Asynchronous event-based 3d vision

Joao Paulo Amaro da Costa Luz Carneiro

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PIERRE ET MARIE CURIE

Spécialité : Robotique

École doctorale : « Sciences mécaniques, acoustique, électronique & robotique de Paris »

réalisée à l'

Institut de la Vision

présentée par

João CARNEIRO

pour obtenir le grade de :

DOCTEUR DE L'UNIVERSITÉ PIERRE ET MARIE CURIE

Sujet de la thèse :

Asynchronous Event-Based 3D Vision

soutenue le 10 February 2014

devant le jury composé de :

M.	Giacomo INDIVERI	Prof UZH	Rapporteur
M.	Patrick RIVES	DR INRIA	Rapporteur
M.	Yves TROTTER	DR CNRS	Examinateur
M.	Stéphane REGNIER	Prof UPMC	Examinateur
M.	Sio-Hoï IENG	MC UPMC	Examinateur
M.	Ryad BENOSMAN	HDR UPMC	Directeur de thèse

Abstract

Reproducing biological vision in a machine is a challenging problem for which scientists have just scratched the surface. Living organisms are able to perform complex tasks in an awestruckly efficient manner. The stereovision is one of these complex mechanisms that computer scientists try to replicate with high resolution cameras. This thesis takes on the stereovision problem in a neuromorphic way by mean of a new generation of vision sensors also called "silicon retinas". These silicon retinas mimic biological retinas by capturing the visual information into the form of asynchronous stream of events that encode contrast change at high temporal precision.

These sensors are used to study the importance of the precise timing and the scene temporal dynamics in solving the stereo correspondence problem. We propose one of the first 3D reconstruction methods which is able to produce 3D models in a truly event-based and asynchronous manner, from event-based visual information. Besides the novelty of proposing a truly temporal- based asynchronous event-driven approach of 3D reconstructions, this work is also able to preserve the native dynamic of the scene.

Time as information medium is proven to have a critical role in stereovision. Time can supplement, compensate and even replace the usual luminance and spatial information. This work lays strong foundations for future research on high temporal and event-based dynamic stereo vision. It also opens new promising perspectives for solving traditional machine vision problems thanks to the use of the new asynchronous vision paradigm.

Abstract

L'implementation de la vision biologique sur machine est un problème majeur que la recherche actuelle a à peine effleuré la surface. Les organismes vivants sont capables de réaliser des tâches visuelles très complexes et de manière très efficace. La stéréovision fait partie de ces mécanismes complexes que les scientifiques tentent de reproduire à l'aide de caméras à haute résolution. Cette thèse aborde le problème de la stéréovision d'une manière neuromorphique par l'intermédiaire d'une nouvelle génération de capteurs de vision appelés "rétines de silicium". Ces rétines de silicium imitent les rétines biologiques en capturant l'information visuelle sous forme de flux asynchrones d'événements codant les changements de contraste avec une grande précision temporelle.

Ces capteurs sont utilisés pour étudier l'importance de la précision et de la dynamique temporelle de la scène dans le problème de mise en correspondance stéréo. Nous proposons une des premières méthodes de reconstruction 3D capable de produire des modèles 3D d'une manière totalement asynchrone, á partir de l'information visuelle. Cette approche, outre son originalité, permet également de préserver la dynamique native de la scène.

Cette thèse montre que le temps en tant que medium d'information, joue un rôle primordial dans la stéréovision. Le temps peut compléter, compenser, voire remplacer l'information apportée habituellement par la luminance et la géométrie. Ce travail établit également les fondations solides des futures recherches en vision stéréo á haute vitesse et haute dynamique, basée sur les événements. Il ouvre également de nouvelles perspectives prometteuses pour la résolution de problèmes traditionels de vision artificielle grâce à l'apport du nouveau paradigme de la vision asynchrone.

Dedication

I dedicate this thesis to my parents Maria do Rosário and Roberto Carneiro, and to my brothers and sisters Pedro, Teresa, Joana, Maria Ana, Marta, Madalena, Inês and António.

Acknowledgements

I would first like to thank my PhD advisors, Professors Ryad Benosman and Sio-Hoï Ieng, for taking me as a PhD candidate and guiding me during these three years which resulted in this thesis. This thesis would not have been possible without their advice and demand.

I am thankful to all other members of the Vision and Natural Computation team, namely Sihem Kime, Xavier Lagorce, David Valeras, Xavier Clady, Christoph Posch for their daily support and not only for keeping a good working atmosphere but also for ensuring a healthy social life. I would also like to express my gratitude to the previous members of the team who I had the pleasure to meet and who were important at different stages of my Phd should namely Henri Lorach, Cédric Meyer, Charles Clercq, Fengchun Dong and Zhenjiang Ni. Furthermore, I would like to thank all other members of the Vision Institute who I had the pleasure to meet and with whom I spent memorable moments from which I would like to particularly mention Alix Trouillet.

I would like to thank tio Artur Fernandes for his continuous support and inspiration.

I thank Emilie Arnault not only for her constant support, but also for pushing me as well when needed during this last year of my PhD.

Finally I would like to thank my family for their endless support during all stages of my PhD. Their constant and daily close presence even at far physical distance played a fundamental role in keeping a positive motivation and persisting to all difficulties.

Contents

Chapter 1

Introduction

> *The nervous system is certainly not a discrete state machine. A small error in the information about the size of a nervous impulse impinging on a neuron may make a large difference to the size of the outgoing impulse. It may be argued that, this being so, one cannot expect to be able to mimic the behaviour of the nervous system with a discrete state system.*
>
> Douglas R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, 1979

1.1 Depth perception and stereovision

While observing their surroundings, humans automatically extract a number of basic critical information for understanding and interacting with the environment. Depth is among the most important information allowing us to perceive the world in three dimensions and determine distances to objects.

We take advantage of several mechanisms which complement each others to achieve depth perception. These can be either psychological or physiological, monocular or binocular. Monocular cues use information from one single eye and include accommodation, motion parallax, retinal image size, linear perspective, texture gradient, overlapping, aerial perspective, shades and shadows. Binocular cues require information from both eyes and include convergence, stereopsis or shadow stereopsis (Howard and Rogers, 2008).

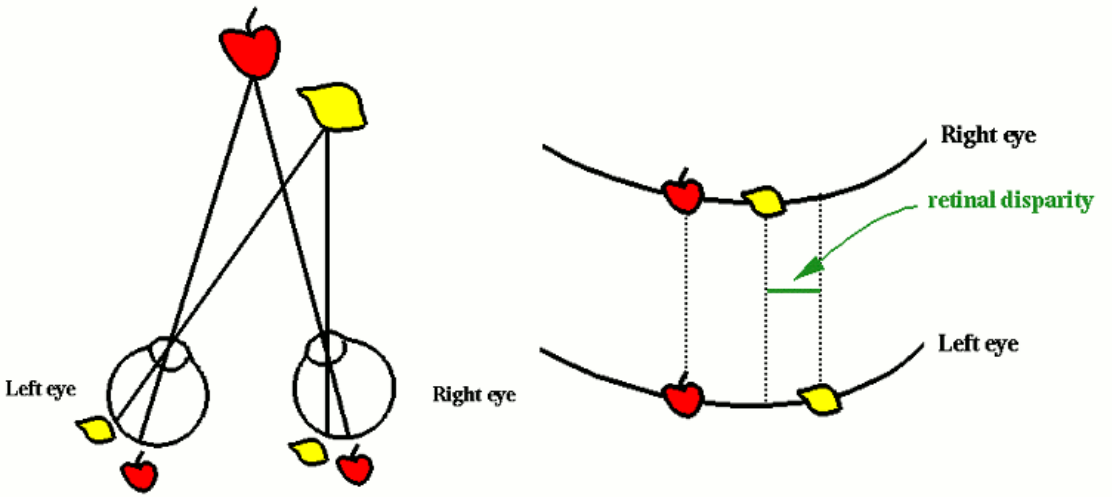


Figure 1.1: Principle of stereovision. Image courtesy of (Read, 2013)

Stereopsis or stereovision (see Fig. 1.1) is a mechanism which uses the fact that both eyes observe the world from slightly different positions giving rise to slightly different views. The magnitude of the difference between images of the same object from both views allows the computation of depth information. Binocular disparity refers to this difference between the views and directly reflects relative distance as closer objects result in larger disparities than distant objects. Stereopsis is a very precise depth perception mechanism and is by itself sufficient to guarantee depth perception even if other cues are removed, as it was first shown by (Julesz, 1960) using random dot stereograms. The use of multiple mechanisms allows removing ambiguities or errors in depth estimation. In particular, vergence is closely coupled and of great importance to stereovision, as it allows

reducing large disparities by moving the eyes in opposite directions (Erkelens, 2001).

Stereovision has been acknowledged as a method for depth perception for centuries with the first theories being proposed by Descartes and Newton (Gonzalez and Perez, 1998). The activity of cortical neurons responding to depth was however only recorded for the first time in 1967 (Bergua and Skrandies, 2000). Since then, several studies aiming at understanding the neurophysiological basis of stereopsis have been developed, successfully showing the cortical response to binocular disparities (Gonzalez and Perez, 1998). To extract depth from stereovision, our brain examines disparities on the two retinal output of an object. To achieve this, it must first be able to determine correspondences between points in both views. This is known as the stereo correspondence problem.

1.2 Stereo matching problem

The stereo correspondence problem is at the core of stereo vision but remains an open issue. Identifying for each point on the left view, its correspondence on the right view is performed effortlessly by humans. However, the solution is far from being trivial and it is still not completely understood even after decades of research. Our brain uses complex cues from the outside world and from knowledge gained through experience to impose additional constraints (e.g. color, opacity, spatial and temporal coherence) in order to solve the stereo matching problem (Read, 2002).

Several models explaining how the brain solves the stereo correspondence problem have been proposed. Cooperative models solve stereo correspondences by combining excitation and inhibition interactions. (Dev, 1975) and (Nelson, 1975) suggested that binocular cells tuned at the same disparity at neighboring spatial locations and inhibits cells tuned to different disparities for the same spatial position. A variant by (Marr and Poggio, 1976) proposes that inhibition across disparities should run along the line of sight, meaning that each monocular input can only give rise to a single binocular disparity. These models

provided the still well-known and widely used smoothness and unicity contraints. (Mayhew and Frisby, 1981) proposed a multi-component algorithm which begins by extracting local edges' location and orientation from the monocular views. Disparity is obtained from edges showing similar orientation and polarity. These cooperative stereo models however fail in depth transparency and depth averaging.

Coarse-to-fine strategies link the disparity processing to the receptive field size. This was originally observed by (Marr and Poggio, 1979), where authors propose a model composed of multiple size spatial filters where small receptive fields process small disparities and large receptive fields process larger disparities. In this model, large receptive fields control vergence eye movements which then conduct to fine disparities. Other models using this coarse-to-fine approach minimize the role of eye movement with large disparity defining the relative position of features to the fixation plane (Quam, 1984) and (Nishihara, 1984).

The most common computational model is the local correlation model proposed in (Cormack et al., 1991) and (Banks et al., 2004). Under this model, disparities are chosen as the ones producing maximal local cross-correlation between both retinal images.

Although these models propose different approaches to solve the stereo matching problem, they all rely on maximizing the interocular correlation i.e. the amount of matches between the left and right view (Howard and Rogers, 2008).

The motivation for understanding the mechanisms behind stereo matching lies on understanding how the brain is able to compute stereo vision so efficiently. If this process is well understood it will then allow large advances in bio-inspired artificial stereo vision systems with applications in robotics, medicine, engineering, or several fields.

In computer vision, several stereo vision algorithms have been developed throughout the last decades in order to estimate depth maps from binocular systems.

Stereovision algorithms can be coarsely classified into two categories illustrated in Fig. 1.2:

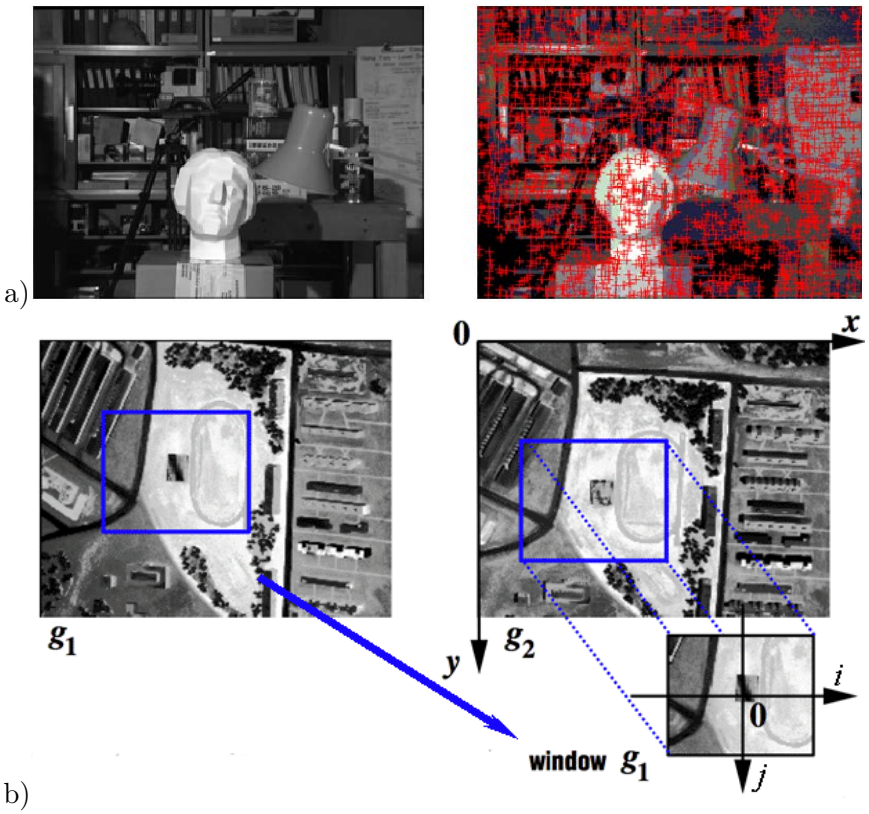


Figure 1.2: a)feature based matching b)area based matching. Image courtesy of (Bhatti, 2012)

- feature-based techniques which consist of matching feature points across images,
- area-based techniques which use image domain similarity metrics for the matching operation.

In (Scharstein and Szeliski, 2002), a taxonomy for binocular stereo dense algorithms is proposed. The authors decompose the stereo algorithms into four steps: matching cost computation, cost (support) aggregation, disparity computation/optimization and disparity refinement.

Significant improvements have been proposed over the last years. The most efficient algorithms tackle the correspondence problem by using disparity optimization methods. The aim is to enforce the smoothness assumption on both vertical and horizontal axes.

Among the recently proposed optimization techniques, graph cut and belief propagation (Szeliski et al., 2008) seem to produce interesting results, but they (Davis et al., 2005; Klaus et al., 2006; Taguchi et al., 2008; Wang and Zheng, 2008; Will and Pennington, 1971; Xu and Jia, 2008; Yang et al., 2007, 2009) are computationally expensive and resource demanding. Other techniques such as scanline optimization (Nayar et al., 2006; Scharstein and Szeliski, 2002) and dynamic programming (Scharstein and Szeliski, 2002; Veksler, 2005; Wang et al., 2006) provide accurate results with reasonable performance (Nayar et al., 2006; Scharstein and Szeliski, 2002; Veksler, 2005; Wang et al., 2006; Young et al., 2007; Zickler et al., 2002). Other reliable techniques use projectors as programmable light sources for active vision techniques using structured light range finding (Curless and Levoy, 1995; Davis et al., 2005; Scharstein and Szeliski, 2003; Will and Pennington, 1971; Young et al., 2007; Zhang et al., 2002), photometry-based reconstruction (Hertzmann and Seitz, 2003; Zickler et al., 2002), relighting (Wenger et al., 2005), light transport analysis (Nayar et al., 2006; Sen et al., 2005) and depth from defocus (Zhang and Nayar, 2006). The main advantage of projecting a set of coloured patterns onto a scene is that it eases the problem of correspondences (Zhang et al., 2002), but the method is inadequate for real-time processing. An evaluation of several algorithms can be found in (Scharstein and Szeliski, 2002).

1.3 Epipolar geometry

Epipolar geometry, sometimes called the geometry of stereovision, studies the relation between 3D points and their projection onto the image plane. Such relations allow enforcing constraints between projected image points easing the search for matching correspondences and are at the base of most computer stereo matching methods.

Let us consider a 3D point $\mathbf{X}$ projected onto the image planes of camera C at $\mathbf{x}$ and

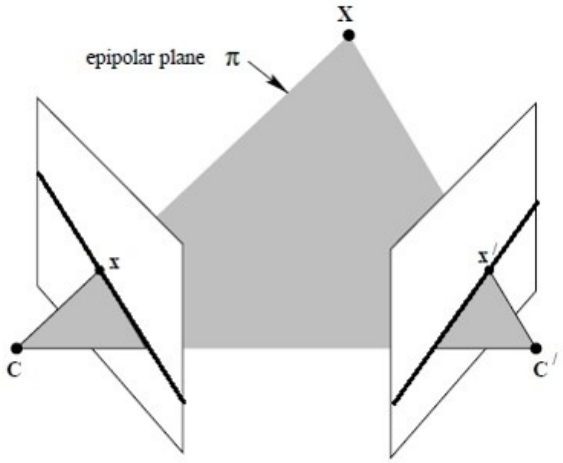


Figure 1.3: A 3D point $\mathbf{X}$ projects onto cameras C and C' as respectively $\mathbf{x}$ and $\mathbf{x}'$. A plane passing through both camera centers and the 3D point will intersect the image planes in a line called epipolar line.

camera C' at $\mathbf{x}'$ according to

$$\mathbf{x} = P\mathbf{X} \tag{1.1}$$

where $P = K \begin{pmatrix} R & -\mathbf{T} \end{pmatrix}$ is the projection matrix of C with extrinsic parameters R, $\mathbf{T}$ and intrinsic parameters K.

The plane passing through $\mathbf{X}$ and the camera centers C and C' is called epipolar plane and intersects both image planes in a line. These lines are called epipolar lines (see Fig. 1.3).

This geometrical relation allows us to define a matrix, called fundamental matrix, which maps each point on one image to a line on the other image according to

$$\mathbf{x}' F \mathbf{x} = 0 \tag{1.2}$$

where F is the 3×3 fundamental matrix. If $\mathbf{F}$ is known, the search for the correct point

$\mathbf{x}'$ in camera C' which matches $\mathbf{x}$ in camera C is reduced to the search over the epipolar line $\mathbf{l}$ defined by:

$$\mathbf{l}' = F\mathbf{x}. \tag{1.3}$$

This map from points to epipolar lines allows us to reduce the stereo matching space from the full frame of pixels to a line in an image. Once corresponding $\mathbf{x}$ and $\mathbf{x}'$ have been identified, the 3D point $\mathbf{X}$ they represent can be obtained by the relation expressed in equation 1.1, by the intersection of the back-projected rays passing through the center of the cameras and respective image points.

This section intended to provide a brief introduction to the elements of epipolar geometry wich are used throughout this thesis. For a more detailed explanation on epipolar geometry reader should refer to (Hartley and Zisserman, 2004).

1.4 3D reconstruction

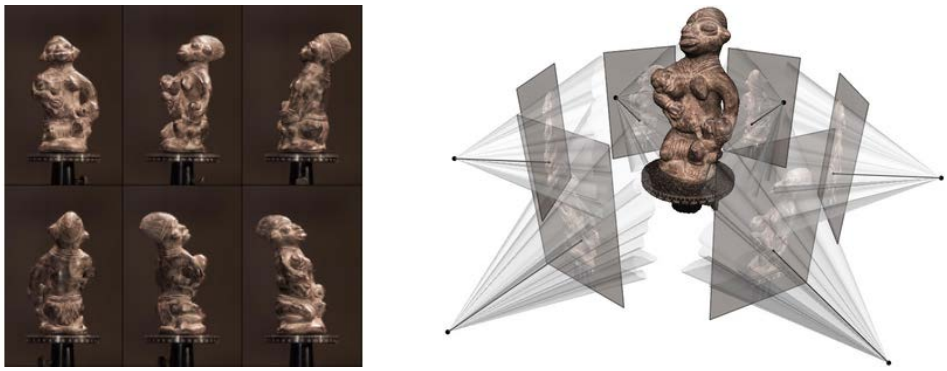


Figure 1.4: 3D reconstruction from multiple views. Image courtesy of (Hernández, 2004)

A major interest of being able to compute depth is the possibility to recover the structure of the scene. While, in binocular stereo, the goal is to produce dense depth maps, here the expected result is to build complete 3D models of the scene (see Fig. 1.4). The

ability to reconstruct an observed environment allows us to build virtual scenes which are exact copies of the real world. This virtualized reality provides, as opposed to computer generated virtual reality scenes, much richer and realistic 3D environments (Kanade et al., 1995). Since the seminal work of creating multi-camera networks (Kanade et al., 1998), tele-immersion became an important element for the next generation of live and interactive 3DTV applications. The goal of these techniques is to allow people at different physical locations to share a virtual environment.

Several methods for achieving 3D reconstruction from multiple views exist. Seitz et. al (Seitz et al., 2006) categorize them into four classes: the first class, includes voxel colouring algorithms which operate by extracting surfaces in a single sweep. A cost is assigned to each voxel of a given volume which is reconstructed if this cost is under a certain threshold (Seitz and Dyer, 1997; Treuille et al., 2004). Variants try to obtain optimal surfaces by using Markov Random Fields and max-flow (Furukawa, 2008; Roy and Cox, 1998; Sinha and Pollefeys, 2005; Vogiatzis et al., 2005) or multi-way graph cut (Kolmogorov and Zabih, 2002). The second class of algorithms includes methods which operate by iteratively refining surfaces through minimization of a cost function. Examples are space carving (Fromherz and Bichsel, 1995; Kutulakos and Seitz, 2000) and variants which progressively refine structures by adding or rejecting voxels to minimize an energy function (Bhotika et al., 2002; Eisert et al., 1999; Kutulakos, 2000; Kutulakos and Seitz, 2000; Saito and Kanade, 1999; Slabaugh et al., 2000, 2004; Yang et al., 2003; Zeng et al., 2005). Level-set techniques start from a large volume which shrinks or expands by minimizing a set of partial differential equations. The third class involves methods that compute sets of depth maps. Image-space methods enforce consistency between depth maps in order to recover a 3D reconstruction of the scene (Gargallo and Sturm, 2005; Kolmogorov and Zabih, 2002; Narayanan et al., 1998; Szeliski, 1999; Zitnick et al., 2004). Finally, the fourth class includes methods that rely on feature extraction. Features are first extracted and matched between viewpoints and a surface fitting method is then

used to reconstruct the surfaces (Faugeras et al., 1990; Manessis et al., 2000; Morris and Kanade, 2000; Taylor, 2003).

As shown, methods for achieving 3D reconstruction have been under intensive research during the last decades. Although much progress has been made, 3D reconstruction and its core problem, stereo matching, still remain fundamental research problems in computer vision. Proposed approaches lack temporal resolution and performance is far from what is provided by the examples we can find in nature such as the 3D vision humans are able to perceive. Classical frame-based cameras capture dynamic scenes as a sequence of static image frames taken at a given frequency typically around 30Hz. The precise temporal dynamic of the scene is therefore lost during the early acquisition phase as the scene is sampled at discrete points in time. Current state-of-the-art methods approach the reconstruction of dynamic scenes as the reconstruction of sequences of static scenes. Produced reconstructions are therefore limited to the low temporal resolutions of the frame-based cameras where the fine temporal resolution of the scene is lost. Currently, real-time 3D reconstruction has been achieved using depth cameras (such as the Microsoft Kinect or the Asus Xtion) however the results are noisy and can only operate in specific lightning condition. Other state-of-the-art methods which are able to produce real-time reconstructions need to find a compromise quality in order to gain in computation speed (Niesner et al., 2013).

In nature, stereovision and 3D reconstruction is achieved effortlessly and is not limited to 30Hz as we perceive the world continuously and not as a sequence of images. Current methods relying on classical cameras are computationally expensive even at low temporal resolution. The reason for such computation difficulties of current methods might come from the way visual information is encoded and the loss of the temporal precision. The next section shows the importance of precise timing in depth perception both in biology and computer vision.

1.5 The importance of time in stereo correspondence

Several studies showed that the temporal information is not only used but is also critical in the stereomatching process of the human visual system. Two temporal factors seem to be particularly important: duration of the stimulus and interocular synchronization (synchronization of images shown to the left and right eyes) (Howard and Rogers, 2008).

Although early studies showed that depth perception could be achieved when exposed to a stimulus for less than 1ms (when eyes were previously converged) suggesting the stimulus duration was not important, further research showed that in fact the stereo matching requires time to solve ambiguities and is more demanding for more complex stimulus such as dense random dot stereograms. A globally accepted idea is that correspondence is achieved by an expensive process of interocular correlation maximization (Cormack et al., 1991) (Howard and Rogers, 2008).

The synchronization between images received by left and right views has also been shown to represent a critical role in stereo matching. Experiments have been conducted where a stimulus presented to an observer had one of the views delayed either using a filter or a computer generated stimulus. Results showed that the disparity-induced depth was still perceived (Howard and Rogers, 2008). The tolerance for interocular delay, representing the amount of tolerated delay between views while still perceiving depth, has been largely studied and shown to be up to 50ms (Mitchell and O'Hagan, 1972)(Ross, 1974)(Howard and Rogers, 2008). Some authors suggest that interocular delay is not only tolerated but can also by itself produce a sensation of depth, calling temporal disparity to this purely temporal stereoscopic disparity as it was first described by Mach and Dvorak (1872) and later by Max Wolf in 1920 (Howard and Rogers, 2008)(Ross, 1974). This effect was called Pulfrich effect and was studied in detail by Carl Pulfrich in 1922 (Gonzalez and Perez, 1998). However, although the claim for the existence of temporal disparities, authors generally explain this phenomenon by assuming that the delay introduced in one

eye interferes with the signals from the other eye (Howard and Rogers, 2008).

An important conclusion should be retained, temporal consistency studies show that higher synchronization between views leads to more accurate depth extraction, whereas interocular delays give rise to non-existent depth and deformed shapes (Chang, 2009).

1.6 Bio-inspired event-based vision

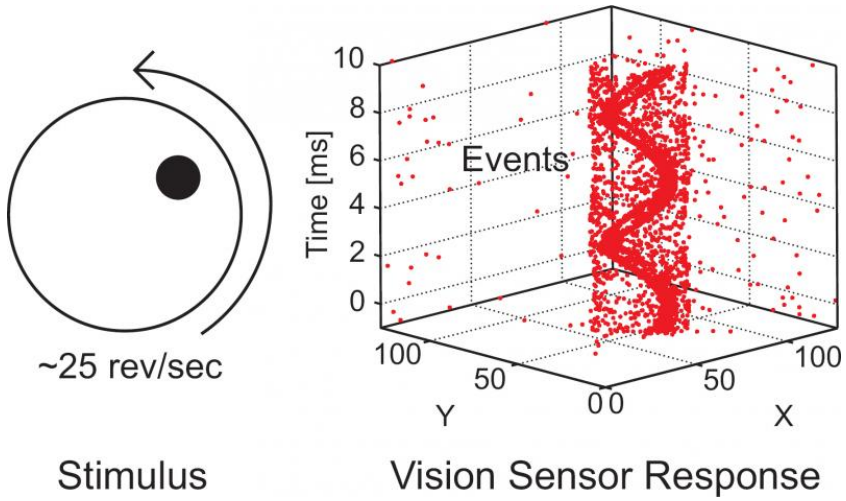


Figure 1.5: Event-based representation of a rotating disk over time. Image courtesy of (Delbruck and Lichtsteiner, 2006)

Biological retinas encode visual information differently from conventional cameras. Frame-based cameras transmit full image frames at constant rates, where each frame contains luminance information for all pixels of the visual sensor. However, biological retinas encode information as a stream of spikes, where each photoreceptor independently generates spikes that encode light intensity changes at millisecond precision (see Fig. 1.5). Therefore, only the information on parts of the scene that change (e.g. luminosity) is encoded, avoiding acquiring and transmitting redundant data while adding precise time information.

In the late 1980s, the first neuromorphic vision sensor mimicking the various be-

haviours of the first three layers of the biological retina was proposed by Mahowald (Mahowald, 1992). It introduced an analog photoreceptor that transforms the perceived light intensity into an output voltage following a logarithmic mapping. Delbruck and Mead improved the design by adding active adaptation (Delbrück and Mead, 1995) and Kramer further added polarity encoding luminosity intensity change (Kramer, 2002). In 2006, the Dynamic Vision Sensor was proposed by Lichtsteiner wich provided the first generation of ready-to-use sensors for asynchronous event-based vision(Lichtsteiner et al., 2006). In 2010 The sensor In 2011 Posch et al. (Posch et al., 2011) proposed a QVGA resolution sensor. Besides increasing by more than four times the resolution of the DVS(Lichtsteiner et al., 2008), the sensor also provides luminance information. Gray-level information of events is encoded as two events representing beginning and end of the exposure measurement. Another recent DVS development (Serrano-Gotarredona and Linares-Barranco, 2013) improves on the contrast sensitivity, allowing for the inclusion of more low-contrast visual information such as texture details. A review of some of the history and recent developments in artificial retina sensors can be found in (Delbruck et al., 2010).

These bio-inspired vision sensors encode the visual information differently from standard frame-based cameras. Their use in stereo vision and the computation of visual tasks allows to study the computation of visual information using previously unexplored temporal properties in a more bio-inspired and event-driven approach. Some examples of recent publications which show the potential in applications of these neuromorphic vision sensors to computer vision tasks include shape tracking (Ni et al., 2012), optical flow estimation (Benosman et al., 2013a) or gesture recognition (Lee et al., 2012b). Examples of biological applications also exist where authors reproduce the spatial and temporal properties of retinal ganglion cells using these bio-inspired sensors (Lorach et al., 2012).

1.7 Stereo-correspondence in neuromorphic engineering

Existing work on stereo vision with neuromorphic sensors is still poorly studied. Mahowald et al. (Mahowald and Delbrück, 1989) implemented cooperative stereovision in a neuromorphic chip in 1989. The resulting sensor was composed of two 1D pixel arrays of 5 neuromorphic pixels each. The use of local inhibition driven along the line of sight implemented the uniqueness constraint (one pixel from one view is associated to only one pixel in the other, except during occlusions), while the lateral excitatory connectivity gave more weight to coplanar solutions to discriminate false matches from correct ones. This method requires a great amount of correlator units to deal with higher resolution sensors.

In 2008, Shimonomura, Kushima and Yagi implemented the biologically inspired disparity energy model to perform stereovision with two silicon retinas (Shimonomura et al., 2008). They simulated elongated receptive fields to extract the disparity of the scene and control the vergence of the cameras. The approach is frame-based and allows to extract coarse disparity measurements to track object in 3D.

Kogler et al. (Kogler et al., 2009) have described a frame-based use of the event-based DVS cameras in 2009. They designed an event-to-frame converter to reconstruct event frames and then tested two conventional stereo vision algorithms: a window-based and a feature-based using center-segment features (Shi and Tomasi, 1993).

Delbruck has implemented a real event-based stereo tracker that tracks the position of a moving object in both views using an event-based median tracker and then reconstructs the position of the object in 3D (Lee et al., 2012a). This efficient and fast method lacks resolution on smaller features and is sensitive to noise when too many objects are present.

In 2011, Rogister et al. (Rogister et al., 2011) (see Fig. 1.6) proposed an asynchronous event-based binocular stereo matching algorithm combining epipolar geometry and timing information. Taking advantage of the high temporal resolution and the epipo-

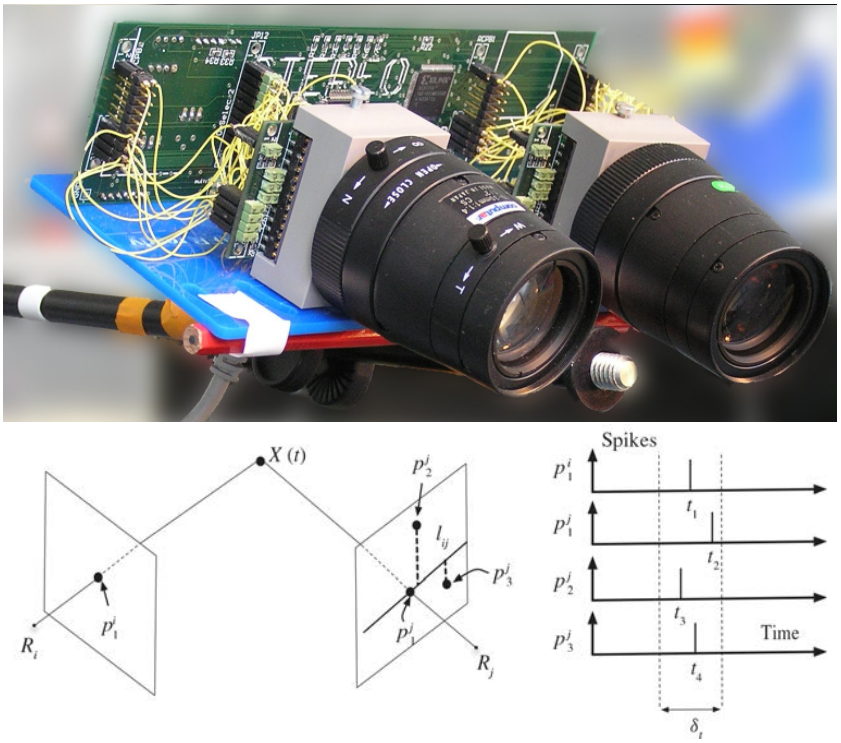


Figure 1.6: Binocular stereo setup composed of two dvs sensors and principle of operation of the asynchronous event-based binocular stereo matching method (courtesy of (Rogister et al., 2011))

lar geometry constraint they provided a truly event-based approach for real-time stereo matching. However the method is very prone to errors as enforced constraints are weak and result in many ambiguities.

1.8 Motivation and contribution

Conventional mainstream computer vision methods rely on frame-based cameras to compute stereovision. Conventional cameras sample scenes at low constant frequencies producing sequences of images. Temporal dynamic and precision are lost while information is encoded in an unnatural way. Computation in that case mainly relies on incomplete representations of the visual information.

The recent advent of asynchronous event-based neuromorphic vision sensors allow to consider visual problem from a new perspective. Information is acquired asynchronously in an event-based manner preserving temporal properties of natural scenes. As this thesis will show, a high temporal resolution is a critical information for visual processing. There are at least two clear direct advantages in using these sensors in visual computation: first it may allow a deeper understanding of biological mechanisms behind visual processing and secondly it provides a way for exploring the temporal dimension of dynamic scenes and its implication in computer vision methods.

The presented work will study the relationship between binocular fusion and simultaneity of stimulation of the two eyes. As we will show, this relationship has considerable theoretical importance as it allows to use time rather than luminance to match visual information. This work also relies on the hypothesis that the simultaneity of arrival of discharges at the cortex from the two retinas is a necessary condition for stereopsis. Evidence supporting the validity of the assumption of simultaneity is now widely accepted (OGLE, 1954a, 1954b). This work shows for the first time from a computational point of view how a simultaneous stimulation of the two eyes is a sufficient condition for stereopsis. It emphasizes the idea that precise times discharges of retinal output resulting from binocular stimulation are massively used in the cortex for binocular fusion.

The assumption of simultaneity will be explored in the context of multi-retina context as well as in binocular systems when dealing with additional information such as motion estimation and light consistency always in an asynchronous event-based framework. All these results show that computationally the assumption that precise cortical arrival of discharges is a necessary condition for the occurrence of stereopsis.

As outlined by biological research, time is a critical information for stereo vision mechanism. With the recently available neuromorphic vision sensors, we have at our disposal a promising tool for exploring and studying thoroughly the exact role played by time in biological machine vision. This thesis aims at building the first theoretical

foundation of an entirely time-based stereo vision system. Because event-based sensing also implies a visual information processing paradigm shift, several scientific obstacles must be solved. In this thesis we address the following questions:

- *Time seems to play an important role in biological stereo vision, how can the temporal precision of neuromorphic retinas be used to tackle the stereo correspondence problem and recover 3D structures?*
- *What are the advantages in using asynchronous event-based encoding of the visual information?*
- *What are the consequences and how to deal with the high event-rate of neuromorphic silicon retinas?*
- *What can temporal dynamics of scenes add to 3D reconstruction?*

This thesis is organized as follows. In chapter 2, a 3D reconstruction algorithm based on asynchronous event-based sensors is presented. Motivated by several characteristics of the event-based visual information such as sparsity of events, temporal accuracy, temporal resolution or co-activation of events, we introduce a method that combines such features with well-known geometrical properties commonly used in classical computer vision. We show that despite the very low spatial resolution of 128×128 pixel sensors we are able to produce 3D reconstructions, computer 3D models which accurately represent the registered scene. However, the method is prone to errors due to low constraints of matching. False matches and consequently noise exists in recovered reconstructions due to the incapacity to disambiguate between correct and incorrect matches. Beyond using higher resolution vision sensors the solution for achieving better reconstructions lies on finding a more constrained formulation where more information could help in solving existing ambiguities.

Knowing how points move in the scene can provide clues to identify and discard noise. Observing if points are consistent with the expected motion can serve as an indicator

as to whether they are correctly reconstructed. The method introduced in chapter 2 provides a stream of asynchronous 3D events containing a high temporal dynamic 3D representation of the scene. We study the 3D scene flow estimation problem in chapter 3 and develop a method to easily extract motion from this stream of asynchronous 3D events. Although explored in an event-based perspective we show that the method can recover 3D motion flow from any sequence of 3D point clouds such as the ones recovered from classical reconstruction methods or other depth sensors. The method relies solely on the spatial-temporal location of 3D points meaning motion can be inferred even if luminance information is not available.

In chapter 4, we use motion and luminance information to further improve asynchronous event-based stereo matching methods. We present four individual constraints (temporal, geometrical, luminance and motion) as independent matching cost functions and formulate our approach as an energy minimization problem minimizing a modular cost function. Penalties are imposed over the set of selected constraint modules composing the cost function. Independence of cost functions allows constraints to be selected according to available information and/or scenario. Luminance information may not always be available depending on the chosen sensor or conditions (e.g. in low illuminated scenes or fast movement the ATIS sensor may fail to provide gray-level information). Under these conditions motion consistency is used and high quality reconstructions can be obtained from the change events alone. However, under certain conditions, such as motion perpendicular to the camera with object located between cameras, cameras see different motions and matching cannot be achieved. When luminance is available photo-consistency can and should be used as this constraint, as opposed to the previous, is motion invariant. Furthermore, gray-level information allows building textured 3D models which provide much more realistic reconstructions of the scene as opposed to only having 3D point clouds of reconstructed edges. In chapter 4, we describe motion and luminance as functions of time. Therefore the minimized energy cost function has a

formulation which is dependent of one single variable: time. This formulation which is solely dependent on time answers several of the questions which motivated the research presented in this thesis. Results using this extended approach show that achieved reconstructions are at a much higher level than what was previously shown with better accuracy and less noise due to its more constrained formulation.

Finally, the last chapter discusses achievements of this thesis and tries to answer the set of questions that were previously raised. Contributions to the domain as well as insights for future work are proposed.

List of Publications

The work presented in this thesis resulted in the following publications:

Journal Papers

- João Carneiro, Sio-Hoï Ieng, Christoph Posch, and Ryad Benosman. Asynchronous event-based 3d reconstruction from neuromorphic retinas. *Neural Networks*, 45(0): 27 – 38, 2013b. ISSN 0893-6080. doi: http://dx.doi.org/10.1016/j.neunet.2013.03.006. URL `http://www.sciencedirect.com/science/article/pii/S0893608013000725`. Neuromorphic Engineering: From Neural Systems to Brain-Like Engineered Systems

- João Carneiro, Sio-Hoï Ieng, Xavier Clady, and Ryad Benosman. Scene flow from 3d point clouds. *International Journal of Computer Vision*, 2013a. Under review

- João Carneiro, Sio-Hoï Ieng, and Ryad Benosman. It's all about time. 2014. Under preparation

Patents

- Ryad Benosman, João Carneiro, and Sio-Hoï Ieng. Method of 3d reconstruction of a scene calling upon asynchronous sensors, 2013b. WO Patent 2,013,083,848

Chapter 2

Asynchronous Event-Based N-Ocular Stereomatching

> *(At that moment, an event-or is "event" the word for it? –takes place which cannot be described, and hence no attempt will be made to describe it.)*
>
> Douglas R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, 1979

2.1 Introduction

State-of-the-art artificial vision systems rely on frame-based acquisition of the visual information. This acquisition strategy is not able to convey the temporal dynamics of most natural scenes and, additionally, produces large amounts of redundant data. Due to these fundamental weaknesses of current visual data acquisition, even the latest developments in stereo computer vision are still far from reaching the performance of

comparatively "simple" and small biological vision systems.

Neuromorphic silicon retinas are vision sensors that mimic the behaviour of biological retinas, asynchronously encoding visual signals pixel-individually and usually at high temporal resolution. The usage of these recently developed devices in stereovision systems enables us to rethink the current approaches to the correspondence problem, supporting the development of spike-based, bio-inspired vision algorithms closely related to neurophysiological models.

In this chapter we present an event-based trinocular stereo matching and reconstruction algorithm for event-based vision data. We use the properties of silicon retina vision sensors, such as high temporal resolution and response to relative light intensity changes, to address the stereo matching problem. We produce accurate 3D reconstructions of visual scenes by applying well-known epipolar geometry in an event-based approach. Furthermore, we show that the combination of trinocular stereo and temporal constraints alone are insufficient to ensure a unique solution to the stereo correspondence problem. We then provide a bayesian inference method for discarding incorrect matches.

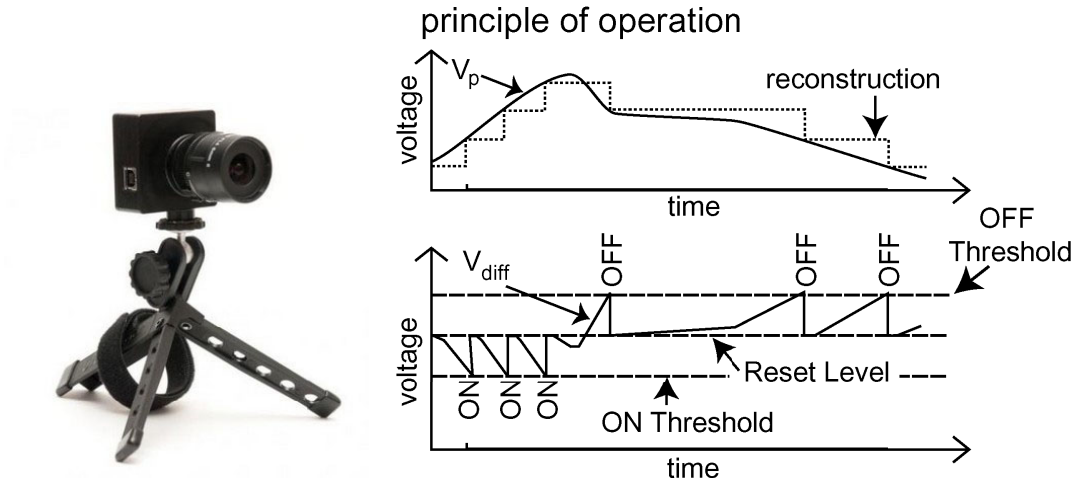


Figure 2.1: Principle of operation of a DVS pixel. Image courtesy of (Lichtsteiner et al., 2008)

The experiments reported in this chapter were conducted using the dynamic vision

sensor (DVS) described in (Lichtsteiner et al., 2008). The DVS is a 128 × 128 pixel resolution Address Event Representation (AER) silicon retina sensor that asynchronously generates response to relative light intensity variations. Pixels operate autonomously and encode temporal contrast, i.e. log intensity changes of a programmable magnitude, into events carrying the active pixel's array address and polarity of change (ON/OFF) (the principle is illustrated in 2.1). The output channel is a parallel, continuous-time, digital bus that asynchronously transmits the Address Events. The data volume of such a self-timed, event-driven sensor depends essentially on the dynamic contents of the target scene as pixels that are not visually stimulated do not produce output. Due to the pixel-autonomous, asynchronous operation, the temporal resolution is not limited by an externally imposed frame rate. However, the asynchronous stream of events carries only change information and does not contain absolute intensity information; there are no conventional image data in the sense of gray-levels.

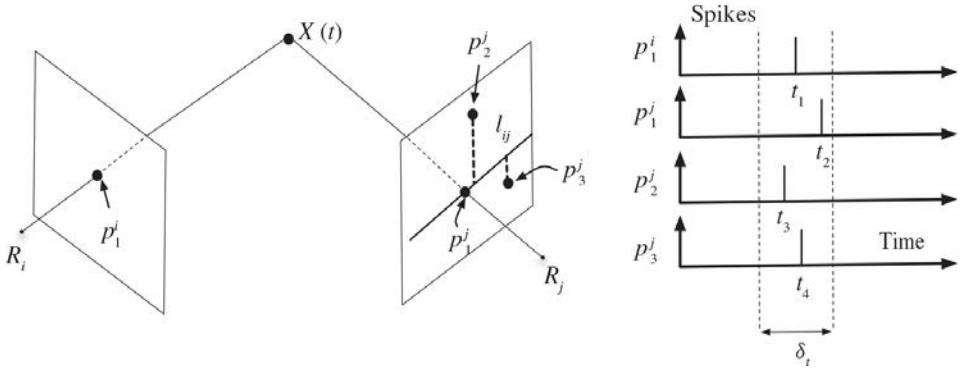


Figure 2.2: Asynchronous event-based binocular stereo matching principle proposed by Rogister et al (Rogister et al., 2011). Events are matched if they respect epipolar geometry and temporal constraints. Image courtesy of (Rogister et al., 2011)

In 2011, Rogister et al. (Rogister et al., 2011) proposed an asynchronous event-based binocular stereo matching algorithm combining epipolar geometry and timing information. The method is illustrated on Fig. 2.2. The main idea is that two pixels observing the same scene point will tend to fire at the same time. The stereo-matching is then performed using temporal coincidences of pixels. Coincidence is not sufficient to pro-

vide accurate matches, therefore the system relies on the epipolar geometry to discard false matches. Other constraints are also added, such as order constraints, unicity of matching...

Rogister's method provided a truly event-based approach for real-time stereo matching. However, the method is prone to errors meaning large amounts of noise exist in the results. This is mainly due to its weak constrained formulation. This chapter extends the previous work by generalizing it to multi-retina matching. The idea is to increase the chances of detecting coincidences by increasing the number of retinas. Namely a match is correct if it appears as a coincidence in more than two artificial retinas. In this extension, at least three instead of two cameras are required to triangulate 3D points. In the case of multi-retinas, epipolar geometry as we will show plays an important role as it allows the propagation of matches in other artificial retinas.

2.2 Asynchronous N-Ocular Stereo Vision

2.2.1 Trinocular geometry

Adding more cameras in stereovision applications is a natural technique for solving the depth recovery problem. Additional sensors not only reduce the occurrence of occlusions but also reinforces the epipolar constraint linking pairs of cameras. If the number of cameras is sufficient, the geometric constraint alone can be used to uniquely define a set of points projected onto each camera.

Figure 2.3 depicts the typical geometric configuration for a set of three cameras. A 3D point seen by the cameras R_i, R_j, R_k is projected onto their respective focal planes in $\mathbf{p}_1^i, \mathbf{p}_1^j, \mathbf{p}_1^k$. If $\mathbf{p}_1^i$ is fixed, then the epipolar constraint states that $\mathbf{p}_1^j$ (respectively $\mathbf{p}_1^k$) lies on an epipolar line in R_j (respectively in R_k). Technical details on the epipolar properties can be found in (Hartley and Zisserman, 2004).

The same property is true if we consider $\mathbf{p}_1^j$ or $\mathbf{p}_1^k$ as fixed. Thus, $\mathbf{p}_1^i, \mathbf{p}_1^j, \mathbf{p}_1^k$ are uniquely

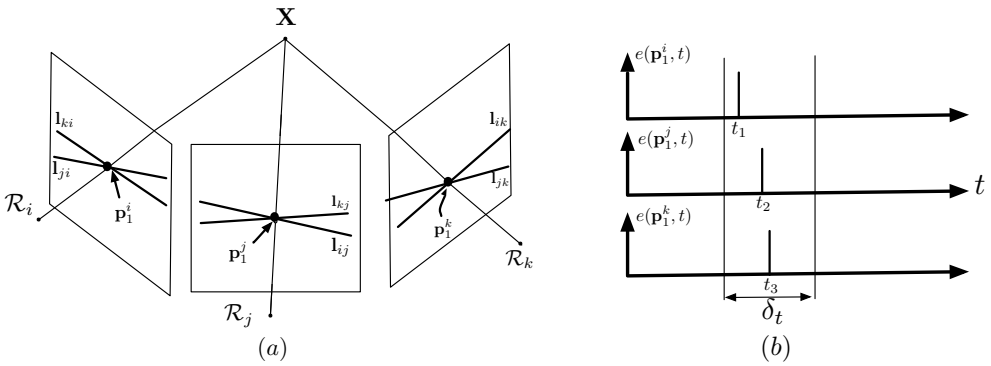


Figure 2.3: (a) Epipolar planes and lines illustrated for three cameras. A 3D point $\mathbf{X}$ is projected onto the three focal planes in $\mathbf{p}_1^i$, $\mathbf{p}_1^j$ and $\mathbf{p}_1^k$. Each of them is at the intersection of two epipolar lines defined by the geometric configuration. (b) Events generated by $\mathbf{X}$ in each camera at time t are usually not recorded with the same date t, but rather different timestamps t_1, t_2, etc. due to a finite precision in synchronizing the cameras.

defined as intersections of two epipolar lines on each focal plane. The unicity of the triplet is only true if the epipolar planes do not overlap. The overlapping happens when all the focal points are coplanar or aligned (which is a special case of coplanarity). These degenerate cases can be reduced by adding more cameras.

The geometrical constraint can be expressed by a homogeneous scalar equation built from the following definitions:

- an event e occurring at time t, observed by the camera $\mathcal{R}_i$ at pixel $\mathbf{p}_u^i = (x, y)^T$ is a function taking value in $\{-1; 1\}$ (the subscript u indexes matched events across the sensor focal planes). Its value is equal to 1 when the contrast increases and -1 when it decreases. The event is therefore defined as $e(\mathbf{p}_u^i, t)$.

- a 3D point $\mathbf{X}$ generating events $e(\mathbf{p}_1^i, t)$, $e(\mathbf{p}_1^j, t)$ and $e(\mathbf{p}_1^k, t)$, is projected respectively as $\mathbf{p}_1^i$, $\mathbf{p}_1^j$ and $\mathbf{p}_1^k$ according to the relation :

$$\begin{pmatrix} \mathbf{p}_1^u \\ 1 \end{pmatrix} = P_u \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix}, u \in \{i, j, k\}. \tag{2.1}$$

$P_u = K_u \begin{pmatrix} R_u & -\mathbf{C}_u \end{pmatrix}$ is the projection matrix of $\mathcal{R}_u$. R_u and $\mathbf{C}_u$ are the extrinsic parameters and K the intrinsic ones (for more details on the projection matrix, the reader can refer to (Hartley and Zisserman, 2004)).

The image point $\mathbf{p}_1^i$ then satisfies the epipolar constraint:

$$(\mathbf{p}_1^j \ 1)^T F_{ij} \begin{pmatrix} \mathbf{p}_1^i \\ 1 \end{pmatrix} = 0, \tag{2.2}$$

F_{ij} is the fundamental matrix establishing the geometric relation linking $\mathcal{R}_i$ to $\mathcal{R}_j$. $F_{ij} \begin{pmatrix} \mathbf{p}_1^i \\ 1 \end{pmatrix} = \mathbf{l}_{ij}(\mathbf{p}_1^i)$ is the epipolar line on $\mathcal{R}_j$, associated to $\mathbf{p}_1^i$. $\mathbf{p}_1^j$ belongs to $\mathbf{l}_{ij}(\mathbf{p}_1^i)$. Using a similar approach, all epipolar lines shown in Fig. 2.3 can be defined. If $\mathbf{p}_1^i$ and $\mathbf{p}_1^j$ are known, and the cameras are calibrated, then $\mathbf{p}_1^k$ can be found as the intersection of the appropriate epipolar lines.

2.2.2 Trinocular spatio-temporal match

Estimating 3D from the cameras requires matching each triplet $\{\mathbf{p}_1^i, \mathbf{p}_1^j, \mathbf{p}_1^k\}$ produced by $\mathbf{X}$ at time t. Since the silicon retina sensors do not provide intensity information, only the geometric property presented in the previous section can be used in conjunction with the highly accurate timing of the events. Let us define the set of events occurring within a time window around time t :

$$S^i(t) = \left\{ e(\mathbf{p}^i, t') | \mathbf{p}^i \in \mathbb{R}^2 \text{ and } t, t' \in \mathbb{R}^+, |t' - t| < \frac{\delta_t}{2} \right\}. \tag{2.3}$$

S^i defines a temporal neighbourhood of events captured by $\mathcal{R}_i$ that occur around t. Such sets are defined for each camera. Because of non-perfect synchronization of the cameras, it is unlikely that matched events are timestamped with the same t (see Fig. 2.3).

In a similar way, we define the set of events geometrically close to $\mathbf{l}_{ij}(\mathbf{p}^i)$:

$$M^j(e(\mathbf{p}^i,t)) = \left\{e(\mathbf{p}^j,t') \in S^j(t) | d(\mathbf{p}^j, \mathbf{l}_{ij}(\mathbf{p}^i)) < \Delta_p\right\}, \tag{2.4}$$

where $d(\mathbf{p}^j, \mathbf{l}_{ij})$ is the euclidean distance of $\mathbf{p}^j$ to $\mathbf{l}_{ij}$. The image points $\mathbf{p}_1^j, \mathbf{p}_1^k$, elements of sets $M^j(\mathbf{p}_1^i, t)$ and $M^k(\mathbf{p}_1^i, t)$ respectively, are matched to $\mathbf{p}_1^i$ if they minimize both $|t - t'|$ and $d(\mathbf{p}_1^i, \mathbf{l}_{ij})$ defined in Eq. (2.3) and (2.4).
Due to the finite precision of the visual acquisition in space and time, the matching process is prone to produce erroneous matches because of additional ambiguities beside the ones induced by degenerate cases. The motivation to use more than just two cameras is also given by (Maas, 1992). The authors show that the use of a third camera reduces the number of ambiguities by a factor of 10 when only geometric constraints can be used. For event-based sensors, the accurate timing adds decisive complementary constraints.

Based on the previous definitions, we design the general trinocular point matching algorithm using temporal and spatial constraint as shown in algorithm 1. This matching algorithm requires a calibrated camera setup. Appropriate calibration can be achieved with the techniques presented in (Benosman et al., 2012) if only the fundamental is needed, or the one from (Svoboda et al., 2005) if the projection matrix is also required. The algorithm can be extended to n cameras with minimal changes.

2.2.3 Stereo match selection using bayesian inference

A triplet of matched events $m_n = \{e(\mathbf{p}_n^i, t), e(\mathbf{p}_n^j, t), e(\mathbf{p}_n^k, t)\}$ is a true match if the events are generated in response to a same stimulus, at the same time. The triplet is mismatched otherwise. For each m_n, a corresponding 3D point $\hat{\mathbf{X}}_\mathbf{n} = (x, y, z)^T$ can be estimated as

Algorithm 1 Trinocular event-based stereo matching algorithm

Require: Three cameras $\mathcal{R}_i$, $\mathcal{R}_j$, $\mathcal{R}_k$
Require: F_{ij}, F_{ik}, F_{jk}, estimations of the fundamental matrix for each pair of cameras

for all events $e(\mathbf{p}_n^i, t)$ in sensor $\mathcal{R}_i$ **do**
  Determine the set of events $S^j(t)$ from sensor $\mathcal{R}_j$
  Determine the set of events $S^k(t)$ from sensor $\mathcal{R}_k$
  Compute the epipolar line $\mathbf{l}_{ij} = F_{ij}\binom{\mathbf{p}_n^i}{1}$
  Compute the epipolar line $\mathbf{l}_{ik} = F_{ik}\binom{\mathbf{p}_n^i}{1}$
  Determine the subset of possible matches $M^j(e(\mathbf{p}_n^i, t)) \subset S^j(t)$
  for all events $e(\mathbf{p}_k^j, t) \in M^j(e(\mathbf{p}_n^i, t))$ **do**
    Compute the epipolar line $\mathbf{l}_{jk} = F_{jk}\binom{\mathbf{p}_k^j}{1}$
    Compute intersection between $\mathbf{l}_{jk}$ and $\mathbf{l}_{ik}$
    if $e(\mathbf{p}_n^i, t) \in S^i(t)$, $e(\mathbf{p}_n^j, t) \in S^j(t)$, $e(\mathbf{p}_n^k, t) \in S^k(t)$ complies to the trinocular constraint **then**
      Create match $m_n = \{e(\mathbf{p}_n^i, t), e(\mathbf{p}_n^j, t), e(\mathbf{p}_n^k, t)\}$ and add it to the list of found matches $T(\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k)$
    end if
  end for
end for
return $T(\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k)$

the intersection of the back-projected rays by "inverting" Eq. 2.1:

$$\hat{\mathbf{X}}_\mathbf{n} = \bigcap_{u \in \{i,j,k\}} \lambda_u R_u^{-1} K_u^{-1} \begin{pmatrix} \mathbf{p}_n \\ 1 \end{pmatrix} + \mathbf{C}_u, \tag{2.5}$$

where λ_u is a scalar.

If a given match m_1 is a wrong match then $e(\mathbf{p}_1^i, t_1)$, $e(\mathbf{p}_1^j, t_2)$ and $e(\mathbf{p}_1^k, t_4)$ are not events induced by the same stimulus in the scene. The set m_1 yields a 3D point which either does not physically exist at time t, or at the location of $\hat{\mathbf{X}}_\mathbf{n}$ in the real scene.

The probability for a set $m_1 = \{\mathbf{p}_1^i, \mathbf{p}_1^j, \mathbf{p}_1^k\}$ at time t, to be a correct match is related to the spatio-temporal neighbourhood of $\hat{\mathbf{X}}_\mathbf{1}$. Because scenes are usually composed by geometric structures which generate edges in the sensors' focal planes, it is unlikely that an isolated 3D point $\hat{\mathbf{X}}_\mathbf{1}$ exists in the scene. We add a statistical constraint using bayesian

inference to sort outliers from correct matches. We first define the set of potential matches contained in a spatio-temporal neighbourhood of m_1:

$$W(m_1) = \left\{ m_n \in T(\mathcal{R}_i, \mathcal{R}_j, \mathcal{R}_k) | d_s(m_1, m_n) \leq \delta_s, \bar{d}_t(m_1, m_n) \leq \delta_t \right\}, \tag{2.6}$$

with

- $d_s(m_1, m_2) = ||\hat{\mathbf{X}}_\mathbf{1} - \hat{\mathbf{X}}_\mathbf{2}||$,
- $\bar{d}_t(m_1, m_2)$, the mean duration between the 6 events defined in m_1 and m_2.

δ_s and δ_t are the spatial and the temporal radii of the neighbourhood (see Fig. 2.4). The two components are decoupled to allow a fine adjustment of the neighbourhood.
Given $W(m_1)$, the probability of m_1 being a correct match is deduced from Bayes' rule :

$$P(m_1|W(m_1)) = \frac{P(W(m_1)|m_1)P(m_1)}{P(W(m_1))}. \tag{2.7}$$

2.2.3.1 Prior

Prior probability is established from the matching algorithm presented in section 2.2.1. The reliability of each match m_n is defined according to how well they comply with the spatio-temporal constraint i.e. how far temporally and spatially the events are from the epipolar intersections given a time t. Typically a gaussian distribution is fitted on the matching results.

2.2.3.2 Likelihood

The Likelihood of having a correctly matched triplet m_n is assumed to increase inversely with its distance to a triplet of matched events that is labelled as correct. Following this hypothesis, the conditional probability of m_n according to its spatio-temporal neighbour

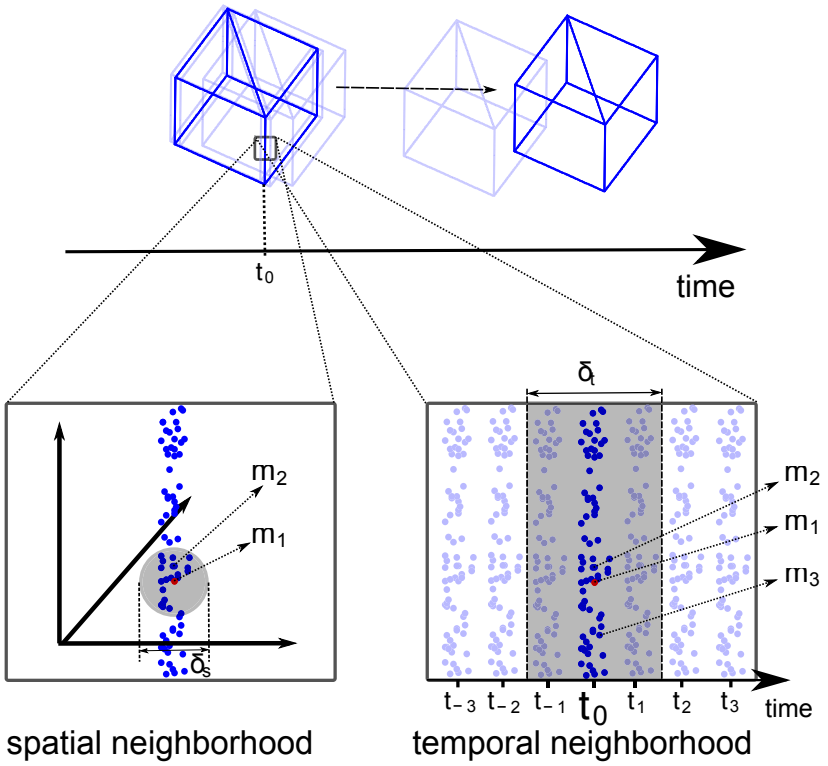


Figure 2.4: Probability of matches in the spatio-temporal neighbourhood of m_1 limited by δ_t and δ_s. m_2 is in the spatio-temporal neighbourhood of m_1 therefore has high probability of being correct. m_3 is in the temporal neighbourhood of m_1 but outside the spatial neighbourhood being therefore probably an incorrect match.

m_1 is defined as:

$$P(m_n = 1|m_1) = \begin{cases} \mathcal{N}(0, \sum) & \text{if } m_1 = 1 \\ k & \text{if } m_1 = 0 \end{cases}, \tag{2.8}$$

where $\mathcal{N}(\mu, \sum)$ is a bivariate gaussian distribution of mean value μ and covariance matrix $\sum$. The probability of having a correct match when its neighbour is not correct ($P(m_n = 1|m_1 = 0)$) is usually small, as isolated 3D points are unlikely to exist in real scenes. k is established based on observations from experimental results.

If we assume that the probability for a given $m_n \in W(m_1)$ to be a correct match, depends only on m_1 (i.e. 2 triplets of events m_i, m_j in $W(m_1)$ are independent), then

the joint probability $P(W(m_1)|m_1)$ is given by:

$$P(W(m_1)|m_1) = \prod_{m_n \in W(m_1)} P(m_n|m_1), \tag{2.9}$$

2.2.3.3 Posterior

The posterior $P(m_1|W(m_1))$ is computed continuously over time according to Eq. (2.7) in order to update the 3D reconstruction model. The 3D structure is therefore progressively reconstructed. During initial stages few sparse matches are observed and the model is poor. As more matches are found in further iterations, matches belonging to edges are given higher probability and the model is progressively refined.

2.2.4 Synchronization

The spatio-temporal matching requires the accurate synchronization of all cameras since matched events result from a common stimulus at time t. The synchronization is achieved using an external trigger signal. However the synchronization accuracy is limited due to several factors:

- non-isotropic stimuli or non identical pixel sensitivities induce different event recording times,
- varying transmission latencies of the sensor output buses due to event collisions. When multiple photoreceptors fire at the same time the sensor's bus arbiters serialize event output, thus delaying the real occurrence and potentially shuffling the firing order of events.

This synchronization uncertainty is referred to as event jitter and can be measured experimentally. We have placed an LED blinking at 10Hz in front of the 6-cameras system. The measured response is shown in Fig. 2.5: all cameras responded within a maximum delay of $631\mu s$ throughout the experiment. In average all cameras responded within a

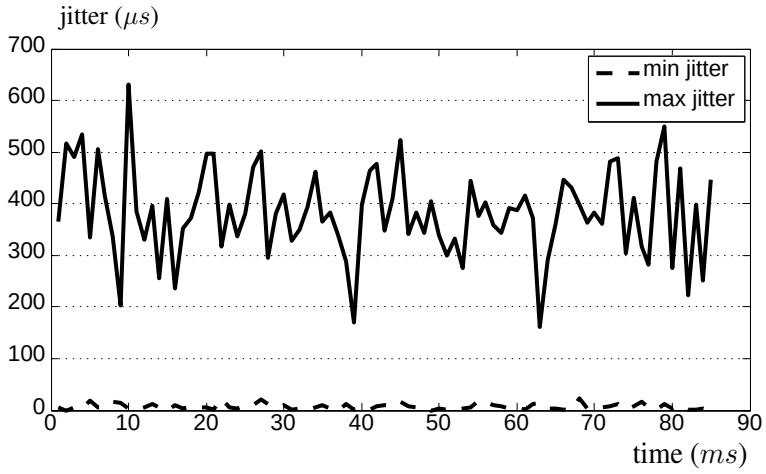


Figure 2.5: Jitter between cameras' response to a blinking led. The curves show the minimum and maximum jitter found between the responses of the cameras to the same stimulus.

$382\mu s$ window.

The variable relative delays between sensors limit the time accuracy at which events are matched. Moreover, the delays are also scene dependent, making the task even more difficult. To achieve correct spatio-temporal matches, prior assumptions about the scene should be made in order to establish the upper bound of the timing accuracy.

2.2.5 N-ocular stereo matching

The trinocular configuration provides the minimal geometric constraint to uniquely identify the set of matched events. However, the matching algorithm presented in section 2.2.1 provides a method which can be extended to any number of cameras.

The purpose of 3D reconstruction methods is to be able to recover the 3D shape of real objects. Increasing the number of different views of a given scene will naturally increase the amount of known information about the objects and decrease the amount of occlusions and will allow to produce more complete and richer reconstructions.

We propose two variations on how the method presented in section 2.2.1 can be applied to n-ocular camera systems which result in different advantages:

- Each camera contributes to enforce the epipolar constraint and the time consistency: matched points at time t are on intersections of a set of epipolar lines. The reliability of matched points increases with the number of used cameras.

- Events are matched by grouping exhaustively all subset of three cameras. For N cameras, $\binom{N}{3}$ unique trinocular configurations exist.

We have shown, in the previous section, that the system has finite temporal precision meaning that sensors may respond differently to the same stimulus not producing corresponding events at the same time. The number of matched events using the first variant decreases with the number of cameras as increasing this number increases the temporal and geometrical constraints. Obtained reconstructions are therefore more reliable as these higher constraints result in fewer incorrectly matched events. However, ensuring that corresponding events are produced by all cameras at the same time becomes increasingly difficult with the increase of the amount of cameras. Therefore the resulting 3D reconstructions often contain too little successfully matched events and are not sufficient to provide complete representations of 3D shapes. In addition, the computational effort increases drastically with the number of cameras as the epipolar lines, respective intersections and geometrical distance errors must be computed for all event candidates of each sensor.

The second variant delivers more matched events resulting in denser reconstructions, including however more errors obtained from incorrect matches. In this case, as all combinations of triplets of cameras are considered, increasing the amount of sensors also increases the probability that matching events are found on at least three sensors. Increasing the number of sensors will therefore result in increasing the amount of reconstructed points and in denser reconstructions. The computational cost however also increases with the number of cameras as more combinations of triplets exist and need to be matched. Nevertheless, in this variant, for each 3D point to be produced only events

from three cameras need to match.

Considering these observations, we chose the second variant combined with the bayesian inference filtering as the best strategy for the event-based 3D reconstruction as it provides the best compromise in terms of reconstruction density and computation cost.

2.3 Experimental results

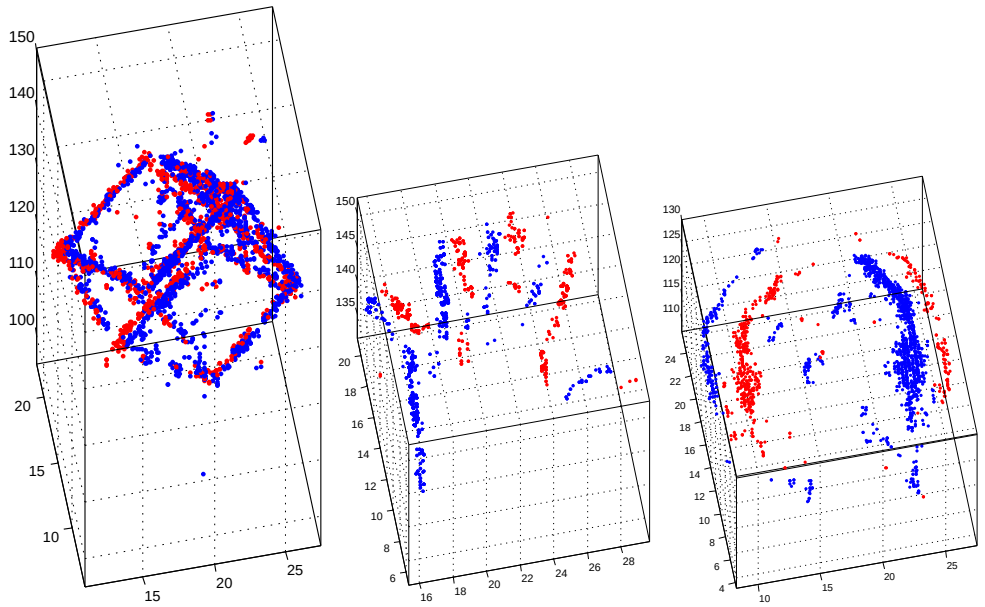


Figure 2.6: Example of reconstructions obtained using the event-based trinocular algorithm presented in section 2.2.5: (left) wireframe cube, (center) hand , (right) human face. The colors encode the polarity of the events producing the reconstructed 3D points. They give a hint to the motion's direction.

Examples of reconstructions are shown in Fig. 2.6 for three objects moving in front of the cameras. The time windows used for matching the events are defined in accordance with the jitter problem presented in section 2.2.4: $500\mu s$ is used for a swinging cube, $1000\mu s$ for a waving hand and $2000\mu s$ for a moving human face.

2.3.1 Experimental Setup

A setup of six DVS sensors (providing 128×128 pixel resolution and $15\mu s$ latency response) (Lichtsteiner et al., 2008) (see Fig. 2.7) has been used to evaluate the spatio-temporal 3D reconstructions principle. The six DVS cameras are synchronized using an external clock. The sensors are also geometrically calibrated using the method given in (Svoboda et al., 2005). The achieved calibration accuracy is sub-pixelic. Due to the low resolution of the sensors, cameras are placed facing inwards and the scene is limited to a $50cm^3$ volume. All experiments were conducted using high illumination provided by a 1000watts halogen lamp. This heavy illumination allows minimizing response time of the DVS sensors. As the sensors respond faster this illumination setup guarantees lower jitters between the sensors' responses and allows minimizing the temporal distance error between matching events.

Figure 2.7: Experimental setup composed of 6 DVS cameras.

2.3.2 Reconstruction Evaluation

Two techniques are proposed to measure reconstruction errors:

- if the ground truth is available, we measure the differences between the reconstructed shapes and the original,
- if the ground truth does not exist, we project the reconstructed shape onto the cameras that were not part of the actual triplet used for reconstruction.

For the wireframe cube, the geometric ground truth is perfectly known and is compared with the reconstructions at each new incoming event. The ground truth's 3D points are first fitted to the reconstructed points using a 3D points set registration algorithm (ICP) (Chetverikov et al., 2005). Then the mean distance, normalized by the edge length c, of all the reconstructed points to the ground truth is computed:

$$\epsilon = \frac{\sum_{i=1}^{n} e_i^2}{n.c^2}. \tag{2.10}$$

(see Fig. 2.8 for an illustration of distances e_i of reconstructed points (circles) to the

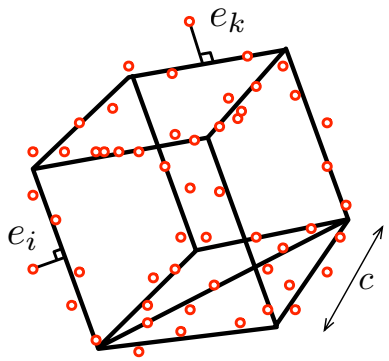


Figure 2.8: Distance of reconstructed points to the ground truth model.

ground truth model (plain curve)).

Figure 2.9 summarizes the normalized reconstruction errors of the moving wireframe cube using all camera combination triplets out of a set of 3 to 6 cameras. The error is computed for the entire sequence with and without Bayesian inference. Using Bayes' rule to filter erroneous matches successfully decreases the relative error by a factor of two. One can see that increasing the number of sensors also increases the number of reconstructed 3D points. We can notice a variability in the amount of obtained points which is related to the movement of the object. When the object moves slower the amount of events generated by the sensors is lower and therefore lower number of points will be reconstructed. Furthermore, as the object enters or leaves the field of view of one or more sensors the number of reconstructed points will also naturally increase or decrease.

The relative mean error gives an idea of how reliable is the reconstructed shape. In this case, the mean error is around 2.5% for a set of 5 sensors using the trinocular algorithm alone. The same error is reduced to 1.5% if the Bayesian inference is applied. For a 6 cameras system, these values are reduced to 1.5% and 0.5% respectively. The reconstruction errors for 2 sensors using the method from (Rogister et al., 2011) are also plotted in order to show the performance of the trinocular algorithm since relative errors never exceed 1% while with the technique given in (Rogister et al., 2011) can reach 400%. 3D reconstruction results shown in Fig. 2.10 give a quick visual assessment of the reconstructions performance: the reconstructed points of the cube (b) is more scattered than the ones using 3 or more cameras.

With regard to this preliminary result, we state that the wireframe cube is usually reconstructed with acceptable accuracy. Any other quality assessment giving scores similar to the cube reconstruction are therefore assumed to correspond to reconstructions of sufficient quality.

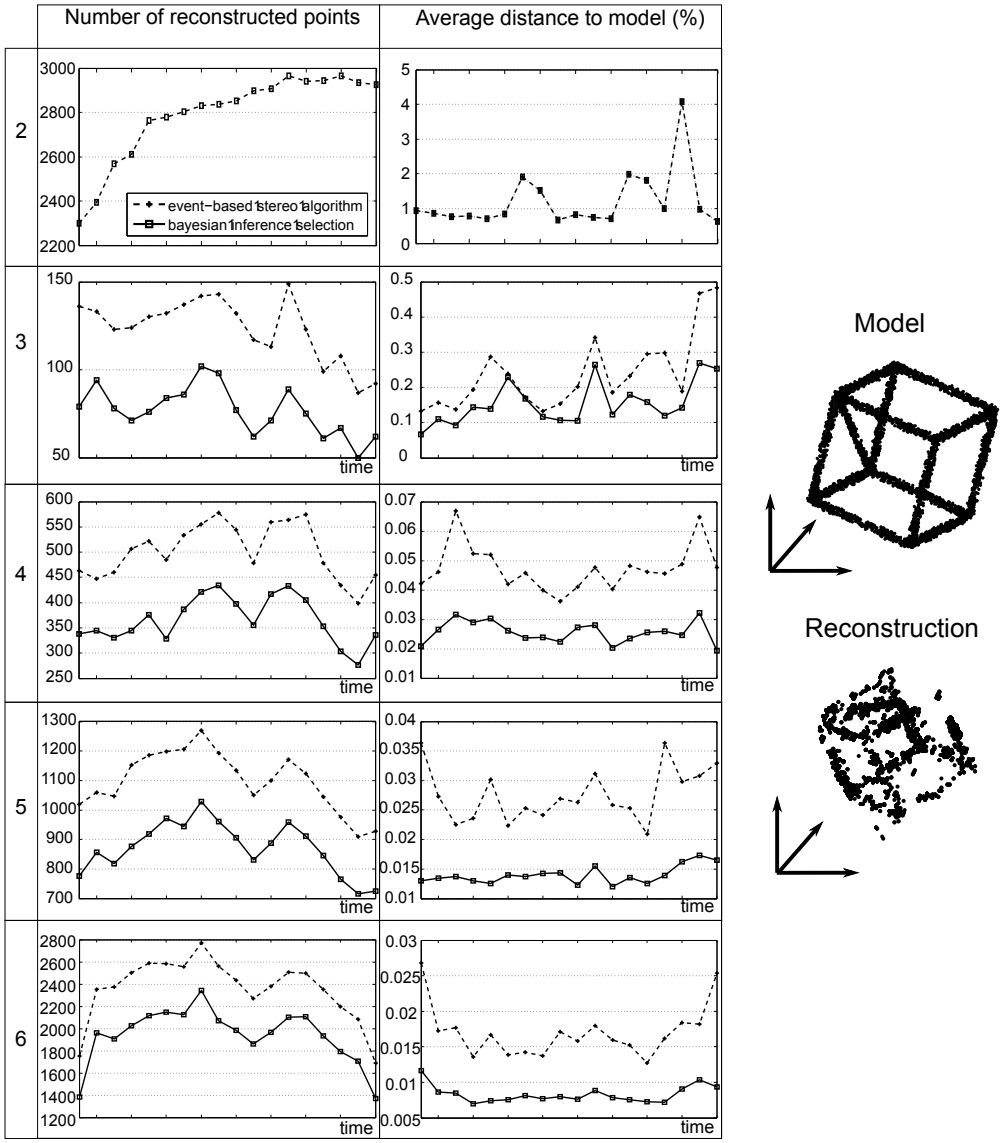


Figure 2.9: Reconstruction errors of the wireframe cube: the first column curves show the number of reconstructed points while in the second column, curves are showing reconstruction's errors. For comparison purpose, we put in the top row, the results achieved by the method explained in (Rogister et al., 2011). Rows 2 to 5 are the results produced by the method we have introduced in this chapter, with the number of cameras increasing from 3 to 6. Reconstructions quality is also measured with (dashed curves) and without (plain curves) Bayesian inference.

Reconstructions for which the ground truth is unavailable require another technique for evaluating their accuracy. We apply a variation of the method presented in (Sinha

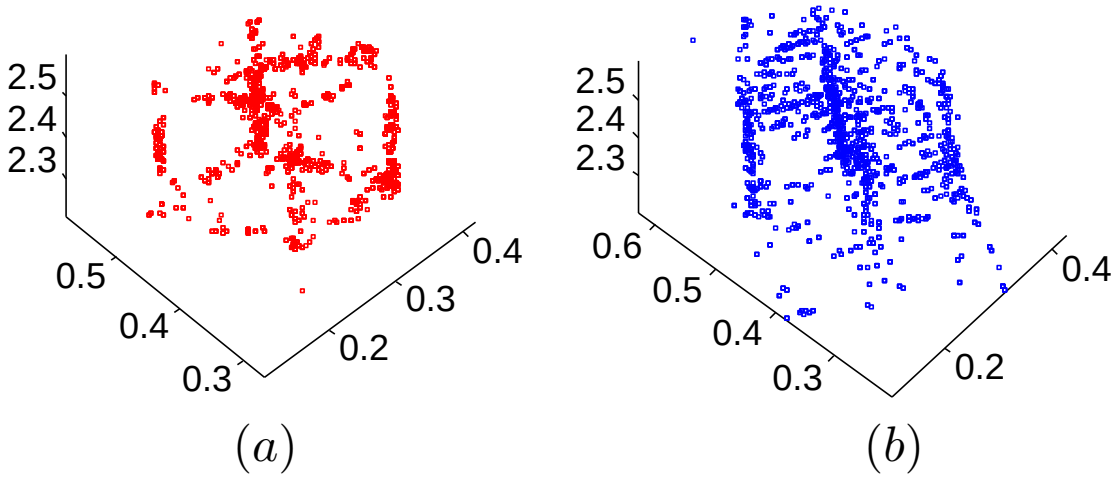


Figure 2.10: Reconstructed cube using the trinocular constrained algorithm (a) and using the initial work of Rogister et al (Rogister et al., 2011) using the event-based coincidence match for two cameras (b). In the second case, the reconstruction result is visibly less accurate.

et al., 2004). Assuming that a set of the 3D object is built from three cameras $\mathcal{R}_j, \mathcal{R}_k, \mathcal{R}_l$ and given another camera $\mathcal{R}_i$ with $i \notin \{j, k, l\}$, the reconstructed objects are evaluated as follows:

- the objects are projected onto $\mathcal{R}_i$.All events arriving in a time window are merged to provide a frame.
- A frame is also built by integrating the events captured by $\mathcal{R}_i$ over the time window defined for the matching algorithm (e.g. $500\mu s$ for the cube).
- The ratio of pixel differences given by subtracting both frames produces the projection error.

Figures (2.11, 2.12, 2.13) show the reconstruction errors for the three sequences.
The estimated error is low for the cube on all cameras: around 3% of error for each sensor. Since the cube reconstruction has already been shown to be accurate, an error of

this magnitude is considered as a good indicator of a reliable reconstruction. For both the hand and the face sequences, the estimated errors have the same order of magnitude (3% for the hand and 5% for the face). We can therefore deduce that the trinocular algorithm is providing sufficiently accurate 3D reconstructions.

2.3.3 Processing time

The processing time is a critical issue especially for real-time applications. In Fig. 2.14, the processing time with respect to the number of reconstructed points for the three sequences are shown for sets of 3 to 6 cameras. The computational effort of the proposed method unsurprisingly increases with the number of used cameras. For each event, possible candidates in all combinations of triplets of cameras are tested resulting in the visible increase in the computation time. In this chapter we proposed a new approach for achieving 3D relying on precise timing and events instead of frame and luminance. The method is visibly much simpler than how classical methods achieve 3D reconstruction. The theoretical framework was proposed but we did not work on achieving high performing implementations (such as compiled languages such as C/C++ or hardware implementations) of our method. However, a remarkable observation is the linearity of the processing time for whatever number of cameras illustrating the linear complexity of the global reconstruction process.

2.4 Conclusion and Discussion

The Event-based matching approach shows the possibility to recover 3D from time and inter-camera geometric consideration only. Two variants of the matching algorithm have been tested for 3D reconstructions: the first method uses all possible combination of

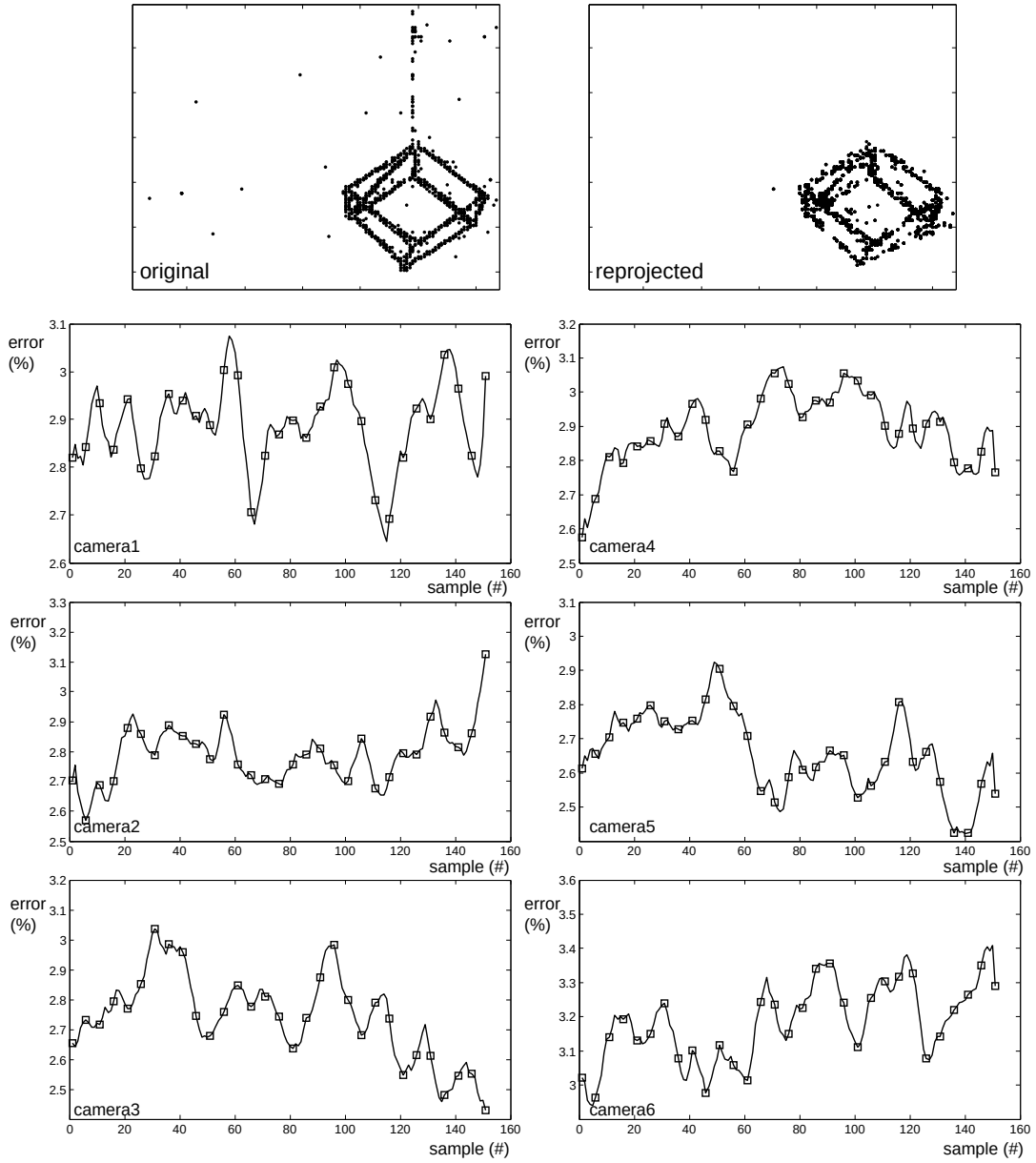


Figure 2.11: Reprojection errors on each of the 6 cameras. For each camera $\mathcal{R}_i$ that is tested, 3D cubes built from any combination of 3 other cameras $\mathcal{R}_{j,k,l}$ are projected onto $\mathcal{R}_i$. The obtained frame is then compared to the frame built by integration. Mean projection errors are around 3%.

three sensors to compute 3D points from events while the second method uses all sensors to enforce the epipolar constraint.

The first method gives the best compromise between the reconstruction accuracy, density

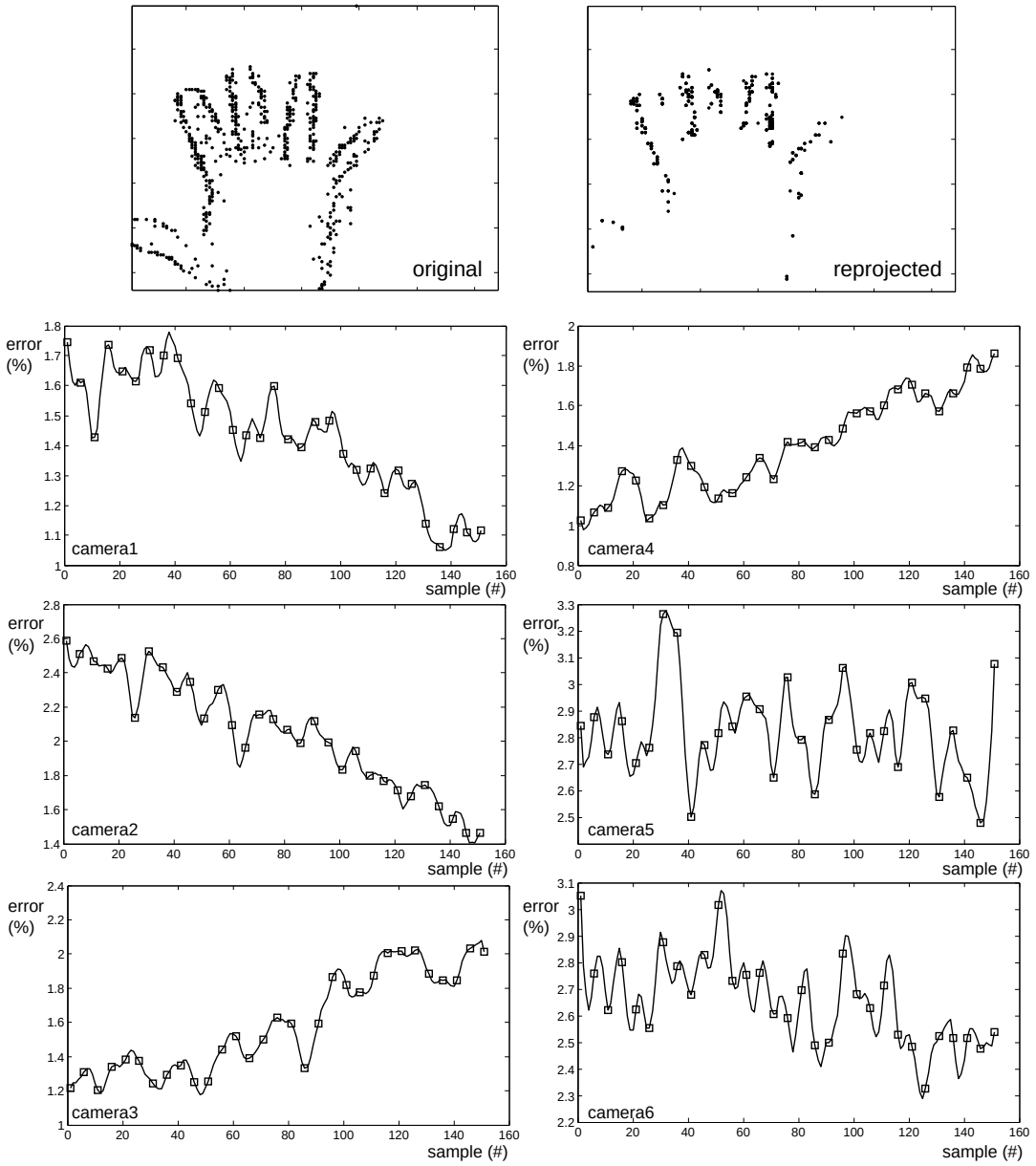


Figure 2.12: Reprojection errors estimated for the hand, using the same method as for the cube. Mean projection errors do not exceed 4%. For the first four cameras (left column and the top right curves), the error curves are showing constant increase/decrease. This is due to the hand leaving/entering the field of view of the cameras.

and computation time. Since the reconstruction complexity is likely linear as suggested in section2.3.3, we expect the algorithm being largely optimizable. The algorithm's runtime is large on non compiled programming languages such as Matlab, however it is very

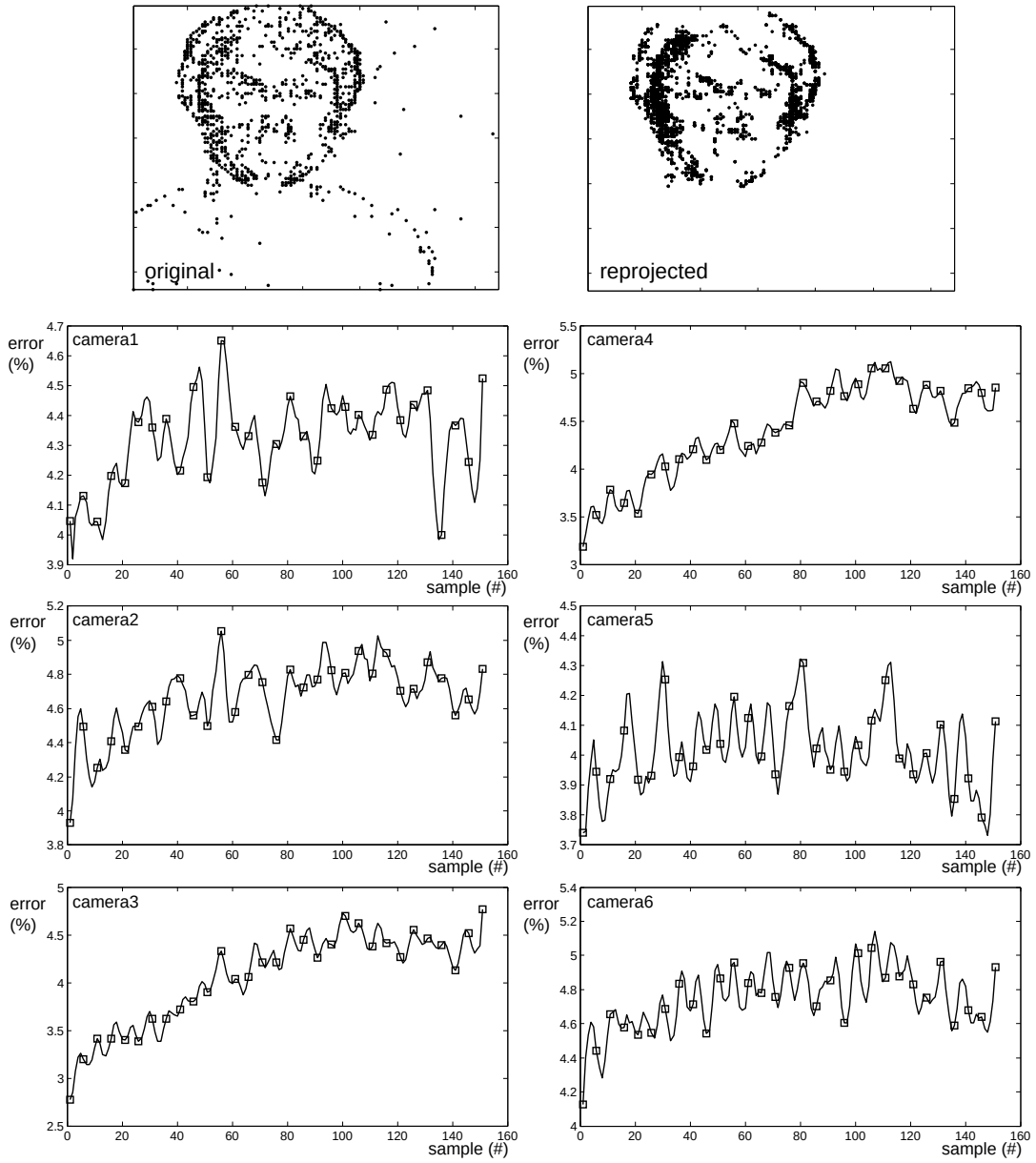


Figure 2.13: Difference between an image frame of events and reprojection of the reconstructed face on each of the 6 cameras. The mean projection errors in this sequence is not exceeding 5%.

likely that processing time can largely be reduced and meet real-time constraints when using a compiled programming language such as C.

Few event-based depth estimation techniques have been proposed in the literature. Ex-

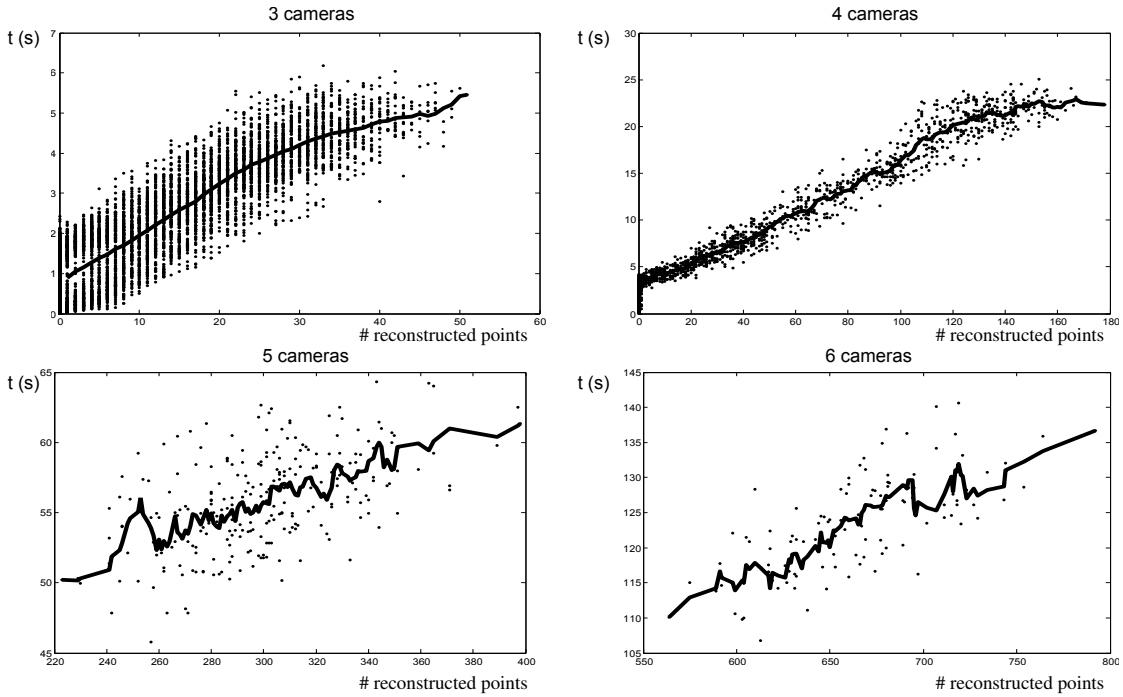


Figure 2.14: Processing time as function of the number of reconstructed points, for 3 to 6 cameras. The mean processing time is represented by the plain curves.

isting ones are still far from being able to achieve similar depth estimation accuracy and real-time 3D models computation. However, it has been shown in this chapter, that although 3D reconstructions can be produced, results are still prone to ambiguities giving rise to noisy 3D point clouds. This hints that both higher spatial resolution sensors and the addition of further constraints to the proposed spatio-temporal error constraint should conduct to higher accuracy and cleaner 3D reconstructions.

This work shows that asynchronous event-based stereo algorithm applied to multi-cameras systems of neuromorphic artificial retina sensors opens new perspectives for recovering 3D information from sparse and asynchronous spatio-temporal signals. The achieved reconstructions are accurate despite the relatively low spatial resolution of the used sensor (128×128 pixels). The precise event timing provides a mean to overcome this limitation, it shows that an event-based vision sensor is the ideal device to capture dynamic scenes. We provide a method which is able to provide reconstructions of dynamic

scenes at 1MHz relying on very simple computation operations. This work extends and pushes to a much higher level the neuromorphic stereo vision formulation initiated in (Rogister et al., 2011).

A major difficulty in establishing spatio-temporal stereo vision is the variable temporal precision of the acquired visual information. Since time constitutes critical information, highly precise event timing is required, hence there is a need to establish that precision at the sensor's level. Effective timing accuracy is limited by physical constraints of the sensor and system hardware such as e.g. CMOS device mismatch leading to intra-chip and inter-chip variations of contrast sensitivity and event latencies, event bus congestions, and timestamping quantization errors (Lichtsteiner et al., 2008). Increasing the number of cameras did as expected lower the number of false matches generating reliable 3D reconstruction. Nevertheless, there are still several false matches that are not removed by the approach, this is due to the low amount of information carried by a single event. An event carries only its location and time of arrival. As we will show in chapter 4, it is compulsory to use additional information. The context of an event must be added, namely what is the background and local activity around it? This additional information seems inevitable and must take into account dynamical information such as the relative timing between neighbouring events, their velocity and when possible even their gray-levels.

We will first inquire how the 3D point clouds generated by the method can be used to refine the stereo matching algorithm. In the next chapter, we will introduce a smoothing approach of generated data based on the local dynamics of reconstructed 3D point clouds. We will explore scene flow estimation which provides information of how 3D points move in the scene and, in a feedback loop, can be used to iteratively refine reconstructed structures.

Chapter 3

Scene flow from 3D point clouds

> *Achilles: The flag is moving.*
> *Tortoise: The wind is moving. (...)*
> *Zeno: (...)Not the wind, not the flag-neither one is moving, nor is anything moving at all. For I have discovered a Great Theorem, which states: "Motion Is Inherently Impossible." And from this Theorem follows an even greater Theorem-Zeno's Theorem: "Motion Unexists."*
>
> Douglas R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, 1979

3.1 Introduction

In this chapter, we study the use of asynchronous event-based 3D reconstruction point clouds to refine focal plane's stereomatching. The approach introduced in the previous chapter is not sufficiently constrained to be robust to wrong matches. We will then introduce a new constraint based on the consistency of motion of reconstructed 3D points.

The recent advent of new sensors such as the RGB-D cameras (Khoshelham and Elberink, 2012) or Time-Of-Flight Range-Imaging sensors (Hansard et al., 2012) allows to directly produce computationally inexpensive 3D information. A 3D scene flow refers to the vector field that maps each 3D scene point to its corresponding instantaneous velocity vector (Vedula et al., 1999). Existing methods conventionally rely on a combined use of luminance and depth information to estimate 3D motion flow (Herbst et al., 2013). This chapter introduces a method to estimate 3D motion flow directly from point clouds without the need of luminance information. We show that a robust estimation of the 3D flow can be decoupled from luminance. Thus, this approach allows to estimate 3D motion flow not only from the event-based cameras system shown in chapter 2 but from all types of sensors that provide depth data such as high definition LiDAR sensors. The minimal requirements are the 3D spatio-temporal coordinates of the reconstructed points namely when and where a 3D point occurred. Points' brightness is not mandatory since only the scene's structure matters. In this chapter, we only assume the non-deformability of local spatio-temporal surfaces. These are used to estimate 3D flow from their local planar orientation. The velocity estimation is reduced to a one-dimensional search over $\mathbb{R}$ and the dense estimation is directly achieved using local spatio-temporal planes. An additional advantage is its ability to determine velocities collinear to moving edges assuming it is possible to identify local 3D structures across the trajectory.

Scene flow is closely related to scenes' structure, estimating one usually implies estimating the other as well. The Structure From Motion (SFM) is one of the classical computer vision problems that was largely researched during the past few decades by the machine vision community (Maybank, 1993). However, SFM's high vulnerability to images' noise and to camera calibration errors raised questions regarding its applicability in real-world scenarios (Tomasi and Zhang, 1995). Currently, with the increasing demand for realistic and high definition 3D content, many ready-to-use sensors are now able to build dense 3D points clouds in real-time, such as laser range-finders, structured light devices, etc. These

devices allow to decouple the structure reconstruction from the motion estimation and to focus the effort on motion extraction and its characterization. To achieve dense scene flow estimation, state-of-the-art techniques consist in building depth maps and computing optical flows for each camera separately and combining them to consistently estimate the 3D flow. This approach parametrizes the motion problem on the image plane i.e. in 2D and is the most commonly found in the existing literature (Isard and MacCormick, 2006; Vedula et al., 1999; Wedel et al., 2011; Zhang et al., 2001). 2D parametrization is however argued being more prone to discontinuities since a smooth 3D signal may be projected into a discontinuous 2D one due to occlusions. 3D parametrization has therefore been employed to bypass this limitation. In (Basha et al., 2013), the depth map and the optical flow are solved simultaneously rather than in a sequential manner, as authors argue, for a better coupling between spatial and temporal information. In (Hadfield and Bowden, 2011; Park et al., 2012), the motion flow is extracted and refined directly from the 3D point clouds by using particle filtering or tensor voting techniques. Optical flows are only estimated for comparison purposes or for initial scene flow estimation. A second requirement for obtaining dense flow estimation is to introduce some form of regularization. For that purpose, one recurrent hypothesis is to assume local rigid body motion and therefore inducing local constant velocity i.e. points on a non-deformable surface will have the same velocity. Regularization is performed often by minimizing an energy function with variational formulation (Huguet and Devernay, 2007; Min and Sohn, 2006; Zhang et al., 2001). Energy minimization has proven to often be a successful technique for both 2D and 3D flow parametrization, however it is also resource and time consuming thus making it difficult to achieve real-time estimation without embedding a dedicated powerful computation unit (e.g. GPU). Scene flow can also be computed from local descriptors of reconstructed surfaces such as the surfel that encodes the local geometry and the reflectance information of reconstructed surfaces (Carceroni and Kutulakos, 2002). Motion is then estimated in an integrative approach by matching descriptors over time.

3.2 Scene flow parametrization

Scenes made of rigid mobile objects provide an important constraint for the motion flow estimation: as objects are non-deformable, the motion flow of points on their surface have identical velocity. This assumption is reasonable for simple objects but it does not well-define more complex objects such as a human body. The same hypothesis is however reasonable if it is applied as a local property. We suggest to subdivide the mobile objects into small non deformable surfaces and points of a same given surface will have the same velocity. The previous assumption is certainly not true for rotations, but under the assumption that infinitesimally small patches can be defined, we assume that rotations can be approximated to small translations, which will later be confirmed by our empirical results. The smaller these surfaces are, the better the proposed technique estimates the velocity and this spatial resolution is only limited by the sensor's accuracy.

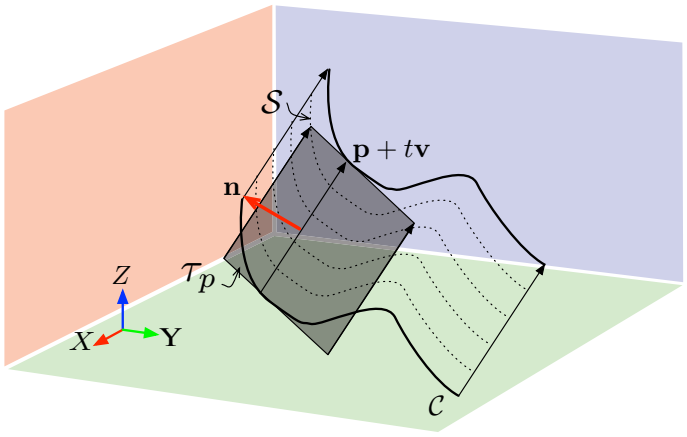


Figure 3.1: The non-deformable surface hypothesis allows to assume the velocity $\mathbf{v}$ is locally constant. The surface $\mathcal{S}$ swept by the edge $\mathcal{C}$ in the direction $\mathbf{v}$ is a ruled surface whose tangent plane τ_p at $\mathbf{p}$ allows recovering $\mathbf{v}$ if sufficient geometric constraints can be derived.

Let us consider a smooth edge $\mathcal{C}$ which can be assumed planar within a small enough spatial neighbourhood. If the velocity of $\mathcal{C}$ is constant, then as time increases, the

edge generates a ruled surface $\mathcal{S}$ in the direction of the velocity $\mathbf{v}$. The surface can be algebraically defined by the equation:

$$\begin{array}{rccc} \mathcal{S}: & \mathbb{R}^3 \times \mathbb{R}^+ & \rightarrow & \mathbb{R} \\ & (x, y, z, t) & \mapsto & \mathcal{S}(\mathbf{p} + t\mathbf{v}) = 0 \end{array} \tag{3.1}$$

where $\mathbf{p} \in \mathcal{C}$. Figure 3.1 shows an illustration of such ruled surface.

The velocity vector is according to Eq.3.1 the directrix of the ruled surface swept by the edge, hence the estimation of $\mathbf{v}$ is equivalent to determine the surface's directrix. In addition to Eq.3.1, if the surface is smooth (i.e. of class $\mathcal{C}^1$ at least), we get a second equation satisfied by $\mathbf{v}$:

$$(\nabla \mathcal{S})^T \mathbf{v} = 0, \tag{3.2}$$

because the directrix $\mathbf{v}$ is contained in the tangent plane $\mathcal{T}_p$ (Sommerville, 1934). $\nabla \mathcal{S}$ refers to the gradient of $\mathcal{S}$. Only the direction of $\mathbf{v}$ can be deduced from the two scalar equations since $\mathbf{v}$ has 3 components. Its norm can be for example set arbitrarily to 1 (e.g. a unit vector). To determine the correct amplitude, additional constraints are required and one possible way to find them is to operate for example a shape registration technique.

The velocity estimation will be achieved in two steps:

- a local fitting of a smooth surface to the 3D point clouds is operated to derive as much equations similar to Eq. 3.1 and 3.2 as possible,
- $\mathbf{v}$ is estimated from the constraint established by the set of equations.

The necessity to have more equations comes from, as mentioned before, the fact that we are short of one equation for recovering $\mathbf{v}$. To solve this problem, we propose to study three surfaces derived from $\mathcal{S}$. Let $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$ be respectively the surfaces built

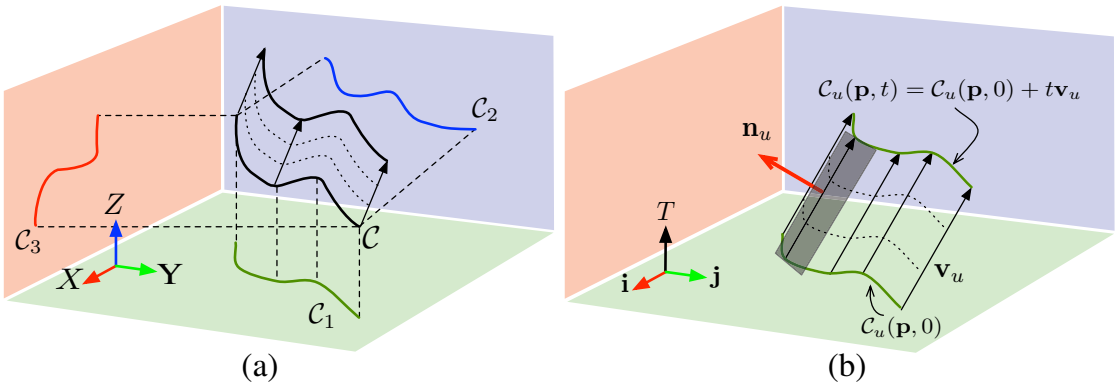


Figure 3.2: (a) A 3D edge $\mathcal{C}$ moving at constant velocity $\mathbf{v}$ is projected as 2D curves in each of the three planes $(O,X,Y),(O,Y,Z)$ and (O,Z,X). (b) Each of the projected curve C_u for $1 \leq u \leq 3$ is also moving at constant speed $\mathbf{v}_u = (v_i, v_j, 1)^T$ in the coordinate frames (ijT) ((i,j) being any element in the set $\{(x,y),(y,z),(z,x)\}$) and is sweeping a ruled surface as t increases.

from Eq. 3.1 in each coordinate frame (X,Y,T), (Y,Z,T) and (Z,X,T). Because of the constant velocity hypothesis, we get three surfaces with implicit equations of the form:

$$\mathcal{S}_k(i,j,t) = \mathcal{S}_k\left(\begin{pmatrix} p_i \\ p_j \\ 0 \end{pmatrix} + t\begin{pmatrix} v_i \\ v_j \\ 1 \end{pmatrix}\right) = 0, \tag{3.3}$$

where (i,j) is any pair of elements in $\{(x,y),(y,z),(z,x)\}$ and k indexes the k^{th} element of this list e.g. if $k=1$, $(i,j)=(x,y)$. This means we are working with the x, y and t components of S.

These surfaces are also ruled surfaces of respective directrices $(v_x,v_y,1)^T$, $(v_y,v_z,1)^T$ and $(v_z,v_x,1)^T$ and their generatrices are the restrictions of $\mathcal{C}$ to (X,Y,T), (Y,Z,T) and (Z,X,T). Following Eq. 3.2 we can establish for each $\mathcal{S}_k$ the equation:

$$(\nabla \mathcal{S}_k)^T \begin{pmatrix} v_i \\ v_j \\ 1 \end{pmatrix} = \frac{\partial \mathcal{S}_k}{\partial i} v_i + \frac{\partial \mathcal{S}_k}{\partial j} v_j + \frac{\partial \mathcal{S}_k}{\partial t} = 0. \tag{3.4}$$

As illustrated by Fig. 3.2, we now have three geometric constraints, which can be rearranged into a matrix form:

$$\underbrace{\begin{pmatrix} \mathcal{S}_{1,x} & \mathcal{S}_{1,y} & 0 \\ 0 & \mathcal{S}_{2,y} & \mathcal{S}_{2,z} \\ \mathcal{S}_{3,x} & 0 & \mathcal{S}_{3,z} \end{pmatrix}}_{M} \mathbf{v} = - \begin{pmatrix} \partial \mathcal{S}_1 / \partial t \\ \partial \mathcal{S}_2 / \partial t \\ \partial \mathcal{S}_3 / \partial t \end{pmatrix}, \tag{3.5}$$

with the convention that $\mathcal{S}_{k,x}$ (respectively y, z) is the partial derivative with respect to x (respectively y,z). To determine $\mathbf{v}$, the ideal case would be to have M invertible i.e. it is full ranked. There is no obvious way to tell from the general expression of M.

3.2.1 Plane approximation

Solving Eq. 3.5 for $\mathbf{v}$ cannot be done without knowing the analytic equations of $\mathcal{S}_k$. We propose to apply a local plane fitting to establish the matrix M. The choice of a plane instead of a more complex surface is motivated by the fitting simplicity and its computational cost even though planes give rise to rank-2 matrices M, as it will be shown further.

Let $\mathbf{\Pi_1}$, $\mathbf{\Pi_2}$ and $\mathbf{\Pi_3}$ be the planes that are fitted locally to the surfaces $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$ respectively. They then can be locally expressed using the plane's implicit equation as:

$$\mathcal{S}_k(i,j,t) = \mathbf{\Pi_1}^T \begin{pmatrix} i \\ j \\ t \\ 1 \end{pmatrix} = 0, \tag{3.6}$$

where $\mathbf{\Pi_k}^T = (a_k, b_k, c_k, d_k)$, for $1 \leq k \leq 3$.

If we derive Eq. 3.6 with respect to each of the spatial and temporal components and for

each $\mathcal{S}_k$, then Eq 3.5 becomes

$$\begin{pmatrix} a_1 & b_1 & 0 \\ 0 & a_2 & b_2 \\ b_3 & 0 & a_3 \end{pmatrix} \mathbf{v} = - \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}. \tag{3.7}$$

3.2.2 Rank of M

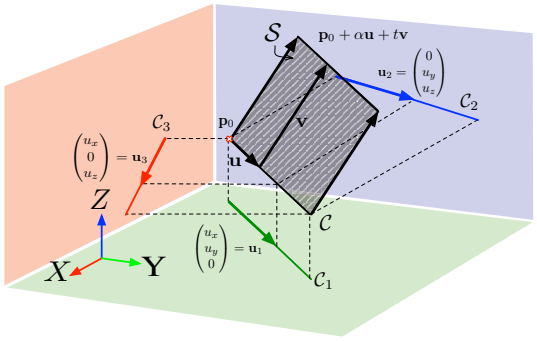


Figure 3.3: The local fitting of a plane to the point cloud allows approximating the plane tangent to the surface swept by an edge as it moves. If the velocity is constant, the so built surface is called ruled surface and the velocity vector $\mathbf{v}$ is its directrix. To estimate $\mathbf{v}$, it is, up to approximation errors, equivalent to work on the tangent plane.

Under the local plane hypothesis we previously made, it is possible to determine the rank of M. For that purpose, we assume the hypothesis that the edge $\mathcal{C}$ is a straight line segment defined by a point $\mathbf{p}_0$, a direction vector $\mathbf{u}$, and parametrized by a real α:

$$\mathbf{p} \in \mathcal{C} \Rightarrow \mathbf{p} = \mathbf{p}_0 + \alpha\mathbf{u}, \tag{3.8}$$

and the equation of $\mathcal{S}$ is changed into:

$$\mathcal{S}(\mathbf{p}, \mathbf{v}, t) = \mathcal{S}(\mathbf{p}_0 + \alpha\mathbf{u} + t\mathbf{v}) = 0. \tag{3.9}$$

Figure 3.3 depicts the case where $\mathcal{C}$ is a line and the resulting ruled surface $\mathcal{S}$, obtained by

sweeping lines in the direction of $\mathbf{v}$ is a plane. The vector $(u_i, u_j, 0)^T$ is by construction parallel to $\mathbf{\Pi}_k$, then:

$$\mathbf{n}_k^T \begin{pmatrix} u_i \\ u_j \\ 0 \end{pmatrix} = 0, \tag{3.10}$$

where $\mathbf{n}_k = (a_k, b_k, c_k)^T$ is the normal to $\mathbf{\Pi}_k$. The three similar equations for the three possible k lead to:

$$M\mathbf{u} = 0. \tag{3.11}$$

This shows $\mathbf{u}$ as an element of the kernel of M. $\mathbf{u}$ is not the null vector because $\mathcal{C}$ is not reduced to a point, thus M is non-invertible and the rank of M is not larger than 2. The rank deficiency of M means we only have two linearly independent scalar equations from Eq. 3.7, however we can still express two of the velocity components as functions of the last one, e.g. v_x:

$$\mathbf{v} = \begin{pmatrix} v_x \\ \frac{-a_1 v_x + c_1}{b_1} \\ \frac{-b_3 v_x + c_3}{a_3} \end{pmatrix} = v_x \underbrace{\begin{pmatrix} 1 \\ -\frac{a_1}{b_1} \\ -\frac{b_3}{a_3} \end{pmatrix}}_{\mathbf{q}} + \underbrace{\begin{pmatrix} 0 \\ \frac{c_1}{b_1} \\ \frac{c_3}{a_3} \end{pmatrix}}_{\mathbf{r}}, \tag{3.12}$$

where $M\mathbf{q} = \begin{pmatrix} 0 \\ \frac{\det(M)}{a_3 b_1} \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$.

This last equation shows that $\mathbf{q}$ is collinear to $\mathbf{u}$ if rank of M is two, hence we deduce from Eq. 3.7 that $M\mathbf{r} = (c_1, c_2, c_3)^T$.

Remark 1 *Expressing* $\mathbf{v}$ *as a one parameter vector fails if and only if the rank of* M *is less than 2 i.e. if edges do not generate planes. However some plane configurations*

require larger effort to achieve the closed form of $\mathbf{v}$ *such as the case where the plane is perpendicular to one of spatial frame axis. For example, when the* X*-axis is normal to the plane, Eq 3.12 is not valid as* b_1 *and* a_3 *are equal to zero. This problem can be solved by expressing* $\mathbf{v}$ *either as a function of* v_y *or* v_z*. In that case, we can see that* $v_x = -c_3/b_3$ *and* v_z *is a function of* v_y*. The problem of finding* $\mathbf{v}$ *is again reduced to the search for the correct value of one of its component.*

3.3 Velocity estimation

As shown in the previous section, from Eq. 3.12, the assumption of local constant velocity motion of straight edges allows to establish a simple linear relation between the velocity vector and the surface swept by the edge points. Estimating the velocity becomes equivalent to identifying the correct real value v_x. To achieve the estimation, we first define the point cloud within a spatio-temporal neighbourhood as a given structure. We then translate it according to vectors $\mathbf{v}$, parametrized by v_x. A matching operation is then performed for several sampled values of v_x. The correct v_x is the one producing the smallest matching error at the time and location given by the velocity vector (see Fig. 3.4).

3.3.1 Error cost function

A point cloud centred on $\mathbf{p}_1$ with luminance I is assumed to be locally non-deformable. Let us recall the luminance constancy constraint first introduced by Horn and Schunck in (Horn and Schunck, 1981), basis of most optical flow estimation methods, that expresses the invariance of I of structures across time. We can therefore state that when the considered point cloud translates from $\mathbf{p}_1$ to $\mathbf{p}_2$, both the local geometric structure and the luminance should be preserved. In that sense, we can formalize the structure

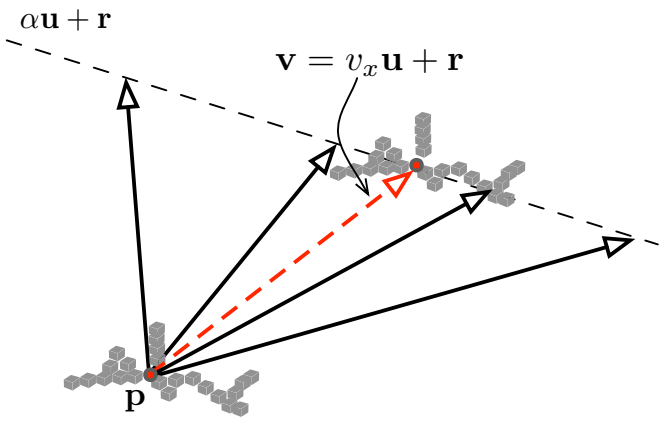


Figure 3.4: The velocity is to be determined locally along a line spanned by $\mathbf{u}$ and passing by $\mathbf{p}+\mathbf{r}$. This is achieved by matching local structure defined by a set of 3D points (gray cubes in the figure).

matching step as an energy minimization problem. Let us define the global energy term E as:

$$E = E_I + E_S, \tag{3.13}$$

where E_I is the luminance error which is minimal if the brightness constancy is satisfied and E_S is the geometric error which is minimal if the structure is not deformed when translating from $\mathbf{p}_1$ to $\mathbf{p}_2$. With the convention that $\mathbf{p}_i$ is the 3D point occurring at time t_i, we define the set $S(\mathbf{p}_i)$ as:

$$S(\mathbf{p}_i) = \{\mathbf{p}_j \in \mathbb{R}^3 |\ ||\mathbf{p}_j - \mathbf{p}_i|| \leq \Delta_s,\ t_j - t_i \leq \Delta_t \text{ and } t_j > t_i\}. \tag{3.14}$$

This set contains all 3D points spatio-temporally close to $\mathbf{p}_i$ i.e. points within a neighbourhood of $\mathbf{p}_i$ of radius Δ_s in space and length Δ_t in time. The energy cost associated to each sampled velocity vector for a given point $\mathbf{p}_0$ is thus computed as follows:

In short, E_I is the sum of the smallest luminance difference between all pairs of $(\mathbf{q}_i, \mathbf{p}_j)$ and E_S is the mean value of the smallest distances of each $\mathbf{p}_i$ to each $\mathbf{q}_j$. It

Algorithm 2 Energy cost computation

Require: $\mathbf{p}_0$, the set $S(\mathbf{p}_0)$, v_x.

1: Apply Eq.3.12 with the given v_x to build vector $\mathbf{v}$.
2: Define $S(\mathbf{p}_0) + \mathbf{v}$, the translated local structure $S(\mathbf{p}_0)$ by $\mathbf{v}$. Define $S(\mathbf{p}_0 + \mathbf{v})$ the set of points that occur in the neighbourhood of $\mathbf{p}_0 + \mathbf{v}$ at $t_i + dt$.
3: With the convention that $\mathbf{p}_i \in S(\mathbf{p}_0) + \mathbf{v}$, and $\mathbf{q}_j \in S(\mathbf{p}_0 + \mathbf{v})$, we compute the energy function $E(\mathbf{v}) = E_I + E_S$ with:

$$E_I = \sum_{i=1}^{n} \min_{\mathbf{q}_j} |I(\mathbf{p}_i) - I(\mathbf{q}_j)|, \tag{3.15}$$

and

$$E_S = \frac{1}{n} \sum_{i=1}^{n} \min_{\mathbf{q}_j} ||\mathbf{p}_i - \mathbf{q}_j||. \tag{3.16}$$

4: Return E.

is also called the mean closest point between both point clouds and is a dissimilarity measure largely used for example in the Iterative Closest Point (ICP) problem (Besl and McKay, 1992). The correct $\mathbf{v}$ is given by the value v_x which minimizes the energy function E:

$$\tilde{v}_x = \underset{v_x \in \mathbb{R}}{\operatorname{argmin}}\ E. \tag{3.17}$$

In order to minimize E with respect to v_x, a coarse to fine strategy is applied to sample possible values of v_x and match local 3D structure accordingly. Let $R = [R_1, R_r]$ be a real interval that is set large enough at the beginning of the search to make sure it contains $\tilde{v}_x$. To determine precisely $\tilde{v}_x$, R is subdivided into r equal length intervals and the centres of all intervals give a set of possible values for v_x. The error cost function is computed for each v_x and the interval producing the smallest E is used to update R (see Algorithm 3). This operation is iterated until E is below a preset (usually experimentally defined) threshold and after a minimum number of iterations. r is usually set to 5, however it can be larger. Estimation accuracy increases with r but at the cost of longer processing time.

3.3.2 Optimal spatio-temporal neighbourhood

The correct estimation of the velocity is conditioned by the spatio-temporal neighbourhood, defined as the spatio-temporal volume of dimensions $(\Delta_x \times \Delta_y \times \Delta_z \times \Delta_t)$, in which the 3D point cloud has moved from time t to $t + dt$. A large neighbourhood will allow to find the correct match, but at the cost of processing a large set of data, on the contrary, a too small one will not allow to match the local structures. The spatio-temporal neighbourhood must also be resized automatically and dynamically in accordance to the 3D points' velocity. In our implementation, we deal with the problem by adjusting a linear function on the neighbourhood size e.g. $\mathbf{s}_k = (\Delta_x, \Delta_y, \Delta_z, \Delta_t)^T$ is a linear combination of the m previous values $\mathbf{s}_{k-1}, ..., \mathbf{s}_{k-m}$:

$$\mathbf{s}_k = \sum_{i=1}^{m} a_i \mathbf{s}_{k-i}, \tag{3.18}$$

where the coefficients a_i are estimated with a standard linear prediction coding scheme (Durbin, 1959). The value of m is usually set to 5 according to experimental results while the initial value $\mathbf{s}_0$ is deduced from the rough estimation of the initial velocity i.e. the mean translation between the first two frames. Thus we have $\mathbf{s}_0 = (\mathbf{v}_0 dt, dt)^T$, assuming $\mathbf{v}_0$ is the initial estimate of the velocity.

The method for 3D flow extraction from point clouds is summarized in the algorithm 3.

Algorithm 3 3D flow algorithm

Require: Stream of 3D points cloud obtained from third-party device/algorithm

1: **for all** 3D point $\mathbf{p}$, at t **do**
2: Determine the spatio-temporal neighbourhood of 3D points close to $\mathbf{p}$.
3: Fit 3 planes $\mathbf{\Pi_1} = (a_1, b_1, c_1, d_1)^T, \mathbf{\Pi_2} = (a_2, b_2, c_2, d_2)^T, \mathbf{\Pi_3} = (a_3, b_3, c_3, d_3)^T$ using a least-square technique to minimize the three scalars:

$$|(p_x, p_y, t, 1)^T \mathbf{\Pi_1}| \ , \ |(p_x, p_z, t, 1)^T \mathbf{\Pi_2}| \ , \ |(p_y, p_z, t, 1)^T \mathbf{\Pi_3}|$$

4: Initialize a large enough interval $R = [R_1, R_r]$ of length L such that $\tilde{v}_x \in R$. Set n=1.
5: **while** $E(\mathbf{v}) >$ threshold **and** $n <$ max-iteration **do**
6: Divide R into r intervals R_k of size $\frac{L}{r}$ and define the set $\{v_k\}$ such that v_k is the center of R_k.
7: **for** each v_k **do**
8: Compute $E(v_k)$ according to Algorithm 2,
9: Update $L \leftarrow L(r-1)/r$,
10: Update R according to the R_k giving the lowest E such that: $R \leftarrow [R_k - \frac{L}{2}, R_k + \frac{L}{2}]$.
11: Compute $\mathbf{v}$:

$$\mathbf{v} = \left(v_k, -\frac{a_1 v_k + c_1}{b_1}, -\frac{b_3 v_k + c_3}{a_3}\right)^T$$

12: Update $n \leftarrow n + 1$.
13: **end for**
14: **end while**
15: **return v**
16: **end for**

3.4 Results

3.4.1 Simulated scene

The experiments are divided in two parts where we assess the estimation technique by means of ready-to-use 3D point clouds. The first set of experiments are performed on synthetic 3D structures moving at predefined velocity and trajectory. Estimated velocity amplitude and direction can be both compared to ground-truth data. Figure 3.5 is a case of smooth translation at constant amplitude for a wire cube. The velocity flow has

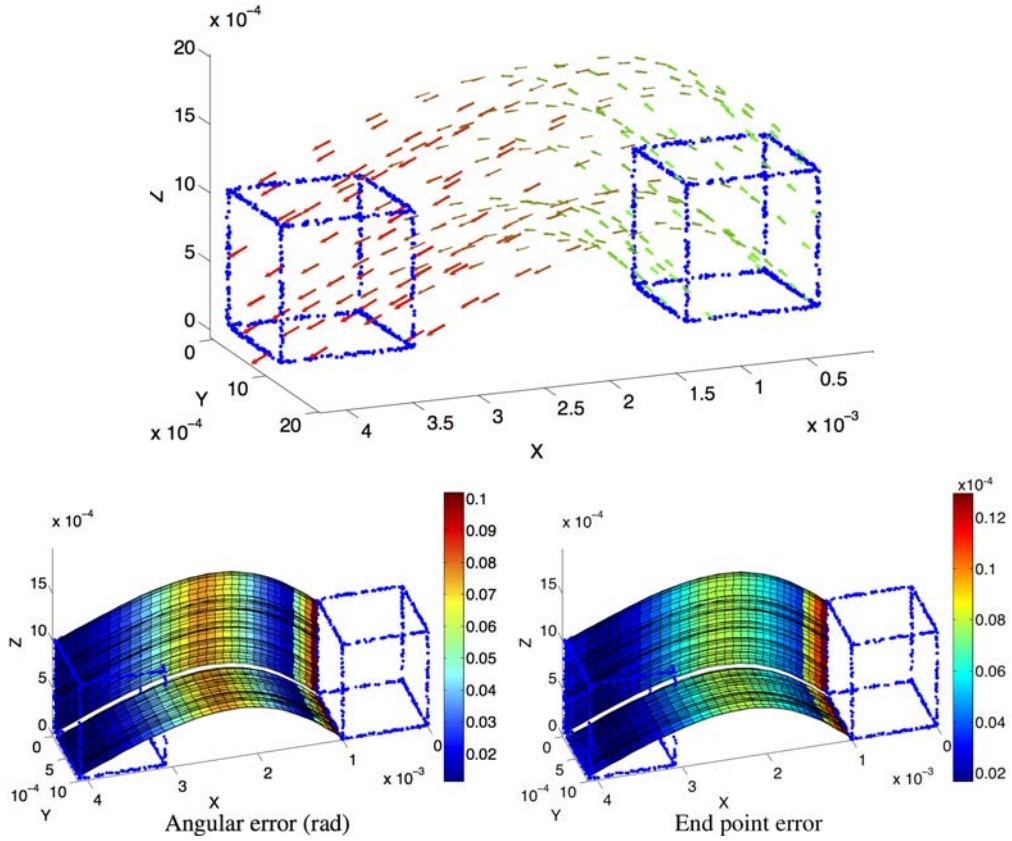


Figure 3.5: (top) Scene flow of a wire cube. The color codes the time as the cube moves from right to left. (center and bottom) Angular and end point errors of the estimated velocity field using local planes fitting for a curved trajectory. The patches of planes are underlined to show the locally constant velocity assumption. For visibility purpose, velocity is only shown for two edges. All axes are expressed in length unit except for the angle color scale.

been computed according to the presented technique with only the geometric structure information: only 3D points' positions and timestamps are given in the simulation. The energy cost function in Algorithm 2 is reduced to E_s. The flow performance is measured by two quantities commonly used in optical flow, the first one is the angular error, which is the angle defined by the estimated normalized velocity vector $\tilde{\mathbf{v}}$ and the ground-truth

$\mathbf{v}$. The angle is given by the inverse cosine of the scalar product of 2 vectors:

$$\arccos\left(\tilde{\mathbf{v}}^T\mathbf{v}/|\tilde{\mathbf{v}}||\mathbf{v}|\right). \tag{3.19}$$

This measure has been introduced by Fleet and Jepson in (Fleet and Jepson, 1990) and it assesses the accuracy of the flow direction. The second performance measure is the endpoint error which is proposed by Otte and Nagel in (Otte and Nagel, 1994) to not favour large displacements over smaller ones. This end point error is the norm of the difference between the estimated velocity and the real one:

$$|\mathbf{v} - \tilde{\mathbf{v}}|. \tag{3.20}$$

Both estimated angular error and end point error are represented with a color scaled representation (see Fig. 3.5). The maximal error occurs at the beginning of the motion and is due to the fitting spatio-temporal neighbourhood, chosen as the best compromise for the entire motion.

These results on synthetic data show the ability for the algorithm to estimate accurately the velocity vectors in a dense and smooth manner. The velocity is estimated with high accuracy since the direction has a maximal angular error of $0.1rad$ ($\sim 5.7°$) and the end point error's maximal value is equal to 1.2×10^{-5} length unit (i.e. at most 12% of the ground-truth value).

3.4.2 Natural scene

The second set of results is obtained from real scenes showing a person as a moving object in the scene. The 3D point clouds are provided by a Kinect sensor that also measures the RGB intensity. In these sequences, the person is a nice example of a deformable target with limbs moving at different non-constant velocities. However, as the results will show, the local constant speed hypothesis holds and is sufficient to allow a smooth

estimation of the scene flow. Scene flows estimations are given as two sets of results. One consists in using only geometric constraints, when the scene luminance is not available for the structure registration operation. The second set exploits the additional information brought by the luminance in addition to the geometry.

The flow estimation for each sequence is assessed in two ways:

1. A reference speed is established using the man's head to compute speed across frames. The head's position at time t is annotated manually to build a reference motion scene. This is then used as ground-truth to evaluate the plane fitting method.

2. If $S(t)$ designates an arbitrary point cloud in the scene at time t then $S(t) + \mathbf{v}dt$ is the morphing of $S(t)$ by translating it by $\mathbf{v}dt$. This morphed point cloud $S(t)+\mathbf{v}dt$ is compared to the corresponding point cloud data $S(t + dt)$. $S(t + dt)$ is obtained directly from the 3D point stream providing a ground-truth to measure the distortion in $S(t)+\mathbf{v}dt$. The shape dissimilarity error is measured using the mean closest point distance between $S(t) + \mathbf{v}dt$ and $S(t + dt)$.

In the first sequence, shown in Fig. 3.6(a), a person walks in front of the cameras at a constant pace. The velocities' amplitudes, and the directions are shown respectively at rows (b) and (d) for an estimation without using luminance information. Rows (c) and (e) are showing the results when luminance is included in the matching operation. For readability reasons, amplitude and direction of the flow are plotted in two separate representations. More importantly, these figures show how well the algorithm behaves in the presence of a deformable object. The limbs, in particular, the legs and the fingertips which are subject to the largest velocity changes show clear phases of acceleration: when the legs reach the end of the step, the speed is close to zero (1st and 3rd images) while it

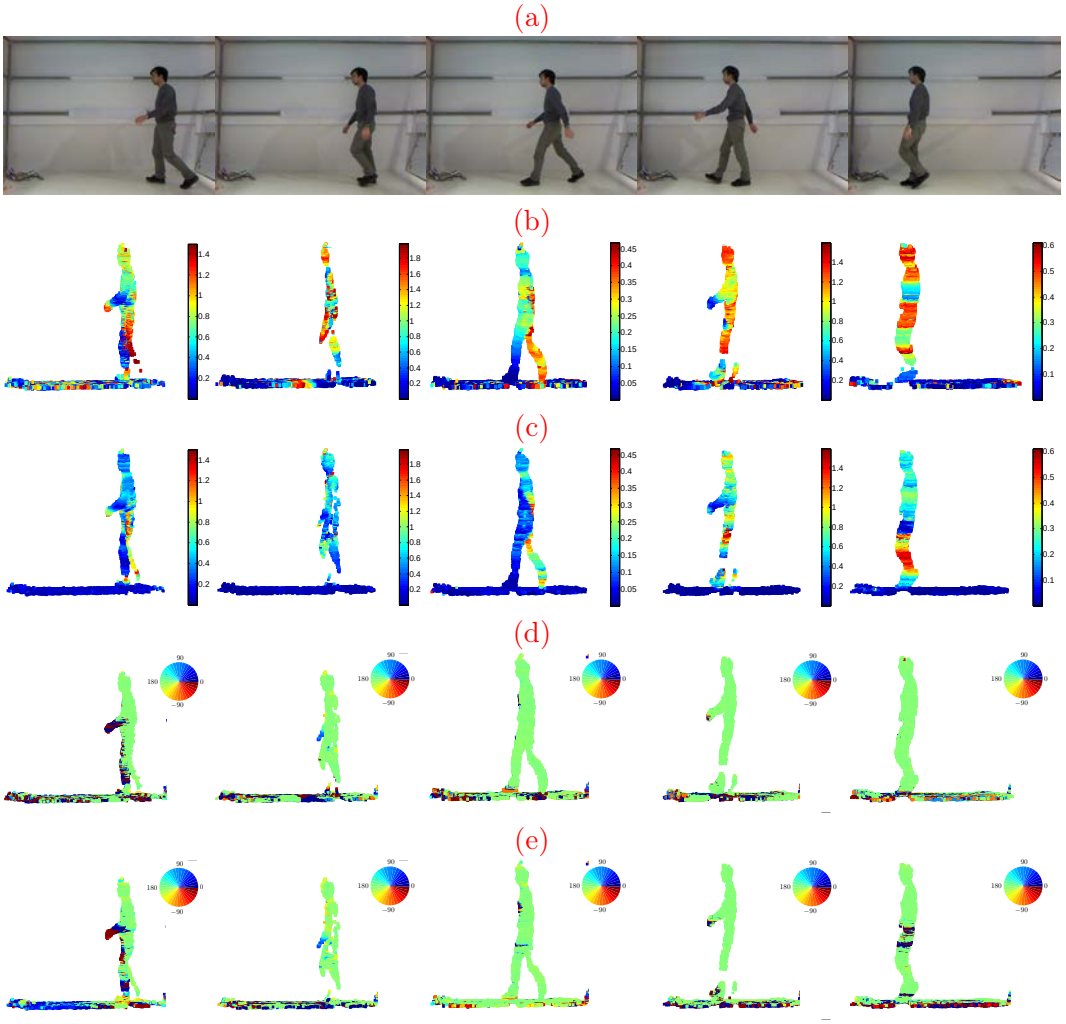


Figure 3.6: (a) Sequence of a person walking at constant speed across the scene. (b and c) The amplitude of each 3D point is color coded and shows that the plane fitting technique is able to estimate non rigid object velocity without (b) and with (c) luminance. (d and e) Colour coded flow directions. One can see the person going from the right to the left as confirmed by the color (green in the color scale i.e. an angle of 180^o). Again, we can observe that directions can be accurately estimated by using only time and geometry (d) and that the inclusion of the luminance is giving slight estimation improvements (e).

reaches a maximal value when the legs are in the middle of the step (5th image). These velocity changes are also visible in the color coded motion directions: the silhouettes are not all green as the hands swing. The floor, as it is scanned by the Kinect sensor, was also processed by the algorithm. The estimated speeds are largely coherent with what

it is expected: they are close to zero, thus negligible with respect to the moving person. The measured velocity variation (in amplitude and direction) from the floor are mainly due to several noise factors coming from the sensor itself, the lighting change induced by the motion, etc. One can also point out the absence of the velocity estimation on the wall due to the shadow. The reason is simply because the wall has been priorly removed before the scene flow estimation was applied.

In this experiment, the man walks across the scene, in front of the cameras at a constant speed of $1.1m/s$. This reference speed is measured by manually segmenting the head's point cloud for each frame. The speed is also extracted for the head from the estimated 3D flow with Algorithm 3. The top row of Fig. 3.8 shows both speed curves, plot together. Square markers represent the reference speed, circle markers show the speed estimated without luminance information while the diamond markers represent the result achieved with the luminance. The speed estimated from the geometric constraint has a mean value of $0.99m.s^{-1}$ and the one using luminance is around $1.2m.s^{-1}$. The relative mean difference between the two estimations is around 0.17%, thus this results show how both estimations are reasonably similar.

The small fluctuations of the estimated speed are not surprising as the trajectory of the head is not a straight translation: body weight transfer happens at each step and it modifies subsequently the head velocity in amplitude and direction. Finally, the color coded flow directions are particularly consistent. The figures show that the flow is pointing at 180°, i.e. from right to left for most of the body except for the person's hands. Floor's directions however have a random distribution but estimated velocities reflect the scanned 3D data accuracy more than the limit of the algorithm itself.

In the second sequence (see Fig. 3.7), a much more complex motion has been tested for the flow estimation. The person jumps in front of the camera and falls back on the ground. The velocity amplitude changes several times throughout the sequence: it increases at the beginning and reaches a maximum, then decreases to 0 when the person

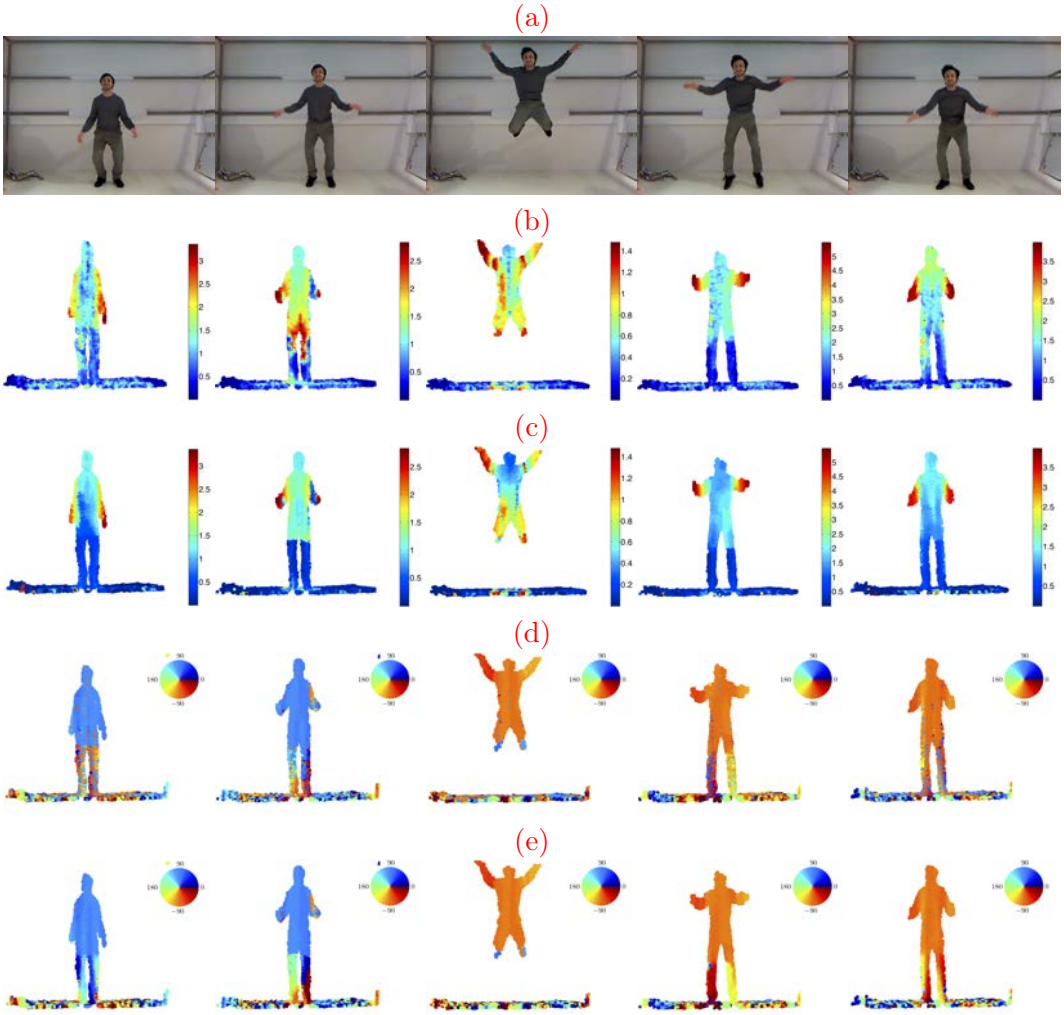


Figure 3.7: (a) Sequence of a jumping person. The target undergoes a complex motion which comprises several rapid changes of the velocity in direction and amplitude. (b and c) Colour-coded amplitude plot of the velocity for each 3D points. Parts of the body can be segmented according to the the speed e.g. the arms, the legs and rest of the body which have distinctive speed amplitude. (b) Amplitude of the velocity estimated from time and geometry only. (c) shows amplitude estimation when luminance is added. (d and e) The color coded flow direction (expressed in degree) are remarkably well estimated as we can see for the whole body, the direction is pointing up (i.e. angle close to 90^o) and pointing to the bottom when the person is falling (i.e. angle around -90^o).

is at the top of its trajectory. Then the amplitude increases again during the fall until he reaches the ground. This sequence of speed change can be seen at the bottom row of

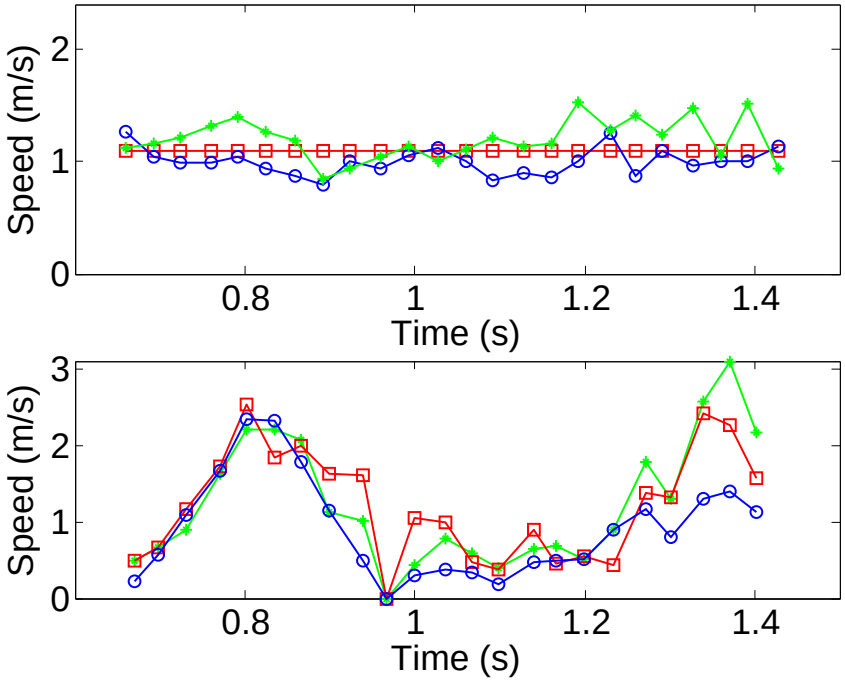


Figure 3.8: Mean velocity curves computed for the head in the walking (top) and the jumping (bottom) sequences. The circle curves are estimations achieved by time and geometric information only. The diamond curves are results one gets when luminance is used for the structure registration. Finally, the square curve represents the velocity of the manually segmented head's 3D points.

Fig. 3.8. Similarly to the walking sequence, both reference speed curves and estimation are shown together. However, in this experiment, it is more difficult to assess the accuracy of the estimation since the reference speed itself is built with a low accuracy. This is due to the difficulty to manually segment the head's 3D point since the speed changes too quickly.

The jumping sequence is an ideal example of a non-rigid body moving at a totally unconstrained speed. Here the amplitude and the direction curves show even more what we have already outlined for the walking sequence: the limbs and the body have their own velocity. The arms, in particular, show the largest velocity changes since the person swings them to gather momentum from the first half of the jump and he folds them back once the body begins to fall. In this sequence one can also observe the velocity estimated

for the floor which is again mostly equal to zero, except right under the jumping point because of the moving shadow of the person. The flow direction is even more interesting, as we can clearly see both the opposite directions: the velocities are pointing mainly up (i.e. angle of 90°) during the ascending phase and pointing down when he is falling (i.e. angle of −90°).

For both walking and jumping sequences, a slightly higher accuracy is achieved in estimating the velocity when luminance is used. Rows (c) and (e) of Figure 3.6 and 3.7 show smoother results when luminance is taking into account.

Morph error (%)				
	Walking		**Jumping**	
Frame	**Without I**	**With I**	**Without I**	**With I**
1	0.0542	0.0214	0.0250	0.0131
2	0.0695	0.0277	0.0635	0.0270
3	0.0786	0.0221	0.0698	0.0414
4	0.0863	0.0424	0.0510	0.0128
5	0.0433	0.0278	0.0481	0.0143
mean	**0.0664**	**0.0283**	**0.0515**	**0.0217**

Table 3.1: Morphed point cloud error.

The second performance assessment consists in measuring the morphing error and is summarized in Table 3.1 for the five frames shown in the sequences. The mean morphing error is below 3% for the walking sequence and slightly higher than 2% for the jumping one when luminance information is used. The estimation performance is slightly lower when the luminance is removed. In these cases, the morphing errors increase respectively to 5% and 7%. Two main observations should be retained from these results: first, morphed point clouds still consist of well defined objects. This shows computed motion is consistent for the full scene as morphing objects do not produce incoherent shapes. Secondly, the estimated scene flow is shown being consistent with the real motion since $S(t)+dt$, the morphed point cloud, matches correctly $S(t+dt)$, the point cloud at $t+dt$.

3.5 Discussion

Several aspects should be outlined from the work we present here. The first is the local regularization achieved by fitting a plane on the 3D points. The choice of a plane implies that the ruled surfaces are being swept by local straight edges. This hypothesis is not satisfied at edges' intersections where the edges have large curvatures. A better fitting strategy would be to extend the plane to a more general smooth surface. Spline curves parametrization is a good generic method to produce a smooth and accurate surface parametrization. Its main defect is the difficulty to properly set the sampling grid. Besides this requirement, the spline curves fitting off the 3D points is expected to produce better estimations since the tangent plane mentioned in section 3.2 is more accurately computed. By replacing the plane fitting by the spline curve fitting, the algorithm will be more expensive in computation. This issue requires a closer analysis.

A secondary aspect of the presented technique is the energy formulation we used for the shape registration operation. The idea was to cumulate both a geometric similarity, measured by the term E_S and a photometric similarity given by the term E_I. However sometimes only the geometric similarity can be estimated e.g. the 3D points measured by a laser range finder provide only geometric information and scene illuminance is not available. In that case, the similarity measure seems prone to ambiguities and allows only a limited mean in achieving shape registration. However if the points are sampled densely enough in time, we could see this technique managed to estimate the velocity with acceptable accuracy. We obtained such results both with the synthetic and natural data where only spatio-temporal information of the 3D points clouds were used. The use of more constraints brings more discriminating criterion in solving registration ambiguities. As it is experimentally shown, geometry and time carry sufficient cue for the velocity estimation. In contrast with most of vision based techniques found in literature,

luminance is no more a mandatory information.

Finally, a last concern may be raised in the case where stereovision rig is used instead of a 3D sensor. In that case, we can argue that it is possible to apply the similar scheme commonly used in scene flow estimation problem which consists in solving the velocity field and the 3D reconstruction problems simultaneously. In the worst case, we are designing a scene flow estimation technique based on local plane fitting which is expected to behave roughly similarly to the state-of-the-art techniques.

3.6 Conclusions

In computer vision, motion inference and 3D reconstruction from multiple cameras are usually coupled tasks. They are solved by stereovision mechanisms which require highly accurate calibration operation. Conventionally, the dense scene flow is estimated and refined from the dense optical flow which represents its projection on the focal plane in an iterative feedback loop scheme. The scene flow computation is therefore a complex problem which basically is an optimization problem under several conflicting constraints. However, with the widespread use of cheap vision sensors able to captured good quality 3D information, it is now possible to decouple the 3D reconstruction problem from the motion estimation itself. In that context, we developed a technique to estimate scene flows from 3D point clouds captured by such a depth sensor (e.g. the Kinect). The proposed technique is based on the local constant motion of the 3D point clouds and on their locally non deformable geometry. These hypotheses, when satisfied, tell us that an object moving in space, locally generates ruled surfaces from which the velocity vectors can be easily extracted. The solution we proposed is simple as it constraints the 3D velocity estimation to a search for a parametrization value over the set of real numbers. To achieve this search, we developed a local 3D structure matching strategy

consisting of using the geometric consistency and when it is available, luminance as an additional constraint to identify structures across time. Experimental results obtained from synthetic and natural images prove the technique to be particularly suitable in estimating the velocity vectors of deformable objects, undergoing arbitrary unconstrained motions. An important result from this method is its ability to accurately estimate 3D scene flow only from point clouds even if luminance information is missing. This approach allows flow estimation from data obtained by sensors that capture the spatio-temporal information but not the luminance. Some examples of such sensors are the range finders (e.g. the LiDAR) or the event-based stereo vision system which provide only 3D point clouds position in space and in time.
A second important result is its ability to provide a dense estimation of the velocity field as an alternative method to using a variational formulation (a very powerful but also highly time consuming technique) for flow estimation. Since the plane fitting we applied for the flow regularization is a computationally cheap operation, the resources are mainly consumed by the structure matching process. This is the main bottleneck of the introduced method that needs an efficient implementation optimization if we are aiming for real-time scene flow computation.

Beyond being applicable to any ready-to-use point cloud representation sequence obtained by systems such as RGBD sensors, the method is particularly interesting in asynchronous event-based 3D vision. Neuromorphic silicon retinas only encode the contrast changing parts of the scene which represent in general the moving objects. The algorithm proposed in the previous chapter outputs a stream of 3D events progressively reconstructed as they appear in the neuromorphic silicon retinas. Only moving points are reconstructed and thus point clouds only exist for structures which express motion. The advantage is clear, the computation of flow is only interesting for moving points and as the 3D event stream only contains moving points, the computation for static objects or background is avoided. Computation is also only performed when motion exists. Fur-

thermore as neuromorphic silicon retinas have high temporal resolution a more accurate and almost continuous 3D scene flow estimation can be recovered.

Chapter 4

It's (all) about time

> *Hofstadter's Law: It always takes longer than you expect, even when you take into account Hofstadter's Law*
>
> Douglas R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, 1979

4.1 Introduction

In chapter 3 we introduced a method to estimate scene flow from 3D point clouds. Our main motivation for estimating 3D motion was its application in refining the reconstructed structures from the method introduced in chapter 2. The idea was to improve reconstructions by using the 3D scene flow as a spatio-temporal predictor enforcing structure consistency across time. Although simple, the proposed method is computationally expensive mainly due to the structure matching step. However, Benosman et al (Benosman et al., 2013a) show a simple method for estimating the optical flow from neuromorphic vision sensors. Optical flow is by definition the projection of the 3D scene flow on

the image plane of the cameras. The paper shows how motion is intrinsically encoded in the local spatio-temporal activity of pixels. Authors show how to extract the optical flow from a simple plane fitting in a spatio-temporal region. We can therefore introduce less computationally expensive motion estimation in the 3D reconstruction method. We show how we can use motion consistency across cameras to further constrain the stereo matching method introduced in chapter 2.

Luminance, through photo-consistency, is also presented as an improvement to 3D asynchronous event-based reconstruction. In (Posch et al., 2011), the author introduces the ATIS, a novel neuromorphic silicon retina which encodes both contrast changes (in the same way as the DVS (Lichtsteiner et al., 2008)) and luminance information as the temporal difference of two exposure measurement events. We assume the scene to be lambertian, this allows us to use the luminance consistency between sensors as an additional constraint to the stereo correspondence problem. As luminance is encoded by time differences, we define a photo-consistency temporal constraint in order to improve the formulation presented in chapter 2.

In this chapter, we introduce two temporal constraints motion and luminance consistency, which can be used to further improve the asynchronous event-based stereo matching. We explore luminance and motion alone and their combination to solve stereo matching ambiguities and decrease the number of false matches thus producing more accurate 3D reconstructions. We define luminance and motion as functions of time and formulate the asynchronous event-based 3D reconstruction method as a minimization problem which is dependent on the single variable t.

To our knowledge this is the first event-based 3D reconstruction method able to recover 3D models of complex shapes. The use of gray-level information allows for the first time to produce textured 3D reconstructions from event-based data.

We present a method to solve 3D reconstruction in a modular approach where selected constraints are used in a minimization function. We first perform stereo matching

purely based on epipolar geometry and time information as introduced in chapter 2. Cost for conflicting candidates can be obtained from existing luminance and/or motion consistency. We then solve it as an energy minimization function that penalizes geometric, temporal, luminance and motion errors. Results show reconstruction errors decrease by 50% from temporal-geometry minimization alone while decreasing noise as well.

4.2 Intensity and motion based stereo matching

Photo-consistency is a typically enforced constraint when matching corresponding pixels. It consists of evaluating how well pixels in one image match another set of pixels in a second image using luminance. A matching cost between possible correspondences is given by a function which evaluates the similarity between regions around the pixels. The most commonly used functions include squared intensity differences, absolute intensity differences or normalized cross correlation. However, other more robust techniques which limit the influence of mismatches have been proposed, examples are the truncated quadratics or contaminated Gaussians. Reader may refer to (Scharstein and Szeliski, 2002) for more details on photo-consistency pixel matching methods.

Motion has also been used to solve depth estimation. Different approaches to solving stereo correspondence problem using optical flow have been proposed: (Hatzitheodorou et al., 2000) proposes computing optical flow between different views and use it to determine corresponding points between images. (Kunii and Chikatsu, 2000) uses the optical flow from the sequence of frames to track lines on each view. Lines are then matched and 3D respective coordinates obtained. In (Nasrabadi et al., 1989) authors propose combining optical flow and intensity information in an energy function modelled using Markov random fields to solve stereo correspondence. A detailed description of several methods using motion as a cue for depth estimation can be found in (Lee et al., 2012c).

4.3 Time encoded imaging

Biomimetic, event-based cameras are a novel type of vision devices that - like their biological role models - are driven by "events" happening within the scene, and not like conventional image sensors by artificially created timing and control signals (e.g. frame clock) that have no relation whatsoever to the source of the visual information (Delbruck et al., 2010). Over the past few years, a variety of these event-based devices have been implemented, including temporal contrast vision sensors that are sensitive to relative light intensity change, gradient-based sensors sensitive to static edges, edge-orientation sensitive devices and optical-flow sensors. Most of these vision sensors output visual information about the scene in the form of asynchronous address events (AER) (Boahen, 2000) and encode the visual information in the dimension of time and not as voltage, charge or current. The presented pattern tracking method is designed to work on the data delivered by such a time-encoding sensors and takes full advantage of the superior characteristics, most importantly the ultra-high temporal resolution and the sparse data representation.

The ATIS ("Asynchronous Time-based Image Sensor") used in this work is a time-domain encoding image sensors with QVGA resolution. (Posch et al., 2011)(Posch et al., 2008). The sensor contains an array of fully autonomous pixels that combine an illuminance change detector circuit and a conditional exposure measurement block.

As shown in the functional diagram of the ATIS pixel in Fig. 4.1, the change detector individually and asynchronously initiates the measurement of an exposure/gray scale value only if - and immediately after - a brightness change of a certain magnitude has been detected in the field-of-view of the respective pixel. The exposure measurement circuit in each pixel individually encodes the absolute instantaneous pixel illuminance into the timing of asynchronous event pulses, more precisely into inter-event intervals.

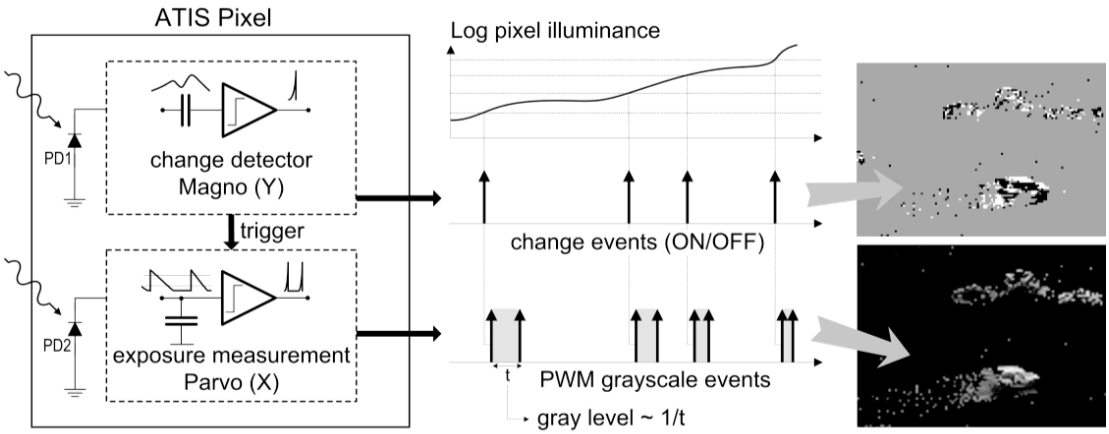


Figure 4.1: Functional diagram of an ATIS pixel (Posch et al., 2011). Two types of asynchronous events, encoding change and brightness information, are generated and transmitted individually by each pixel in the imaging array.

Since the ATIS is not clocked like conventional cameras, the timing of events can be conveyed with a very accurate temporal resolution at the order of microseconds. The time-domain encoding of the intensity information automatically optimizes the exposure time separately for each pixel instead of imposing a fixed integration time for the entire array, resulting in an exceptionally high dynamic range and improved signal to noise ratio. The pixel-individual change detector driven operation yields almost ideal temporal redundancy suppression, resulting in a maximally sparse encoding of the image data.

Figure 4.2 shows the general principle of asynchronous imaging spaces. Frames are absent from this acquisition process. They can however be reconstructed, when needed, at frequencies limited only by the temporal resolution of the pixel circuits (up to hundreds of kiloframes per second) (see Fig. 4.2 top). Static objects and background information, if required, can be recorded as a snapshot at the start of an acquisition henceforward moving objects in the visual scene describe a spatio-temporal surface at very high temporal resolution (see Fig. 4.2 bottom).

Let us consider the output of a neuromorphic vision sensor such as the ATIS (Posch et al., 2011). Visual information is encoded as a stream of events where each event is

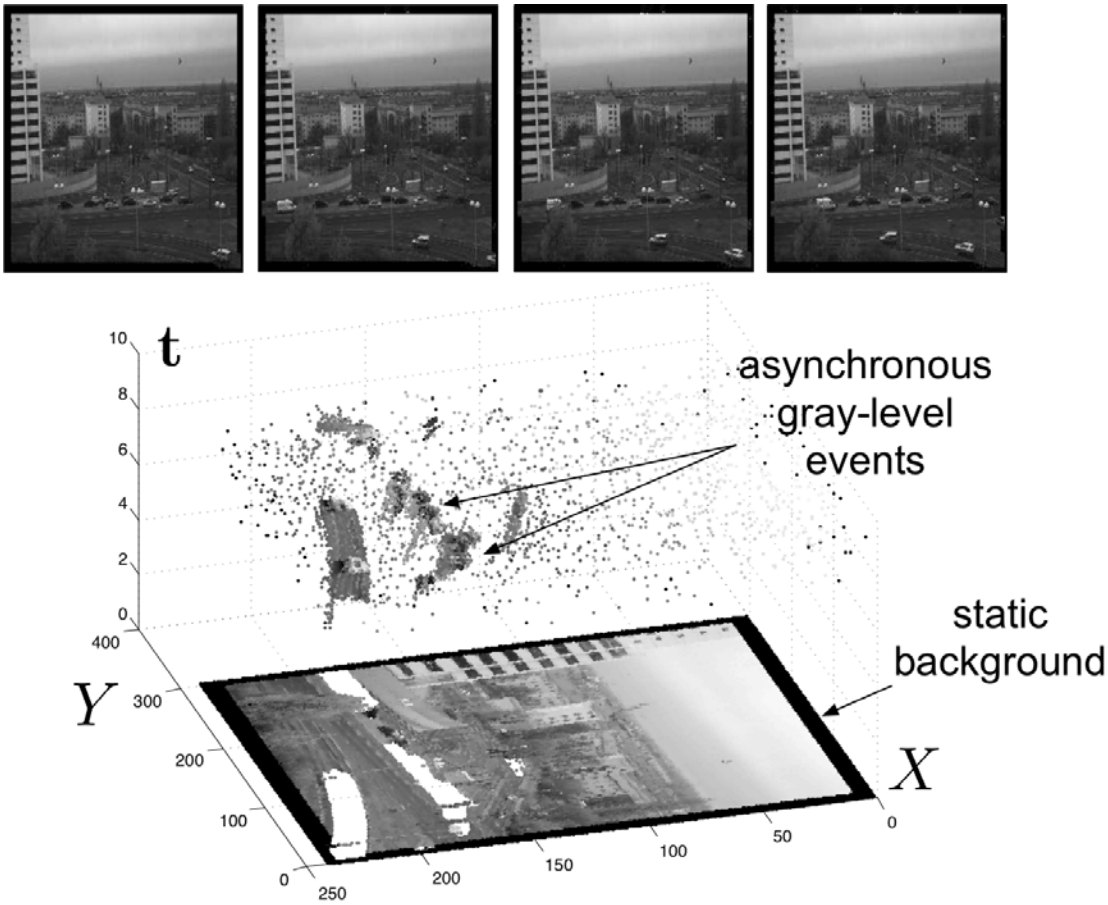


Figure 4.2: (Lower part) The spatio-temporal space of imaging events: Static objects and scene background are acquired first. Then, dynamic objects trigger pixel-individual, asynchronous gray-level events after each change. Frames are absent from this acquisition process. Samples of generated images from the presented spatio-temporal space are shown in the upper part of the figure.

defined by its (x,y) image coordinates, time, polarity and luminance information. Events provide a two-dimensional representation of the perceived luminance variation of the scene on the sensor's image plane. A light intensity variation at a given 3D point $\mathbf{X}$ will be projected on to the image plane $\mathcal{R}_u$ of sensor u at position $\mathbf{p}$ according to:

$$\begin{pmatrix} \mathbf{p}^u \\ 1 \end{pmatrix} = P_u \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix}. \tag{4.1}$$

If we ignore the existing latency between the stimulus and sensor's response time then the stimulus ($X(t)$) at time t produces an event $e(\mathbf{p}^u, t)$ in retina $\mathcal{R}_u$. Latency is variable and dependent on several environmental aspects such as scene's illumination, sensor's bias settings or induced contrast change. Reader may refer to (Rogister et al., 2011) and chapter 2 for more information on the neuromorphic vision sensor's latency and particularly its implications in event-time based stereo matching. Furthermore, if several pixels are simultaneously activated, arbitration will output events with random order and variable timestamp delay thus introducing jitter between cameras. If sensor u perceives a change of the 3D scene point $\mathbf{X}(t)$, an event $e(\mathbf{p}^u, t)$ at t is produced. The event $e(\mathbf{p}^u, t)$ characterizes the nature of the stimulus in terms of:

- polarity - representing the direction of the induced luminance change from its previous value. Events can therefore assume a single value 1 or -1 if they represent respectively an increase or decrease in luminance change.

- luminance - corresponding to the gray-level perceived by the retina at pixel $\mathbf{p}$. As described in (Posch et al., 2011), ATIS sensors encode luminance information in terms of exposure time. A contrast change triggers the luminance measurement generating events at the beginning and at the end of the measurement. The difference between these exposure events gives a duration which is inversely proportional to the absolute luminance of the scene.

An event $e(\mathbf{p}^u, t)$ can be defined as

$$e(\mathbf{p}^u, t) = \begin{cases} \mathbf{p}^u = (x, y)^T \\ pol = sign(I(e(\mathbf{p}^u, t-1)) - I(e(\mathbf{p}^u, t))) \\ I(e(\mathbf{p}^u, t)) = t_{e^+} - t_{e^-} \end{cases} \tag{4.2}$$

where $(x, y)^T$ is the sensor coordinate where the event occurred and *pol* is the polarity

representing the direction of the contrast change assuming values -1 or 1. $I(e(\mathbf{p}^u, t))$ represents the inverse luminance of pixel $\mathbf{p}^u$ at time t and is encoded as the time difference between exposure measurement events $I(e(\mathbf{p}^u, t)) = t_{e^+} - t_{e^-}$, where t_{e^-} represents the starting and t_{e^+} the finishing timestamp of the integration.

4.4 Event-based stereo matching

We develop our approach using the spatio-temporal stereo correspondence method formulated in chapter 2 and illustrated in Fig. 4.3.

4.4.1 Geometrical error

The matching of events between cameras is based on the combination of the epipolar geometry and temporal matching exploring the sparsity and precise timing of events provided by the sensors. Lets consider F_{uv} as the fundamental matrix that maps events between cameras u and v, $\mathbf{l}^v(\mathbf{p}^u)$ is the epipolar line on the image plane $\mathcal{R}_v$ defined as:

$$\begin{array}{rcl} \mathbb{R}^2 & \rightarrow & \mathbb{R}^2 \\ \mathbf{p}^u & \mapsto & \mathbf{l}^{uv}(\mathbf{p}^u) = F_{uv} \begin{pmatrix} \mathbf{p}^u \\ 1 \end{pmatrix} \end{array} \tag{4.3}$$

.

m the triple of matching events produced by three sensors is geometrically defined as the triple $\{e^1 = e(\mathbf{p}^u, t^u), e^2 = e(\mathbf{p}^v, t^v), e^3 = e(\mathbf{p}^w, t^w)\}$, that represent points lying on the intersection of the epipolar lines on respectively image planes $\mathcal{R}_u$, $\mathcal{R}_v$ and $\mathcal{R}_w$, such

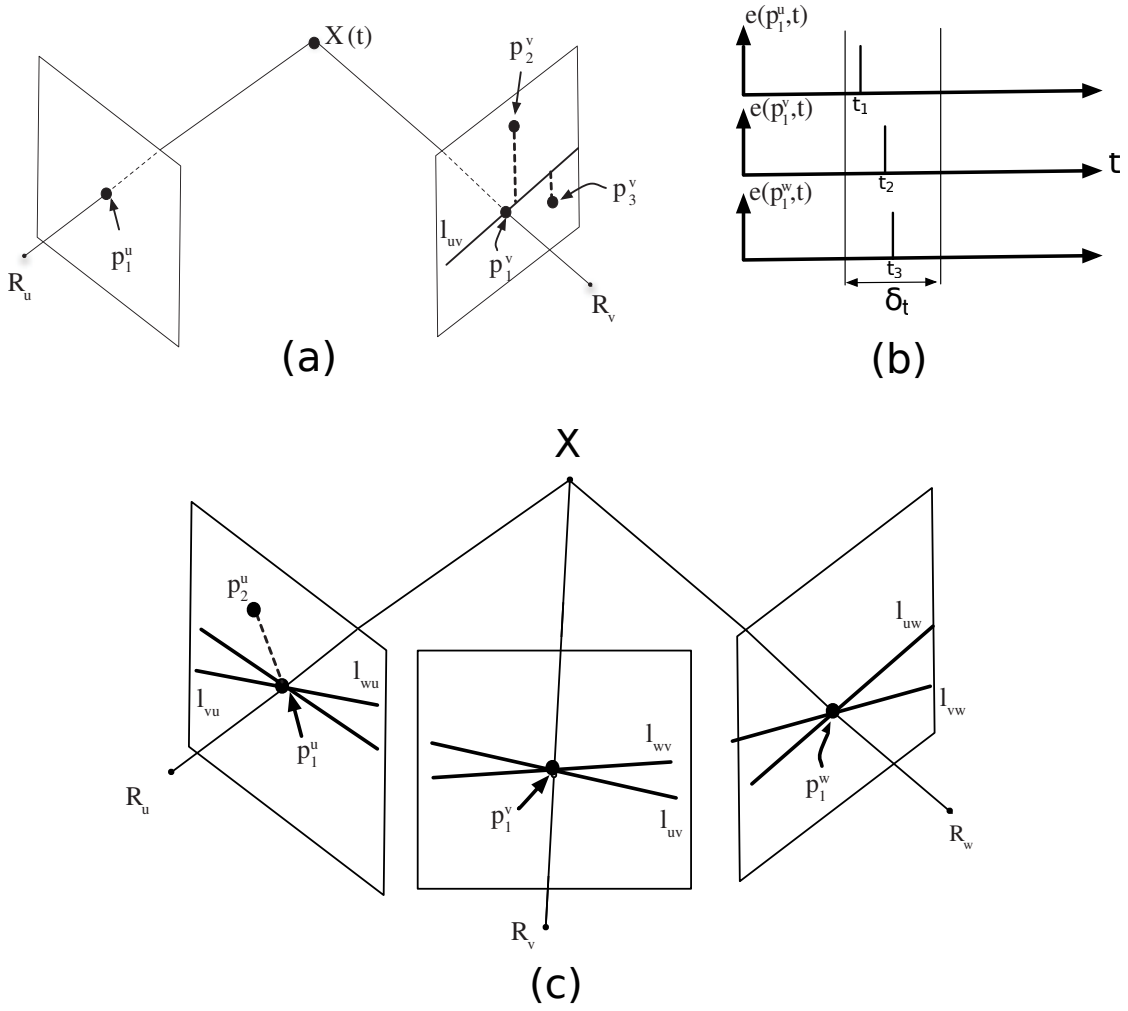


Figure 4.3: (a,c) Epipolar illustrated for two and three cameras. A 3D point $\mathbf{X}$ is projected onto the focal planes as $\mathbf{p}_1^u$, $\mathbf{p}_1^v$ and $\mathbf{p}_1^w$. Each of them is close to the epipolar line (binocular case) or the intersection of two epipolar lines (trinocular case) defined by the geometric configuration. (b) Events generated by $\mathbf{X}$ in each camera at time t are usually not recorded with the same date t, but rather different timestamps t_1, t_2, etc. due to a finite precision in synchronizing the cameras.

that

$$\forall m(e^1, e^2, e^3), \begin{cases} \hat{\mathbf{p}}^u = \mathbf{l}^{vu}(\mathbf{p}^v) \cap \mathbf{l}^{wu}(\mathbf{p}^w) \\ \hat{\mathbf{p}}^v = \mathbf{l}^{uv}(\mathbf{p}^u) \cap \mathbf{l}^{wv}(\mathbf{p}^w) \\ \hat{\mathbf{p}}^w = \mathbf{l}^{uw}(\mathbf{p}^u) \cap \mathbf{l}^{vw}(\mathbf{p}^v) \end{cases} \tag{4.4}$$

where due to geometrical errors $\hat{\mathbf{p}}^i \approx \mathbf{p}^i$ with $i \in \{u, v, w\}$.

The geometrical error for a given match is given as the average distance between the intersection of epipolar lines and the matched point at each retina and it reflects how well a match respects the epipolar constraints:

$$E_G(m) = \frac{1}{3\epsilon_g} \sum_{i \in \{u,v,w\}} ||\mathbf{p}^i - \hat{\mathbf{p}}^i|| \tag{4.5}$$

ϵ_g is a normalizing scalar which represents the maximum allowed geometric distance. This maximum allowed distance defines which events are considered as potential candidates and therefore if $||\mathbf{p}^i - \hat{\mathbf{p}}^i|| > \epsilon_g$, the match is discarded automatically. When the binocular method is used, the geometrical error is given from the distance from candidate points to epipolar lines such and therefore in this case:

$$E_G(m) = \frac{d(\mathbf{p}^v, \mathbf{l}^{uv}(\mathbf{p}^u)) + d(\mathbf{p}^u, \mathbf{l}^{vu}(\mathbf{p}^v))}{2\epsilon_g} \tag{4.6}$$

$d(\mathbf{p}, \mathbf{l})$ is the perpendicular distance from point $\mathbf{p}$ to the epipolar line $\mathbf{l}$.

4.4.2 Temporal error

On the time domain, matching is achieved by identifying events which occur at the same time on all sensors. If a given stimulus $\mathbf{X}(t)$ is detected by sensors u, v and w, events $e(\mathbf{p}^u, t^u)$, $e(\mathbf{p}^u, t^v)$ and $e(\mathbf{p}^u, t^w)$ will be generated but due to variable sensor latency, $t \neq t^u \neq t^v \neq t^w$. However, we can define matching events as the ones generated at the closest temporal distance by minimizing the temporal matching error

$$E_T(m) = \frac{\sum_{i \in \{v,w\}} |t^u - t^i|}{2\epsilon_t} \tag{4.7}$$

where ϵ_t is a normalizing scalar which represents the maximum temporal distance error. Similarly, in the binocular case we have:

$$E_T(m) = \frac{|t^u - t^v|}{\epsilon_t} \tag{4.8}$$

4.4.3 Time-coded intensity matching

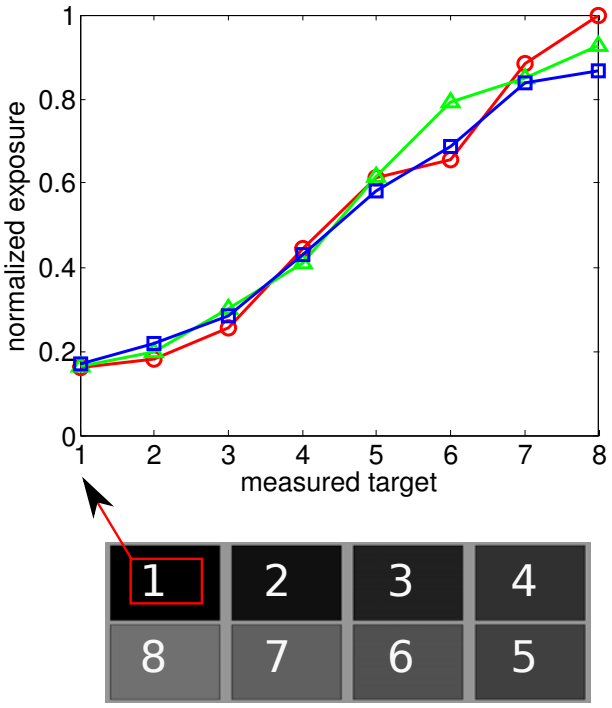


Figure 4.4: Exposure measurement of different gray level targets by three sensors represented respectively in red, green and blue.

We previously defined $I(e(\mathbf{p}^u, t))$ as the inverse luminance value of pixel $\mathbf{p}^u$ when the measurement is triggered by $e(\mathbf{p}^u, t)$. At any given time t, $I(\mathbf{p}^u, t)$ gives the value of the last luminance measurement of pixel $\mathbf{p}^u = (x, y)^T$.

Lets take the assumption that scenes are composed by lambertian surfaces. Luminance does not change with the view angle and all corresponding pixels have the same intensity. If a 3D point $\mathbf{X}(t)$ generates events $e^u(\mathbf{p}^u, t)$, $e^v(\mathbf{p}^v, t)$ and $e^w(\mathbf{p}^w, t)$ respectively

on $\mathcal{R}_u$, $\mathcal{R}_v$ and $\mathcal{R}_w$, we can assume that $I(\mathbf{p}^u, t) = I(\mathbf{p}^v, t) = I(\mathbf{p}^w, t)$. Photo-consistency allows us to further constrain the stereo matching method by assigning a matching score according to the luminance disparity between corresponding pixels. Figure 4.4 shows three ATIS sensors' exposure measurement for eight different gray levels. Values are obtained as the average measurement over each level and show that measurements are consistent across cameras and may be used as a matching constraint.

The ATIS sensor provides luminance measurements for pixels where changes were

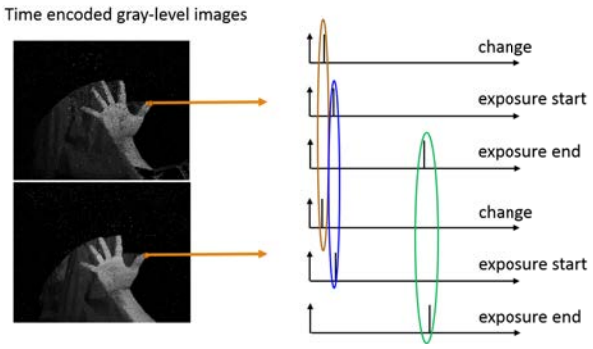


Figure 4.5: Matching of two events using the time-coded intensity. Matching is achieved throuh the temporal coincidence of the three generated timestamps (corresponding to change and exposure measurements events).

detected. This luminance is given as a pair of events representing beginning and ending of exposure measurement. We extend the even-based matching method to include luminance information and ensure temporal consistency of both change and exposure measurement events. Matching a given event consists therefore in matching 3 coincident events as shown in Fig. 4.5.

As briefly explained in section 4.2, intensity based pixel matching is achieved by operating a matching cost function over a neighbourhood support region. Aggregation and matching cost functions have been largely studied for decades and several have been proposed with different advantages. Reader may refer to (Tombari et al., 2008) and (Scharstein and Szeliski, 2002) for more information and evaluations.

We use the normalized cross correlation (NCC) as similarity measure operating it over luminance information of the support regions of matching pixels. Although slower and more complex than other functions such as the sum of absolute differences, NCC has been chosen experimentally as it produced good results. For simplicity we chose a fixed square support region of size $\delta_s \times \delta_s$.

If $A(\mathbf{p}^u, t)$ is the array of luminance values for pixels which are part of the support region around $\mathbf{p}^u$, we can define it as:

$$A(\mathbf{p}^u, t) = \{I(\mathbf{p}^u_{i,j}, t) | \mathbf{p}^u_{i,j} = (x+i, y+j)^T, |i-x| \leq \delta_s, |j-y| \leq \delta_s\}. \tag{4.9}$$

We can define an energy cost function expressing how well pixels are correlated as:

$$E_I(m) = 1 - \frac{1}{2} \sum_{c \in \{v,w\}} \frac{\sum_i \sum_j ((I(p^u_{ij}, t) - \bar{A}(\mathbf{p}^u, t))(I(p^c_{ij}, t) - \bar{A}(\mathbf{p}^c, t))}{\sqrt{(\sum_i \sum_j (I(p^u_{ij}, t) - \bar{A}(\mathbf{p}^u, t)^2)(\sum_i \sum_j (I(p^c_{ij}, t) - \bar{A}(\mathbf{p}^c, t)^2)}} \tag{4.10}$$

where

$$\bar{A}(\mathbf{p}^u, t) = \frac{1}{\delta_s^2} \sum_i \sum_j I(\mathbf{p}^u_{ij}, t) \tag{4.11}$$

with $\mathbf{p}_{ij}$ neighbourhood pixels of $\mathbf{p}^u$ such that $\mathbf{p} = (x, y)^T$, $|i-x| \leq \frac{\delta_s}{2}$ and $|j-y| \leq \frac{\delta_s}{2}$. $I(\mathbf{p^u_{i,j}})$ is the last known intensity value of pixel $\mathbf{p} = (x+i, y+j)^T$ in camera u. $\bar{A}(\mathbf{p}^u, t)$ represents the mean luminance of the support region. The error is given as the average correlation of support regions in pairs of cameras and gives values in $0 \leq E_I(m) \leq 2$.

Figure 4.6 shows the matching principle used in the photo-consistency constraint. Support regions around matching events are correlated in order to evaluate how well events match. Two matching regions are underlined representing an example which produces high correlation value and low luminance disparity error.

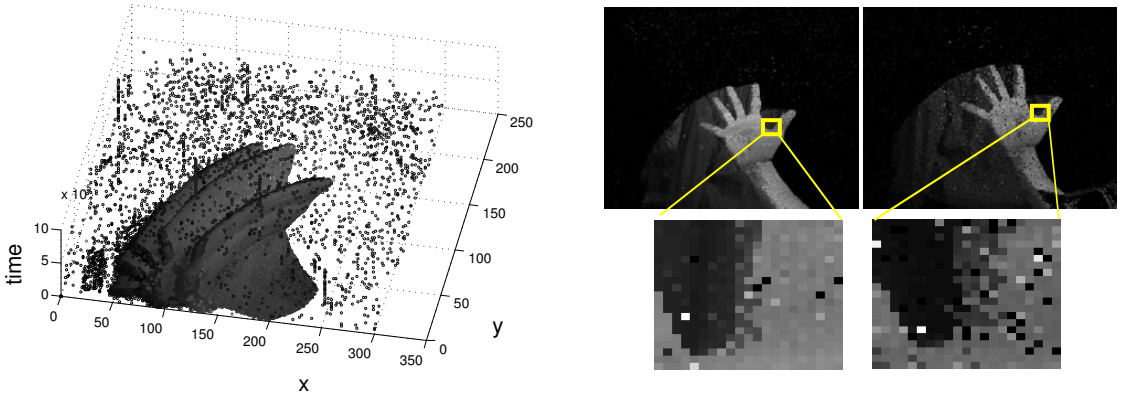


Figure 4.6: Luminance based matching between two sensors. Left image shows luminance information of waving hand across time. Right image shows luminance of the support regions of matching pixels in two sensors.

4.4.4 Motion matching

Motion information is directly encoded by the time of occurrence of events. In (Benosman et al., 2013a) authors propose an event-based visual flow estimation method where optical flow is estimated directly from the time of occurrence of events. $\sum_e$ is defined as the function that maps to $\mathbf{p}$ the time t:

$$\begin{aligned} \textstyle\sum_e : \mathbb{N} &\rightarrow \mathbb{R} \\ \mathbf{p} &\rightarrow \textstyle\sum_e(\mathbf{p}) = t \end{aligned} \tag{4.12}$$

and authors show its gradient $\nabla \sum_e(\mathbf{p})$ is the vector that measures the rate and direction of change of time with respect to space and is related to velocity vectors as:

$$\nabla \textstyle\sum_e(\mathbf{p}) = (\frac{1}{\mathbf{v}_x}, \frac{1}{\mathbf{v}_y})^T. \tag{4.13}$$

Evaluating the occurrence of events for each pixel provides a way to recover motion information. Figure 4.7 shows the temporal information for events generated by a waving hand showing the relation of motion and time. Events occurring at the same time show similar color. We assume that if two sensors are close to each other, a moving object

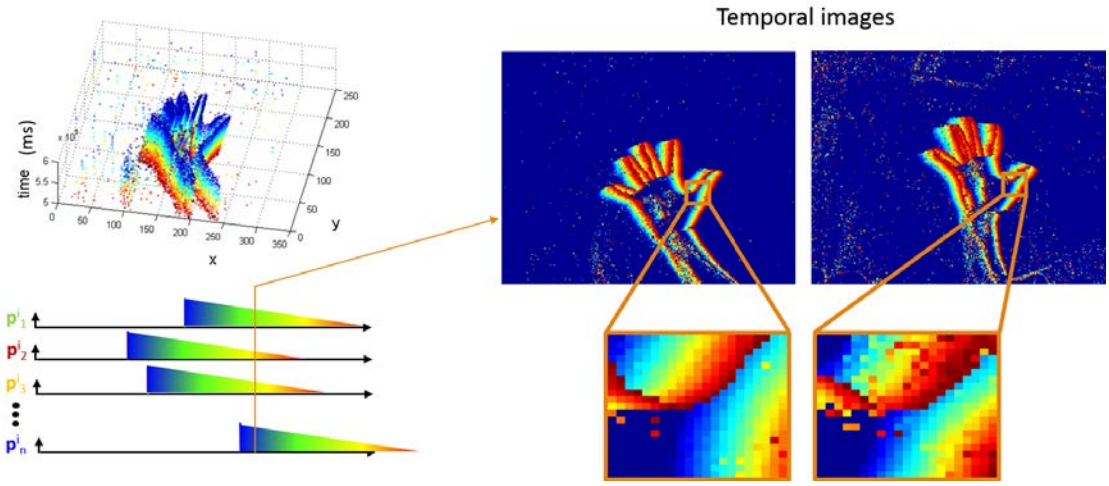


Figure 4.7: Motion matching principle of a waving hand. Figures represent images of time at a given instant in time, obtained by the pixel activity at small temporal regions on two cameras.

will generate similar optical flows in respective focal planes. This is not valid for all scenarios, namely objects located in the center and moving perpendicularly to the stereo rig produce uncorrelated optical flows. However, when verified it provides a powerful method to select matches even when luminance information is not available (e.g. when using the DVS sensor).

Temporal activity region provides a spatio-temporal feature for each event as it encodes spatial direction and rate of the temporal change. Figure 4.7 shows two matching regions of a waving hand. Let us define the spatio-temporal region $A(\sum_e(\mathbf{p}^u))$ of size $\delta_s \times \delta_s$ around $\sum_e(\mathbf{p}^u)$:

$$A(\textstyle\sum_e(\mathbf{p}^u)) = \{\sum_e(\mathbf{p}^u_{i,j})|\mathbf{p}^u_{i,j} = (x+i, y+j)^T, |i-x| \leq \delta_s, |j-y| \leq \delta_s\}. \quad (4.14)$$

We verify motion consistency between matched events by correlating their corresponding surrounding regions. An energy cost function can be defined imposing penalties on

motion disparity such that:

$$E_M(m) = 1 - \frac{1}{2} \sum_{c \in \{v,w\}} \frac{\sum_i \sum_j ((T(p_{ij}^u) - \bar{A}(T(\mathbf{p}^u))))(T(p_{ij}^c) - \bar{A}(T(\mathbf{p}^u))))}{\sqrt{(\sum_i \sum_j (T(p_{ij}^u) - \bar{A}(T(\mathbf{p}^u)))^2)(\sum_i \sum_j (T(p_{ij}^c) - \bar{A}(T(\mathbf{p}^u)))^2)}} \tag{4.15}$$

for visibility $\sum_e(\mathbf{p}^u) = T(\mathbf{p}^u)$ and

$$\bar{A}(T(\mathbf{p}^u)))) = \frac{1}{\delta_s^2} \sum_i \sum_j T(\mathbf{p}_{ij}^u) \tag{4.16}$$

with p_{ij} neighbourhood pixels of $\mathbf{p}^u$ such that $\mathbf{p} = (x, y)$, $|i-x| \leq \delta_s$ and $|j-y| \leq \delta_s$. E_M reflects how well points match based on their motion consistency. Motion flow defined as the local activity of pixels in a given spatio-temporal region provides a spatio-temporal which allows better characterizing individual events. Matching temporal activity neighbourhoods ensures temporal consistency and intrinsically motion consistency as well.

4.4.5 Error minimization

We presented four independent cost functions (equations 4.5,4.7,4.10,4.15) which can be seen as individual modules for evaluating stereo matches. Each function expresses the matching cost by penalizing errors on its respective constraint geometry (E_G), time (E_T), luminance (E_I) or motion (E_M). Geometry and time costs are normalized to give values between 0, best match, and 1, maximum error. Luminance and motion cost functions give values between 0, maximum correlation of support regions and best match, and 2, completely anti-correlated regions and therefore maximum error.

We can now present the asynchronous event-based stereo matching method as a modular energy cost function composed by any combination of the four cost functions. The basic asynchronous event-based method proposed in chapter 2 (or (Rogister et al., 2011) for the binocular case) based on the spatio-temporal errors can therefore be formulated

as an energy cost function composed by the normalized geometrical and temporal errors. Events are selected among matching candidates as the n-tuples minimizing the temporal-geometrical cost function:

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \left(E_G(m_i) + E_T(m_i)\right) \tag{4.17}$$

where $\mathcal{M}$ is the set containing all match candidates to a given event e^u.

This modular approach allows us to complete the previous formulation with combinations of motion and when available luminance matching cost. An energy cost function with all four constraints is given by:

$$E(m) = E_G(m) + E_T(m) + E_M(m) + E_I(m). \tag{4.18}$$

It is important to notice that while geometrical and temporal costs vary between 0 and 1, motion and intensity costs assume values between 0 and 2. Costs larger than 1 represent anti-correlated regions. This choice of giving a higher impact of anti-correlated regions on the global matching cost allows to robustly impose a higher cost and thus reject anti-correlated matches. Motion and luminance are matched based on their support region of surrounding neighbourhood providing much richer and robust information than the information of single pixels.

Let us recall E_G defined in equations 4.5 and 4.6, where ϵ_g defines the maximum matching pixel error for any candidate. When the matching method is operated ϵ_g is typically set to 1 if subpixelic calibration is achieved. Let us now consider the case where we fix $\epsilon_g = \alpha$ (matches up to α pixels distance are considered plausible candidates) but ignore the value of E_G. Under this situation, geometrical error is only considered in the matching process where the threshold $\epsilon_g = \alpha$ exists but E_G has no influence in the

minimization step. This allows us defining an energy cost function:

$$\tilde{E}(m) = E_T(m) + E_M(m) + E_I(m) \text{ with } \forall E_G(m), \epsilon_g = \alpha, \tag{4.19}$$

meaning we accept matches with any pixel distance smaller than ϵ_g which minimize all other constraints, such that

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \ (E_T(m_i) + E_M(m_i) + E_I(m_i)). \tag{4.20}$$

The main purpose of excluding the geometrical error from the energy minimization formulation is being able to separate the spatial and temporal variables. Doing so, allows us to define the stereo-correspondence method as a purely temporal dependent minimization problem. The minimized function is therefore dependent on a single variable: time.

The four time-dependent energy cost minimizations which can be built from the proposed set of cost functions are:

- temporal:

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \ E_T(m_i) \tag{4.21}$$

- temporal and motion:

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \ (E_T(m_i) + E_M(m_i)) \tag{4.22}$$

- temporal and luminance:

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \ (E_T(m_i) + E_I(m_i)) \tag{4.23}$$

- temporal, motion and luminance

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \ (E_T(m_i) + E_M(m_i) + E_I(m_i)). \tag{4.24}$$

The three last minimizations, as it will be shown, produce the best results with however, those including motion being vulnerable to possible different optical flows on different cameras (as previously explained in section 4.4.4). The complete asynchronous event-based trinocular stereo matching method is summarized in algorithm 4:

Algorithm 4 Trinocular event-based stereo matching algorithm with motion and luminance minimization

Require: Three cameras $\mathcal{R}_u$, $\mathcal{R}_v$, $\mathcal{R}_w$
Require: F_{uv}, F_{uw}, F_{vw}, estimations of the fundamental matrix for each pair of cameras

1: **for all** events $e(\mathbf{p}^u, t)$ in sensor $\mathcal{R}_u$ **do**
2: Determine the set of events $S^v(t)$,$S^w(t)$ from respectively sensors $\mathcal{R}_v$,$\mathcal{R}_w$ occurring at maximum temporal distance ϵ_t
3: Determine the subset of possible matches

$$M = \{m_n = \{e(\mathbf{p}^u_n, t), e(\mathbf{p}^v_n, t), e(\mathbf{p}^w_n, t)\} | e(\mathbf{p}^v_n, t) \in S^v(t), e(\mathbf{p}^w_n, t) \in S^w(t)\}$$

which comply to the trinocular constraint with maximum pixel distance ϵ_g
4: **for all** match candidate $m_n \in M$ **do**
5: Select match m which minimizes temporal, motion and luminance errors

$$m(e^u) = \underset{m_i \in \mathcal{M}}{\operatorname{argmin}} \ (E_T(m_i) + E_M(m_i) + E_I(m_i))$$

6: **end for**
7: **end for**

4.5 Results

A moving box, a waving hand and a moving face were captured using our multi-camera system. Figure 4.8 shows 3D reconstructions of the three scenes. Results were obtained by operating the asynchronous event-based trinocular stereo matching algorithm pre-

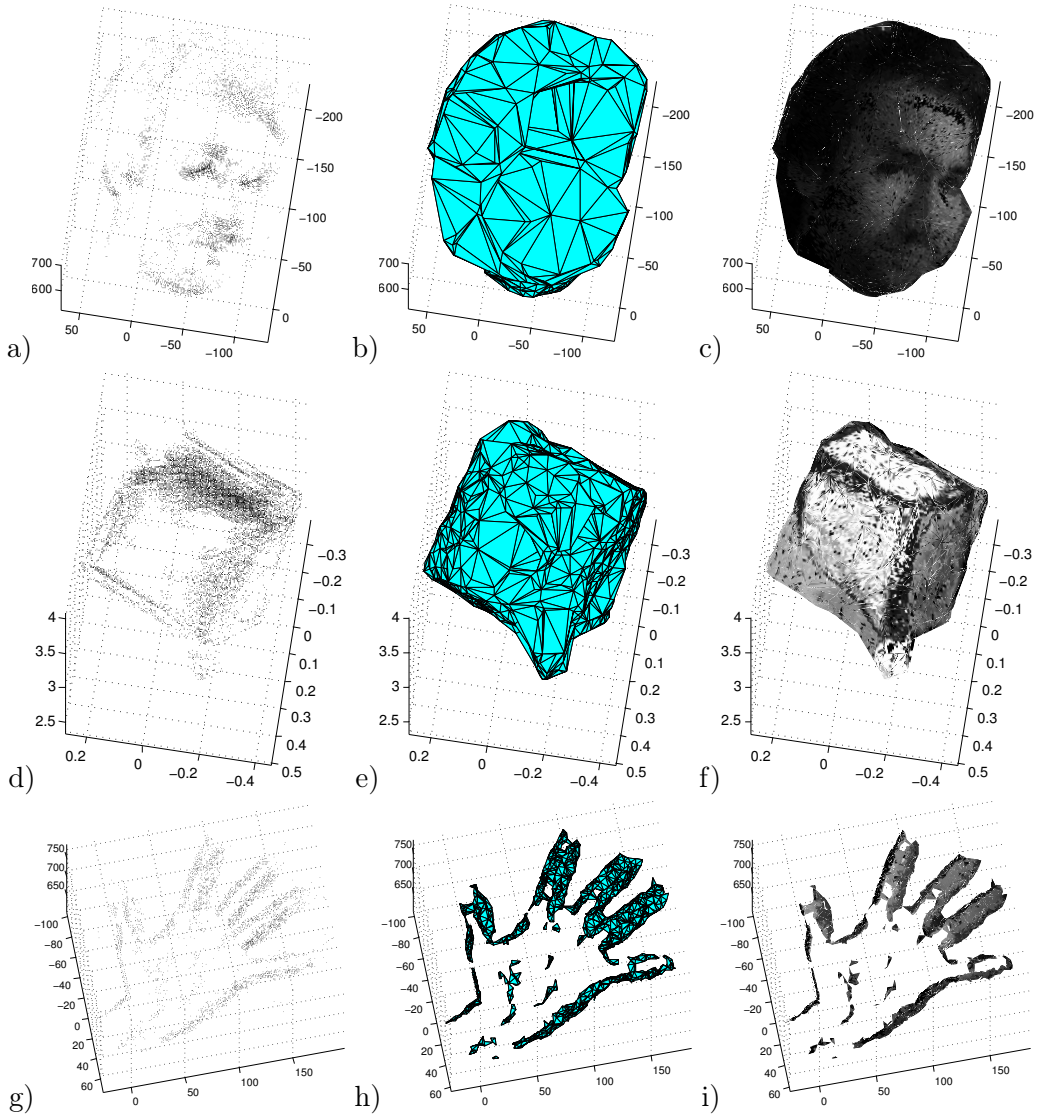


Figure 4.8: 3D reconstruction of a face and box. a,d,g)point cloud b,e,h)alpha volume ($\alpha < 35$) (Edelsbrunner et al., 2006) c,f,i)textured reconstruction

sented in chapter 2 with luminance minimization. All three objects (box, hand and face) were successfully reconstructed. Estimated 3D shapes provide recognizable representations of the object they represent. Figure 4.8 shows examples of reconstructions as well as corresponding computed alpha shape and textured shape. An alpha shape is a collection of piecewise linear simple curves in the Euclidean plane associated with the shape

of a finite set of points as first defined by Edelsbrunner in (Edelsbrunner et al., 2006). The alpha shape is therefore obtained by triangulating clusters of points at a maximum radius alpha.

The ATIS sensor provides gray-level information for generated contrast change events. If we consider an object passing in front of the sensor, the front edge - triggered by the contrast change from the background to the object luminance - will give gray-levels corresponding to the interior of the object, while the back edge - triggered when going from the object surface to background luminance - will give gray values for the background of the scene. Therefore, if an object passes by the sensor (stimulating all pixels) and last gray-level measurements are kept, we are left with a full frame of pixels containing the background of the scene. At any point in time, the gray-level information for any pixel is therefore given by the last luminance measurement at that pixel (if it exists). We map each triangle of the alpha shape to the corresponding image plane pixels obtaining gray-level information. Each triangle of the alpha shape is individually textured with its corresponding gray-level luminance information resulting in the realistic 3D models shown in Fig. 4.8c)f) and i).

4.5.1 Experimental setup

The experimental setup consists of a multi-camera rig composed by 4 ATIS cameras and a Microsoft Kinect sensor. Cameras are synchronized and calibrated with subpixel precision using Yves Bouguet's (Bouguet, 2008) toolbox. The Microsoft Kinect sensor is also calibrated with the multi-camera system and provides ground-truth 3D information of the scene. Synchronization between ATIS and Kinect is achieved by software. In order to maximize the quality of the results with the resolution provided by the QVGA (304×240 pixel resolution) ATIS sensor, cameras are turned inwards such that the total observed scene is limited to a $1m^3$ volume. Illuminating the scene with a 1000watts halogen lamp ensures minimal jitter between the cameras' response.

Figure 4.9: Multicamera setup composed of three ATIS sensors and a Microsoft Kinect sensor for ground-truth.

4.5.2 Method evaluation

A scene of a bouncing human head is captured by all sensors. Both asynchronous event-based matching algorithms (binocular (Rogister et al., 2011) and trinocular) are operated and 3D reconstructions obtained. We evaluate the accuracy of 3D reconstructions obtained when minimizing proposed energy cost functions.

In event-based binocular (Rogister et al., 2011) and trinocular stereo matching method proposed in chapter 2, two input parameters exist and required critical tuning for achieving acceptable 3D reconstructions: matching time-window and geometrical error (distance of matching pixels to epipolar lines). We test the influence of these parameters independently and present 3D reconstruction results for temporal matching windows between $1ms$ and $7ms$ and matching pixel errors of $1px$ to $4px$. Relaxation of temporal and

geometrical constraints allows observing the robustness and sensitivity of the algorithm to parameters' choice.

3D reconstructions are recovered and evaluated in terms of accuracy for four different energy cost minimizations ($1 \leq \epsilon_t \leq 7$ and $1 \leq \epsilon_g \leq 4$):

- spatio-temporal $E = E_G + E_T$
- motion $E = E_M$
- luminance $E = E_I$
- motion and luminance $E = E_M + E_I$

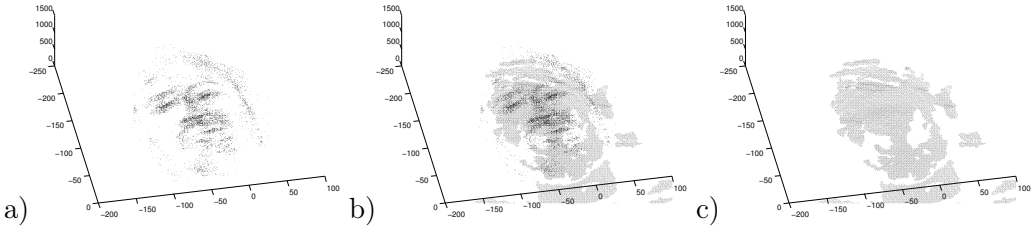


Figure 4.10: 3D reconstruction accuracy evaluation method. Computed 3D points shown in a) are compared to corresponding kinect's depth data in c). By superposing a) to c) we can measure existing closest points for each reconstructed point in order to evaluate the reconstructed shape's accuracy.

In order to evaluate the accuracy of results, 3D reconstructions are compared against the 3D point clouds recovered by the Microsoft kinect sensor (see Fig. 4.10). Pairs of closest point between 3D reconstructions and kinect's point clouds are identified. Two measures quantify the accuracy of the computed reconstructions:

- The reconstruction error is given by the mean distance between the pairs of closest points and is normalized by the maximum width of the object. This value tells how close the computed reconstruction is from the kinect's shape.

- The number of false matches is given by the total number of points whose distance to its corresponding closest point in the kinect's point cloud is larger than 10%. This measure evaluates the amount of noise surrounding the recovered shape.

Last subsection of results shows performance evaluation for a C++ implementation of the asynchronous event-based stereo algorithm running on an Intel(R) Core(TM) i7-2630QM CPU @ 2.00GHz laptop with 8GB DDR3 RAM.

4.5.3 Binocular matching

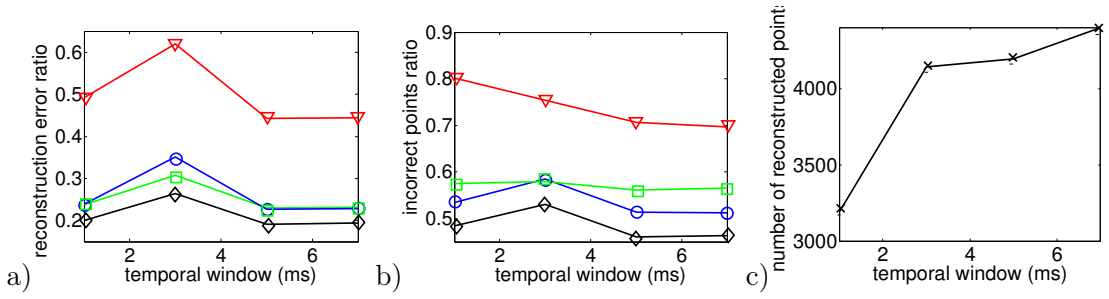


Figure 4.11: Comparison of reconstruction accuracy and errors with each matching cost function and variable matching time window applied to the asynchronous event-based binocular stereo matching (Rogister et al., 2011). Red line (triangle) represents geometrical and temporal minimization (E_G+E_T), green line (square) represents motion minimization (E_M), blue line (circle) represents luminance minimization (E_L) and black (diamond) represents motion and luminance minimization (E_M+E_L). a)Accuracy of 3D reconstruction b)Amount of incorrect points c)Total amount of reconstructed points with different matching time windows.

3D reconstruction is obtained from operating the asynchronous event-based binocular stereo matching algorithm over the sequence encoding a moving human face. The maximum matching geometrical distance is fixed such that $\epsilon_g = 1px$ while the matching time-window parameter is tested for values ranging $1ms \leq \epsilon_t \leq 7ms$ at fixed intervals of $2ms$. 3D reconstructions are obtained with minimization of each of the four proposed energy cost functions E_G+E_T, E_M, E_L and E_M+E_L. Results are presented in Fig. 4.11 showing the influence of the matching time window size when using the binocular match-

ing and each of the proposed matching costs.

Figure 4.11 a) shows that reconstruction accuracy remains almost constant with time window variations for all energy costs. Minimization of $E_G + E_T$ (shown by the red curve) corresponds to the method introduced by Rogister et al in (Rogister et al., 2011), achieves very poor results with reconstruction accuracy around 50%. Furthermore b) shows us that 75% of recovered 3D points are wrong matches meaning that this method produces very noisy and inaccurate point clouds.

The use of any of the other proposed cost functions shows far better results. E_M or E_L minimization gives similar results in terms of accuracy with average reconstruction error around 25% but however E_M seems to produce slightly more wrong matches than E_L with respectively 57% average amount of wrong matches against 53% given by the second function. Energy minimization combining motion and flow $E_M + E_L$ shows the best results with an average 20% reconstruction error and under 50% of wrong matches. Although showing better results than other matching functions, $E_M + E_L$ still shows large amounts of noise with half of the reconstructed points being wrong matches.

The increase of the time-window does not seem to have an influence on the accuracy or the percentage of wrong matches as both are shown constant. However, the number of reconstructed points increases with time window due to relaxation of this maximum temporal distance error. As the number of reconstructed points increases but proportion of wrong matches remains constant, noise in the form of scattered points also accumulates in the area surrounding the object. If the amount of noise is too high the reconstruction becomes unrecognisable even if correct points also exist in the reconstruction.

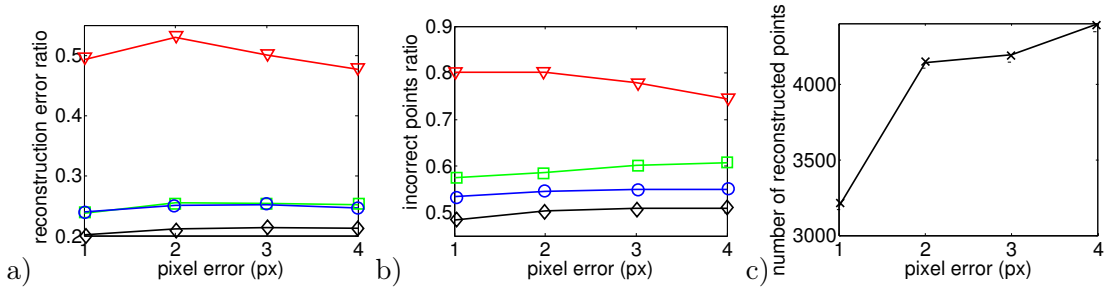


Figure 4.12: Comparison of reconstruction accuracy with each constraint and variable picel geometrical distances with asynchronous event-based binocular stereo matching. Red line (triangle) represents geometrical and temporal minimization ($E_G + E_T$), green line (square) represents motion minimization (E_M), blue line (circle) represents luminance minimization (E_L) and black (diamond) represents motion and luminance minimization ($E_M + E_L$). a)Accuracy of 3D reconstruction b)Amount of incorrect points c)Total amount of reconstructed points

Very similar results are obtained while testing the matching pixel distance error. In this case, the maximum matching temporal distance is fixed such that $\epsilon_t = 1ms$ while the matching pixel error parameter is tested for values ranging $1px \leq \epsilon_g \leq 4ms$. Results are shown in Fig. 4.12. Reconstruction accuracy and amount of false matches remains constant with variable maximum pixel distance error. The amount of reconstructed points increases with the relaxation of the geometrical error, and the same conclusions on the effects of noise are applicable.

Pixel and temporal matching distances do not seem to have an effect on the accuracy of the reconstruction. However, as these constraints are relaxed the number of reconstructed points and amount of produced noise increases. If the density of noise in the reconstructed scene is high identifying the 3D model becomes difficult. Noise can however be filtered up to some level by imposing a threshold on the maximum accepted cost, keeping in mind that if the limit is too tight correct points will be discarded as well.

ϵ_t(ms)	$E_G + E_T$	E_M	E_I	$E_M + E_I$
1				
3				
5				
7				

Table 4.1: Influence of matching timewindow on reconstruction results using the trinocular event-based method. Reconstruction was achieved with geometrical stereo matching of 1px and figures were created from 50ms of 3D events.

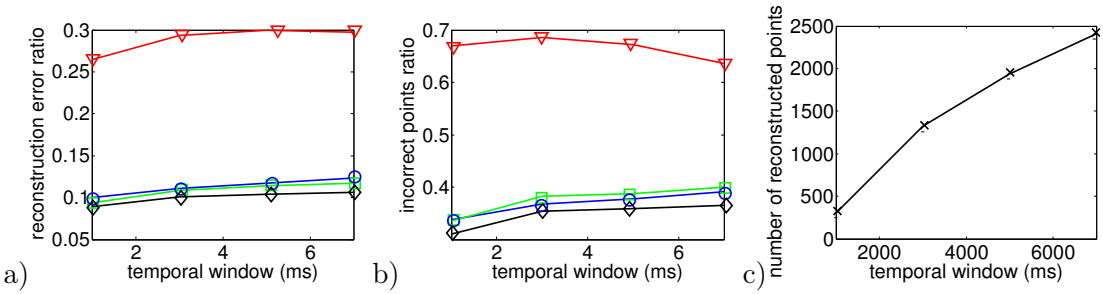


Figure 4.13: Comparison of reconstruction accuracy and errors with each constraint and variable matching time window size with asynchronous event-based trinocular stereo matching. Red line (triangle) represents geometrical and temporal minimization ($E_G + E_T$), green line (square) represents motion minimization (E_M), blue line (circle) represents luminance minimization (E_L) and black (diamond) represents motion and luminance minimization ($E_M + E_L$). a)Accuracy of 3D reconstruction b)Amount of incorrect points c)Total amount of reconstructed points

4.5.4 Trinocular matching

The same evaluation is performed for the proposed energy cost functions ($E_G + E_T$, E_M, E_L and $E_M + E_L$) and independently varying ϵ_t time-window and ϵ_g pixel error. Table 4.1 shows reconstruction of the human face sequence with matching time-windows $1ms \leq \epsilon_t \leq 7ms$ at fixed intervals of $2ms$ and fixed $\epsilon_g = 1px$. Quantitative evaluation is resumed in Fig. 4.13.

Enlarging the matching time-window increases the amount of reconstructed points. The result is visible in two ways: the density of the point cloud with more well-defined shapes but also in the amount of generated noise.

Comparing to the results obtained from the binocular case we can see that the accuracy of 3D reconstructions almost doubles with the trinocular method, with reconstruction errors decreasing from 50% to 25-30% for $E_G + E_T$ minimization and decreasing from 20-25% to 10% for the remaining energy functions. The amount of noise also decreases in the trinocular case with 65% of false matches for the original method $E_G + E_T$ as proposed in chapter 2 and around 35-40% for the improve formulation with any of the additional constraints. An improvement obtained with the trinocular formulation was already expected as the approach consists of a much more constrained geometrical algorithm, however this also largely reduces the amount of reconstructed points as the method enforces the existence of corresponding events in three sensors. Under these circumstances we should recall the solution of increasing the number of sensors as suggested in chapter 2 with the double benefit of increasing the density of 3D reconstructions and minimizing occlusions by providing more viewpoints.

Finally, table 4.2 presents reconstructions using the asynchronous event-based trinocular stereo matching algorithm with varying maximal geometrical distance $1px \leq \epsilon_g \leq 4px$ and fixed time window $\epsilon_t = 1ms$ for each energy cost minimization.

Identically to previous experiments, relaxing the maximum matching pixel error ϵ_g results in noisier reconstructions particularly noticeable in $E_G + E_T$. Figure 4.14 shows

ϵ_g(px)	$E_G + E_T$	E_M	E_I	$E_M + E_I$
1				
2				
3				
4				

Table 4.2: Influence of matching pixel error on reconstruction results using the trinocular event-based method. Reconstruction was achieved with maximal temporal error 1ms for the stereo matching and figures were created from 50ms of 3D events.

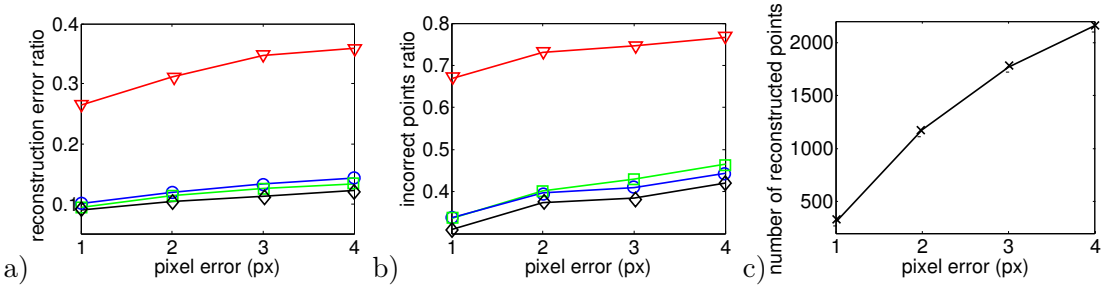


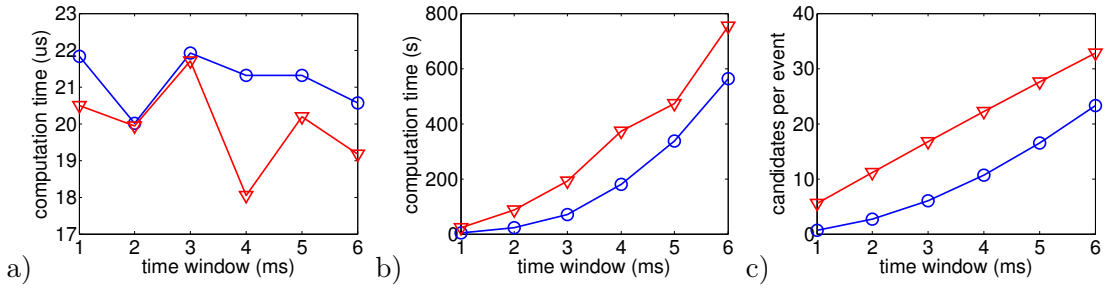
Figure 4.14: Comparison of reconstruction accuracy with each constraint and variable geometrical distances with asynchronous event-based trinocular stereo matching. Red line (triangle) represents geometrical and temporal minimization ($E_G + E_T$), green line (square) represents motion minimization (E_M), blue line (circle) represents luminance minimization (E_L) and black (diamond) represents motion and luminance minimization ($E_M + E_L$). a)Accuracy of 3D reconstruction b)Amount of incorrect points c)Total amount of reconstructed points

the evaluation of obtained 3D point clouds against kinect's depth data. The results show the same expectable increase in accuracy as what was shown for variable matching time-window with larger errors for $E_G + E_T$ and double accuracy and less noise for E_M, E_L and $E_M + E_L$. However it is interesting to notice that the reconstruction error and noise seems to increase with the pixel distance suggesting the trinocular algorithm is more sensitive to the relaxation of the geometrical constraint than to the temporal distance constraint.

Finally, we should highlight the most important observed result from all binocular and trinocular experiments: the use of any of the proposed additional energy minimization functions at least doubles the accuracy of the reconstruction when compared to the corresponding basic spatio-temporal stereo matching method (for both binocular and trinocular methods). Furthermore the noise of reconstructions introduced by false matches is also reduced by 20% to 50% meaning cleaner reconstructions are obtained.

4.5.5 Performance evaluation

Depending on its application purpose, computation performance of 3D reconstruction can be critical. In robot navigation or tele-immersion real-time is a requirement but however in post processing of 3D scenes computation time becomes less critical.



a) b) c)

Figure 4.15: Performance evaluation of the 3D reconstruction algorithm. Binocular matching is shown in red (triangle) and trinocular in blue (circle). a)Mean computation time per candidate b)Mean computation time per 1 second of data c)Mean number of candidates per event.

Increasing the matching time-window results in an increase of potential match candidates and reconstructed 3D points as it was shown in the previous sections. Computation time is however not related to the event rate of the input streams alone but to the event rate of matching events. Performance is therefore conditioned by the amount of candidate matches. Figure 4.15 shows computation time for binocular (red curves) and trinocular (blue curves) matching. Processing time is shown per 3D match candidate a) and for per second of data b) with several different time windows. Mean computation time per candidate remains constant at $20\mu s$ for binocular and $21\mu s$ for trinocular regardless of the matching time-window size. The total computation time increases for larger time windows due to larger amounts of candidate matches. Moreover, for the same reasons the 3D reconstruction for a given sequence becomes longer using the binocular matching method than with the trinocular method.

According to these results, current implementation of the proposed algorithm is able to compute in real-time if the number of event match candidates per second remains under 47000. Taking into account the curve shown in Fig. 4.15c), which shows the mean number of candidates per event with different matching time windows we can estimate the event rates at which real-time computation is possible. Let us assume as example a $4ms$ time-window, where 10 candidates are found per event. In this case real-time is possible only if the event-rate remains under 4700 events per sensor. Event-rate is higly dependant on the scene, light conditions and bias settings, but however for reference, the three reconstructions shown in Fig. 4.8 have average event rates of 452 kevents/second(face), 235 kevents/second(box) and 229 kevents/second(hand). This shows that although a noticeable performance increase was obtained, the algorithm is still far from achieving real-time reconstruction in real scenes.

Another interesting result which should be highlighted is that classically, going from a binocular to a trinocular setup has been reported to increase the computation time by 25% while decreasing ambiguities by 50% (Dhond and Aggarwal, 1990). However, in

our event-based implementation we obtain a decrease of ambiguities (for a $4ms$ matching time window) of 50% while also seeing a decrease in computation time around 50%. This can be easily explained by the amount of generated match candidates in the binocular or trinocular case. The binocular case, generates more than double of 3D points (see fig 4.11,4.12,4.13,4.14) and about double the amount of match candidates (for time window of 4ms), meaning more processing is needed due to both higher number of correlations (due to the larger number of candidates per 3D point) and higher number of minimizations to perform (due to larger number of reconstructed points).

4.6 Discussion

The use of the ATIS camera, a much higher resolution (≈ 4.5 times than the 128×128 pixel resolution DVS) sensor, leads by itself to incomparably better results from what was seen in chapter 2 and other previous work using the DVS sensor. This increase in pixel resolution means better accuracy in geometrical matching as smaller pixels map fewer 3D points and thus fewer match candidates exist.

Results with both binocular and trinocular asynchronous event-based stereo matching are able to provide acceptable 3D reconstruction of complex shapes when using motion or luminance minimization. Results with the trinocular method provide cleaner and more accurate reconstructions than binocular solution while requiring lower computation cost. Motion and luminance minimization seem to provide similar results in terms of reconstruction accuracy and may be used according to circumstances. Motion consistency is sensitive to the motion of the scene meaning that scenes where objects move at the center of the stereo rig and with a movement perpendicular to the baseline will result in divergent optical flows and thus matching will be unsuccessful. Luminance is in this case better as it is motion invariant. However, in situations where luminance information is not available, best reconstruction is achieved by minimizing the motion consistency

error.

The accuracy of reconstructed shapes does not seem to be radically affected by variations of maximal temporal or geometrical errors. However, as the number of reconstructed events increases and the percentage of wrong matches remains constant, the noise in the reconstruction will increase. Furthermore as larger temporal and geometrical distances are allowed in the stereo matching process, more matching candidates will exist and computation cost quickly grows.

4.6.1 3D Structure refinement using point cloud prediction

The event-based trinocular stereo correspondence problem was solved as the minimization of an energy cost function with penalties on geometrical, temporal, motion and intensity errors. The result of operating the proposed algorithm over the event streams of a multi-camera system is a 3D event stream containing 3D reconstructed points observed by the cameras over time. However, matching errors occur and the 3D event stream may contain incorrectly reconstructed events.

In chapter 3 we proposed a method for 3D scene flow estimation from point clouds where motion is directly extracted from position and time information of 3D points. The main motivation behind this method was its application to an iterative refinement of 3D structures. Although not explicitly explored throughout this chapter (where its projection, the optical flow, was used instead), we will anyhow present how the 3D scene flow could be applied and provide a formulation to be used in future work. We make use of this motion information to predict future location of 3D structures. 3D events are compared to predictions as to where the point cloud exists. Motion and reconstruction are used in a closed feedback loop where reconstructed points are used for motion estimation and motion information is used to validate new 3D events.

Information about the 3D motion flow of points in a scene provides valuable information that may be added to further increase the quality and robustness of 3D structures.

While estimating velocity vectors, a shape registration method allows to match structures across time and to formulate an energy minimization solution for motion estimation. Here we use the same principle of moving point clouds according to estimated velocities to predict the position of new events. Reconstructed events which match the predicted structure can be given higher credibility.

The formulation of the asynchronous event-based scene flow presented in the previous chapter provides a way to extract motion from a spatio-temporal neighbourhood $N(e(\mathbf{P},t))$, containing the 3D events spatially defined around $\mathbf{P}$ and temporally defined after t. We now intend to use the motion flow information to predict the location of events in the future where $e(\mathbf{P_{+1}},t_{+1}) = e(\mathbf{P}+\mathbf{v}(e(\mathbf{P},t)),t)$. In fact, motion $\mathbf{v}(e(\mathbf{P},t))$ is estimated from $N(e(\mathbf{P},t))$ defined in the future of t. Here, we use instead $\hat{\mathbf{v}}(e(\mathbf{P},t)) = \mathbf{v}(e(\mathbf{P_{-1}},t_{-1}))$ an approximation of velocity $\mathbf{v}(e(\mathbf{P},t))$. As structures do not appear abruptly across time we know that all events consist of temporal evolution of previous events at a different location (exceptions for occlusions or objects entering/leaving the scene):

$$\forall e(\mathbf{P_{+1}},t_{+1}) \exists e(\mathbf{P},t) : \mathbf{P_{+1}} = \mathbf{P} + \mathbf{v}(e(\mathbf{P_{-1}},t_{-1})) \tag{4.25}$$

We can therefore evaluate new events according to their consistency to shape and motion of the scene according to:

$$\mathbf{P_{+1}} = \mathbf{P} + \mathbf{v}(e(\mathbf{P_{-1}},t_{-1})) + \varepsilon_{\mathbf{v}} \tag{4.26}$$

where $||\varepsilon_{\mathbf{v}}||$ is the inconsistency error which inversely reflects the credibility of a given event.

However identifying corresponding $\mathbf{P_{+1}}$ and $\mathbf{P}$ is difficult and we cannot guarantee that both will exist respectively at $t+1$ and t. Moreover a single 3D point distance does not provide a robust measurement. We therefore instead use ε_v as the mean closest point

distance between the existing structures surrounding $\mathbf{P_{+1}}$ and $\mathbf{P}$, similarly to what is done in the previous chapter for motion estimation. If $N(\mathbf{P})$ is the set of points spatially located around $\mathbf{P}$ then

$$\varepsilon_v = \frac{1}{n}\sum_{i=1}^{n} \min ||\mathbf{P_i} - \mathbf{P_j}|| \text{ with } \mathbf{P_i} \in N(\mathbf{P}), \mathbf{P_j} \in N(\mathbf{P_{+1}}). \tag{4.27}$$

4.7 Conclusion

3D reconstruction from multiple views is one of the most important problems in computer vision as it allows recovering tridimensional structures from multiple two-dimensional views of a given scene. Its importance and numerous applications have motivated researchers to continuously propose new methods throughout the last decades. Traditionally, 3D reconstruction from multiple views is achieved through a process of pixel matching between different views. Finding correspondences in different views is a complex problem which can easily conduct to ambiguities. Computer vision methods typically solve these ambiguities by adding spatial constraints (such as photo-constancy). The temporal dimension is still largely unexplored even though the important role of time in 3D estimation has been shown both in biological and computer stereo vision. Researchers reported incorrect depth perception occurs from temporal disparities between views but however most existing computer methods, relying on low temporal dynamic frame-based acquisition, cannot take this property into account.

The introduction of neuromorphic silicon retinas, bio-inspired vision sensors which encode visual information as a stream of events provides a new way to address the stereo correspondence problem. Early solutions such as (Rogister et al., 2011) and the method we proposed in chapter 2 used classical epipolar geometry and the precise timing of these sensors to match events and recover depth in an asynchronous event-based fashion. However, these methods were prone to errors as ambiguities could not be solved from

co-activation and geometry alone.

We studied temporal-based constraints with luminance and motion information expressed in terms of time. We proposed independent energy cost functions for each of the four constraints geometry, time, motion and luminance. We introduced a modular formulation of an energy cost function composed by any combination of the available matching cost functions. This modular approach has the advantage of allowing to chose energy cost functions according to available information or performance concerns. Furthermore proposed constraints (luminance and motion) were defined as functions of time allowing the asynchronous event-based stereo correspondence problem to be described as the minimization of an energy cost function solely dependent of the variable time.

We show that the added luminance constancy and motion consistency cost functions greatly increase accuracy of reconstructions while reducing the amount of false matches and noise in both binocular and trinocular versions. Results prove that complex shapes can be reconstructed with high accuracy when luminance or motion minimization are used.

In this chapter we presented the first asynchronous event-based 3D reconstruction method able to recover the 3D structure of complex shapes from neuromorphic vision sensors. The method is able to produce realistic representations of the scene in the form of textured 3D models. Furthermore, due to the high temporal resolution of the neuromorphic vision sensors, our method is also a pioneer in high frequency 3D reconstruction being able to recover shapes at up to 1MHz with current sensor. The method is therefore ideal for recovering 3D from fast moving scenes where frame-based cameras are not able to cope with the dynamics of the scene.

Chapter 5

Discussion

> *Sometimes it seems as though each new step towards AI, rather than producing something which everyone agrees is real intelligence, merely reveals what real intelligence is not.*
>
> Douglas R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, 1979

Neuromorphic vision sensors introduce a new way of encoding visual information where scenes are no longer represented as frames but as a continuous stream of asynchronous events. This high temporal resolution and precise timing of events allows keeping the temporal dynamic of visual scenes usually lost in conventional computer vision. This new representation of visual information introduces a paradigm shift in computer vision. Visual computation can be based only on the occurrence of single asynchronous events. Event-based visual computation opens new way for innovative research and new methodology to solve computer vision problems. In this thesis we explored the potential of using time as a computation feature in depth estimation and 3D reconstruction and studied its properties.

Stereo vision has been chosen as a research topic because it is still an important and not totally solved problem in computer vision. Currently real-time dense methods are computationally expensive and do not exceed 60Hz. Several frame-based methods for 3D reconstruction have been developed, however the temporal dimension is never used nor seen as a critical variable.

Time seems to play an important role in biological stereo vision, how can the temporal precision of neuromorphic retinas be used to tackle the stereo correspondence problem and recover 3D structures? Previous event-based work approached the problem using single event matching and binocular epipolar geometry. The formulation showed an innovative approach but depth estimation was coarse and prone to errors. To solve this question we introduced a more geometrically constrained method by adding the implications of adding a third sensor. The approach relied on single event matching based on temporal consistency and trinocular epipolar geometry. The method is scalable to any number of cameras and showed innovative results in asynchronous event-based 3D vision. It is the first and only method able to produce reliable 3D reconstructions using time. In our approach the role of precise timing is critical as events are matched based on their co-activation. We can arguably say that high temporal precision is an essential feature of 3D processing.

We may then wonder *what are the advantages in using asynchronous event-based encoding of the visual information?* Neuromorphic silicon retinas encode visual information as events that represent relative contrast changes of the scene. If only changing information (such as moving objects) is transmitted, the amount of data to be computed is substantially smaller. Asynchronous event-based information allows matching single events using coactivation of pixels. The task of finding correspondences between cameras is then obviously far more efficient than searching for correspondences in the whole array of pixels.

What are the consequences and how to deal with the high event-rate of neuromorphic

silicon retinas? The methods proposed in this thesis were formulated in an event-based methodology in order to take advantage of the temporal properties provided by the event-based encoding format of neuromorphic silicon retinas. However, these approaches were developed and implemented using conventional Von-Neumann computer processors which are inadequate to process event-based information. Promising neuromorphic neural network chips exist, they are able to process event-based data in a highly parallel way. A neuromorphic hardware implementation of asynchronous event-based stereo matching methods is out of the scope of this thesis, its use would lead to a more natural computation without the need to time-stamp events and serialize them.

The high temporal resolution of the sensor cannot be efficiently processed by classical computer but *how can the temporal dynamics of scenes contribute to 3D reconstruction?* This temporal dynamics provided by the neuromorphic visual sensor allows neuromorphic computer vision methods to produce a high amount of temporal accurate results. The result has a much higher dynamic 3D representation of the scene. As motion is estimated from the position of an object in time this continuous information can be used to help solving ambiguities in stereo-matching methods. We showed that using event-based information allows to create spatial-temporal features that highly improve the event-based matching.

We may then wonder *how is asynchronous event-based stereo matching related to biological stereo vision?* Although we can not derive direct assumptions about the use of precise timing to solve stereo correspondence. We can for sure emphasize that without time processing disparity would be more computationally expensive. Neurons are known to be coincidence detectors. In this context our approach fully matches this property. However if direct matching is performed only on event coincidence between two views, we have shown that this approach produces a high number of false matches that lead to a poor estimated shape. The combined use of higher level information such as motion and grey-level appears to be essential to estimate correct shapes. We can then conjecture

that 3D is a hierarchical process. At a low level in V1 simple coincidence detector are computed, this information is then sent in a second stage sent to higher cortical areas such as MT where they are refined. Even though we have not shown how these structure interact, it is most probable that a feedback loop exists to lower areas of the visual system in order to inhibit neurons generating wrong matches. Other smoothing constraints must also play an important role, such as local disparity differences. It is interesting to notice that different time scales operate at the same time. A direct immediate inaccurate computation and more long term accurate information.

The presented work provides the basis for further research providing clues for other event-based stereo matching studies form a computational and physiological point of view. Event-based acquisition allows to fulfil David Marr's dream to merge computational and biological studies of the visual system into a unified framework. Current developments at the lab are exploring this path. They have started using brain imaging techniques and psychophysics tests to study the hypothesis derived from our work to produce more realistic and computationally efficient algorithms to estimate depth.

References

Martin S. Banks, Sergei Gepshtein, and Michael S. Landy. Why is spatial stereoresolution so low? *JOURNAL OF NEUROSCIENCE*, 24(9):2077–2089, 2004. 4

T. Basha, Y. Moses, and N. Kiryati. Multi-view scene flow estimation: A view centered variational approach. *International Journal of Computer Vision*, 101-1:6–21, 2013. 48

R. Benosman, C. Clercq, X. Lagorce, S.-H. Ieng, and C. Bartolozzi. Event-based visual flow. *Neural Networks and Learning Systems, IEEE Transactions on*, PP(99):1–1, 2013a. ISSN 2162-237X. doi: 10.1109/TNNLS.2013.2273537. 13, 72, 85

Ryad Benosman, Sio-Hoï Ieng, Charles Clercq, Chiara Bartolozzi, and Mandyam Srinivasan. Asynchronous frameless event-based optical flow. *Neural Netw.*, 27:32–37, March 2012. ISSN 0893-6080. doi: 10.1016/j.neunet.2011.11.001. URL `http://dx.doi.org/10.1016/j.neunet.2011.11.001`. 27

Ryad Benosman, João Carneiro, and Sio-Hoï Ieng. Method of 3d reconstruction of a scene calling upon asynchronous sensors, 2013b. WO Patent 2,013,083,848.

A Bergua and W Skrandies. An early antecedent to modern random dot stereograms - 'the secret stereoscopic writing' of ramon y cajal. *International Journal of Psychophysiology*, 36(1):69–72, 2000. URL `http://www.ncbi.nlm.nih.gov/pubmed/10700624`. 3

P.J. Besl and H.D. McKay. A method for registration of 3-d shapes. *Pattern Analysis*

and Machine Intelligence, IEEE Transactions on, 14(2):239 –256, feb 1992. ISSN 0162-8828. doi: 10.1109/34.121791. 57

Asim Bhatti. *Current Advancements in Stereo Vision*. InTech, 2012. ISBN 978-953-51-0660-9. doi: 10.5772/2611. URL `http://www.intechopen.com/books/export/citation/BibTex/current-advancements-in-stereo-vision/stereo-matching-from-the-basis-to-neuromorphic-engineering`. 5

Rahul Bhotika, David J. Fleet, and Kiriakos N. Kutulakos. A probabilistic theory of occupancy and emptiness. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *ECCV (3)*, volume 2352 of *Lecture Notes in Computer Science*, pages 112–132. Springer, 2002. ISBN 3-540-43746-0. URL `http://dblp.uni-trier.de/db/conf/eccv/eccv2002-3.html#BhotikaFK02`. 9

K.A. Boahen. Point-to-point connectivity between neuromorphic chips using address events. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 47(5):416–434, 2000. ISSN 1057-7130. doi: 10.1109/82.842110. 75

J. Y. Bouguet. Camera calibration toolbox for Matlab, 2008. URL `http://www.vision.caltech.edu/bouguetj/calib_doc/`. 92

R. Carceroni and K. Kutulakos. Multi-view scene capture by surfel sampling: From video streams to non-rigid 3d motion, shape and reflectance. *International Journal of Computer Vision*, 49:175–214, 2002. 48

João Carneiro, Sio-Hoï Ieng, Xavier Clady, and Ryad Benosman. Scene flow from 3d point clouds. *International Journal of Computer Vision*, 2013a. Under review.

João Carneiro, Sio-Hoï Ieng, Christoph Posch, and Ryad Benosman. Asynchronous event-based 3d reconstruction from neuromorphic retinas. *Neural Networks*, 45(0):27 – 38, 2013b. ISSN 0893-6080. doi: http://dx.doi.org/10.1016/j.neunet.2013.03.006. URL

http://www.sciencedirect.com/science/article/pii/S0893608013000725. Neuromorphic Engineering: From Neural Systems to Brain-Like Engineered Systems.

João Carneiro, Sio-Hoï Ieng, and Ryad Benosman. It's all about time. 2014. Under preparation.

R. Chang. *Etude des réseaux de caméras non synchronisées ou non calibrées*. Thèse de doctorat, Université Pierre et Marie Curie, Paris 6, 4 place jussieu, 75005 PARIS, Jan. 2009. 12

D. Chetverikov, D. Stepanov, and P. Krsek. Robust euclidean alignment of 3d point sets: the trimmed iterative closest point algorithm. *Image and Vision Computing*, 23: 299–309, 2005. 36

Lawrence K. Cormack, Scott B. Stevenson, and Clifton M. Schor. Interocular correlation, luminance contrast and cyclopean processing. *Vision Research*, 31(12):2195 – 2207, 1991. ISSN 0042-6989. doi: http://dx.doi.org/10.1016/0042-6989(91)90172-2. URL http://www.sciencedirect.com/science/article/pii/0042698991901722. 4, 11

B. Curless and M. Levoy. Better optical triangulation through spacetime analysis. In *International Conference on Computer Vision*, Stanford, CA, USA, 1995. Stanford University. 6

James Davis, Diego Nehab, Ravi Ramamoorthi, and Szymon Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(2):296–302, February 2005. ISSN 0162-8828. doi: 10.1109/TPAMI.2005.37. URL http://dx.doi.org/10.1109/TPAMI.2005.37. 6

T. Delbrück and C. A. Mead. Analog VLSI phototransduction by continuous-time, adaptive, logarithmic photoreceptor circuits. In C. Koch and H. Li, editors, *Vision Chips: Implementing vision algorithms with analog VLSI circuits*, pages 139–161. IEEE Computer Society Press, Los Alamitos, CA, 1995. 13

T. Delbruck, B. Linares-Barranco, E. Culurciello, and C. Posch. Activity-driven, event-based vision sensors. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 2426–2429, 2010. doi: 10.1109/ISCAS.2010.5537149. 13, 75

Tobi Delbruck and Patrick Lichtsteiner. Freeing vision from frames. *The Neuromorphic Engineer*, 2006. doi: 10.2417/1200605.0037. URL `http://www.ine-news.org/view.php?source=0037-2006-05-01`. 12

Parvati Dev. Perception of depth surfaces in random-dot stereograms : a neural model. *International Journal of Man-Machine Studies*, 7(4):511 – 528, 1975. ISSN 0020-7373. doi: http://dx.doi.org/10.1016/S0020-7373(75)80030-7. URL `http://www.sciencedirect.com/science/article/pii/S0020737375800307`. 3

U.R. Dhond and J.K. Aggarwal. Binocular versus trinocular stereo. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pages 2045–2050 vol.3, 1990. doi: 10.1109/ROBOT.1990.126306. 102

J. Durbin. Efficient estimation of parameters in moving average models. *Biometrica*, 46: 306–317, 1959. 58

H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Trans. Inf. Theor.*, 29(4):551–559, September 2006. ISSN 0018-9448. 91, 92

Peter Eisert, Eckehard Steinbach, and Bernd Girod. Multihypothesis volumetric reconstruction of 3-d objects from multiple calibrated camera views. In *Proc. International Conference onAcoustics, Speech, and Signal Processing ICASSP'99*, pages 3509–3512, 1999. 9

Casper J. Erkelens. Organisation of signals involved in binocular perception and vergence control. *Vision Research*, 41(25):3497 – 3503, 2001. ISSN 0042-6989. doi: http://

dx.doi.org/10.1016/S0042-6989(01)00004-9. URL http://www.sciencedirect.com/science/article/pii/S0042698901000049. 3

Olivier D. Faugeras, Elisabeth Le Bras-Mehlman, and Jean-Daniel Boissonnat. Representing stereo data with the delaunay triangulation. *Artif. Intell.*, 44(1-2):41–87, 1990. URL http://dblp.uni-trier.de/db/journals/ai/ai44.html#FaugerasBB90. 10

DavidJ. Fleet and AllanD. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990. ISSN 0920-5691. doi: 10.1007/BF00056772. URL http://dx.doi.org/10.1007/BF00056772. 61

Thomas Fromherz and Martin Bichsel. Shape from multiple cues: Integrating local brightness information. In *Proceedings of the Fourth International Conference for Young Computer Scientists, ICYCS 95*, pages 855–862, 1995. 9

Yasutaka Furukawa. *High-fidelity image-based modeling*. PhD thesis, University of Illinois at Urbana-Champaign, Champaign, IL, USA, 2008. AAI3314770. 9

Pau Gargallo and Peter F. Sturm. Bayesian 3d modeling from images using multiple depth maps. In *CVPR (2)*, pages 885–891. IEEE Computer Society, 2005. ISBN 0-7695-2372-2. URL http://dblp.uni-trier.de/db/conf/cvpr/cvpr2005-2.html#GargalloS05. 9

F Gonzalez and R Perez. Neural mechanisms underlying stereoscopic vision. *Progress in Neurobiology*, 55(3):191 – 224, 1998. ISSN 0301-0082. doi: 10.1016/S0301-0082(98)00012-4. URL http://www.sciencedirect.com/science/article/pii/S0301008298000124. 3, 11

Simon Hadfield and Richard Bowden. Kinecting the dots: Particle based scene flow from depth sensors. In *Proceedings, International Conference on Computer Vision*, pages 2290 – 2295, Barcelona, Spain, 6-13 Nov 2011. doi: 10.1109/ICCV.2011.6126509. 48

Miles Hansard, Seungkyu Lee, Ouk Choi, and Radu Horaud. *Time-of-flight cameras: Principles, Methods and Applications*. Springer Briefs in Computer Science, 2012. ISBN 978-1-4471-4657-5. 47

R.I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 8, 24, 26

M. Hatzitheodorou, E. A. Karabassi, G. Papaioannou, A. Boehm, and T. Theoharis. Stereo matching using optic flow. *Real-Time Imaging*, 6(4):251–266, August 2000. ISSN 1077-2014. doi: 10.1006/rtim.1998.0141. URL `http://dx.doi.org/10.1006/rtim.1998.0141`. 74

Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d flow: Dense 3-d motion estimation using color and depth. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2013. 47

C. Hernández. *Stereo and Silhouette Fusion for 3D Object Modeling from Uncalibrated Images Under Circular Motion*. PhD thesis, Ecole Nationale Supŕieure des Télécommunications, May 2004. 8

Aaron Hertzmann and Steven M. Seitz. Shape and materials by example: a photometric stereo approach. In *Proceedings of the 2003 IEEE computer society conference on Computer vision and pattern recognition*, CVPR'03, pages 533–540, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1900-8, 978-0-7695-1900-5. URL `http://dl.acm.org/citation.cfm?id=1965841.1965911`. 6

Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, August 1981. ISSN 00043702. doi: 10.1016/0004-3702(81)90024-2. URL `http://dspace.mit.edu/handle/1721.1/6337`. 55

Ian P. Howard and Brian J. Rogers. *Seeing in Depth Volume 2: Depth Perception*, volume 2. Oxford University, 2008. 2, 4, 11, 12

Frederik Huguet and Frederic Devernay. A variational method for scene flow estimation from stereo sequences. In *ICCV*, pages 1–7. IEEE, 2007. 48

Michael Isard and John MacCormick. Dense motion and disparity estimation via loopy belief propagation. In *Proceedings of the 7th Asian conference on Computer Vision - Volume Part II*, ACCV'06, pages 32–41, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-31244-7, 978-3-540-31244-4. 48

B. Julesz. Binocular Depth Perception of Computer-Generated Patterns. *Bell System Tech.*, 39(5):1125–1161, September 1960. 2

T. Kanade, P. J. Narayanan, and P. W. Rander. Virtualized reality: concepts and early results. In *Proceedings of the IEEE Workshop on Representation of Visual Scenes*, VSR '95, pages 69–, Washington, DC, USA, 1995. IEEE Computer Society. ISBN 0-8186-7122-X. URL `http://dl.acm.org/citation.cfm?id=832295.836234`. 9

T. Kanade, H. Saito, and S. Vedula. The 3d room: digitizing time-varying 3d events by synchronized multiple video streams. Technical report, CMU, Dec. 1998. 9

K. Khoshelham and S. O. Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012. 47

Andreas Klaus, Mario Sormann, and Konrad Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 03*, ICPR '06, pages 15–18, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2521-0. doi: 10.1109/ICPR.2006.1033. URL `http://dx.doi.org/10.1109/ICPR.2006.1033`. 6

J Kogler, Christoph Sulzbachner, and Wilfried Kubinger. Bio-inspired stereo vision system with silicon retina imagers. *Computer Vision Systems*, pages 174–183, 2009. URL `http://www.springerlink.com/index/T280653VJQU70473.pdf`. 14

Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *ECCV (3)*, volume 2352 of *Lecture Notes in Computer Science*, pages 82–96. Springer, 2002. ISBN 3-540-43746-0. URL `http://dblp.uni-trier.de/db/conf/eccv/eccv2002-3.html#KolmogorovZ02a`. 9

Jörg Kramer. An on/off transient imager with event-driven, asynchronous read-out. In *in Proc. IEEE International Symposium on Circuits and Systems - ISCAS 2002*, pages 165–168, 2002. 13

Y. Kunii and H. Chikatsu. Automatic stereo matching using optical flow for 3d object modeling. In *Proceedings, International Archives of Photogrammetry and Remote Sensing*, volume 33, pages 459–465, Amsterdam, NL, 2000. International Society for Photogrammetry and Remote Sensing. 74

Kiriakos N. Kutulakos. Approximate n-view stereo. In *in Proc. European Conf. on Computer Vision*, pages 67–83, 2000. 9

Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000. URL `http://dblp.uni-trier.de/db/journals/ijcv/ijcv38.html#KutulakosS00`. 9

J. Lee, T. Delbruck, P.K.J. Park, M. Pfeiffer, C.W. Shin, H. Ryu, and B.C. Kang. Gesture-based remote control using stereo pair of dynamic vision sensors. In *IEEE International Symposium on Circuits and Systems - ISCAS 2012*, 2012a. URL `http://siliconretina.ini.uzh.ch/wiki/index.php`. 14

Junhaeng Lee, T. Delbruck, P.K.J. Park, M. Pfeiffer, Chang-Woo Shin, Hyunsurk Ryu, and Byung-Chang Kang. Live demonstration: Gesture-based remote control using stereo pair of dynamic vision sensors. In *Circuits and Systems (ISCAS), 2012 IEEE*

International Symposium on, pages 741–745, May 2012b. doi: 10.1109/ISCAS.2012.6272144. 13

Zucheul Lee, R. Khoshabeh, J. Juang, and T.Q. Nguyen. Local stereo matching using motion cue and modified census in video disparity estimation. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 1114–1118, 2012c. 74

P. Lichtsteiner, C. Posch, and T. Delbruck. A 128 x 128 120db 30mw asynchronous vision sensor that responds to relative intensity change. In *Solid-State Circuits Conference, 2006. ISSCC 2006. Digest of Technical Papers. IEEE International*, pages 2060–2069, Feb 2006. doi: 10.1109/ISSCC.2006.1696265. 13

Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128x128 120 db 15 us latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid State Circuits*, 43(2):566–576, 2008. URL `http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4444573`. 13, 22, 23, 35, 45, 73

Henri Lorach, Ryad Benosman, Olivier Marre, Sio-Hoi Ieng, José A Sahel, and Serge Picaud. Artificial retina: the multichannel processing of the mammalian retina achieved with a neuromorphic asynchronous light acquisition device. *Journal of Neural Engineering*, 9(6):066004, 2012. URL `http://stacks.iop.org/1741-2552/9/i=6/a=066004`. 13

H.-G. Maas. Complexity analysis for the establishment of image correspondences of dense spatial target fields. *International Archives of Photogrammetry and Remote Sensing*, 29:102–107, 1992. 27

M. Mahowald. *VLSI Analogs of Neuronal Visual Processing: A Synthesis of Form and Function*. PhD thesis, California Institut of Technology, 1992. URL `http://caltechcstr.library.caltech.edu/591/`. 13

M. Mahowald and T. Delbrück. Cooperative stereo matching using static and dynamic image features. In C. Mead and M. Ismail, editors, *Analog VLSI Implementation of Neural Systems*, pages 213–238. Kluwer Academic Publishers, 1989. 14

Anastasios Manessis, Adrian Hilton, Phil Palmer, Philip F. McLauchlan, and Xinquan Shen. Reconstruction of scene models from sparse 3d structure. In *CVPR*, pages 2666–2673. IEEE Computer Society, 2000. ISBN 0-7695-0662-3. URL `http://dblp.uni-trier.de/db/conf/cvpr/cvpr2000.html#ManessisHPMS00`. 10

D Marr and T Poggio. Cooperative computation of stereo disparity. *Science*, 194(4262): 283–287, 1976. doi: 10.1126/science.968482. URL `http://www.sciencemag.org/content/194/4262/283.abstract`. 3

D. Marr and T. Poggio. A Computational Theory of Human Stereo Vision. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 204(1156):301–328, 1979. ISSN 00804649. doi: 10.2307/77509. URL `http://dx.doi.org/10.2307/77509`. 4

S. Maybank. *Theory of reconstruction from image motion*. Springer series in information sciences. Springer-Verlag, 1993. ISBN 9780387555379. URL `http://books.google.fr/books?id=GigZAQAAIAAJ`. 47

John E. W. Mayhew and John P. Frisby. Psychophysical and computational studies towards a theory of human stereopsis. *Artif. Intell.*, 17(1-3):349–385, 1981. URL `http://dblp.uni-trier.de/db/journals/ai/ai17.html#MayhewF81`. 4

Dongbo Min and Kwanghoon Sohn. Edge-preserving simultaneous joint motion-disparity estimation. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 02*, ICPR '06, pages 74–77, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2521-0. doi: 10.1109/ICPR.2006.470. URL `http://dx.doi.org/10.1109/ICPR.2006.470`. 48

Donald E. Mitchell and Steven O'Hagan. Accuracy of stereoscopic localization of small line segments that differ in size or orientation for the two eyes. *Vision Research*, 12(3):437 – 454, 1972. ISSN 0042-6989. doi: http://dx.doi.org/10.1016/0042-6989(72)90088-0. URL `http://www.sciencedirect.com/science/article/pii/0042698972900880`. 11

Daniel D. Morris and Takeo Kanade. Image-consistent surface triangulation. In *CVPR*, pages 1332–1338. IEEE Computer Society, 2000. ISBN 0-7695-0662-3. URL `http://dblp.uni-trier.de/db/conf/cvpr/cvpr2000.html#MorrisK00`. 10

P. J. Narayanan, Peter Rander, and Takeo Kanade. Constructing virtual worlds using dense stereo. In *ICCV*, pages 3–10, 1998. URL `http://dblp.uni-trier.de/db/conf/iccv/iccv1998.html#NarayananRK98`. 9

Nasser M. Nasrabadi, Sandra P. Clifford, and Yi Liu. Integration of stereo vision and optical flow by using an energy-minimization approach. *J. Opt. Soc. Am. A*, 6(6): 900–907, Jun 1989. doi: 10.1364/JOSAA.6.000900. URL `http://josaa.osa.org/abstract.cfm?URI=josaa-6-6-900`. 74

Shree K. Nayar, Gurunandan Krishnan, Michael D. Grossberg, and Ramesh Raskar. Fast separation of direct and global components of a scene using high frequency illumination. In *ACM SIGGRAPH 2006 Papers*, pages 935–944, 2006. ISBN 1-59593-364-6. doi: 10.1145/1179352.1141977. URL `http://doi.acm.org/10.1145/1179352.1141977`. 6

Jeremiah I. Nelson. Globality and stereoscopic fusion in binocular vision. *Journal of Theoretical Biology*, 49(1):1 – 88, 1975. ISSN 0022-5193. doi: http://dx.doi.org/10.1016/S0022-5193(75)80020-8. URL `http://www.sciencedirect.com/science/article/pii/S0022519375800208`. 3

Z. Ni, A. Bolopion, J. Agnus, R. Benosman, and S. Régnier. Asynchronous event-based

visual shape tracking for stable haptic feedback in microrobotics. *IEEE Transactions on Robotics (T-RO)*, 28(5):1081–1089, 2012. 13

Matthias Niesner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 32(6):169:1–169:11, November 2013. ISSN 0730-0301. doi: 10.1145/2508363.2508374. URL `http://doi.acm.org/10.1145/2508363.2508374`. 10

H. K. Nishihara. Prism: A practical real-time imaging stereo matcher. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1984. 4

M. Otte and H.-H. Nagel. Optical flow estimation: Advances and comparisons. In Jan-Olof Eklundh, editor, *Computer Vision âĂŤ ECCV '94*, volume 800 of *Lecture Notes in Computer Science*, pages 49–60. Springer Berlin Heidelberg, 1994. ISBN 978-3-540-57956-4. 61

J. Park, T Oh, J. Jung, Y. Tai, and I Kweon. A tensor voting approach for multi-view 3d scene flow estimation and refinement. In *European Converence on Computer VIsion*, 2012. 48

C. Posch, D. Matolin, and R. Wohlgenannt. An asynchronous time-based image sensor. In *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, pages 2130–2133, 2008. doi: 10.1109/ISCAS.2008.4541871. 75

Christoph Posch, Daniel Matolin, and Rainer Wohlgenannt. A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds. *IEEE Journal of Solid-State Circuits*, 46:259–275, February 2011. 13, 73, 75, 76, 78

L. H. Quam. Hierarchical warp stereo. In *Image Understanding Workshop*, New Orleans, Lousiana,, December 1984. 4

Heather Read. Visual receptive fields: Physiology, organization and development. http://read.uconn.edu/PSYC3501/Lecture07/, 2013. [Online; accessed 15-december-2013]. 2

Jenny C. A. Read. A bayesian approach to the stereo correspondence problem. *Neural Comput.*, 14:1371–1392, June 2002. ISSN 0899-7667. doi: http://dx.doi.org/10.1162/089976602753712981. URL http://dx.doi.org/10.1162/089976602753712981. 3

P. Rogister, R. Benosman, S.H. Ieng, P. Lichtsteiner, and T. Delbruck. Asynchronous event-based binocular stereo matching. *IEEE Transactions on Neural Networks*, 23: 347–353, 2011. 14, 15, 23, 37, 38, 39, 45, 78, 87, 93, 95, 96, 106

John Ross. Stereopsis by binocular delay. *Nature*, 248(5446):363–364, March 1974. doi: 10.1038/248363a0. URL http://dx.doi.org/10.1038/248363a0. 11

Sébastien Roy and Ingemar J. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. In *ICCV*, pages 492–502, 1998. URL http://dblp.uni-trier.de/db/conf/iccv/iccv1998.html#RoyC98. 9

Hideo Saito and Takeo Kanade. Shape reconstruction in projective grid space from large number of images. In *CVPR*, pages 2049–2054. IEEE Computer Society, 1999. ISBN 0-7695-0149-4. URL http://dblp.uni-trier.de/db/conf/cvpr/cvpr1999.html#SaitoK99. 9

Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vision*, 47:7–42, April 2002. ISSN 0920-5691. doi: 10.1023/A:1014573219977. URL http://dl.acm.org/citation.cfm?id=598429.598475. 5, 6, 74, 83

Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. In *Proceedings of the 2003 IEEE computer society conference on Computer vision and pattern recognition*, CVPR'03, pages 195–202, Washington, DC,

USA, 2003. IEEE Computer Society. ISBN 0-7695-1900-8, 978-0-7695-1900-5. URL http://dl.acm.org/citation.cfm?id=1965841.1965865. 6

Steven M. Seitz and Charles R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 1067–, Washington, DC, USA, 1997. IEEE Computer Society. ISBN 0-8186-7822-4. URL http://dl.acm.org/citation.cfm?id=794189.794361. 9

Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR '06, pages 519–528, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2597-0. doi: 10.1109/CVPR.2006.19. URL http://dx.doi.org/10.1109/CVPR.2006.19. 9

Pradeep Sen, Billy Chen, Gaurav Garg, Stephen R. Marschner, Mark Horowitz, Marc Levoy, and Hendrik P. A. Lensch. Dual photography. *ACM Trans. Graph.*, 24(3): 745–755, July 2005. ISSN 0730-0301. doi: 10.1145/1073204.1073257. URL http://doi.acm.org/10.1145/1073204.1073257. 6

Teresa Serrano-Gotarredona and Bernabé Linares-Barranco. A 128×128 1.5% contrast sensitivity 0.9% fpn $3\mu s$ latency $4mw$ asynchronous frame-free dynamic vision sensor using transimpedance preamplifiers. *J. Solid-State Circuits*, 48(3): 827–838, 2013. URL http://dblp.uni-trier.de/db/journals/jssc/jssc48.html#Serrano-GotarredonaL13. 13

J. Shi and C. Tomasi. Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition*, Ithaca, NY, USA, 1993. Cornell University. 14

Kazuhiro Shimonomura, Takayuki Kushima, and Tetsuya Yagi. Binocular robot vision

emulating disparity computation in the primary visual cortex. *Neural Networks*, 21: 331 – 340, 2008. ISSN 0893-6080. doi: 10.1016/j.neunet.2007.12.033. URL `http://www.sciencedirect.com/science/article/pii/S089360800700247X`. 14

S.N. Sinha, M. Pollefeys, and L. Mcmillan. Camera network calibration from dynamic silhouettes. In *IEEE conference on Computer Vision and Pattern Recognition*, pages 195–202, 2004. 38

Sudipta N. Sinha and Marc Pollefeys. Multi-view reconstruction using photo-consistency and exact silhouette constraints: A maximum-flow formulation. In *ICCV*, pages 349–356. IEEE Computer Society, 2005. ISBN 0-7695-2334-X. URL `http://dblp.uni-trier.de/db/conf/iccv/iccv2005-1.html#SinhaP05`. 9

Gregory G. Slabaugh, Thomas Malzbender, W. Bruce Culbertson, and Ronald W. Schafer. Improved voxel coloring via volumetric optimization. Technical report, Siemens Corporate Research, 2000. 9

Gregory G. Slabaugh, W. Bruce Culbertson, Thomas Malzbender, Mark R. Stevens, and Ronald W. Schafer. Methods for volumetric reconstruction of visual scenes. *International Journal of Computer Vision*, 57(3):179–199, 2004. URL `http://dblp.uni-trier.de/db/journals/ijcv/ijcv57.html#SlabaughCMSS04`. 9

D.M.Y. Sommerville. *Analytic Geometry of Three Dimensions*. Cambridge University Press, 1934. 50

T. Svoboda, D. Martinec, and T. Pajdla. A convenient multi-camera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14: 407–422, 2005. 27, 35

Richard Szeliski. A multi-view approach to motion and stereo. In *CVPR*, pages 1157–1163. IEEE Computer Society, 1999. ISBN 0-7695-0149-4. URL `http://dblp.uni-trier.de/db/conf/cvpr/cvpr1999.html#Szeliski99`. 9

Richard Szeliski, Ramin Zabih, Daniel Scharstein, Olga Veksler, Vladimir Kolmogorov, Aseem Agarwala, Marshall Tappen, and Carsten Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(6):1068–1080, June 2008. ISSN 0162-8828. doi: 10.1109/TPAMI.2007.70844. URL `http://dx.doi.org/10.1109/TPAMI.2007.70844`. 6

Y. Taguchi, B. Wilburn, and C. L. Zitnick. Stereo reconstruction with mixed pixels using adaptive over-segmentation. *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008. doi: 10.1109/CVPR.2008.4587691. URL `http://dx.doi.org/10.1109/CVPR.2008.4587691`. 6

Camillo J. Taylor. Surface reconstruction from feature based stereo. In *ICCV*, pages 184–190. IEEE Computer Society, 2003. ISBN 0-7695-1950-4. URL `http://dblp.uni-trier.de/db/conf/iccv/iccv2003-1.html#Taylor03`. 10

C. Tomasi and J. Zhang. Is structure-from-motion worth pursuing? In *Proceedings of the Seventh International Symposium on Robotics Research*, 1995. 47

F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008. doi: 10.1109/CVPR.2008.4587677. 83

Adrien Treuille, Aaron Hertzmann, and Steven M. Seitz. Example-based stereo with general brdfs. In Tomás Pajdla and Jiri Matas, editors, *ECCV (2)*, volume 3022 of *Lecture Notes in Computer Science*, pages 457–469. Springer, 2004. ISBN 3-540-21983-8. URL `http://dblp.uni-trier.de/db/conf/eccv/eccv2004-2.html#TreuilleHS04`. 9

S. Vedula, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. In

International Conference on Computer Vision, volume 27-3, pages 475–480, 1999. 47, 48

Olga Veksler. Stereo correspondence by dynamic programming on a tree. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2 of *CVPR '05*, pages 384–390, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2372-2. doi: 10.1109/CVPR.2005.334. URL http://dx.doi.org/10.1109/CVPR.2005.334. 6

George Vogiatzis, Philip H. S. Torr, and Roberto Cipolla. Multi-view stereo via volumetric graph-cuts. In *CVPR (2)*, pages 391–398. IEEE Computer Society, 2005. ISBN 0-7695-2372-2. URL http://dblp.uni-trier.de/db/conf/cvpr/cvpr2005-2.html#VogiatzisTC05. 9

Liang Wang, Miao Liao, Minglun Gong, Ruigang Yang, and David Nister. High-quality real-time stereo using adaptive cost aggregation and dynamic programming. In *Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pages 798–805, 2006. ISBN 0-7695-2825-2. doi: 10.1109/3DPVT.2006.75. URL http://dx.doi.org/10.1109/3DPVT.2006.75. 6

Zeng-Fu Wang and Zhi-Gang Zheng. A region based stereo matching algorithm using cooperative optimization. In *IEEE Conference on Computer Vision and Pattern Recognition - CVPR 2008*, pages 1–8, jun 2008. doi: 10.1109/CVPR.2008.4587456. URL http://dx.doi.org/10.1109/CVPR.2008.4587456. 6

Andreas Wedel, Thomas Brox, Tobi Vaudrey, Clemens Rabe, Uwe Franke, and Daniel Cremers. Stereoscopic scene flow computation for 3d motion understanding. *Int. J. Comput. Vision*, 95(1):29–51, October 2011. ISSN 0920-5691. doi: 10.1007/s11263-010-0404-0. URL http://dx.doi.org/10.1007/s11263-010-0404-0. 48

A. Wenger, A. Gardner, C. Tchou, J. Unger, T. Hawkins, and P. Debevec. Performance

relighting and reflectance transformation with time-multiplexed illumination. In *ACM SIGGRAPH 2005 Papers*, pages 756–764, 2005. doi: 10.1145/1186822.1073258. URL http://doi.acm.org/10.1145/1186822.1073258. 6

P. M. Will and K. S. Pennington. Grid coding: a preprocessing technique for robot and machine vision. In *Proceedings of the 2nd international joint conference on Artificial intelligence*, IJCAI'71, pages 66–70, San Francisco, CA, USA, 1971. Morgan Kaufmann Publishers Inc. URL http://dl.acm.org/citation.cfm?id=1622876.1622884. 6

L. Xu and J. Jia. Stereo matching: An outlier confidence approach. In David Forsyth, Philip Torr, and Andrew Zisserman, editors, *IEEE on European Conference on Computer Vision.*, volume 5305 of *Lecture Notes in Computer Science*, chapter 57, pages 775–787. Springer Berlin, Berlin, Heidelberg, 2008. ISBN 978-3-540-88692-1. doi: 10.1007/978-3-540-88693-8_57. URL http://www.cse.cuhk.edu.hk/~{}leojia/all_final_papers/stereo_eccv08.pdf. 6

Q. Yang, R. Yang, J. Davis, and D. Nistér. Spatial-depth super resolution for range images. In *IEEE Conference on Computer Vision and Pattern Recognition - CVPR 2007*, 2007. 6

Qingxiong Yang, Liang Wang, Ruigang Yang, Henrik Stewénius, and David Nistér. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *IEEE Transation on Pattern Analysis and Machine Intelligence*, 31(3):492–504, March 2009. ISSN 0162-8828. doi: 10.1109/TPAMI.2008.99. URL http://dx.doi.org/10.1109/TPAMI.2008.99. 6

Ruigang Yang, Marc Pollefeys, and Greg Welch. Dealing with textureless regions and specular highlights - a progressive space carving scheme using a novel photo-consistency measure. In *ICCV*, pages 576–584. IEEE Computer Society, 2003. ISBN 0-7695-1950-4. URL http://dblp.uni-trier.de/db/conf/iccv/iccv2003-1.html#YangPW03. 9

Mark Young, Erik Beeson, James Davis, Szymon Rusinkiewicz, and Ravi Ramamoorthi. Viewpoint-coded structured light. In *IEEE Conference on Computer Vision and Pattern Recognition - CVPR 2007*, pages 1–8, June 2007. URL `http://graphics.cs.berkeley.edu/papers/Young-VCS-2007-06/`. 6

Gang Zeng, Sylvain Paris, Long Quan, and François X. Sillion. Progressive surface reconstruction from images using a local prior. In *ICCV*, pages 1230–1237. IEEE Computer Society, 2005. ISBN 0-7695-2334-X. URL `http://dblp.uni-trier.de/db/conf/iccv/iccv2005-2.html#ZengPQS05`. 9

Li Zhang and Shree Nayar. Projection defocus analysis for scene capture and image display. In *ACM SIGGRAPH 2006 Papers*, pages 907–915, 2006. ISBN 1-59593-364-6. doi: 10.1145/1179352.1141974. URL `http://doi.acm.org/10.1145/1179352.1141974`. 6

Li Zhang, B. Curless, and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *First International Symposium on 3D Data Processing Visualization and Transmission - 2002*, pages 24–36, 2002. doi: 10.1109/TDPVT.2002.1024035. URL `http://dx.doi.org/10.1109/TDPVT.2002.1024035`. 6

Ye Zhang, Chandra Kambhamettu, and Ra Kambhamettu. On 3d scene flow and structure estimation. In *In IEEE Conf. on Computer Vision and Pattern Recognition*, pages 778–785, 2001. 48

Todd E. Zickler, Peter N. Belhumeur, and David J. Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. *Int. J. Comput. Vision*, 49(2-3): 215–227, September 2002. ISSN 0920-5691. doi: 10.1023/A:1020149707513. URL `http://dx.doi.org/10.1023/A:1020149707513`. 6

C. Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon A. J. Winder, and Richard Szeliski. High-quality video view interpolation using a layered representa-

tion. *ACM Trans. Graph.*, 23(3):600–608, 2004. URL `http://dblp.uni-trier.de/db/journals/tog/tog23.html#ZitnickKUWS04`. 9