kopf, Michael Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 96, New York, NY, USA, 2007. ACM. 59, 102

[KCTT08] Young Min Kim, Derek Chan, Christian Theobalt, and Sebastian Thrun. Design and calibration of a multi-view TOF sensor fusion system. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–7, 2008. 42

[KH10] Yun-Suk Kang and Yo-Sung Ho. High-quality multi-view depth generation using multiple color and depth cameras. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1405–1410, July 2010. 14

[KI07] Timo Kahlmann and Hilmar Ingensand. Increased Accuracy of 3D Range Imaging Camera by Means of Calibration. In *Optical 3-D Measurement Techniques VIII (Eds.: Grn, Kahmen)*, pages 101–108, 2007. 32

[KS06] Klaus-Dieter Kuhnert and Martin Stommel. Fusion of Stereo-Camera and PMD-Camera Data for Real-Time Suited Precise 3D Environment Reconstruction. In *International Conference on Intelligent Robots and Systems*, pages 4780–4785, 2006. 14

[KTD$^+$09] Young Min Kim, Christian Theobalt, James Diebel, Jana Kosecka, Branislav Miscusik, and Sebastian Thrun. Multi-view Image and ToF Sensor Fusion for Dense 3D Reconstruction. In *IEEE Workshop on 3-D Digital Imaging and Modeling, 3DIM*, 2009. 14, 42

[LH10a] Eun-Kyung Lee and Yo-Sung Ho. Generation of multi-view video using a fusion camera system for 3D displays. *IEEE Transactions on Consumer Electronics*, 56(4):2797–2805, November 2010. 16

[LH10b] Eun-Kyung Lee and Yo-Sung Ho. Generation of multi-view video using a fusion camera system for 3D displays. *IEEE Transactions on Consumer Electronics*, 56(4):2797–2805, November 2010. 46

[Lin10] Marvin Lindner. *Calibration and Real-Time Processing of Time-of-Flight Range Data*. PhD thesis, Department of Electrical Engineering and Computer Science. University of Siegen, 2010. 8, 26, 32

[LK81] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence*, volume 2, pages 674–679. Morgan Kaufmann Publishers Inc., 1981. 102

[LKR08] Marvin Lindner, Andreas Kolb, and Thorsten Ringbeck. New Insights into the Calibration of TOF Sensors. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–5, 2008. 31

[LMW+11] T. Leyvand, C. Meekhof, Yi-Chen Wei, Jian Sun, and Baining Guo. Kinect identity: Technology and experience. *Computer*, 44(4):94–96, April 2011. 13

[Loc67] E. H. Lockwood. *A Book of Curves.* Cambridge University Press, 1st edition, 1967. 92

[LS99] R.C. Luo and K.L. Su. A review of high-level multisensor fusion: approaches and applications. In *IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 25–31, 1999. 15

[LS01] Robert Lange and Peter Seitz. Solid-State Time-of-Flight Range Camera. *IEEE Journal of Quantum Electronics*, 37:390–397, March 2001. 1, 5, 7, 9, 11, 83

[LSBL00] R. Lange, P. Seitz, A. Biber, and St. Lauxtermann. Demodulation Pixels in CCD and CMOS Technologies for Time-of-Flight Ranging. In *In Proceedings of SPIE*, volume 3965A, pages 177–188, January 2000. 6, 11, 129

[Mid11] Middlebury stereo datasets. http://vision.middlebury.edu/stereo/data/, October 2011. 73, 78, 96

[MP98] E. Memin and P. Perez. A multigrid approach for hierarchical motion estimation. In *International Conference on Computer Vision*, pages 933–938, January 1998. 102

[Nag90] H. H. Nagel. Extending the ‘oriented smoothness constraint’ into the temporal domain and the estimation of derivatives of optical flow. In *In Proceedings of European Conference on Computer Vision (ECCV)*, pages 139–148. Springer-Verlag New York, Inc., 1990. 102

[NE86] Hans-Hellmut Nagel and Wilfried Enkelmann. An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 8(5):565–593, September 1986. 102

[NFK$^+$08] C. Niclass, C. Favi, T. Kluter, F. Monnier, and E. Charbon. Single-photon synchronous detection. In *European Solid-State Circuits Conference (ESSCIRC)*, pages 114–117, September 2008. 12

[Nip11] The Nipon Signal Co., LTD. http://www.signal.co.jp, August 2011. 41, 114, 118

[NN94] S.K. Nayar and Y. Nakagawa. Shape from focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 16(8):824–831, August 1994. 3

[NRBC05] C. Niclass, A. Rochas, P.-A. Besse, and E. Charbon. Design and characterization of a CMOS 3-D image sensor based on single photon avalanche diodes. *IEEE Journal of Solid-State Circuits*, 40(9):1847–1854, September 2005. 12

[NSS$^+$08] Koba Natroshvili, Michael Schmid, Martin Stephan, Andreas Stiegler, and Thomas Schamm. Real time pedestrian detection by fusing pmd and cmos cameras. In *Intelligent Vehicles Symposium*, pages 925–929, 2008. 15

[OLK$^+$04] Thierry Oggier, Michael Lehmann, Rolf Kaufmann, Matthias Schweizer, Michael Richter, Peter Metzler, Graham Lang, Felix Lustenberger, and Nicolas Blanc. An all-solid-state optical range camera for 3D real-time imaging with sub-centimeter depth resolution (SwissRanger). In *In Proceedings of SPIE*, volume 5249, 2004. 1, 6, 7

[PAH$^+$04] Georg Petschnigg, Maneesh Agrawala, Hugues Hoppe, Richard Szeliski, Michael Cohen, and Kentaro Toyama. Digital Photography with Flash and No-Flash Image Pairs. In *Proceedings of Siggraph, ACM Transactions on Graphics*, 2004. 59

[PD09] Sylvain Paris and Frédo Durand. A fast approximation of the bilateral filter using a signal processing approach. In *International Journal of Computer Vision*, volume 81, pages 24–52. Kluwer Academic Publishers, 2009. 58, 68, 70, 89, 97, 115

[Poi11] Point Grey Research, Inc. http://www.ptgrey.com, August 2011. 30, 123

[Por08] F. Porikli. Constant time o(1) bilateral filtering. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008. 58, 68

[RCM04] A.N. Rajagopalan, S. Chaudhuri, and Uma Mudenagudi. Depth estimation and image restoration using defocused stereo pairs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(11):1521–1525, November 2004. 3

[RDP$^+$11] Malcolm Reynolds, Jozef Dobos, Leto Peely, Tim Weyrich, and Gabriel J. Brostow. Capturing Time-of-Flight Data with Confidence. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 945–952, 2011. 117

[RZFM92] Charles A. Rothwell, Andrew Zisserman, David A. Forsyth, and Joseph L. Mundy. Canonical frames for planar object recognition. In *In Proceedings of European Conference on Computer Vision (ECCV)*, pages 757–772. Springer Berlin / Heidelberg, 1992. 34

[SAB$^+$07] E. Stoykova, A.A. Alatan, P. Benzie, N. Grammalidis, S. Malassiotis, J. Ostermann, S. Piekh, V. Sainov, C. Theobalt, T. Thevar, and X. Zabulis. 3-D Time-Varying Scene Capture Technologies: A Survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(11):1568–1586, November 2007. 1, 3

[SFPL10] Joaquim Salvi, Sergio Fernandez, Tomislav Pribanic, and Xavier Llado. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition*, 43:2666–2680, August 2010. 3, 5

[SK98] Y.Y. Schechner and N. Kiryati. Depth from defocus vs. stereo: how different really are they? In *International Conference on Pattern Recognition (ICPR)*, volume 2, pages 1784–1786, August 1998. 3

[SM05] M. Sarifuddin and R. Missaoui. A new perceptually uniform color space with associated color similarity measure for contentbased image and video retrieval. In *Proceedings of Multimedia Information Retrieval Workshop*, pages 3–7, 2005. 90, 96

[SPS+07] David Stoppa, Lucio Pancheri, Mauro Scandiuzzo, Lorenzo Gonzo, Gian-Franco Dalla Betta, and Andrea Simoni. A CMOS 3-D Imager Based on Single Photon Avalanche Diode. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 54(1):4–12, January 2007. 12

[SS02] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1–3):7–42, April–June 2002. 1, 4

[SSVH95] T. Spirig, P. Seitz, O. Vietze, and F. Heitger. The lock-in CCD-two-dimensional synchronous detection of light. *IEEE Journal of Quantum Electronics*, 31(9):1705–1708, September 1995. 2, 6, 9, 12

[ST94] Jianbo Shi and C. Tomasi. Good features to track. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, June 1994. 102

[ST97] G. Sharma and H.J. Trussell. Digital Color Imaging. *IEEE Transactions on Image Processing*, 6(7):901–932, 1997. 90

[ST06] P. Sand and S. Teller. Particle Video: Long-Range Motion Estimation using Point Trajectories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 2195–2202, 2006. 104

[STDT08] Sebastian Schuon, Christian Theobalt, James Davis, and Sebastian Thrun. High-quality scanning using time-of-flight depth superresolution. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–7, 2008. 14

[TM98] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998. 14, 57

[Tri11] TriDiCam. http://www.tridicam.net, August 2011. 6

[Tsa87] Roger Y. Tsai. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation*, RA-3:323–344, 1987. 32

[TV98] Emanuele Trucco and Alessandro Verri. *Introductory Techniques for 3-D Computer Vision.* Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998. 3, 23, 24, 25, 27, 28, 31, 42

[VD10] A. Vahdat and M. S. Drew. Colour From Grey by Optimized Colour Ordering. In *Proceedings of Color Imaging Conference: Color Science and Engineering Systems, Technologies, and Applications*, pages 17–21, November 2010. 90, 91

[Vis11] Chari Visesh. High accuracy optical flow. http://www.mathworks.com/matlabcentral/fileexchange/17500, October 2011. 104

[VT86] A. Verri and V. Torre. Absolute depth estimate in stereopsis. In *J. Optical Society of America A (JOSA A)*, volume 3, 1986. 3

[Wan08] S. Wang. Depth from Shading Based on 2D Maximum Entropy. In *International Conference on Intelligent Computation Technology and Automation (ICICTA)*, volume 2, pages 119 –121, Octobre 2008. 3

[WBSS04] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. In *IEEE TIP*, volume 13–4, April 2004. 97

[WCH92] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 14(10):965–980, Octobre 1992. 26

[WFH$^+$10] Yu Wei, F. Franchetti, J.C. Hoe, Chang Yao-Jen, and Chen Tsuhan. Fast bilateral filtering by adapting block size. In *ICIP*, pages 3281–3284, 2010. 58

[WS01] Joachim Weickert and Christoph Schnörr. Variational Optic Flow Computation with a Spatio-Temporal Smoothness Constraint. *Journal of Mathematical Imaging and Vision*, 14:245–255, May 2001. 102

[XCS$^+$06] Jiangjian Xiao, Hui Cheng, Harpreet Sawhney, Cen Rao, Michael Isnardi, and Sarnoff Corporation. Bilateral filtering-based optical flow estimation with occlusion detection. In *In Proceedings of European Conference on Computer Vision (ECCV)*, volume 1, pages 211–224, 2006. 104

[YMH06] Yun Yuan, Zhenjiang Miao, and Shaohai Hu. Real-Time Human Behavior Recognition in Intelligent Environment. In *International Conference on Signal Processing*, volume 3, pages 16–20, 2006. 2

[YTA09] Qingxiong Yang, Kar-Han Tan, and N. Ahuja. Real-time O(1) bilateral filtering. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 557–564, 2009. 58, 68, 115

[YYDN07] Qingxiong Yang, Ruigang Yang, James Davis, and David Nistér. Spatial-depth super resolution for range images. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007. 14, 16, 57

[ZBSS04] Wang Zhou, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004. 83

[Zha00] Zhengyou Zhang. A flexible new technique for camera calibration. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 22, pages 1330–1334, Novembre 2000. 32

[ZWY$^+$10] J. Zhu, L. Wang, R. Yang, J. Davis, and Z. Pan. Reliability Fusion of Time-of-Flight Depth and Stereo for High Quality Depth Maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (99):1, 2010. 13, 16