Poznań University of Technology
Faculty of Electronics and Telecommunications
Chair of Multimedia Telecommunication and Microelectronics

Doctoral Dissertation

# Hybrid techniques of depth map estimation and their application in three-dimensional video systems

Maciej Kurc

Supervisor: Prof. dr hab. inż. Marek Domański

Poznań, 2019

Politechnika Poznańska
Wydział Elektroniki i Telekomunikacji
Katedra Telekomunikacji Multimedialnej i Mikroelektroniki

Rozprawa Doktorska

# Hybrydowe techniki wyznaczania map głębi i ich wykorzystanie w systemach obrazu trójwymiarowego

Maciej Kurc

Promotor: Prof. dr hab. inż. Marek Domański

Poznań, 2019

I would like to thank all important people in my life, especially my beloved Parents, who have always been in the right place and supported me in difficult moments, especially during realization of this work.

I would like to express special thanks and appreciation to professor Marek Domański, for his time, help and ideas that have guided me towards completing this dissertation.

Chciałbym podziękować wszystkim ważnym osobom w moim życiu, szczególnie moim Rodzicom, które zawsze były we właściwym miejscu i wspierały mnie w trudnych chwilach, w szczególności podczas realizacji niniejszej pracy.

Chciałbym również wyrazić szczególne podziękowania oraz wyrazy wdzięczności panu profesorowi Markowi Domańskiemu, za jego czas, pomoc oraz pomysły, które doprowadziły mnie do ukończenia tej rozprawy.

# Table of contents

# List of concepts, symbols and abbreviations

***Concepts:***

| | |
|---|---|
| Active depth sensing | A distance/depth acquisition technique based on a form of active illumination of the scene followed by its observation by a sensor. |
| Camera calibration | A process in which the intrinsic, extrinsic or both sets of parameters of a camera are estimated. |
| Camera distribution line | A line in 3D space that passes through the optical centres of all cameras of a linear multi-camera system |
| Confidence map | A 2D array, provided by a depth camera, in which each sample represents confidence of a distance measurement. |
| Depth camera | A camera capable of registering distance, possibly with additional information such as confidence and/or monochromatic image. |
| Depth estimation | A process of computation of a depth map using information from two or more video cameras and possibly additional data. |
| Depth map | A 2D array in which every sample represents a distance measured along the camera optical axis. |
| Distance map | A 2D array, provided by a depth camera, in which each sample represents a physical distance to its corresponding point of the scene. |
| Intensity image | An image, provided by a ToF camera, formed by reflected light wave amplitude measurements. |
| Inter-view consistency | A feature of multiple depth maps that indicates consistency of a 3D scene representation between them. |
| Multi-camera system | A system that consists of multiple video and/or depth cameras. |
| Linear multi-camera system | A multi-camera system in which video cameras are arranged on a straight line and are oriented toward the same direction. |
| Multi-view video sequence | A video sequence recorded using a multi-camera system. It consists of a number of individual video sequences captured by each camera. |
| Rectification | A process of geometrical transformation of views of a multi-view sequence that virtually modifies camera rotation and translation. |
| Structured lighting | An active depth sensing method in which a pattern is projected onto the scene. The pattern is observed by one or more cameras and the depth is inferred from its distortions. |
| Time-of-Flight | An active depth sending technique in which a light emitter sends a modulated light wave. The light wave reflects from the scene back to the camera's image sensor. The depth is derived from phase difference of the modulating signal measured between the transmitted and reflected light wave. |
| View | A colour image acquired from a camera, which is a part of a multi-camera system. |
| Virtual view | An artificially generated image that represents a 3D scene seen from a different point of view than any of the cameras. |

***Abbreviations:***

| | |
|---|---|
| AVC | Advanced Video Coding |
| DERS | Depth Estimation Reference Software |
| DIBR | Depth Image Based Rendering |
| DLT | Direct Linear Transformation |
| FPGA | Field Programmable Gate Array |
| HEVC | High Efficiency Video Coding |
| LiDAR | Light Detection and Ranging |
| LED | Light Emitting Diode |
| SVD | Singular Value Decomposition |
| MVC | Multi-View Video Coding |
| MV-HEVC | Multi-View High Efficiency Video Coding |
| PSNR | Peak Signal to Noise Ratio |
| SAD | Sum of Absolute Differences |
| SSD | Sum of Squared Differences |
| SSIM | Structural Similarity |
| ToF | Time-of-Flight |
| VSRS | View Synthesis Reference Software |

***Mathematical nomenclature:***

| | |
|---|---|
| $\mathbf{A}$ | matrix A |
| $a_{ij}$ | element i,j of matrix A |
| $\vec{v}$ | vector v |
| $v_x$ | component x of vector v |
| $k, K$ | scalars k and K |
| $\mathbf{A}^t, a_{ij}^t, \vec{v}^t, v_x^t, k^t, K^t$ | an element with time index t |

# 1. Introduction

## 1.1 Scope of the dissertation

This dissertation addresses problems related to acquisition of depth, fusion of video and depth data, and to the compression of multi-view plus depth video sequences.

A depth map can be computed using visual information from a stereo camera pair or from a multi-camera system. The process is known as depth estimation. It is based on finding correspondences between two or more images from neighbouring cameras. These correspondences, in conjunction with the camera parameters, are used to obtain the distance information.

A different method of obtaining depth information is direct distance measurement using active sensing techniques such as e.g. Structured Lighting [Szeląg_01][Scharstein_02] or Time-of-Flight (ToF) [Horaud_01]. Structured Lighting depth acquisition techniques are based on the projection of a light pattern which varies in space and/or time. Observations of distortions of the pattern are used to infer about the distance. One of a widely used application of structured lighting depth acquisition is digitization of cultural heritage artefacts [Sitnik_01]. On the other hand, Time-of-Flight techniques are based on measurements of light wave travel time.

Distance measurement approach based on Time-of-Flight has its application in e.g. Light Detection and Ranging (LiDAR) devices [Liu_01] and ToF cameras [Horaud_01]. A LiDAR device consists of one or multiple lased distance measuring units which are based on the ToF principle. Usually either a rotating mirror is used or the whole LiDAR device is rotating continuously to extend its field of view. LiDARs find its applications mostly in autonomous vehicle navigation [Gargoum_01] and geodetic measurements [Glennie_01].

In this dissertation, **the author focuses on Time-of-Flight (ToF) depth cameras**. A ToF camera uses a light emitter to illuminate the scene with an amplitude modulated light wave [SR4000]. The camera provides two types of information derived from the reflected light: modulating signal amplitude and the modulating signal phase. Amplitude data can be treated as an intensity image that corresponds to the operating wavelength of the camera. The signal modulation phase carries information about the distance. The distance is derived from phase difference measured between the transmitted and reflected light wave. A ToF camera can also provide a third kind of information, namely confidence, which is derived from the amplitude data. It provides information about how certain a particular distance measurement is. Confidence information is useful as it allows to identify uncertain measurements.

Both the depth estimation and depth acquisition method have major drawbacks. The estimated depth maps are often inaccurate due to insufficient amount of reliable image features that can be matched [Mieloch_01][Sterp_01]. Also the process of depth estimation requires large amounts of

memory and computing power. Therefore depth estimation is very slow and cannot be performed during video sequence acquisition – it has to be done afterwards. The depth maps, acquired using Structured Lighting methods, are accurate but the acquisition process is too slow to be used for acquiring video sequences. Moreover, it requires multiple images to be captured. On the other hand depth maps, acquired using a Time-of-Flight camera, yield low spatial resolution and usually contain a significant amount of noise.

In this dissertation the author proposes a method of **video and acquired depth data fusion** in order to obtain **higher quality depth maps** that can be achieved using either the estimation or the acquisition technique. Higher quality means that a depth map is more accurate in terms of conformance of the distance it represents with the actual physical distance in 3D space. Moreover, the author focuses also on improvements to algorithms related to multi-camera system calibration and video and depth data processing.

A multi-camera system is required in order to record multi-view plus depth sequences. The system must operate synchronously, i.e. each camera captures a frame at precisely the same time instant. Different types of cameras require different types of external synchronisation signal; therefore, a synchronisation signal conversion is required. The author proposes **a method of synchronisation-signal conversion** which allows to synchronise **ToF depth cameras with video cameras**.

The author also addresses problems related to camera **parameter estimation of multi-camera systems with video and depth cameras**. In this dissertation the author focuses on linear multi-camera systems only. Accurate camera parameters are crucial for the depth estimation process and for video and depth data fusion; therefore the author proposes a set of improvements to the state-of-the-art camera parameter estimation parameters and to the multi-view rectification algorithms.

Another major issue that is addressed in this dissertation is **multi-view plus depth video compression**. State-of-the-art multi-view compression techniques such as MVC [AVC_02], MV-HEVC [HEVC_02] and 3D-HEVC [HEVC_03] strongly rely on inter-view similarities of input video and depth data. Inter-view similarities are present in multi-view video sequences as they represent the same scene. This creates a redundancy which is exploited by a multi-view codec. Unfortunately, depth data usually exhibit low inter-view similarity (or consistency) because each depth map is estimated independently by using a different subset of available views. In this dissertation the author introduces a **new method of improvement of depth map inter-view consistency** for multi-view plus depth map sequences. The method is aimed at improving multi-view compression of depth information by exchanging information between the depth maps of neighbouring cameras.

## 1.2 Goals and thesis of the dissertation

The primary goal of this dissertation was to develop a method of **fusion of video and acquired depth data.** The fusion will allow to obtain a higher quality depth map that can be estimated using stereo correspondence information or acquired using a depth acquisition technique only.

There are also other goals of the dissertation that address issues related to the preparation of depth and video data for the fusion process. These issues include: **synchronisation of different types of cameras**, **video camera parameter estimation**, **depth camera parameter estimation, image rectification** and **noise reduction in ToF depth data**.

**Improvement of multi-view compression of depth maps** is another goal of this dissertation. Multi-view oriented compression algorithms rely on similarities in neighbouring views; unfortunately, estimated depth maps usually do not exhibit sufficient similarities. The author will address this problem by proposing an algorithm that will improve the inter-view consistency of existing depth maps.

There are two theses considered in this dissertation:

> "Relatively simple design of a system for hybrid depth acquisition with the use of time-of-flight depth cameras and video cameras allows to obtain higher quality depth maps as compared to systems based on either time-of-flight cameras or video analysis only."

> "Improvement of inter-view depth map consistency that increases depth map quality and increases the compression efficiency of compression algorithms which exploit inter-view relations of depth maps."

## 1.3 Overview of the dissertation

Proving the theses of the dissertation required additional effort regarding construction of a multi-camera system with ToF cameras, making video sequence recordings and processing acquired data. All the experiments took place in years 2011 - 2012. The state-of-the-art data processing algorithms regarding camera parameter estimation, image rectification and ToF camera calibration at that time were not suitable for the kind of processing required by the author. Therefore the author had to propose additional modifications to them which are included in this dissertation as secondary goals.

All of the proposed modifications were investigated by the author but not thoroughly enough to consider investigation conclusions as exhaustive. The experiments made were constrained by available equipment and other technical means (e.g. availability of only one type of ToF camera). The author is aware that more work is required to fully investigate all the proposed modifications. Nevertheless, detailed experiment descriptions and results followed by conclusions are presented in this dissertation.

### *1.3.1 Chapter 2*

In chapter 2, the author presents his original **modifications to state-of-the-art camera parameter estimation and image rectification algorithms** for linear multi-camera systems constructed using video and ToF depth cameras.

The methods that are presented in chapter 2 are not the main goal of this dissertation; however, they are necessary for correct data preparation for video and depth map fusion. Thus these modifications were not as thoroughly investigated by the author as the data fusion process itself. The results presented here confirm the ideas of the proposed improvements but may not be enough to state that the proposed methods are general.

An ideal linear multi-camera system has all optical centres of the cameras equally spaced on a straight line. Their optical axes are parallel to one another and perpendicular to that line. It is also often assumed that all of the cameras are identical (have identical parameters). These hard constraints are almost impossible to meet when constructing such a system, thus the goal of the image rectification process is to create a multi-view image set as it would have been captured by an ideal linear multi-camera system by using images captured by an existing, non-ideal one.

For a real linear multi-camera system it is essential to estimate the camera distribution line so that it passes as close to all of the optical centres of the cameras as possible. The author suggests using a 3D linear regression for this problem. Because the resulting line will not pass through the camera positions directly, an appropriate image rectification procedure is suggested that uses image feature points. If all optical axes of the cameras are not perpendicular to the camera distribution line, then during the rectification process some of the image data will be shifted out of the image frame. In order to prevent this, the author proposes a technique that modifies the distribution line direction so that it is perpendicular to the mean optical axis direction.

Different types of cameras require different approaches to their parameter estimation. ToF cameras exhibit very low resolution which makes image features more difficult to be precisely localised. Depth cameras provide additional information, namely the depth map, which the author suggests to use for their parameter estimation. By knowing the 3D coordinates of points in the scene it is possible to estimate relative extrinsic parameters between two or more depth cameras directly instead of using optimisation algorithms that use only 2D image correspondences.

### *1.3.2 Chapter 3*

Chapter 3 is devoted to ToF camera **depth data preprocessing**. The author addresses problems related to distance measurement calibration and distance measurement correction based on the fact that a depth camera measures the distance along a direct light wave propagation path while a depth map represents distances measured along the camera optical axis. The proposed preprocessing methods also include noise parameter estimation used in the motion-aware depth map noise reduction algorithm.

The author proposes a method of **distance measurement compensation for ToF cameras**. The method allows to compensate for systematic error hypothetically introduced by a ToF camera. The approach is based on distance measurements obtained using both ToF and image of a 2D calibration pattern of a known structure and dimensions.

Depth maps from ToF cameras cannot be used directly for multi-view processing as their representation differs from the commonly known representation of a depth map. A ToF camera measures the distance between its sensor and an object along the shortest path of light wave propagation; however, a depth map represents the Euclidean distance measured along the camera optical axis. The author proposes a method of translation between these two representations that is based on the camera's intrinsic parameters.

A depth map provided by a ToF camera is noisy, especially for low reflective objects. The author introduces an **efficient noise reduction algorithm** that consists of two independent filtration procedures. The first step uses a motion-aware temporal filter that derives motion information from amplitude information using estimated camera noise parameters. The second step incorporates a spatial, edge-adaptive bilateral filter that operates on distance and amplitude data.

### *1.3.3 Chapter 4*

Chapter 4 is dedicated to the problem of **synchronisation in multi-camera systems** and related issues. The author introduces a method of synchronisation between television video cameras that use a Genlock signal for synchronisation and industrial cameras (such as ToF depth cameras) which require a single electrical trigger pulse per frame.

The author provides a detailed description of his own constructed hardware device that provides the means for conversion from a Genlock input signal to the trigger signal. The described device was

used by the author to capture multi-view video plus depth test sequences that were used for an evaluation of the proposed algorithms for video and depth data fusion.

### *1.3.4 Chapter 5*

In chapter 5 **the author presents his most important achievement, which is depth map estimation via fusion of video data from video cameras and depth data from ToF depth cameras.**

The author proposes to modify the state-of-the-art depth estimation algorithm that is based on energy minimisation via global optimisation algorithms such as Belief Propagation or Graph Cuts. The author introduces a modification to the global cost function so that not only information from stereo correspondence is used but also additional depth cues from the depth cameras. Moreover, the author also proposes another modification that allows to take advantage of video camera image edges. The use of image edges allows to estimate depth maps that have very sharp edges on object boundaries, which is a much desired feature.

### *1.3.5 Chapter 6*

In chapter 6 the author addresses **multi-view video plus depth compression**. Video plus depth compression algorithms, such as the state-of-the-art HEVC-3D [HEVC_03] and 3D-AVC [AVC_04] algorithms, use inter-view video and depth inter-view similarities. The compression ratio is highly dependent on inter-view video and depth consistency. Images from video cameras are consistent as the scene looks very similar from different points of view that are close to one another; however, the same cannot be said about depth maps, which are estimated independently by using information from different video cameras – and this is the source of inconsistencies between the views.

The author proposes an innovative **depth map inter-view consistency improvement algorithm** that allows to improve consistency without having to re-estimate them. The proposed algorithm is based on an iterative inter-view information exchange between all depth maps of a multi-view sequence. The results have proven the positive influence of the algorithm on the multi-view compression ratio without any quality degradation based on a representative set of multi-view plus depth sequences.

### *1.3.6 Chapter 7*

Chapter 7 summarises the dissertation by indicating all of the author's original achievements along with possible future research paths.

# 2. Estimation of parameters in a multi-camera system and image rectification

## 2.1 Introduction

In order to perform a depth estimation process on a multi-view video sequence, the parameters of a multi-camera system used for their acquisition must be known. The camera parameters can be divided into two groups: **intrinsic parameters** and **extrinsic parameters**. The **intrinsic parameters define the characteristics of each camera itself**. On the other hand, the **extrinsic parameters define the camera's placement with relation to the other cameras in a common global coordinate system**. Both the intrinsic and extrinsic parameters must be known for each camera in order to fully characterise a multi-camera system.

In this dissertation, the author focuses on linear multi-camera systems with video and depth cameras. Such systems consist of a number of video cameras which are spaced evenly on a straight line. Each video camera has its optical axis perpendicular to the line. Video camera orientations are identical. On the other hand, depth cameras need not to be placed on the same line as video cameras. The can be placed anywhere provided that their fields of view overlap with field of views of the video cameras.

The author is going to consider depth estimation algorithms that require multi-view sequences with rectified and lens distortion free images. The goal of the multi-view rectification process is to simplify stereo correspondence search by performing geometrical transformations of all the images. In a rectified multi-view sequence, the stereo correspondence search can be limited to horizontal direction only. This, in turn, allows to reduce overall computational complexity of the depth estimation process.

The accuracy of camera parameter estimation is crucial for the depth estimation algorithms. Inaccurate camera parameters will cause depth estimation algorithms to perform stereo matching on non-corresponding regions of the images. This, in turn, will lead to incorrect depth maps; therefore, the author focused on improvement of camera parameter estimation and image rectification algorithms in order to increase estimated parameters accuracy.

The proposed improvements of a multi-camera system calibration includes estimation of the camera distribution line direction, correction of the line direction and rectification of a multi-view video sequence using camera calibration pattern feature points.

The author paid special attention to the estimation of depth camera parameters. Depth cameras differ in their characteristics from video cameras. Usually, their image resolution is much lower than the resolution of a video camera. The depth cameras provide distance information which can be used to estimate their extrinsic parameters. In this chapter the author proposes a method which allows to

estimate the relative extrinsic parameters of two or more depth cameras using the distance information they provide.

## 2.2 Camera model and parameters

### 2.2.1 Intrinsic parameters

The projective camera model was used to model the camera in 3D space. The model is also known as the **Pinhole Camera Model**. It provides a base to formulate most of the relations in a modelled multi-camera system. It is well known from the literature and widely used for stereoscopic and multi-camera systems [Hartley_02][Cyganek_01][Weng_01][Zhang_01].

The model assumes that all light rays passes to a single point in 3D space known as the principal point. This corresponds to a camera which has an infinitely small aperture instead of a lens. The model is correct also for cameras with lenses provided that all visible objects are much more distant than the focal length of the lens. Figure 2.2.1 illustrates a model of a thin lens [Hecht_01].

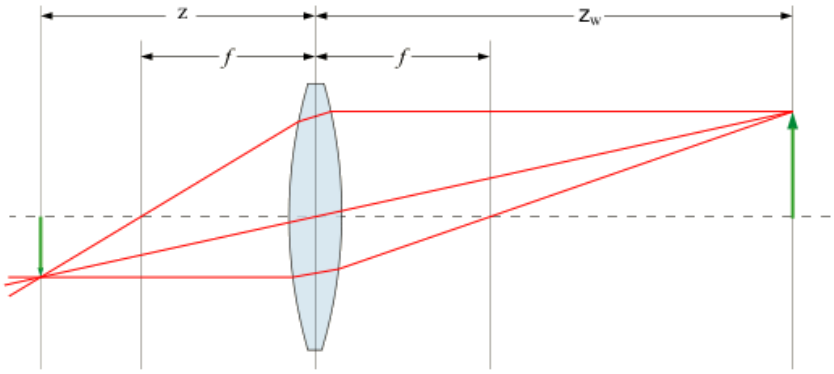



Figure 2.2.1 – Illustration of the thin lens model [Hecht_01].

The thin lens formula is given by the Equation 2.2.1 :

$$\frac{1}{z}+\frac{1}{z_w}=\frac{1}{f}, \tag{2.2.1}$$

where $z$ denotes distance from image sensor to the lens, $z_w$ denotes distance from an object to the lens and $f$ denotes the focal length. Once it is assumed that $z_w >> f$, the term $\frac{1}{z_w}$ can be neglected which allows to conclude that $z=f$. This means that images of all objects are formed exactly on the camera sensor.

Figure 2.2.2 shows the idea of the pinhole camera model.

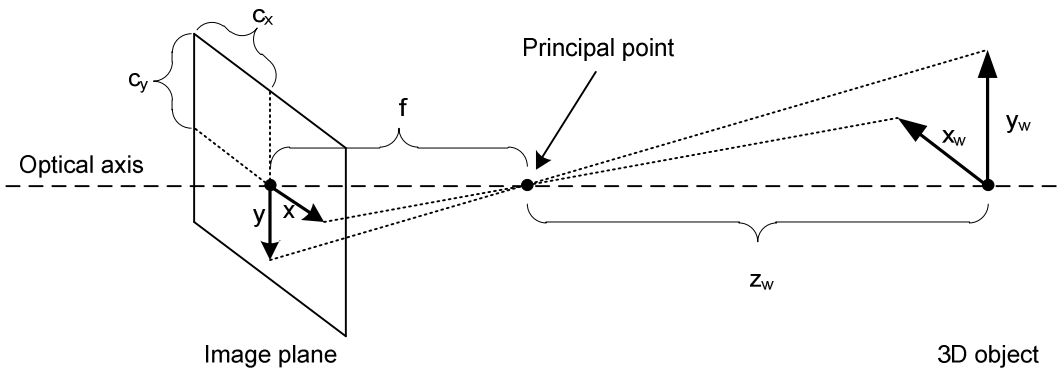



Figure 2.2.2 – Illustration of the pinhole camera model.

The transformation between a point in 3D space and a corresponding point on the image plane is given by the equation:

$$z \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{K} \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix}, \ \mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.2.2}$$

where $x_w$, $y_w$, $z_w$ are the coordinates of a point in the 3D space, $x$, $y$ are its projection coordinates on the image plane and $z$ is a scaling factor. The matrix **K** in Equation 2.2.1 is called the **intrinsic matrix**. The parameters $f_x$, $f_y$ define the **focal length** for projection in each dimension and $c_x$, $c_y$ define the **principal point** position within the image. Focal lengths $f_x$, $f_y$ are expressed in multiples of the horizontal and vertical spatial sampling periods of the image sensor, respectively. These two focal lengths may not be equal for an image sensor with non-equal horizontal and vertical sampling periods. The difference may also be caused by a non-spherical (e.g. anamorphic or cylindrical) camera lens [Maxwell_01].

An extension of the pinhole camera model is the **lens distortion model**, which defines the non-linear geometric image transformation that reflects the distortion caused by the camera lens or the lens system [Brown_01]. It is defined by the following equations:

$$x_d = x \cdot \frac{1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6}{1 + k_4 \cdot r^2 + k_5 \cdot r^4 + k_6 \cdot r^6} + 2 \cdot p_1 \cdot x_u \cdot y_u + p_2 \cdot (r^2 + 2 \cdot x_u^2), \tag{2.2.3}$$

$$y_d = y \cdot \frac{1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6}{1 + k_4 \cdot r^2 + k_5 \cdot r^4 + k_6 \cdot r^6} + p_1 \cdot (r^2 + 2 \cdot y_u^2) + 2 \cdot p_2 \cdot x_u \cdot y_u, \tag{2.2.4}$$

and $r$ is given by:

$$r = \sqrt{x_u^2 + y_u^2}\,. \tag{2.2.5}$$

Variables $x_u$ and $y_u$ denote undistorted pixel coordinates, $x_d$ and $y_d$ denote lens-distorted pixel coordinates. The lens distortion model can be characterised by coefficients $k_1$ to $k_6$ and $p_1$ to $p_2$. Coefficients $k_1$ to $k_6$ define the 1st, 2nd and 3rd order radial distortion, while coefficients $p_1$ to $p_2$ define the tangential distortion. Depending on the required model complexity, some of them may be set to zero, thus indicating that a particular type of distortion is not present.

The lens distortion model is independent from camera parameters such as focal length and principal point position. Equations 2.2.3 and 2.2.4 define solely radial and tangential lens distortion effects. It must be noted, that the lens distortion model does not affect the scaling factor $z$. It remains the same after lens distortion removal.

The lens-distorted pixel coordinates $x_d$ and $y_d$ are defined in normalised image space. The normalised image coordinate system assumes that the principal point is located at position (0, 0) and the image extends from (-1,-1) to (+1,+1). The perspective projection from 3D world coordinates to 2D normalised image coordinates is given by the following equation [Hartley_02] :

$$z \cdot \begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix}. \tag{2.2.6}$$

Equation 2.2.6 defines a perspective projection with unit focal length and principal point at position (0,0). In order to de-normalise the pixel coordinates after application of the lens distortion model, equation 2.2.7 needs to be applied:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{K} \cdot \begin{bmatrix} x_u \\ y_u \\ 1 \end{bmatrix}. \tag{2.2.7}$$

In the Equation 2.2.7 the actual intrinsic matrix of the camera is used [Hartley_01], [Cyganek_01]

### 2.2.2 Extrinsic parameters

The extrinsic parameters of a camera are not directly related with the pinhole camera model. The extrinsic parameter matrix defines the relation between the global coordinate system and the local coordinate system of a camera. The relation is defined by the following equation [Hartley_01]:

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} \cdot \left[\mathbf{R}|\vec{t}\right]^{-1} = \begin{bmatrix} x_g \\ y_g \\ z_g \\ 1 \end{bmatrix}, \ \mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \ \vec{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}, \tag{2.2.7}$$

where $x_g$, $y_g$, $z_g$ are coordinates in the global coordinate system, $x_w$, $y_w$, $z_w$ are coordinates in the camera coordinate system, $r_{ij}$ are the **rotation-related** coefficients and $t_i$ are the **translation** coefficients. The matrix in equation 2.2.7 is known as the **extrinsic matrix**.

## 2.3 Camera parameters estimation

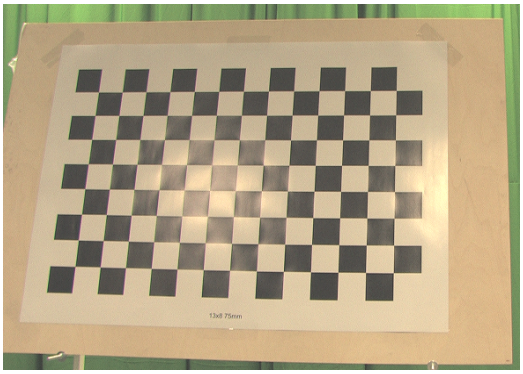
All parameters of the camera model correspond to the physical dimensions and placement of the camera. The simplest way to estimate the camera model parameters is to directly measure all of them; however, due to finite manufacturing precision and difficulties in accessing some elements of the camera, the measured parameters might exhibit too low accuracy to be used. Thus, it is better to estimate the camera parameters using information derived from the spatial structure of a calibration object with known dimensions and its image formed by the camera.

Numerous techniques of estimation of the camera model parameters are known from the literature. Most of them incorporate the use of a dedicated calibration pattern that is shown to the camera. The calibration pattern provides a set of feature points which can be easily detected on the image. The feature points locations on the pattern in conjunction with their corresponding locations on the image are then used to estimate the camera parameters [Zhang_01][Tsai_01]. Different kinds of techniques make use of the sole scene features [Dwarakanath_01][Xu_01][Liu_01] by using well-known feature point detectors such as SIFT [Lowe_01] or SURF [Bay_01]. Unfortunately these methods are sensitive to errors due to possible ambiguities during scene features identification. The technique that incorporates a calibration pattern board [Zhang_01] proves to be the most useful for multi-camera systems, thus it is going to be used for the mixed video plus depth multi-camera system as proposed by the author.

The pattern-based parameter estimation techniques mostly use a planar pattern which defines an *M*-by-*N* grid of feature points. The author has used two types of planar calibration patterns:

- Checkerboard pattern – A rectangular grid of alternating black and white squares. Feature points are defined by corners of adjacent squares.
- Circle grid pattern – A rectangular grid with black circles on white background. Features are defined by centres of the black circles.

Figure 2.3.1 shows photographs of the calibration patterns that were used by the author. The table 2.3.1 summarises parameters of those patterns.



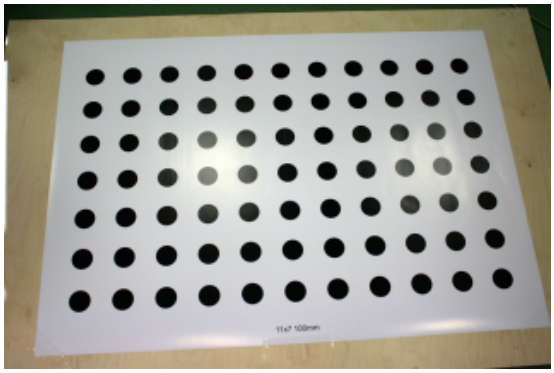


Figure 2.3.1 – Checkerboard calibration pattern (left) and a circle grid pattern (right) used by the author.

Table 2.3.1 – Parameters of calibration patterns used by the author

| **Pattern name** / **Parameter** | **Checkerboard pattern** | **Circle grid pattern** |
|---|---|---|
| Number of features | 13 by 8 | 11 by 7 |
| Feature spacing | 75 mm | 100 mm |
| Board dimensions | 1.1 m x 0.84 m (A0 paper sheet) | |

As the relations between the positions of the calibration pattern features and their locations on an image are known, it is possible to estimate both the intrinsic and extrinsic parameters of the camera. Unfortunately, a single image is not enough to solve the mathematical problem of parameter estimation; therefore, multiple images are needed. Each image must represent the same calibration pattern but rotated by a different angle, i.e. they cannot be co-planar [Zhang_01].

An image of a calibration pattern must represent it with a high enough spatial resolution in order for the features of the pattern to be accurately located. The accuracy of the locations of these features is crucial for the accuracy of further camera parameter estimation, i.e. the higher the resolution, the more accurate the camera parameters. The author's experience shows, that for a high-resolution

camera, it is better to use a checkerboard pattern. High image resolution allows precise localisation of the checkerboard corners' positions. For a low-resolution camera (such as for a modern ToF depth camera), it is better to use a circle grid pattern. On a low-resolution image, accurate localisation of the centres of the coloured circles can be done with better accuracy than localisation of the checkerboard corners. Therefore, the author proposes to use the circle grid pattern for low-resolution cameras and the checkerboard for high-resolution cameras.

The camera parameters are estimated with accuracy up to the scaling factor because the distance between the camera and the pattern is not known. By viewing a 2D calibration pattern, it is not possible to determine whether it is small and located close to the camera, or whether it is large and located far away from the camera. The knowledge regarding the physical dimensions of the calibration pattern allows to connect the estimated parameters such as focal length and camera translation to their physical counterparts in the real world.

The extrinsic parameters are estimated by finding the camera position with respect to the local coordinate system of the calibration pattern. When the same calibration pattern is seen by several cameras it is possible to determine their relative translations and rotations, which together form the relative extrinsic parameters. The relative extrinsic parameters of a camera pair can be found by knowing the relation between each camera coordinate system and the coordinate system of the calibration pattern.

## 2.4 Image rectification

The goal of the image rectification is to transform images captured by all of the video cameras to look as if they were captured by an ideal multi-camera system [Cyganek_01]. These transformations include:

- lens distortion correction,
- camera rotation correction,
- camera translation correction.

The rectification process ensures that corresponding points are placed on the same horizontal line of each image. Then the correspondence search in a stereo matching algorithm can be constrained to the horizontal direction only. The multi-view video system rectification process itself is a set of geometrical image transformations.

An ideal linear multi-camera system has all cameras uniformly spaced on a line (namely the camera distribution line) with their optical axes perpendicular to this line. Moreover, intrinsic parameters of all the cameras (focal length and principal point position) are meant to be equal.

Due to unavoidable misalignments during the production of individual cameras and their arrangement in the system, it is extremely difficult to have a set of cameras whose positions and parameters meet these constraints. This situation is shown in Figure 2.4.1. The left figure shows the situation for an ideal multi-camera system while the right one shows the situation for a real multi-camera system.

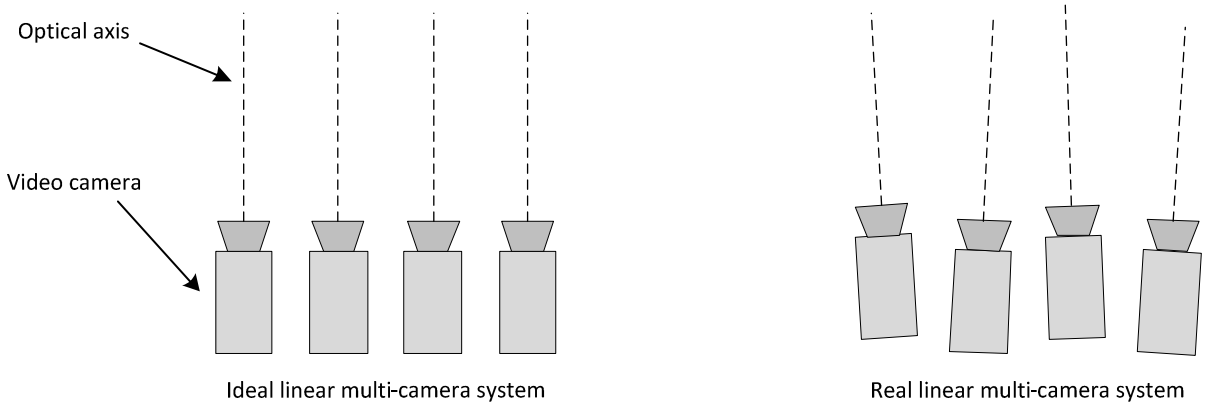



Figure 2.4.1 – Camera positions in a linear multi-camera system. On the left is an ideal system; on the right is a real one.

The data provided by the depth cameras do not require rectification as the images are not used for stereo correspondence search. Depth cameras provide their own depth maps. The only required image transformation process is lens distortion removal in order to make the registered depth map compliant with the pinhole camera model.

There are multiple known techniques of image rectification, yet most of them are meant for a stereo camera pair rather than for a multi-camera system [Huihuang_01][Lin_01]. These techniques are based on epipolar geometry which describes the geometrical relation of a pair of cameras [Hartley_01][Liansheng_01]. Unfortunately, none of those methods can be applied directly to a multi-view system because the rectification has to be done on more than two cameras. Nozick presented his method of multi-view rectification [Nozick_01]; however, the described algorithm does modify the focal length of the camera model, which makes the camera parameters them non-conformant to an ideal linear multi-camera system that assumes all of the focal lengths are equal. A similar method was presented by Kang with an identical drawback [Kang_01].

In his work Perek et. al. describes an algorithm that allows to rectify a stereoscopic pair of images without knowing the camera parameters [Perek_01]. The algorithm takes advantage of local image features. Both images are divided into a two dimensional $m$ by $n$ mesh grid, then vertical disparities for each vertex of the are found. Finally both images are transformed geometrically in such a way that after the rectification, vertical disparities of the grid vertices are equal to zero.

Unfortunately, the described rectification method cannot be used by the author as it operates on a stereoscopic image pairs only.

The algorithm described by Iqbal et. al. also operates on stereoscopic pairs and uses image features [Iqbal_01]. The features are estimated by the SURF algorithm [Bay_01]. The algorithm estimates camera parameters by finding the fundamental matrix of the stereoscopic pair using the eight point algorithm [Hartley_02]. Once the fundamental matrix is known it is decomposed to intrinsic and extrinsic camera parameters. Those are then used to find a geometrical transformation of images that allows to rectify them. The method introduced by Iqbal et. al. is meant for a stereoscopic pair of cameras. Therefore the author cannot use it for a multi-camera system, especially with depth cameras alongside video cameras.

A method that allows rectifying images from a multi-camera system was introduced by Stankowski et.al. in his work [Stankowski_01]. The method was used to postprocess multi-view video sequences acquired at Poznan University of Technology. Those video sequences were later submitted as a response to call for contribution by the MPEG group [MPEG2008/N9468] and became a part of standardised multi-view test video sequences defined by common test conditions for multi-view compression evaluation provided by the MPEG group [CTC].

The multi-view rectification algorithm introduced by Stankowski et. al. is based on estimation of intrinsic and extrinsic camera parameters using Zhang's algorithm [Zhang_01]. Estimated camera parameters are then used for finding a geometrical transformation of all the images which leads to their rectification. The author has taken a similar approach to [Stankowski_01] in his work followed by modification of the original algorithm.

The **author proposes several modifications to the image rectification algorithms** known from the literature. The proposed modifications are aimed at preserving the connection between camera parameters and the physical world. This connection is necessary in order to be able to re-project the depth map captured by the depth camera into the video camera image space that is required for video and depth data fusion.

The author proposes the following modifications and improvements to algorithms known from the literature:

- A new method for estimation of camera distribution line to be used in algorithm described in [Stankowski_01] and [Kang_01].
- A new method of correction of previously estimated distribution line direction in order to compensate for systematic misalignment of video camera rotations. To be used in conjunction with algorithms described in [Stankowski_01] and [Kang_01].

- A modified method for compensation of video camera translation misalignment based on calibration pattern features, also to be used along with algorithm described in [Stankowski_01].

Moreover, the author proposes a new method of estimation of depth camera relative extrinsic parameters using video plus depth information which was not previously mentioned in literature.

## 2.5. Proposed modifications of camera parameters estimation and image rectification algorithms

### 2.5.1 Computation of the camera distribution line of a linear multi-camera system

The image rectification procedure requires that all camera positions and rotations are defined in a common global coordinate system. Because the multi-camera system is assumed to be linear, it is convenient to define one axis of the coordinate system as the direction of the camera distribution line. The line should pass as close to all optical centres of the cameras as possible. The reason for this is that in an ideal linear multi-camera system all the optical centres are located directly on that line. Image rectification corrects their misplacement by applying a geometric transformation to the views. The smaller the distance between the actual optical centre position and the camera distribution line, the less image correction is required.

Because it is impossible to draw a straight line through more than two points in 3D space, the distribution line needs to be approximated. Rectification algorithms, as known from the literature, require certain assumptions about the direction of the distribution. In their work, Kang et al. suggest to estimate the camera distribution line direction by analysing each camera local coordinate system *x* axis direction expressed in a common global coordinate system [Kang_01]. The directions of the *x* axes of each camera coordinate system are iteratively averaged. After each iteration, those directions for which one or more component values do not fall into a specified interval are rejected. At the end, the distribution line direction is assumed to be equal to the direction closest to the average. The drawback of the algorithm by Kang et al. is that it assumes that the camera distribution line direction is close to the direction of the *x* axis of each camera of the system. This assumption is not true in general.

**The author proposes a different approach based on the use of 3D linear regression directly.** The proposed method uses the least mean-squared minimisation of distance between each camera position and the camera distribution line. Equation 2.5.1 shows the error function to be minimised:

$$E(k_a, k_b, k_c, k_d) = \sum_{i=1}^{N} \frac{(k_a \cdot x_i + k_b \cdot y_i + k_c \cdot z_i + k_d)^2}{{k_a}^2 + {k_b}^2 + {k_c}^2}, \tag{2.5.1}$$

where $E$ is the error function. Each term being summed in the equation 2.5.1 corresponds to squared distance between a point and a line in 3D space [Ballantine_01]. The point is defined by the coordinates $x_i$, $y_i$, $z_i$ and corresponds to the position of the $i$-th camera optical centre. Coefficients $k_a$, $k_b$, $k_c$ and $k_d$ define the line in general form.

The line, computed using 3D linear regression, will not pass exactly through all the optical centres of the cameras. This condition, however, is required by the definition of an ideal linear multi-camera system. This implies the need for correction of translation vector of each camera. The correction is aimed at cancelling the offset between camera optical centre and the distribution line.

The problem is that the appropriate image transformation that would reflect this correction cannot be done without knowing the $z$ coordinates of pixels forming the image. The $z$ coordinates can be derived from the depth map, which has not been estimated yet. Because the rectification process is performed prior to the depth estimation, it is impossible to rectify the images in a general situation. Fortunately, translation compensation can be approximated when the distance between the camera system and the scene is much greater than the focal length of each camera. For such a case it can be assumed that all image pixels have an unknown but identical depth value. The image transformation may then be approximated by an affine transformation instead of a full DIBR view synthesis process [DIBR].

Because depth cameras are not subjected to the rectification process, the distribution line direction is estimated using the video cameras' positions only. However, the line defines a new global coordinate system, so the depth camera extrinsic parameters need to be transformed to the new coordinate system accordingly in order to maintain consistency within the rest of the multi-camera system.

### 2.5.2 Correction of the camera distribution line direction

In this chapter the author proposes to modify the camera distribution line direction. A linear multi-camera system is expected to have all of the cameras oriented perpendicularly to their distribution line; therefore a part of the image rectification process is the camera **rotation correction**. The goal of correction of the cameras' relative rotations is to **make the cameras oriented in the same direction**, perpendicular to the line.

If there is a systematic camera orientation misalignment with respect to the desired optical axis direction, image rectification will lead to a loss of information. The image transformation that is intended to correct the camera rotation will move a significant part of the image pixels outside of the image frame.

The situation is shown in Figures 2.5.1 and 2.5.2. The figures show the cameras as seen from above for better clarity. The grey area indicates the image data of an un-rectified image.

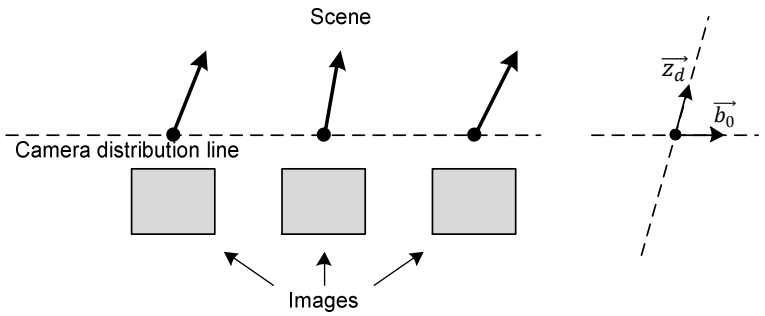



Figure 2.5.1– Camera positions and optical axes directions with corresponding images before rotation correction.

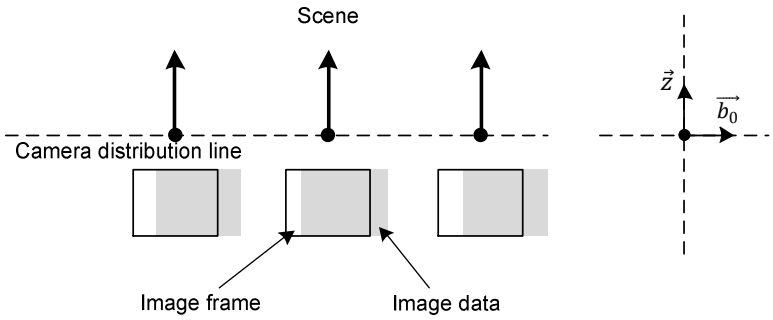



Figure 2.5.2 – Camera positions and optical axes directions with the corresponding images after rotation correction. A part of the image data was moved outside the image frame.

The direction vector $\overrightarrow{b_0}$ denotes estimated direction of the camera distribution line. The direction vector $\overrightarrow{z_d}$ corresponds to mean optical axis direction for all video cameras. Depth cameras are not taken into account as they are not subjected to the rectification as described in chapter 2.4. Finally, the direction vector $\vec{z}$ corresponds to the Z axis in the coordinate system after the image rectification.

The author proposes a solution which is intended **to modify the line direction itself** in order for it to be perpendicular to the average camera orientation, as is shown in Figure 2.5.3.

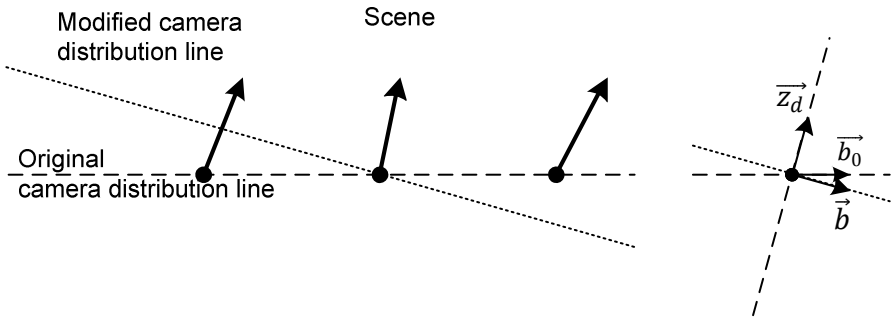



Figure 2.5.3 – Derivation of a new line according to the optical axes directions of the cameras.

The direction vector $\vec{b}$ indicates direction of the new camera distribution line. By changing the distribution line direction so that it is perpendicular to the average orientation of the camera system, only the relative rotations of the cameras have to be compensated for. The procedure proposed by the author is to do a calculation of the mean optical axis direction vector, i.e. the orientation of the system, and then make it orthogonal to the line direction vector according to formula 2.5.2:

$$\vec{b} = \overrightarrow{b_0} - (\overrightarrow{b_0} \cdot \overrightarrow{z_d})\overrightarrow{z_d}\,, \tag{2.5.2}$$

where $\vec{b}$ is the new distribution line direction vector, $\overrightarrow{b_0}$ is the original distribution line direction vector and $\overrightarrow{z_d}$ is the mean optical axis direction vector computed for all of the video cameras (but not for the depth cameras).

The proposed camera distribution line modification technique leads to the elimination of image data loss caused by rotation correction; however, the change in the distribution line direction from its optimal direction causes the cameras to be positioned further away from it. This, in turn, implies the need for 3D translation correction which cannot be performed without a depth map. This state is shown in Figure 2.5.4.

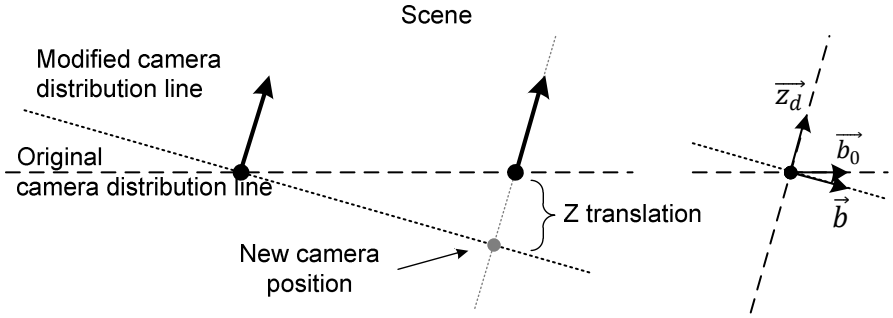



Figure 2.5.4 – Illustration of the translation correction problem when the direction of the camera distribution line is changed.

For small translation corrections (measured in millimetres) related to the distance between the camera and the objects (measured in meters) in the scene, the correction can be approximated by a geometric image transformation instead of the full DIBR view synthesis [DIBR]. The DIBR algorithm cannot be used because the depth map is not yet known during image rectification. The author proposes his method for geometrical image transformation, which is based on location of camera calibration pattern features. The technique is described in detail in chapter 2.5.4.

### 2.5.3 Experiments related to the proposed method

The proposed modification of the camera distribution line direction allows to fit more pixels from the original image into the rectified image, thus preventing loss of data. The author conducted an experiment to observe the scale of this effect.

The experiment consisted of two setups of a full HD camera stereo pair. In the first setup the cameras were oriented perpendicularly to the system distribution line (90 degree angle), while in the second setup their optical axes formed an approximately 95 degree angle with the line, as is shown in Figure 2.5.5.

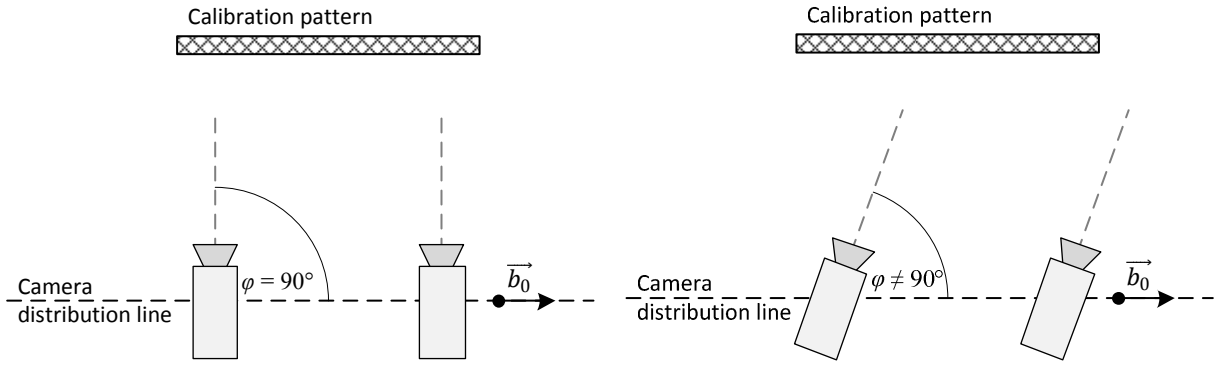



Figure 2.5.5 – Two experimental setups of the cameras. On the left, the optical axes are perpendicular to the distribution line; on the right, they form a specific angle $\varphi$ with the line.

The 95 degree angle was chosen because it is a realistic case of camera misalignment. Choice of an angle closer to 90 degrees would not allow to demonstrate advantages of the proposed method. On the other hand angle smaller than 95 degrees would be and unrealistic example. Despite that the proposed method is not limited to the angle chosen.

The cameras were oriented toward the same calibration pattern. The checkerboard pattern was used with parameters summarised in table 2.3.1. A series of three test sequences were captured for each setup. Each sequence represents the calibration pattern placed at a different position with respect to the camera system. Its images were used to estimate the extrinsic parameters of the cameras. The relative extrinsic parameters of the cameras were estimated for each calibration pattern position independently and then averaged in order to remove the influence of noise. The intrinsic parameters of both cameras were assumed to be known (they were estimated independently before the experiment).

In the experiment, two rectification transformations were computed for both of the camera setups. The first transformation was based on an estimation of the camera system distribution line direction that uses only the camera positions given in their extrinsic parameters. The second transformation incorporated the same distribution line estimation method followed by its direction modification as proposed by the author. For the second rectification method, camera translation

corrections were necessary as the modified line does not pass through the camera positions directly. New camera positions were found by projecting their original translation vectors onto the new line. The translation correction was performed using feature positions taken from calibration pattern images. A detailed description of the procedure is given in chapter 2.5.4.

Figures 2.5.6 and 2.5.7 show the example rectification results of the experiment, where the optical axis angle $\varphi$ was equal to 95 degrees. Figure 2.5.6 shows the rectification result using the original, unmodified camera distribution line, while Figure 2.5.7 shows the result when the distribution line was modified using the algorithm proposed by the author.

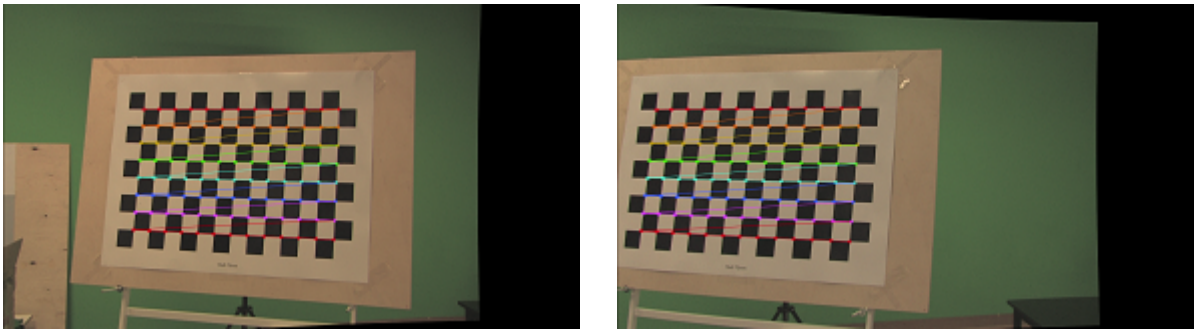


Figure 2.5.6 – Rectification result using an original, unmodified camera distribution line.

Figure 2.5.7 – Rectification result using the modified camera distribution line according to the author's proposal.

The rectified images, shown in Figure 2.5.6, exhibit a large area near their right edges where there is no image data. Some parts of the images were lost as the pixels near the left edges of the original images were projected outside of the image frame. The images, rectified using the proposed algorithm, do not exhibit such data loss.

In order to measure the number of pixels lost during the rectification transformation, the author suggests to use an objective measure. The measure is defined as a percentage of the image frame that was not filled with image data after the rectification. The measure is defined according to equation 2.5.3:

$$r_{empty} = \frac{n(\Omega)}{N} \cdot 100\%, \tag{2.5.3}$$

where $r_{empty}$ denotes the percentage of image frame not filled with usable data, Ω is the set of pixels not assigned to any actual colour value, $n(\Omega)$ denotes the cardinality of set Ω while *N* represents the number of pixels in the image.

Table 2.5.1 summarises the measure value for each test case of the experiment.

Table 2.5.1 – Percentage of the image frame that was not filled with data after the rectification.

| Angle of optical axes of the cameras and their distribution line | Value of $r_{empty}$ for the original distribution line $\overrightarrow{b_o}$ | | Value of $r_{empty}$ for the modified distribution line $\vec{b}$ | |
|---|---|---|---|---|
| φ = 90º | Right camera:<br>Left camera:<br>Average: | 1.2%<br>3.0%<br>2.1% | Right camera:<br>Left camera:<br>Average: | 1.2%<br>0.9%<br>1.3% |
| φ = 95 º | Right camera:<br>Left camera:<br>Average: | 20.2%<br>17.3%<br>18.7% | Right camera:<br>Left camera:<br>Average: | 1.9%<br>0.3%<br>1.1% |

According to the experimental results, the algorithm proposed by the author allows to significantly reduce data loss during the rotation correction stage of the image rectification process. The average value of $r_{empty}$ ranged about 1%, which is much less than for rectification without the proposed modification. The 5 degree systematic error of the camera orientation alignment was chosen purely to prove the proposed algorithm's ability to properly rectify images in such cases.

The amount of data loss depends mostly on the misalignment of the optical axes of the camera system. Results of the experiment show that for the assumed 90 degree angle the optical axes were not perfectly perpendicular to the distribution line as they supposed to be. This yielded in about 2% of empty area of the image frame according to the proposed $r_{empty}$ metric. The proposed algorithm allowed to reduce the empty space in that case to about 1% which shows its potential.

The number of pixels that lies outside the image frame will never reach zero, as the rectification process needs to perform a geometrical correction of the image, which implies that the transformed image will not be rectangular. The correction of lens distortion also introduces data loss that cannot be counteracted when using the proposed method.

Although the results presented in table 2.5.1 indicates that the proposed algorithm allows to reduce number of lost pixels (as measured using the $r_{empty}$ metric) , the single experiment is not enough

to consider those results to be general. The author is aware that the proposed method requires more thorough investigation which would reach beyond the scope of the dissertation.

### 2.5.4 Correction of camera translation with respect to the modified distribution line

Camera translation vector describes its position in the global coordinate system in 3D space. It is one of the camera extrinsic parameters. Once a translation vector of a camera is found, it requires a correction so that it describes position of the optical centre of the camera on the camera distribution line.

As it was stated in chapter 2.5.1 it is generally not possible to draw a straight line through more than two points in 3D space. Therefore optical centres of all the cameras will not be positioned directly on the camera distribution line. This implies a need for a translation correction. Moreover the method proposed by the author deliberately changes direction of the camera distribution line pushing it even further from optical centres of the cameras**. In this chapter author proposes his own approach to the camera translation correction** based on techniques known from the literature.

The assumptions about an ideal multi-camera system made in this dissertation states that all the cameras must have equal focal lengths. Fulfilment of this assumption is not possible in practise. Therefore **the algorithm proposed by the author in this chapter allows correcting not only camera translations but also non-equal focal lengths.**

During rectification, the translation is performed by computing an adequate 2D geometric transformation of an image that reflects camera translation correction in 3D space. The translation of a camera in 3D space causes a **change of the point of view**. In order to transform the image accordingly, a **depth map is required**. The depth map defines the distance between the camera and each observed point of the scene, measured along the optical axis of the camera. Because some parts of the observed scene are located closer to the camera than others, translations of their counterparts on a 2D image plane will be different. This is the main idea of the DIBR view synthesis algorithm [DIBR]. Without knowing the depth map it is impossible to define the translation for each point, hence it is impossible to transform the image.

Fortunately, if the distance between the camera and the scene is much greater than the focal length of the camera, the change to the point of view is insignificant. It is assumed then that all image pixels have the same depth. The method proposed by the author is based on that assumption.

There are several techniques known from the literature that are aimed at rectification of multi-view images. For example Kang et. al. describes a method of rectification of multi-camera system images which corrects for vertical image misalignment by modifying intrinsic parameters of each camera [Kang_01]. The modification changes principal point positions. The author did not take advantage of the described method as during the rectification. In his proposed approach intrinsic

parameters of all the cameras should remain identical and the images are to be transformed to become conformant with those parameters.

Stankowski et. al. introduce an image rectification algorithm that operates in two steps [Stankowski_01]. During the first step the geometrical image transformation, necessary for the rectification, is computed using estimated extrinsic parameters of a camera. Then, during the second step, a 2D perspective transformation of the image is computed. Parameters of the transformation are estimated using four feature points of the calibration pattern that was used to find extrinsic parameters of the camera.

Unfortunately, none of the methods found in the literature takes advantage of the major assumption of the linear multi-camera system - the cameras are uniformly spaced along their distribution line. Therefore the author proposes to incorporate this assumption into his proposed rectification method.

The goal of the image transformation is to:

- have all the vertical coordinates of all the feature points equal for all views and
- have horizontal positions compliant with the camera locations along the distribution line.

The transformation, which meets these requirements, will also correct uneven focal lengths of all the cameras implicitly.

The author proposes a three-step correction procedure which consists of:

- estimation of vertical image shift based on calibration pattern features,
- estimation of horizontal image shift based on calibration pattern features and assumption about uniform spacing of the cameras along their distribution line,
- computation and application of a 2D homographic transformation to the image which does the actual rectification.

An important assumption of the proposed method is that the multi-camera system is linear and that all of the cameras are equally spaced on the line. The proposed method is not suitable for a general case of camera arrangement.

***Correction of vertical image shift***

According to the method proposed by the author, the new vertical coordinate $y_i$ of each feature point can be computed as an average of its vertical positions from all views according to the formula 2.5.4:

$$y_i = \frac{1}{N} \cdot \sum_{j=0}^{N-1} y_j \,, \tag{2.5.4}$$

where $y_i$ is the vertical coordinate of a feature point on the *i*-th camera image and *N* is the number of cameras. Figure 2.5.8 illustrates the method of correction of a vertical image shift; $\Delta y$ denotes the necessary vertical correction of the vertical position of a feature point.

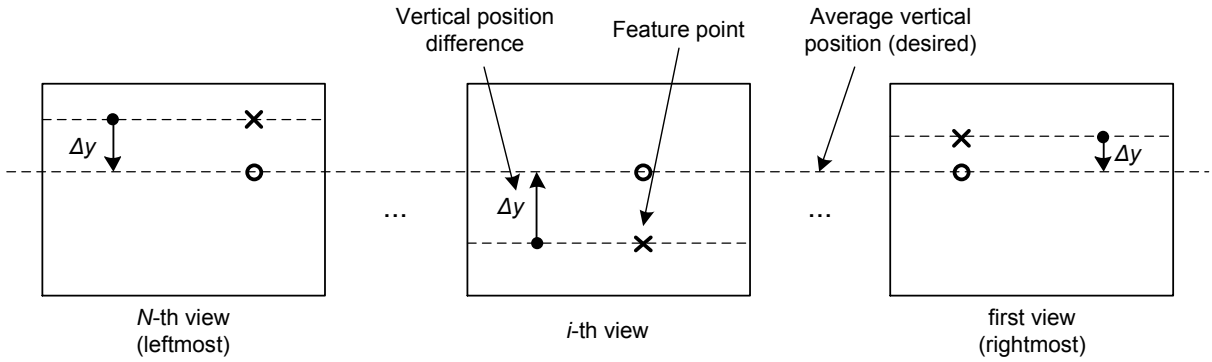



Figure 2.5.8 – Illustration of the proposed method of correction of a vertical image shift. The "X" symbol indicates an observed position of a feature point while the "O" symbol denotes its desired position.

***Correction of horizontal image shift***

When the cameras are uniformly spaced on the distribution line and the distribution line direction is parallel to the *x* axis, the relative position of a feature point with respect to each camera in 3D space differs only in the *x* coordinate while the other coordinates remain identical. Assuming that all of the cameras are positioned exactly on the distribution line and their intrinsic parameters are equal, we can write the following formula for the projection of the *i*-th feature point. The formula is derived from the pinhole camera model (equation 2.2.2):

$$x_i = \frac{x_w - t_{xi}}{z_w} - c_x \,, \tag{2.5.5}$$

where $x_i$ is the horizontal coordinate of projection of a feature point for the *i*-th camera, $x_w$,$y_w$ and $z_w$ are its 3D coordinates, $t_{xi}$ is the translation of the *i*-th camera along the distribution line in 3D space taken from its extrinsic parameters. The variable $c_x$ is the horizontal coordinate of the optical centre location on the image plane.

Let us assume that the camera with index 0 is the rightmost one, i.e. it has the lowest value of $t_{ix}$ among all the cameras. Similarly, the *N-th* camera is the leftmost one, i.e. the value of $t_{ix}$ is the greatest. Because all variables except for $t_{ix}$ are constant, the relation becomes linear. This means that if the cameras are equally spaced between the *0-th* and the *N-th* one, projections of a feature point should also be uniformly spread between the *0-th* and the *N-th* view. This observation is the basis of the desired horizontal coordinate estimation for the feature points. For each feature point, its leftmost and rightmost position is found among all the views, then for the rest of the views its value is interpolated linearly according to the following formula:

$$t_{xi} = \frac{i}{N-1} \cdot (t_{xN} - t_{x0}) + t_{x0}\ , \tag{2.5.6}$$

where $t_{xi}$ is the translation along the distribution line of the *i*-th camera and *N* is the number of cameras in the system. Figure 2.5.9 illustrates the principle of the proposed method.

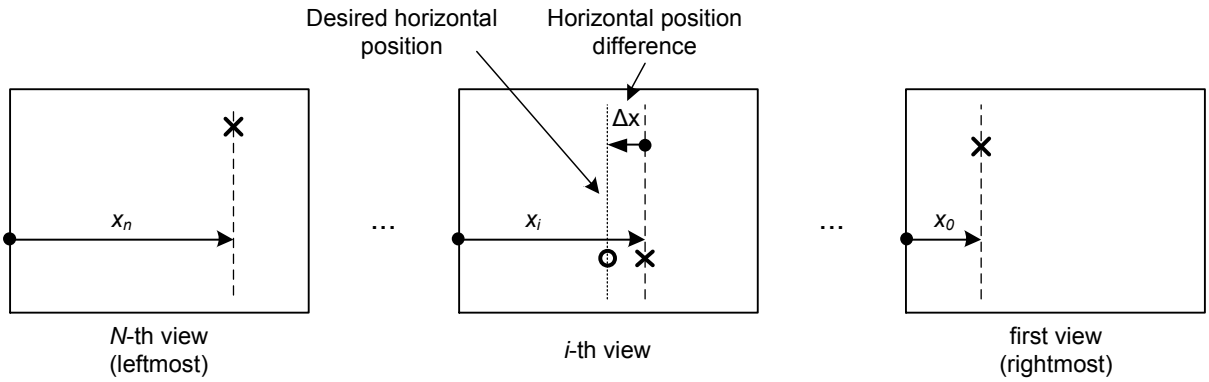



Figure 2.5.9 – Illustration of the proposed method of horizontal image shift correction. The "X" symbol indicates an observed position of a feature point while the "O" symbol denotes its desired position.

***Geometric image transformation***

Once the relations between the detected feature points' locations and their desired locations are known, the image transform can be computed using the direct linear transform (DLT) method [Hartley_02]. The image transformation has the form of a 2D homographic transformation. It is defined by the equation:

$$w \cdot \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{2.5.7}$$

where *x, y* are the spatial coordinates of an un-rectified image pixel and *x', y'* are the desired coordinates of a rectified image pixel and *w* is a scaling factor. Coefficients $h_{ij}$ form a 2D homographic transformation matrix.

Unfortunately, the DLT algorithm is very sensitive to the presence of outliers in the input data set (point correspondences that do not comply with the others). A solution to this problem as proposed by the author is the use of the Random Sample Consensus (RANSAC) algorithm which is based on a random selection of the exact number of correspondences needed for the DLT solution [Fischler_01].

The method proposed by the author has an important advantage over those known from the literature. It allows correcting for horizontal misalignment of the cameras. The author has verified the method on a linear multi-camera system with uniform camera spacing. The proposed method can also be extended to a case, when the cameras are not uniformly spaced along the distribution line, provided that their assumed positions with respect to the leftmost and rightmost camera are known.

## 2.6 Estimation of depth camera extrinsic parameters

### 2.6.1 Introduction

Zhang's algorithm can be used to estimate extrinsic parameters of a video multi-camera system [Zhang_01]. However, the algorithm only takes advantage of 2D coordinates the calibration pattern features. A depth camera can provide additional distance feature. The distance information can be used to either enhance Zhang's algorithm or to develop a new one. **In this chapter the author proposes a method of estimation of extrinsic parameters for two depth cameras by using depth information they provide.**

The Zhang's algorithm requires a single calibration pattern which is visible by all the cameras. Images of the pattern allow estimating transformations between coordinate system of the pattern and each coordinate system of each camera. Those are extrinsic parameters of the cameras.

It is convenient to have one camera as a reference. In such a case extrinsic parameters of all other cameras will define a transformation between the reference camera coordinate system and their local coordinate systems. Such parameters are the relative extrinsic parameters.

Relative extrinsic parameters are computed in the following way. Extrinsic parameters of the *i*-th camera are given by its rotation $\mathbf{R}(i)$ matrix and translation vector $\vec{t}(i)$. The translation vector is defined in local coordinate system of the camera. Let's assume that the reference camera is the 0-th one. Then, the relative rotation matrix and translation vector for the *i*-th camera with respect to the 0-th camera are given by equations 2.6.1 and 2.6.2 [Stankowski_01]:

$$\mathbf{R}_{rel}(i) = \mathbf{R}(0) \cdot \mathbf{R}(i)^{-1}, \tag{2.6.1}$$

$$\vec{t}_{rel}(i) = \mathbf{R}_{rel}(i) \cdot \vec{t}(i) - \vec{t}(0)\,. \tag{2.6.2}$$

The process of finding relative extrinsic parameters is identical for video and depth cameras.

### 2.6.2 Solutions known from the literature

There are several techniques, known from the literature, that are dedicated to the estimation of depth camera parameters:

Fuchs et al. introduced a method that provides a means of calibrating distance measurements combined with extrinsic parameter estimation [Fuchs_01]. Their experimental setup consisted of a ToF camera that was mounted on a robotic arm and a checkerboard calibration pattern. Unfortunately, the authors of the publication focused mostly on distance measurement calibration, and the topic of extrinsic camera parameter estimation was generally not dealt with.

Jiyoung et al. suggested using a special calibration pattern for depth cameras that do not provide intensity information [Jiyoung_01]. The pattern has the form of a flat board with holes. These holes can easily be detected on a depth map. Their image coordinates were used in Zhang's algorithm [Zhang_01] for the estimation of parameters of the camera. The method did not differ from the one used for video cameras, and only the calibration pattern was different.

Ningbo et al. propose to use a 2D calibration pattern in order to find extrinsic parameters of a depth camera and photo camera [Ningbo_01]. Their method uses Harris corner detector [Harris_01] for checkerboard feature extraction on low resolution ToF amplitude image.

There are also camera calibration algorithms, known from the literature, which are based on 3D calibration patterns. Heikkila suggests using two planar calibration patterns that form a 90 degree angle [Heikkila_01]. Although the pattern was three-dimensional, only its 2D projection was used for the camera parameter estimation. The proposed method did not require depth information.

Herrera et al. describe a method of calibration of two colour cameras (different resolutions) and one depth camera [Herrera_01]. Their method is also based on the Zhang's algorithm and does not take advantage of both intensity and depth information. Feature points of the calibration pattern are needed to be marked manually on the acquired depth map. Also a situation with more than one depth camera is not addressed.

Matusiak et al. propose a technique for matching features, that can be extracted from an image with depth map [Matusiak_01]. The technique is based on modified SIFT algorithm [Lowe_01] which takes advantage of depth information. The feature matching algorithm is also based on SIFT keypoint descriptors. It is not concluded in the publication that the method can be used to match together multiple ToF images (as monochromatic intensity image with ToF depth data) but such application is

possible. Nevertheless the author has proposed his technique which uses a calibration marker instead of features detected in the scene image.

A method of extrinsic parameter estimation for low resolution ToF camera and high resolution video camera is presented by Pertile et al. [Pertile_01]. The publication focuses on enhancement to calibration pattern feature localisation – in this case a checkerboard corners. However, the author proposes a different approach by using a different calibration pattern for low resolution ToF camera which does not require the solution proposed by Pertile et al.

The use of a checkerboard pattern is also suggested by Botezatu et al. [Botezatu_01]. In their systems Botezatu el al. use a stereo camera together with a depth camera capable of registering intensity image in infrared light. Unfortunately, the calibration procedure described in the publication does not relate to taking advantage of depth cues provided by the depth camera used.

A very similar approach, to the one proposed by the author, was introduced by Fukushima [Fukushima_01]. In the publication, a use of the iterative closest point (ICP) method is proposed. The ICP method allows to iteratively match together two set of points in 3D space. However, the author found that for his ToF sensor such method would be unreliable due to presence of noise; hence the author proposed a different approach described in chapter 2.6.3

### 2.6.3 A solution proposed by the author

The author proposes to estimate extrinsic parameters of two depth cameras by **computing a 3D rigid transformation.** A rigid transformation is a concatenation of 3D rotation and 3D translation. The transformation connects two feature point sets, with each one defined in the coordinate system of a different camera. In order to fully characterise a 3D rigid transformation, at least four 3D correspondences are required. By knowing how to transform a point set from one coordinate system to another, the transformation between the coordinate systems themselves can be found. The transformation will then define the relative extrinsic parameters of those two cameras.

The 3D coordinates provided by a depth camera are defined in the coordinate system of a depth map. An inverse projection needs to be applied in order to derive their 3D positions in the local coordinate system of the camera. The 3D coordinates of each sample of a depth map are given by equation 2.6.3. The author assumes no lens distortion or that the lens distortion was removed prior to the extrinsic parameter estimation.

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = \mathbf{K}^{-1} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} z \,, \tag{2.6.3}$$

where $x$, $y$ are the sample coordinates on the depth map, $z$ is its corresponding distance as measured by the depth camera, **K** is the intrinsic matrix of the camera and $x_w$, $y_w$ ,$z_w$ are the point coordinates in the 3D space. As each point is defined in the local coordinate system of the respective camera, a transformation between the corresponding points from two depth cameras also defines the transformation between their coordinate systems.

The rigid transformation can then be used to compute the relative extrinsic parameters for a selected camera pair [Besl_01]. The 3D rigid transformation is given by the equation:

$$\begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix} = \mathbf{R} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} + \vec{t}\,, \tag{2.6.4}$$

where $\vec{p}$ is a point in one ToF camera space and $\vec{q}$ is a point in the other ToF camera space. The matrix **R** defines the rotation part of the rigid transformation while the vector $\vec{t}$ defines the translation in the reference camera coordinate system.

For a system with more than two depth cameras, the extrinsic parameters of each camera are defined as a transformation between its coordinate system and the reference camera coordinate system.

Given a two point sets, each originating from different depth cameras, the translation vector can be computed as a difference between positions of their centroids according to the equation:

$$\vec{t} = \overrightarrow{C_q} - \overrightarrow{C_p}\,, \tag{2.6.5}$$

where $\overrightarrow{C_p}$and $\overrightarrow{C_q}$ are centroid positions of two point sets. It must be noted that a translation computed in this manner is defined in the reference camera coordinate system; however, an extrinsic camera matrix must contain the translation defined in the coordinate system of that camera. Therefore, the translation needs to be transformed according to equation 2.6.6 prior to the extrinsic camera matrix construction:

$$\overrightarrow{t_{local}} = -\mathbf{R} \cdot \overrightarrow{t_{global}}\,, \tag{2.6.6}$$

where $\overrightarrow{t_{local}}$ is the translation vector in the local coordinate system of a camera, $\overrightarrow{t_{global}}$ is the translation vector in a global (or reference) coordinate system and **R** is the rotation matrix of the camera.

The optimal rotation transformation can be computed through the Singular Value Decomposition (SVD) of the coordinate covariance matrix of the two point sets. The covariance matrix is given by the equation:

$$\mathbf{H} = \sum_{j=0}^{N} \left(\overrightarrow{P_j} - \overrightarrow{C_p}\right) \cdot \left(\overrightarrow{Q_j} - \overrightarrow{C_q}\right)^T, \tag{2.6.7}$$

where **H** is the covariance matrix, $\overrightarrow{C_p}$and $\overrightarrow{C_q}$ are the centroid positions and $\overrightarrow{P_j}$ and $\overrightarrow{Q_j}$ are individual point positions in 3D space.

The rotation matrix **R** can be computed by SVD decomposition of the covariance matrix **H**. The SVD decomposition of the covariance matrix is given by the equation 2.6.8 [Arun_01]:

$$\mathbf{H} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T, \tag{2.6.8}$$

where matrices **U**, **S** and **V** are result of the SVD decomposition.

The rotation matrix **R** can be computed according to the equation 2.6.9:

$$\mathbf{R} = \mathbf{V} \cdot \mathbf{U}^T. \tag{2.6.9}$$

Estimation of a rigid transform requires at least four point correspondences. The point correspondence sets, obtained using a depth camera, usually contain many thousands of correspondences, and incorporating all of the correspondences into the rigid transform estimation via the least mean square (LMS) minimisation algorithm yields poor results. This is due to the fact that the LMS solution is prone to outliers, thus the author proposes to use the RANSAC algorithm instead [Fischler_01].

The Random Sample Consensus algorithm chooses four random point correspondences from the whole set, which yield the most accurate transformation. A new 3D rigid transformation is computed during each iteration of the RANSAC algorithm. The point set from the first depth camera remains unchanged, while the other one is modified using that transformation. For an accurate transformation, these two point sets should overlap. The author proposes to measure the accuracy by taking the

Euclidean distance between two corresponding points from both sets which are positioned the farthest apart. The algorithm allows finding the most accurate transformation effectively.

Because only depth cameras provide depth information, the method cannot be used to determine the relative positions of a video and depth camera. The video camera does not provide the required depth data, thus Zhang's algorithm and a calibration pattern must be used in this case.

A set of multiple point correspondences in 3D space is required for estimation of a 3D rigid transformation. In order to precisely localise a point in 3D space, the author proposes to use a calibration marker. The proposed marker provides one feature point, i.e. it must be constructed in such a way that the feature point position can be precisely determined on a low-resolution intensity. As the depth distance measurement quality highly depends on the reflectivity of an object, the marker must exhibit high reflectivity so that the distance can be measured with very high confidence.

The author proposes to use a marker pattern that has the form of a black ring on a white background. The centre of the ring can be found by calculating the centroid position over pixels that form the interior of the ring. The distance can be measured accurately due to its high reflectivity. Figure 2.6.1 shows an image of the marker as proposed by the author.

Figure 2.6.1 – Proposed marker used by the author for depth camera extrinsic parameter estimation.

The detection of marker features requires fusion of the intensity image and the depth map provided by the depth camera. The intensity image provides the means of feature point location while the depth map allows estimating its distance.

***Experimental results***

The author conducted an experiment to prove that the proposed extrinsic calibration method yields better accuracy than solutions that incorporate the 2D calibration patterns known from the literature [Zhang_01], [Jiyoung_01].

The experimental setup consisted of two ToF cameras placed in arbitrary locations. The cameras were oriented in roughly the same direction and their fields of view mostly overlapped. The setup is illustrated in figure 2.6.2.

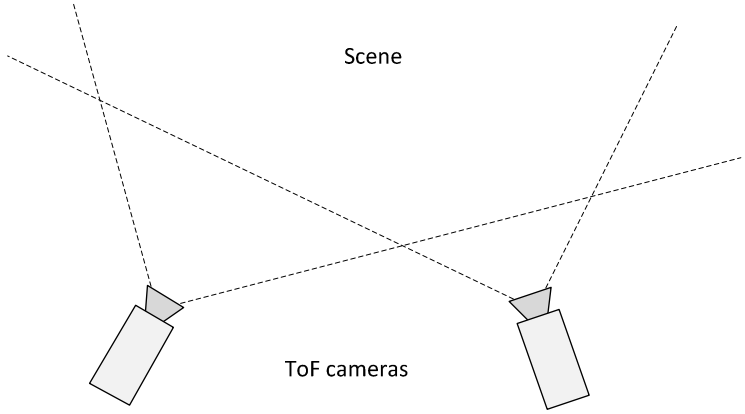



Figure 2.6.2 – ToF cameras' positions for the purpose of the experiment.

The ToF cameras used were the MesaImaging SR4000 [SR4000]. Parameters of those cameras are summarised in the table 2.6.1.

Table 2.6.1 – Summary of relevant parameters of the SR4000 ToF camera.

| **Parameter name** | **Parameter value** |
|---|---|
| Resolution | 176 x 144 pixels |
| Pixel pitch | 40 µm x 40 µm |
| Field of view | 69º horizontal, 56º vertical |
| Frame rate | Up to 50 Hz (depending on settings) |
| Frame acquisition time | 0.3 ms to 25.8 ms |
| Measurement range | 0.8 m to 10.0 m (depending on chosen modulation frequency) |
| Calibrated range | 0.8 m to 8.0m (depending on chosen modulation frequency) |
| Accuracy | +/- 10 mm |
| Modulation frequency | 14.5.0 / 15.0 / 15.5 / 29.0 / 30.0 / 31.0 MHz (selectable) |
| Illumination wavelength | 850 nm |
| Phase data bit depth | 15-bit |
| Amplitude data bit depth | 14-bit |
| External synchronization | Trigger signal input |

A total of four test sequences were recorded. The test sequences were divided into two groups. The first group contained sequences of a 2D calibration pattern board placed about 2 m apart from the

cameras. Each one represented the pattern seen from a different location. A total of three different views of the board were recorded. Multiple views were used to estimate the relative extrinsic parameters between the camera pair. They were then averaged to increase accuracy and reduce the influence of the noise. In the experiment, a 2D circular grid pattern was used. Parameters of the calibration pattern are summarised in table 2.3.1.

The second group contained sequences with the marker proposed by the author. The marker was placed at roughly the same distance from the cameras as the calibration pattern. Each sequence shows the marker placed in a different location in the 3D space. The marker defines a single reference point in the 3D space which is used for extrinsic parameter estimation using the method proposed by the author.

Figure 2.6.3 shows the intensity and distance images for the first case, i.e. where the calibration pattern board is used. Figure 2.6.4 shows the frames for the second case when the proposed marker is used.

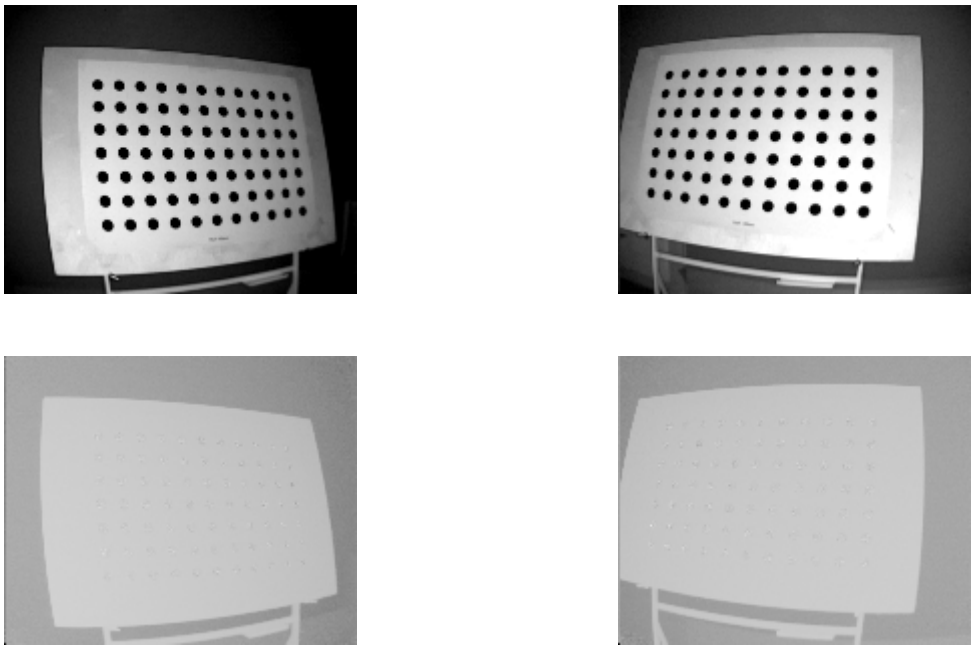


Figure 2.6.3 – Intensity image and distance map for the left and right ToF camera, respectively, of the 2D calibration pattern.

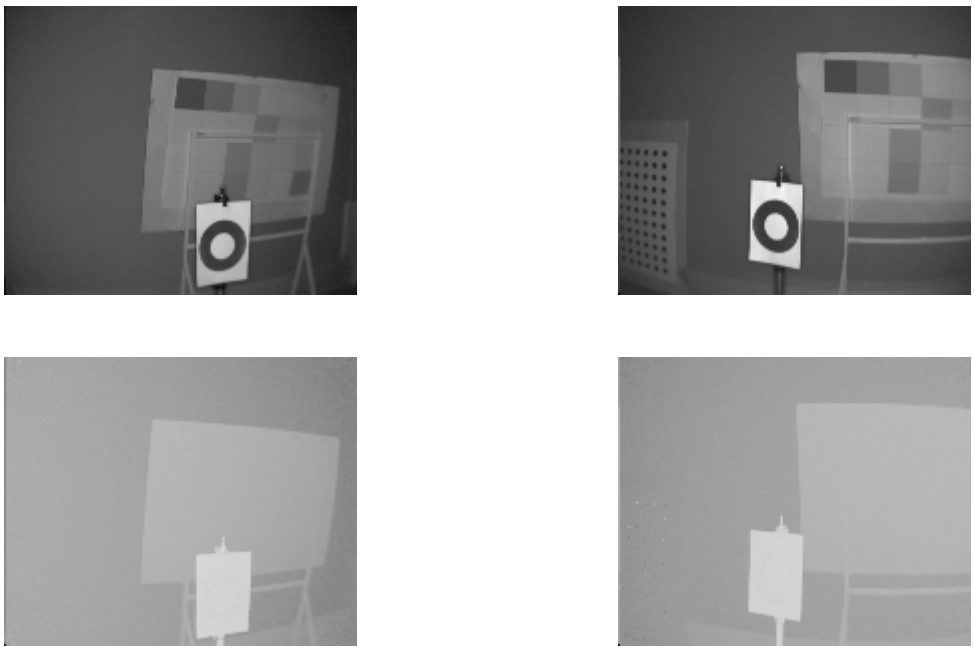


Figure 2.6.4 – Intensity image and distance map for the left and right ToF camera, respectively, of the marker proposed by the author.

For sequences with the calibration pattern, the extrinsic parameters were estimated using Zhang's algorithm [Zhang_01] followed by computation of the relative extrinsic parameters, which, obtained for each board placement, were averaged to improve their accuracy and reduce noise. The second set of extrinsic parameters was estimated using 3D point sets obtained from the images of the marker and distance measurements according to the method proposed by the author. For each marker placement, its position on the image was estimated with sub-pixel precision using a semi-automatic pattern matching technique; the details of this technique are beyond the topic of this dissertation. The distance was taken by sampling the depth map, provided by the ToF camera, at the coordinates of the feature point using a bi-linear interpolation filter.

**In order to assess the quality of the estimated extrinsic parameters objectively, an additional test sequence was recorded** using the SR4000 ToF camera with parameters summarised in table 2.6.1. Those sequences contain a view of the calibration pattern (as described in table 2.3.1) placed about 2 m from the ToF camera. The sequence provided independent sets of points which were not used for the parameter estimation. The 3D coordinates of each point were computed using equation 2.6.1. A set of points was assembled for each ToF camera. All of the point sets were used for quality measurement of extrinsic parameters.

The point set from the first camera was left unchanged; the point set from the second camera was transformed to the coordinate system of the first ToF camera. For perfectly accurate relative

extrinsic parameters, these two point sets should overlap. The higher the parameter inaccuracy, the farther away the corresponding points are located.

The author proposes to assess the quality of extrinsic parameter estimation by computing the root mean squared error (RMSE) over the Euclidean distances between the points in corresponding point pairs. Along with the RMSE, the maximum Euclidean distance (MAX) was also computed. Table 2.6.1 summarises the root mean squared error (RMSE) and the maximum error (MAX) expressed in millimetres.

Table 2.6.1– RMS error and maximum error of the feature point transformation using estimated ToF camera extrinsic parameters.

| **Zhang's method [Zhang_01]** | | **The proposed method** | |
|---|---|---|---|
| RMSE: | 52.8 mm | RMSE: | 21.2 mm |
| MAX: | 72.9 mm | MAX: | 55.5 mm |

According to the data presented in Table 2.6.1, the proposed algorithm allows to estimate the depth camera extrinsic parameters more precisely than the method used by Zhang [Zhang_01], which is confirmed by the numbers. The RMS error of point set transformation was reduced from 52.8mm to 21.2mm, while the maximum distance between two points that should overlap was reduced from 72.9mm to 55.5mm.

The proposed method allows estimating the extrinsic parameters of a depth camera more accurately than the algorithms based on 2D calibration patterns. The proposed algorithm requires a minimum of four 3D point correspondences in order to estimate the relative positions of the two depth cameras. When more than four point correspondences are available, the use of the RANSAC algorithm [Fischler_01] allows choosing those that yield the best parameter accuracy. The least mean squares solution (LMS) of the problem is also possible; however, the LMS solution is more susceptible to the influence of outlier correspondences [Fischler_01].

Unfortunately, the proposed method is applicable only to depth cameras, i.e. it is not possible to estimate the relative extrinsic parameters of a video and depth camera since the video camera does not provide depth information. In this case the method with a 2D calibration pattern must be used (eg. Zhang's algorithm [Zhang_01]); however, when there is more than one depth camera in the system it is possible to increase its accuracy by estimating the relative positions of the video camera and of each depth camera. Then the relative position between the video camera and the reference depth camera can be found. Once multiple sets of the relative extrinsic parameters of these two cameras are known they can be averaged. Averaging the relative extrinsic parameters allows improving their accuracy.

## 2.7 Conclusions

The author has thoroughly investigated the state-of-the-art camera parameter estimation algorithms and managed to incorporate many innovative modifications to them. These modifications allow for accurate parameter estimation of depth cameras as well as of video cameras.

The camera parameter estimation technique which incorporates a 2D calibration pattern proved to be sufficiently accurate for a linear multi-camera system; however, different pattern types should be used for high-resolution video cameras and low-resolution depth cameras. Modifications to the image rectification algorithms as proposed by the author allow rectifying the multi-view sequence with better accuracy than by using the state-of-the-art techniques. The distance measurements provided by depth cameras can be used for their extrinsic parameter estimation, which yields better parameter accuracy than that obtained by using methods with a 2D calibration pattern.

The experiment has proven that the proposed method allows to estimate extrinsic parameters of multiple ToF cameras more accurately than when using the Zhang's algorithm [Zhang_01]. Unfortunately these results are not enough to prove, that the improvement can be observed in general. The author knows that this topic requires further research and investigation. Unfortunately lack of availability of different ToF camera models and constraints regarding scope of the dissertation prevented the author from further experiments.

# 3. Time-of-Flight depth camera measurement correction

## 3.1 Introduction

The raw data that comes from a ToF camera does not have sufficient quality to be used in multi-view systems directly since a significant amount of noise is present. Furthermore, the distance measurements exhibit large systematic errors that need to be corrected.

In this chapter the author proposes algorithms aimed at correcting ToF camera measurements. The proposed algorithms are:

- geometrical correction of distance measurements
- calibration of ToF measurements
- noise reduction for distance data.

The geometrical correction of the distance measurement consists of transformation of the raw measurements from a ToF camera into a depth representation that is commonly used. A depth map represents distance measured parallel to the optical axis of the camera. Unfortunately, ToF cameras measure the distance directly along the light wave propagation path; therefore, a geometrical transformation is required to convert the distance measurement from one representation to another. Usually a ToF camera has a built-in algorithm that performs the correction; however, the algorithm itself and the camera parameters that are used are not available to the user. The lack of availability of these parameters prevents the camera to be used in a multi-camera system. Therefore, the author proposes to do the correction independently, using camera parameters estimated on his own.

The proposed method of ToF camera calibration is aimed at correcting systematic errors of distance measurements. A ToF camera provides internal conversion from phase of the light wave modulating signal to an actual physical distance. However, due to incorrect calibration parameters, the conversion may not be accurate; therefore, the author proposes to use Zhang's algorithm [Zhang_01] to estimate the relative camera position with respect to a calibration board. The relative position will then be used as a reference for ToF distance measurement calibration.

The proposed noise reduction algorithm provides an effective way of reducing the spatial and temporal noise of the acquired depth map. The algorithm uses a motion-adaptive temporal IIR filter together with a spatial bilateral FIR filter. Together, these two filtrations provide an effective means of temporal and spatial noise removal. The method is not destructive to important depth map features such as smooth surfaces and sharp edges, which are crucial for accurate representation of objects.

## 3.2 Geometrical correction of distance measurements

A ToF camera measures modulating signal phase difference of a light wave that originates in the light source of the camera, reflects from objects in the scene and returns through the lens to the image sensor [Hansard_01]. The measured phase difference represents the distance between each pixel of the sensor and a point in the scene that the pixel corresponds to; however, a different distance representation is used for a depth map. A depth map represents the distance measured parallel to the optical axis of the camera. This is illustrated in Figure 3.2.1.

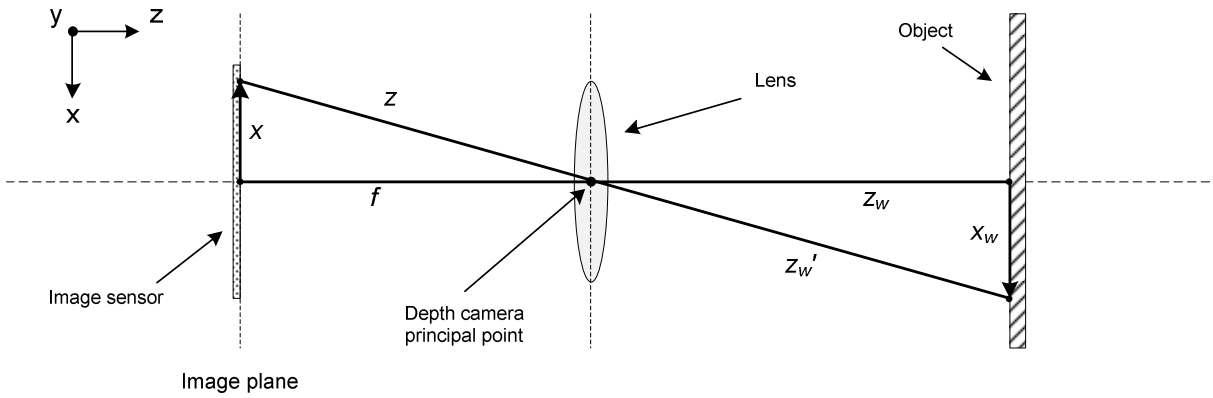



Figure 3.2.1 – ToF camera distance measurement.

The illustration above shows the projection in the *x* dimension only for better clarity, but the presented situation is identical for the *y* dimension. The model assumes no lens distortion or that the lens distortion is fully corrected.

The measured distance value $z_m$ is equal to the distance between the lens and the point in scene $z_w'$ plus the distance between the lens and the corresponding pixel on image sensor $z$; hence $z_m = (z_w' + z)$. However, the depth map is a representation of the value of $z_w$. In order to derive the value of $z_w$ from the measured distance $z_m$, a geometrical correction is required. The geometrical correction method is not aimed at correcting the distance measurement itself but rather the path of the measurement.

There are several methods known from the literature that tend to correct distance measurements themselves [Belhedi_01][Falie_01][Falie_02][Chiabrando_01], but none of them is related directly to correction of the representation of distance measurement. This issue is rarely mentioned in the literature. For most depth cameras the manufacturers provide their own measurement correction procedures [MesaImaging][SR4000]. Unfortunately, these procedures are not documented. The intrinsic parameters are usually embedded in the algorithms and are also not available to the user. Without knowing how intrinsic parameters were used for the correction it is not possible to use identical parameters for video and depth fusion; therefore, **the author proposes to perform the**

**necessary corrections by using the intrinsic parameters estimated using the method proposed by him** [Kurc_02].

Knowing the intrinsic parameters of a ToF camera makes it possible to determine the relation between $z_w'$ and $z_w$ according to the pinhole camera model. As triangles formed by the $z_w$, $x_w$, $z_w'$ and $f$, $x$, $z$ line segments are geometrically similar, we can write an equation, which defines the relation between $z_w$ and $z_w'$:

$$\frac{z_w}{z_w'} = \frac{f}{z}. \tag{3.2.1}$$

By solving the equation for $z_w$ using known geometric relations in a triangle along with the Pythagorean Theorem, the final correctional formula emerges:

$$z_w = z_m \cdot \frac{f}{\sqrt[2]{x^2 + f^2}} - f'. \tag{3.2.2}$$

The first term represents a multiplicative correction of the measured distance. It provides a transformation between the ToF distance that is measured and its corresponding value when measuring along the optical axis of the camera. The second term $f'$ corrects for the additional distance that the light wave has to travel. It is the distance between the lens and the image sensor of the camera. The value of $f'$ needs to be expressed in distance units (the same as the value of $z_m$). As the pinhole model of a camera defines the focal length in multiples of the spatial sampling period, the value of $f'$ can be computed according to the following formula:

$$f' = fs, \text{ and usually } f' \ll z_w \tag{3.2.3}$$

where $s$ is the spatial sampling period. For a projection in two dimensions, the correction transformation as described by formula 3.2.2 needs to be performed for each dimension separately. formula 3.2.2 will then contain an additional multiplicative correction factor (for the second dimension) and a modified additive correction term $f''$:

$$z_w = z_m \cdot \frac{f_x}{\sqrt[2]{x^2 + f_x^2}} \cdot \frac{f_y}{\sqrt[2]{y^2 + f_y^2}} - f'', \tag{3.2.4}$$

where $f_x$ and $f_y$ represent focal lengths in the $x$ and $y$ dimension respectively.

In order to estimate the focal length value $f''$, the author proposes to average both of the focal lengths obtained by combining model parameters $f_x$ and $f_y$ with the camera spatial horizontal and vertical sampling periods $s_x$ and $s_y$ according to formula 3.2.5. The author assumes that the lens of a ToF camera is spherical. Only for a spherical lens both of the pinhole model parameters $f_x$ and $f_y$ represent the same physical distance between the sensor and the lens.

$$f'' = \frac{f_x \cdot s_x + f_y \cdot s_y}{2}. \tag{3.2.5}$$

***Experimental results***

The author conducted an experiment to assess the quality of the proposed geometrical distance measurement correction method. The experiment incorporated a ToF camera and a flat surface used as a reference.

Each ToF depth map of the reference planar surface should represent a perfectly flat plane after application of the geometrical correction algorithm as proposed by the author. The surface should be planar regardless of the angle between the reference surface and the optical axis of the camera. The author proposes to measure the geometrical correction quality by calculating the RMS error of the plane model fitting into data from the corrected depth map. The lower the RMS error, the more accurate the correction is.

The camera used for the experiment was the Mesa Imaging SR4000 [SR4000]. For its parameters please refer to table 2.6.1. For purpose of the experiment the camera was placed at about 2.5 - 3 meters from reference flat surface. A sufficiently large flat wall was used as the reference. The optical axis of the camera and the wall formed the angle $\varphi$. Multiple combinations of distance and angle values were tested in order to determine the proposed correction method accuracy under different circumstances. Figure 3.2.2 illustrates the schematic of the experimental setup (as viewed from above).

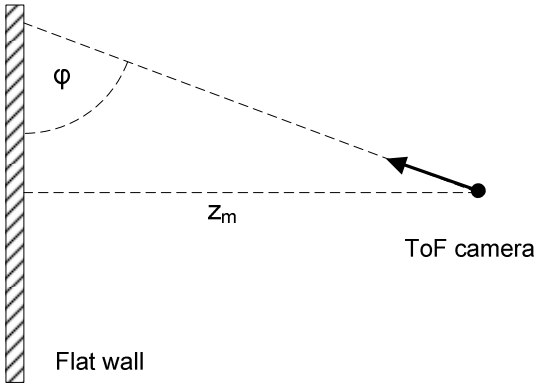



Figure 3.2.2 – Experimental setup of the ToF camera facing a flat surface.

The value of $z_m$ defines the distance between the wall and the camera while the value of $\varphi$ defines angle between the camera optical axis and the wall sufrace. A total of 16 measurements were taken, each one consisting of 50 frames of the ToF depth map. The frames were averaged in order to remove temporal noise.

The distance correction method requires that the intrinsic parameters and lens distortion parameters of the ToF camera to be known. Both the intrinsic and the lens distortion parameters were estimated using Zhang's algorithm [Zhang_02]. The calibration pattern with a regular black circle grid (Figure 2.3.1) was used. The geometrical lens distortions needed to be removed from the captured depth maps prior to the proposed distance measurement correction. According to the manufacturer's datasheet [SR4000], the spatial sampling period of the SR4000 camera sensor is 40μm by 40μm. The value was used in Equation 3.2.5.

The following Figures 3.2.3 and 3.2.4 show two-dimensional and three-dimensional visualisations of the measured distance for the case when the camera was facing the reference surface with 0° angle. The left-hand image shows a raw depth map (without correction). As was expected, the distance is greater in the depth map corners than in the centre. The right-hand image shows the same data set after performing the geometrical correction. The corrected depth map now represents a planar surface.

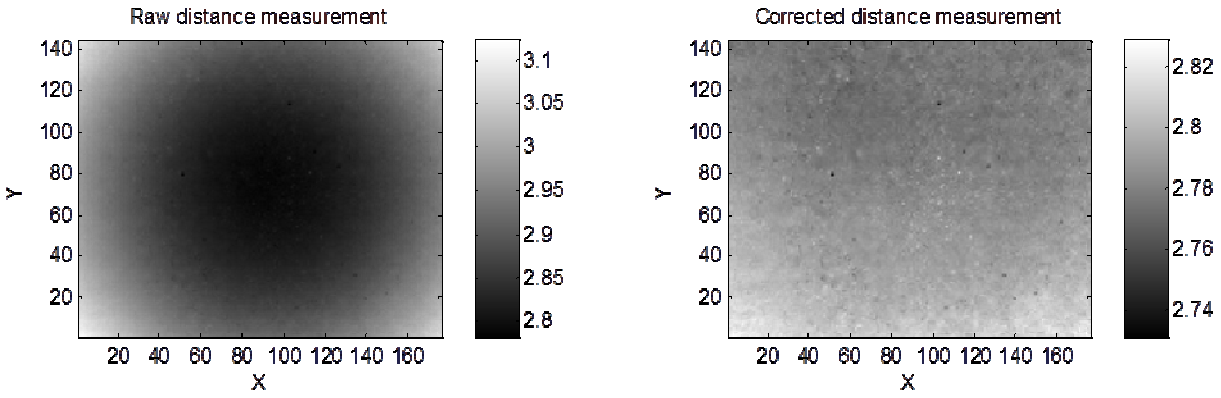



Figure 3.2.3 – Two-dimensional visualisation of the ToF depth map before and after correction.

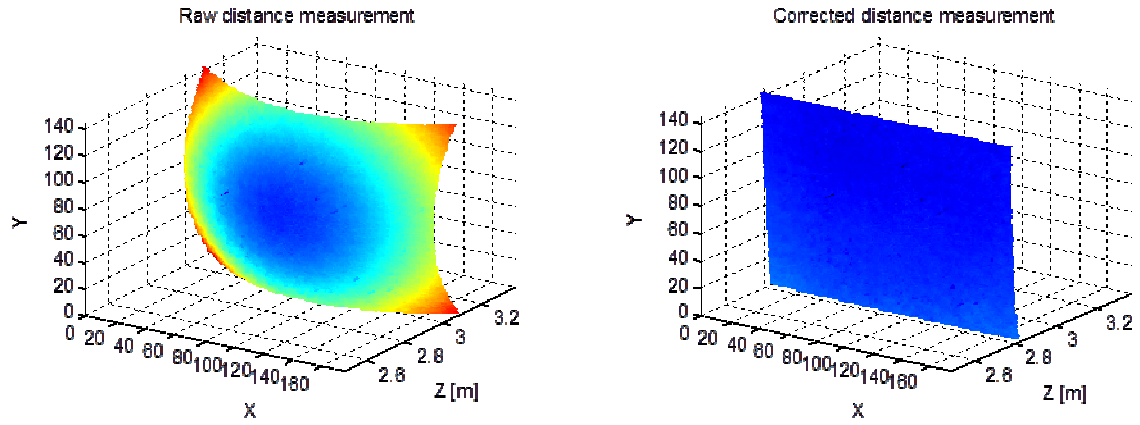



Figure 3.2.4 – Three-dimensional visualisation of the ToF measurement before and after correction.

In order to assess the quality of the proposed method, the author took advantage of the available ground-truth information. It states that the surface seen by the ToF camera is planar; therefore, it can be approximated by a model of a 2D plane in a 3D space. The model of the plane is defined by formula 3.2.6:

$$z_{plane}(x, y) = a \cdot x + b \cdot y + c\,. \tag{3.2.6}$$

The value of $z_{plane}(x, y)$ represents the distance along the optical axis of the camera, $x$ and $y$ are the coordinates of each sample of the captured depth map and $a$, $b$ and c are the plane equation coefficients.

The fitting error function of the plane model to the ToF data is defined by the equation:

$$E(x,y) = \left| Z_{plane} - Z_{ToF} \right| . \tag{3.2.7}$$

The value of error is expressed in meters. For each case the RMS error and the maximum error of the plane model fitting was computed. These two error measures are defined according to equations 3.2.8 and 3.2.9:

$$E_{RMS} = \sqrt{\frac{\sum_{x=0}^{M} \sum_{y=0}^{N} E(x,y)^2}{M \cdot N}}, \tag{3.2.8}$$

$$E_{max} = max\big(E(x,y)\big), \tag{3.2.9}$$

where $M$ is the width of the ToF image and $N$ is the height of the ToF image.

The diagrams in Figures 3.2.5 and 3.2.6 show charts of the RMS error ($E_{rms}$) and the maximum error ($E_{max}$) for all test cases. The chart on the left-hand side shows the error values before correction while the chart on the right-hand side shows the error after correction.

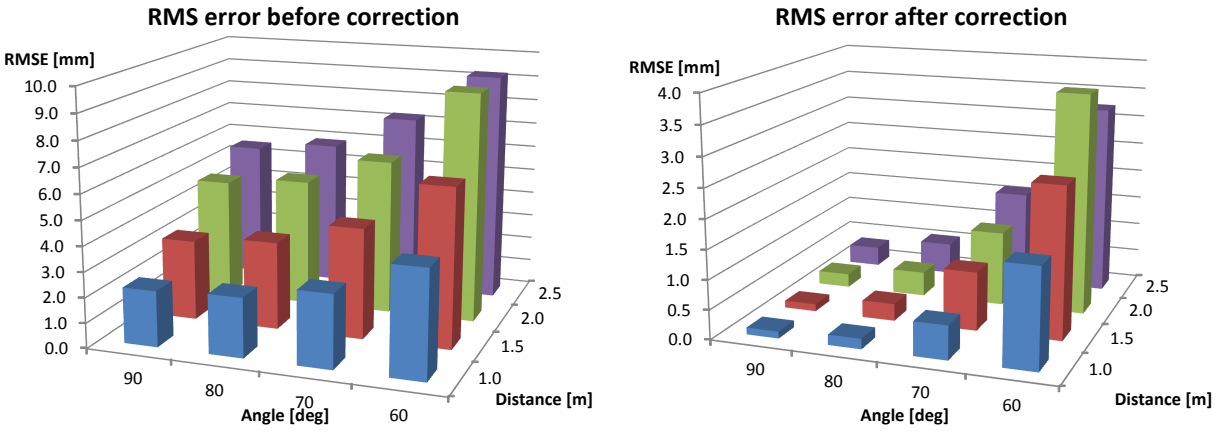



Figure 3.2.5 – RMS error ($E_{rms}$) according to the equation 3.2.8 before and after correction using the proposed algorithm.

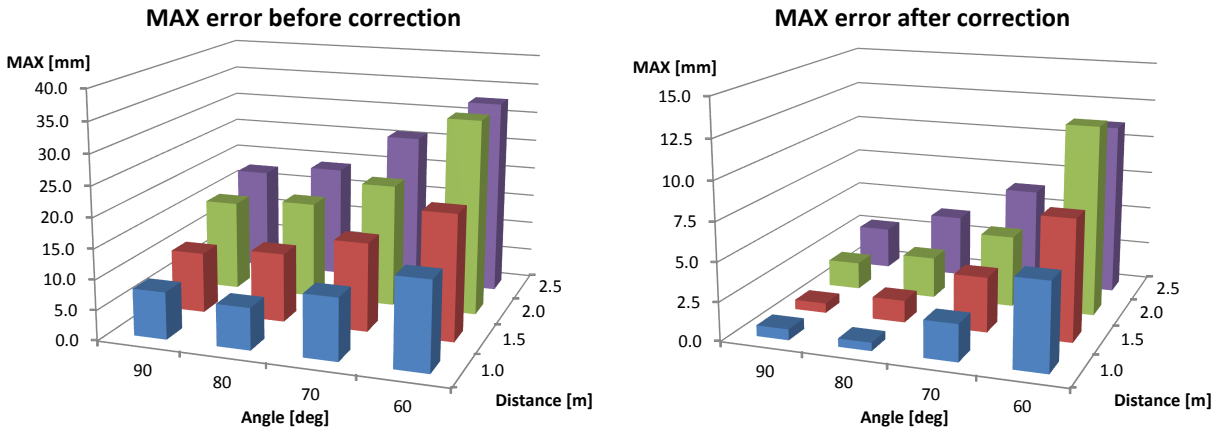



Figure 3.2.5 – Maximum error ($E_{max}$) according to the equation 3.2.9 before and after correction using the proposed algorithm.

The proposed geometrical correction algorithm allows to change the representation of the depth map, which makes it usable for DIBR rendering [DIBR] and for video and depth data fusion. The proposed algorithm requires only the intrinsic parameters of the ToF camera. It does not depend on any internal calibration algorithm provided by a ToF camera manufacturer. The procedure is very simple and requires no time consuming calculations.

As it was mentioned above, the manufacturer of the tested ToF camera provided own correction procedure; however, the intrinsic parameters used for the correction are not available to the user. Therefore the author was unable to do a comparison of the proposed method and the one used by the manufacturer. The lack of availability of intrinsic parameters used by the camera manufacturer disqualifies using of the corrected distance data for video and depth data fusion.

The method proposed by the author is general as the idea emerges from fundamental geometry. Unfortunately the experiment using only one type of a ToF camera is not enough to prove that the improvement can be attained in every case. The author is aware of this drawback and knows that the topic requires further research and experimentation.

## 3.3 Calibration of ToF distance measurement

The distance measurements of a ToF camera should be equal to the physical distances between the camera and the measured objects. Unfortunately, raw measurements of modulating signal phase difference do not correspond to the physical distances directly. Modern ToF cameras provide a built-in conversion algorithm that translates phase data to distance data. Unfortunately, due to possible incorrect internal calibration of the ToF camera, the distance measurements may also be incorrect and may exhibit a systematic error.

The author proposes a method to calibrate ToF measurements by finding a relation between the distances provided by the ToF camera and the physical distances that can be measured using a distance measurement device of another kind. The author focuses on distance measurements only.

Several methods of ToF distance calibration are described in the literature. Lindner et al. recommended to take a series of ToF distance measurements of a flat, semi-reflective panel along with the reference distance measurements [Lindner_01]. They used a B-spline curve to represent the relation between the ToF measurements and the physical distance. The distance measurement error was highly non-linear and exhibited periodicity; however, in the publication a reference was made to an old ToF device that did not provide built-in measurement correction. The drawback of this calibration method is that it requires a set of reference measurements made by an independent distance measuring device.

A similar method that was used by Kahlmann et al. incorporates a calibration pattern made of infra-red LED diodes [Kahlmann_01]. The infra-red diode pattern can be easily detected on a ToF intensity image and the depth map as the light-emitting diodes interfere with the camera operation. In the publication, the proposed ToF calibration system consisted of a ToF camera mounted on a moving trolley, which was coupled with a distance measurement unit based on an interferometer. It was used to provide reference distance measurements with relative accuracy up to a few microns. Kahlmann et al. proposed to construct a look-up table (LUT) using reference measurements and ToF measurements. No distortion model was proposed. The method described in the literature is very accurate but it requires expensive equipment and a set of very dense reference measurements.

Both methods as described in the literature require a very sophisticated mechanical setup and a reference distance measuring device. The method proposed by the author of this dissertation requires only a camera calibration board and a calibration pattern.

***The proposed method***

The principle of the method proposed here is based on the relation between the distance values computed using the camera model and the actual distance measurements. When using a calibration pattern, the transformation between the coordinate system of the pattern and the coordinate system of the camera can be computed. The intrinsic parameters of the camera must be known. The method uses the same algorithm as for the extrinsic camera parameter estimation [Zhang_01], but in this case only the relation between the pattern and the camera is interesting. By knowing the transformation and the pattern structure it is possible to compute the position of each feature point of the pattern in the camera coordinate space. The computed $z$ coordinates of the points define their distances from the ToF camera. These distances can be used as a reference, and there is no need to use any external distance measuring device.

The author proposes to use the following linear model for the distortion correction:

$$z_c = a \cdot z_m + b\,, \tag{3.3.1}$$

where $z_c$ is the corrected distance value, $z_m$ is the measured distance and $a$ and $b$ are the model parameters.

The reason for choosing the linear model is the fact that modern ToF cameras are compensated internally. The internal compensation accounts for all non-linear phenomena that are a part of the demodulation of the reflected light wave signal. What may not be correctly compensated is the modulation frequency of the carrier light wave. This frequency is required for conversion from a phase difference to an actual distance. The conversion from phase to frequency is a linear multiplicative operation.

***The experiment setup***

The author conducted an experiment which goal was to verify his proposed method of correction of ToF distance measurement. As the result of the experiment parameters of the assumed distortion model were estimated for a particular ToF camera. The Mesa Imaging SR4000 ToF camera was used [SR4000]. Please refer to the table 2.6.1 for the camera parameters.

The experimental setup consisted of a ToF camera and a planar calibration pattern mounted on a movable rig. The pattern used consisted of rectangular circle grid. Its dimensions are specified in the table 2.3.1.

A total of three calibration sequences were recorded. Each sequence contains a view of a stationary planar calibration pattern placed at a different distance from the ToF camera. The board with the calibration pattern was tilted with respect to the optical axis of the camera. When a calibration board is tilted, the distance between the camera and each feature point of the pattern is different. This allows to obtain more different distance measurements from a single calibration pattern view.

In order to remove noise, all frames of each sequence were averaged. The resulting three images were used to estimate the distance measurement distortion.

Figure 3.3.1 shows the intensity images of the calibration pattern and Figure 3.3.2 shows the corresponding distance maps.

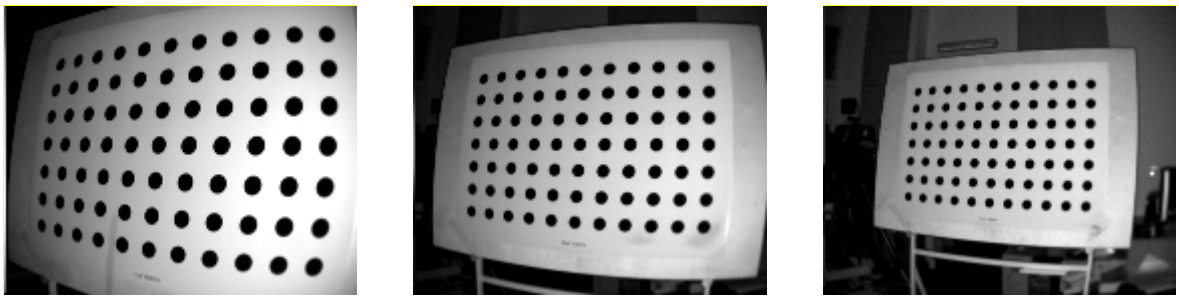


Figure 3.3.1 – Intensity images from the three test sequences of the calibration pattern.

Figure 3.3.2 – Sample distance maps from the three test sequences of the calibration pattern.

For each of the images, the distance to each feature point of the calibration pattern was computed according to Zhang's algorithm [Zhang_01].

The corresponding measured distance values were determined by sampling the depth map as provided by the ToF camera. Unfortunately black circles, which are the feature points, exhibit very low reflectivity. The reflectivity was low enough to cause distance measurements to be very uncertain. Therefore instead of sampling the distance map directly, the samples came from a model of a 3D plane fitted into the measured distance data.

The plane model was build according to equation 3.2.6. Parameters *a*,*b* and *c* of the equation were computed using least mean square minimization of distance between the plane and 3D coordinates of all points (according to equation 3.2.7) that constitute for the calibration pattern board, not just to the feature points. This allowed to take advantage of more accurate distance measurements, taken from between the pattern features, where the board reflectivity is significantly higher.

***The experiment results***

The relation between distances measured by the ToF camera and those computed using the pinhole camera model is shown in Figure 3.3.3.

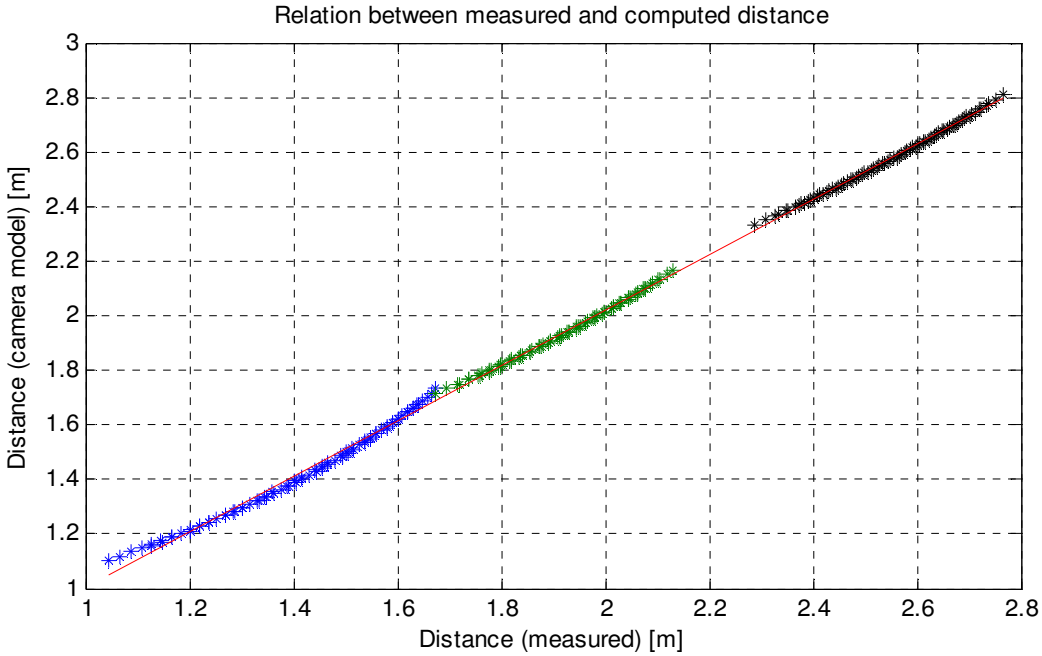



Figure 3.3.3 – Relation between the distance measured by the ToF camera and that computed from its model.

There are three groups of points in the figure above. Each one corresponds to one calibration image. The calibration pattern board was tilted by ca. 45 degrees with respect to the camera optical axis, so that the distance to each point of the calibration pattern is different. As was expected, the relation is almost perfectly linear; the non-linearity within the point group marked in blue may be explained by inaccuracy of estimation of the pattern feature points or by an unwanted bend of the calibration pattern board.

The distance correspondences were used to compute the parameters of the distance distortion model according to Equation 3.3.1. The LMS criterion was used. The resulting model parameters for the SR4000 ToF camera are ***a*=1.011** and ***b*=-0.008**. In an ideal case the parameter *a* would be equal to 1.0 while the parameter *b* equal to 0.0.

The results of the experiment show that a particular ToF camera provides the correct distance measurements even without the calibration. The multiplicative factor *a* is very close to 1.0 (an ideal case), which indicates a direct one-to-one relation, while the additive parameter *b* indicates a systematic offset of 8 mm.

The method proposed by the author allows to correct the discrepancy between the actual ToF distance measurements and the distances computed using the pinhole camera model. This greatly increases the accuracy of the acquired depth maps. Furthermore, the method can be used to translate

directly between the phase difference of the light modulating signal and the distance without knowing the ToF modulation frequency. The modulation frequency can be computed from the parameters of the distance distortion model. This makes the proposed technique a universal tool for calibrating ToF measurements for applications in multi-camera systems.

The proposed method was verified using only one type of a ToF depth camera. This is not sufficient to consider the method general and to conclude that it is suitable for any ToF camera. The author is aware of that fact. The lack of availability of multiple types of ToF cameras prevented him for further investigation of this topic. Nevertheless, in the future such investigation shall be conducted.

## 3.4 Depth map noise reduction

Measured distance data provided by a ToF camera is noisy. Modern ToF cameras provide distance measurements with a significant amount of spatial and temporal noise. The amount of noise depends mostly on the reflectivity of objects' surfaces. The lower the reflectivity, the higher the noise level as the camera is unable to reliably estimate phase of modulation signal of a reflected light wave when its amplitude is low.

A depth map spatial noise impairs the smooth surface and object edges representation, which is the key feature of a depth map. When performing video and depth data fusion, a change to the depth of a pixel will cause a misalignment; therefore, it is important to have noise-free depth data.

Because the noise is mostly exposed when the light wave amplitude is low and the integration time of the camera is short, an increase in the integration time seems to be the simplest solution; however, this increase has its limitation. The whole multi-camera system operates with a certain frame rate and the integration time cannot be longer than the duration of a single frame. The increase in the integration time will also cause a motion blur that can be observed on moving objects.

The author thus proposes to deal with the noise by using a dedicated distance data filtration algorithm. A spatial **bilateral filter along with a motion-adaptive temporal filter** was thus incorporated in the proposed algorithm. Their purpose is to **reduce the noise while preserving sharp edges of objects**.

The depth noise reduction results, as described in the literature, show a significant reduction of the noise level; however, the described methods apply only to static scenes, as no temporal filtration is taken into account.

As an example, Ma et al. [Ma_01] proposed to perform a triangulation [Porikli_01] of the measured data points prior to the application of a spatial bilateral filter to the mesh vertices.

Huhle et al. [Huhle_01] proposed the use of a non-local means filter [Buades_01] for distance data filtration. The non-local means filter is a variation of a bilateral filter. In this filter, the weights are based on similarity measure(s) between small neighbouring patches of the filtered image rather than

on individual pixels. The described filter also incorporates intensity information as an additional cue. Unfortunately, the method does not incorporate any temporal filtration.

Georgiev et al. [Georgiev_01] proposed a spatio-temporal non-local means filter that incorporates data from multiple depth frames. The filter provides a joint spatio-temporal filtration, although it is not motion-adaptive. The filtration algorithm was supposed to operate in real time, which suggests its low complexity; however, the analysis of the algorithm leads one to the conclusion that it might not preserve the object edges as well as the algorithm proposed by the author.

A solution for ToF depth data denoising, for situations when the ToF camera must operate under low power conditions, is also proposed by Georgiev et al. [Georgiev_02] [Georgiev_03]. The low power conditions are defined in the publications as low power scene illumination and short integration time. Georgiev et al. propose to first project the ToF data to the 3D space and to perform the denoising of the 3D mesh afterwards [Georgiev_02]. In their second publications a non-local denoising approach working in complex domain is proposed [Georgiev_03]. There is also a suggestion about estimation and removal of fixed pattern noise. However, the author hasn't observed such a phenomenon on the ToF equipment that he worked with.

There is also a convolutional neural network (CNN) approach of ToF camera noise modelling proposed by Bolsee et al. [Bolsee_01]. A number of cascaded convolutional neural networks is trained using artificially generated noisy ToF data – both amplitude and distance. As a result the neural network allows to estimate noise characteristics of new ToF data and also provide a denoised version of it.

The distance data filtration algorithm as proposed by the author consists of two filtrations. The first filtration is meant to remove temporal noise by using a **motion-adaptive IIR temporal filter**. The motion adaptation is based on independent, per-pixel motion detection. Pixels, which are considered as moving, are not subjected to temporal filtration. The second filter is a **spatial bilateral filter** [Porikli_01]. The goal of spatial filtration is to remove noise from smooth surfaces while keeping the object edges intact. Sharp object edges are the key element of a depth map. The block diagram of the proposed algorithm is shown in Figure 3.4.1.

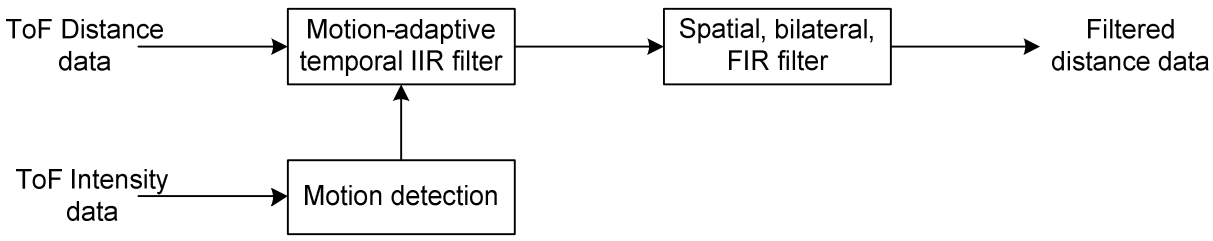



Figure 3.4.1– Block diagram of the distance data filtration algorithm proposed by the author.

Motion detection is performed on intensity data using temporal frame difference instead of background estimation [Stauffer_01] and [Kim_01]. An intensity image carries some texture information which is meaningful for motion detection. The distance data does not contain any texture information, hence object motion cannot be reliably detected.

### 3.4.1 Motion-adaptive temporal IIR filter

The motion-adaptive IIR filter is the first stage of the filtration procedure. The filter operates on each sample of the distance map independently. There are as many filters as samples in the distance map. When no motion is present, the proposed filter behaves like a first-order low-pass IIR filter. The transfer function of the filter is described by Formula 3.4.1:

$$H(z) = \frac{1-k}{1-k \cdot z^{-1}}, \tag{3.4.1}$$

where $k$ is the control parameter that controls the passband of the filter [Mayer-Baese_01] and $z$ is a complex argument.. Parameter $k$ ranges from 0 to 1. The higher the value of $k$, the narrower the passband of the filter is.

When a motion is detected, the delay element is immediately loaded with the distance from the current measurement. This breaks the IIR filter feedback loop and causes the input distance value to be present at the output of the filter immediately. The momentary bypass of the temporal filter loop ensures that moving objects do not get blurred by the filtration, while noise in stationary regions of the scene is properly reduced.

The author has chosen the first order IIR filter structure due to its simplicity and robustness. His experiments show that first order filtration is sufficient for temporal noise removal.

Figure 3.4.2 shows a detailed block diagram of the proposed temporal filter.

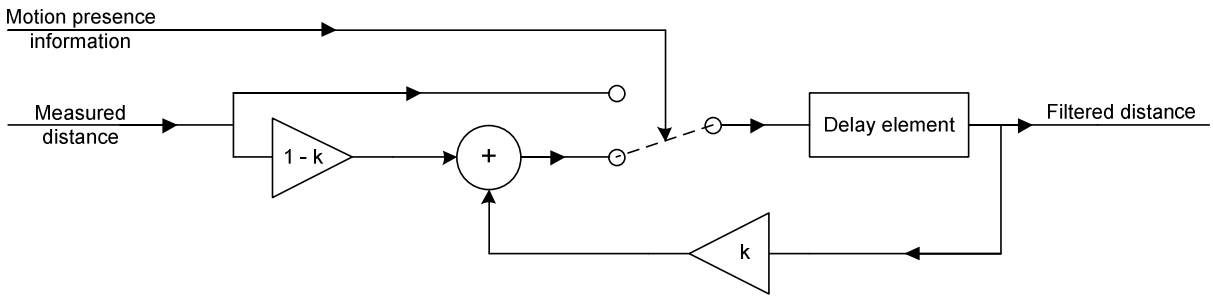



Figure 3.4.2 – Block diagram of the motion-adaptive temporal filter proposed by the author for a single distance map sample.

It is very important to reliably detect motion in the depth map for the temporal filtration to work as expected. Failure to do so will either cause moving objects to get blurred or the stationary objects to exhibit temporal noise. Reliable motion detection is difficult because the depth map does not provide any colour information; therefore, the author proposes to use an intensity image for motion detection. The intensity image represents scene luminance in the infra-red wavelength range. Most textures that are visible to the naked eye are also visible near the infra-red band; therefore, object motion can reliably be detected using texture information from the intensity image.

Motion detection in the proposed motion-adaptive IIR filter is based on an analysis of the difference computed over consecutive intensity frames. The author chose this technique because of its simplicity and robustness. There are other motion detection techniques known from the literature, e.g. Stauffer et al. and Kim et al. described a technique based on background estimation [Stauffer_01][Kim_01]. The estimated background image is then subtracted from each analysed frame. Then the difference is analysed and a decision is made whether a pixel belongs to an object or to the background. These techniques do not consider camera motion or panning, so the camera needs to be stationary. The method proposed by the author of this dissertation does not require this.

The proposed motion detection algorithm works as follows: for each sample of the intensity image the difference is computed between its intensity from the current and the previous frame. The absolute value of the difference is then thresholded. Samples where the absolute difference value exceeds the threshold are considered as belonging to a moving object while others are considered as the stationary background. The decision-making rule is illustrated by formula 3.4.2:

$$M_{i,j}^{t} = \begin{cases} 1 & \left|A_{i,j}^{t} - A_{i,j}^{t-1}\right| > th \\ 0 & otherwise \end{cases}, \tag{3.4.2}$$

where $A_{i,j}^{t}$ and $A_{i,j}^{t-1}$ are the intensities for a sample at *j,i* position for a current and previous frame and *th* is the threshold parameter. $M^t$ is a binary mask, where the non-zero value indicates the presence of motion. The method has low computation complexity and, what is more important, it uses a short observation time window (only two consecutive frames). A short time window is desired because it detects the beginning of object movement without any time delay.

The chosen threshold parameter *th* must have a value low enough in order not to omit any possible motion. On the other hand, its value must be high enough in order not to consider an intensity change, caused by the temporal noise, as an indication of motion. For the purpose of estimating the threshold, the author has used the **additive Gaussian noise model** of the temporal noise [Boyat_01] [Rakhshanfar_01].

According to the model, the intensity of each pixel can be described by a sum of two variables: the actual intensity of a pixel $I_0$, that cannot be observed directly, and a random variable with Gaussian distribution of variance $\sigma^2$. The model is defined by the equation 3.4.3.

$$I = I_0 + N(0, \sigma^2)\,, \tag{3.4.3}$$

where $I$ is the observed pixel intensity and $N(\mu, \sigma^2)$ denotes a random variable of normal distribution with parameters $\mu$ and $\sigma^2$ (here it is assumed that $\mu = 0$) The unknown actual intensity of a pixel $I_0$ can be approximated by the arithmetic mean $\mu$ computed over a number of consecutive video frames. The variance $\sigma^2$ informs us about the spread of observed intensity values. Unfortunately, the value of noise variance is unknown and difficult to predict, as the noise is caused by a combination of many physical processes that take place during image acquisition. What is more, the variance may depend on the pixel coordinates and on the observed mean intensity $\mu$. This complicates the problem of choosing the correct threshold parameter *th*. Therefore **the author introduces an extended noise model in which noise variance depends on the observed mean intensity of a pixel**:

$$I = I_0 + N\big(0, \sigma^2(\mu)\big)\,, \tag{3.4.4}$$

Both values of $\mu$ and $\sigma^2$ can be estimated by using a series of captured images and assuming a static background, which can easily be done in practice. The author proposes to determine the particular ToF camera noise variance and its dependence of observed pixel intensity.

The author chose to set the threshold *th* to be equal to $3\sigma$, which will cause a rejection of most false motion detections (statistically) [Kazmeir_01].

***Estimation of temporal noise variance***

There are methods, known from literature, regarding estimation of camera noise parameters. For example Rakhshanfar et al. propose estimation of camera noise parameters independently for homogenous regions of an image [Rakhshanfar_01]. The noise model proposed by Rakhshanfar et al. is, however, very complex and incorporates non-linear image processing effects which are not applicable in ToF cameras.

Jin-chao et al. propose much simpler camera noise model which assume only one noise parameter [Jin-chao_01]. The parameter is independent from observed signal intensity and spatial pixel location within image. However, the author has observed that such a dependency on intensity exists therefore the model proposed by Jin-chao et al. is not suitable for the ToF camera used by him.

The author conducted an experiment to determine the relation between an average pixel intensity and its corresponding Gaussian noise variance. The goal was **to determine the relation between the average intensity of each pixel and its corresponding intensity variance.** The knowledge about this relation will allow to perform more accurate motion detection. That, in turn, will allow to remove more temporal noise while preserving moving objects shapes.

The estimation method used by the author is based on capturing a number of ToF intensity images of a static scene. As the scene is static, the only cause for a pixel intensity change is the temporal noise. Statistical analysis of multiple consecutive intensity images will allow to determine relation between $\mu$ and $\sigma^2$.

The author used the Mesa Imaging SR4000 ToF camera [SR4000] which parameters are summarised in the table 2.6.1. The ToF camera was placed in front of a scene setup which contained a variety of objects that exhibited different reflectivity. This wide variety of objects' reflectivity allowed to capture a sequence in which many different intensity levels were present. This allowed to determine the variance for a largest span of possible average intensity values. Figure 3.4.3 shows a single intensity frame used for noise parameter estimation.

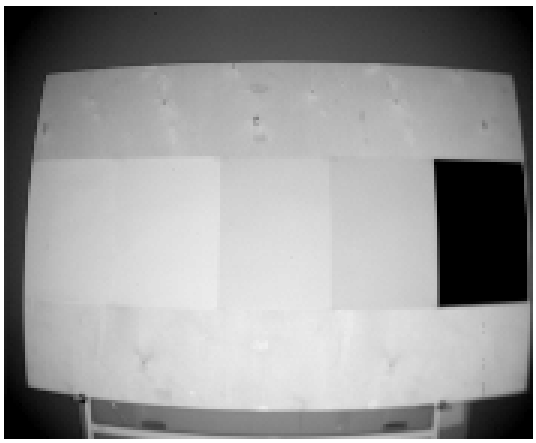


Figure 3.4.3 –Intensity image of the experimental scene setup captured by the ToF camera.

For each pixel, the average intensity $\mu$ and the intensity variance $\sigma^2$ was computed by taking the intensity measurements from the whole video sequence. Spatial pixel coordinates were not taken into account. Each pair ($\mu$, $\sigma^2$) defines a relation between these two parameters regardless of which pixel they come from. The ToF camera provides the intensity data with 14-bit resolution. For detailed SR4000 camera parameters please refer to the table 2.6.1. The scatter plot of each ($\mu$, $\sigma^2$) pair is shown in Figure 3.4.4. Figure 3.4.5 shows the density plot of the occurrence of particular ($\mu$, $\sigma^2$) pairs. The occurrence is shown in a logarithmic scale.

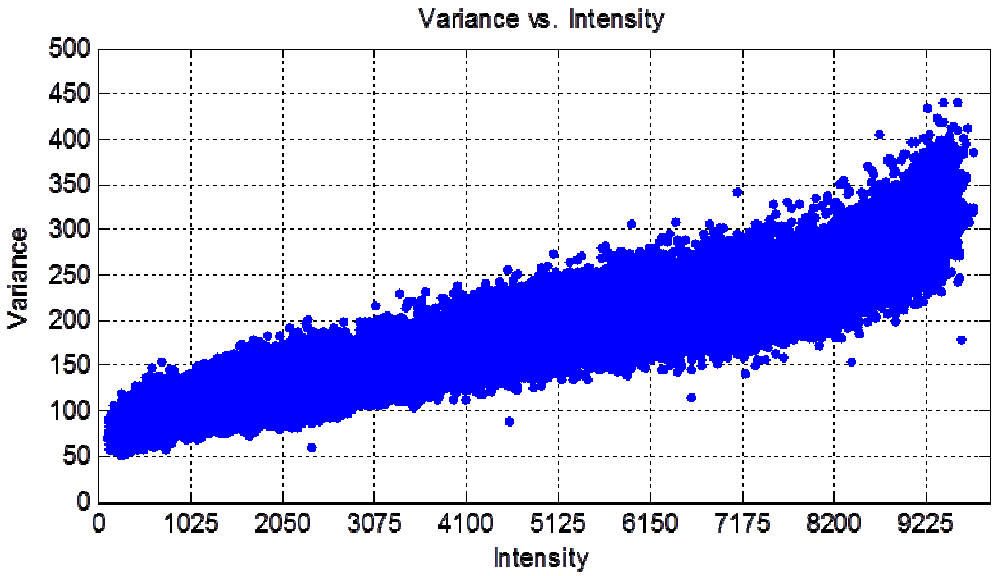



Figure 3.4.4 – Scatter plot of the relation between average pixel intensity and its intensity variance.

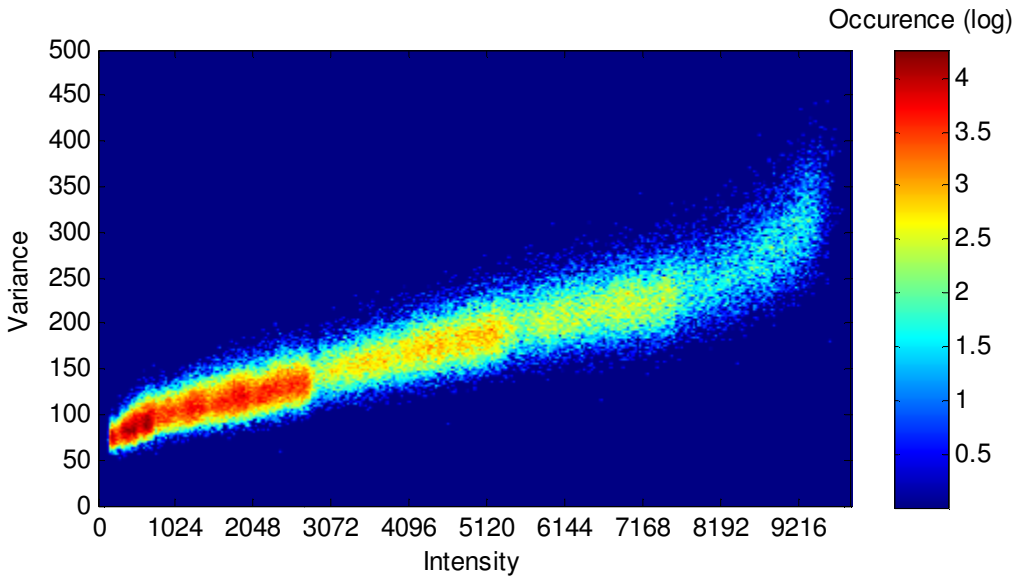



Figure 3.4.5 – Occurrence density plot of the relative average pixel intensity and variance pairs.

After an analysis of the experimental data, the author has proposed a model of the relation between the average intensity and its corresponding noise variance. The proposed model has a form of a polynomial curve. The degree of the polynomial was chosen to be 1, 2 or 3. The experiments have shown that increasing the polynomial order above 3 does not influence significantly the approximation accuracy. The proposed model of variance is described by equation 3.4.5:

$$\sigma^2(I) = a_3 \cdot I^3 + a_2 \cdot I^2 + a_1 \cdot I \quad + a_0 \,, \tag{3.4.5}$$

where $\sigma_2$ is noise variance, $I$ corresponds to the infrared image intensity and variables $a_0$ to $a_3$ are the coefficients of the approximation polynomial.

The parameters of each model were estimated using the least mean square (LMS) criterion using the same data set. Data points which exhibit their mean value lesser than tripled standard deviation were rejected; such data points might indicate that the intensity can be negative. Table 3.4.1 summarises the estimated polynomial coefficients for each polynomial degree along with their RMS fitting errors. The presented results are correct for the Mesa Imaging SR4000 ToF camera [SR4000] and should not be generalized.

Table 3.4.1 – Regression polynomial coefficients along with the RMS error of fit for the SR4000 ToF camera.

| Coefficients / Degree | $a_3$ | $a_2$ | $a_1$ | $a_0$ | RMSE |
|---|---|---|---|---|---|
| 1 | | | 2.16e-2 | 8.04e+1 | 18.87 |
| 2 | | 2.85e-7 | 1.91e-2 | 8.37e+1 | 18.79 |
| 3 | 2.85e-10 | -3.58e-6 | 3.30e-2 | 7.30e+1 | 18.32 |

The modelling curve with the lowest RMS error (among all of the tested curves) is the third-degree polynomial; however, the linear and quadratic models do not exhibit a significantly larger fitting error. As the RMSE informs us about the average difference between the curve and each data point, there may be areas where the local error is much greater than the average. The plot of the three modelling curves overlaid on the occurrence density plot is shown in Figure 3.4.6.

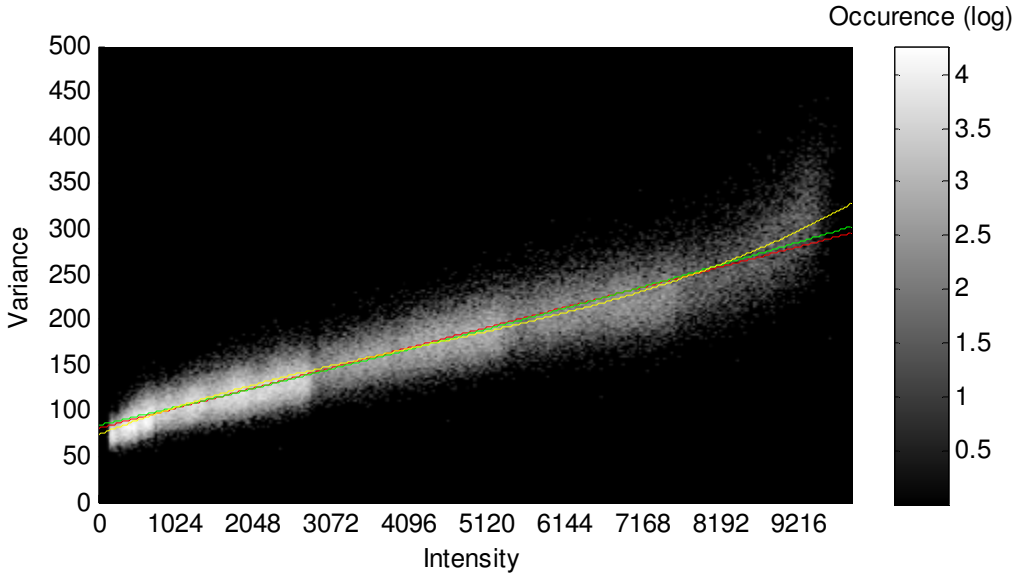



Figure 3.4.6 – Occurrence density plot with overlaid curves of the regression polynomials: 1st degree (red), 2nd degree(green) and 3rd degree (yellow).

As it was expected, the 3rd-degree polynomial (yellow) fits into the data set with the largest accuracy. However, due to the large spread of possible variance values for a particular mean intensity, the other two models perform almost equally well. During the author's experiments it was proven that the linear model is sufficient to model the relation to be used for motion detection.

It must be noted that the method allows to determine the general relation between mean intensity and its variance caused by noise while assuming that all pixels of the sensor have identical characteristics. Each pixel of the camera may have different noise characteristics than the others. In order to determine the $\mu$ and $\sigma^2$ relation for each pixel independently, each pixel should be observed for every possible average intensity. The task is difficult in practice as the ToF camera does not provide any intensity regulation of the light emitter, so a large number of uniformly reflective scene setups is required. Therefore the author has assumed that all pixels have equal noise characteristics.

Moreover, the author is aware that the presented results are true for the SR4000 ToF camera and that they may be different for a different camera model. The topic of camera noise and especially ToF camera noise which operates differently than a video camera requires more research and extends beyond the scope of this dissertation. Therefore the results of this experiment shall not be considered as general.

### 3.4.2 Spatial bilateral FIR filter

The distance data is subjected to the spatial filter after the motion-adaptive temporal filtration. The spatial filtration algorithm as proposed by the author incorporates a spatial bilateral FIR filter [Tomasi_01][Paris_01]. The proposed filter operates on measured distances instead of depths. This allows for more control over the filtration than when using depth or disparity values. It is possible to set the filtration parameters according to the physical features of the scene. The proposed filter is defined by the equation:

$$z_{flt}(x,y) = \sum_{j=0}^{M}\sum_{i=0}^{N} z_{org}(x-i, y-j) \cdot w\left(z_{org}, x-i, y-j\right), \tag{3.4.6}$$

where $z_{org}$ denotes input distance map, $z_{flt}$ is the output filtered distance map, $w$ is the filter weight mask that depends on input data, $x$ and $y$ are sample spatial coordinates and $M$ and $N$ are the filter mask dimensions.

The filter weights are adapted according to local features of the distance map gradient. The weight adaptation procedure is designed in such a way so that the filter can distinguish between weak and strong edges. The weak edges are usually caused by spatial noise while the strong edges are most likely to be the actual object edges.

**The author proposes to use local gradient features** rather than the image patches used in the non-local means filter [Buades_01] [Huhle_01]. The reason is that the depth map does not contain any texture information and the object texture features in the intensity image do not always correspond to the actual object boundaries in the scene.

The filter uses a square mask of a fixed size ($M = N$). The size of the mask is a control parameter, which defines the spatial filtration strength of the filter. An individual filter mask is computed for each sample of the input distance map. The goal is to find such a mask that in the case of the presence of a strong edge the samples lying on the same side of the edge as the currently filtered sample are assigned greater weights than samples lying on the opposite side of that edge.

For each point of the mask the shortest discrete path between its location and the location of the central point is chosen. The path represents the way that the filtration algorithm needs to traverse to "reach" the corresponding sample. If such a path crosses a strong edge, the weight assigned to that sample should be low, otherwise it should be high. Each point is assigned a weight which corresponds to the maximum edge strength along the path.

The author proposed the edge strength to be measured as the absolute difference between the distance values of neighbouring points along the path. These differences reflect the local gradient amplitude of the distance map. An illustration of the filter mask construction procedure is shown in Figure 3.4.7.

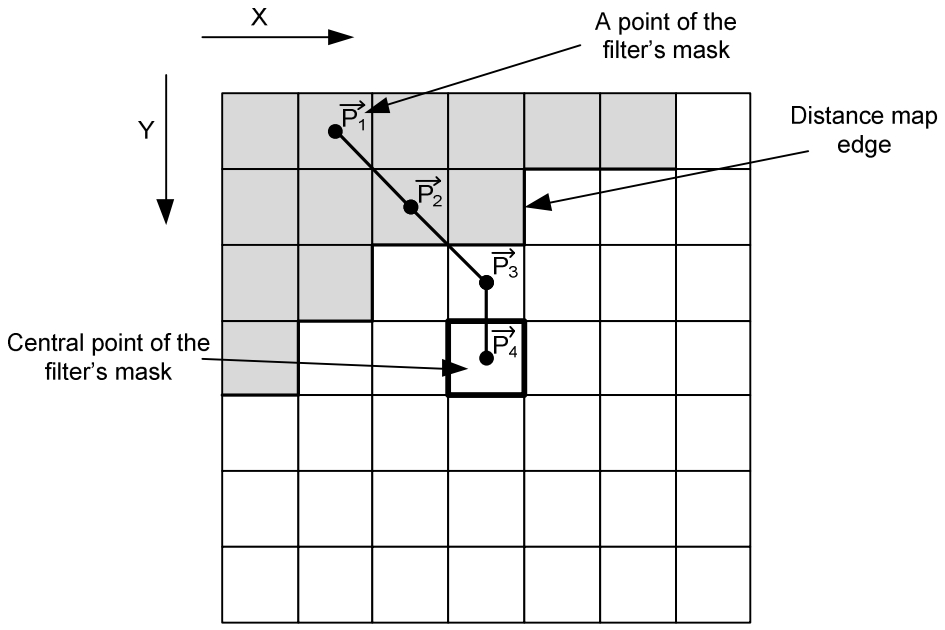



Figure 3.4.7 – Example illustration of computation of a filter mask weight for a specified point $\overrightarrow{p1}$ while performing filtration for point $\overrightarrow{p4}$.

The maximum gradient amplitude value along the path, as shown in Figure 3.4.7, is computed according to formula 3.4.7

$$|g_{max}| = max\{|z(\overrightarrow{p1}) - z(\overrightarrow{p2})|, |z(\overrightarrow{p2}) - z(\overrightarrow{p3})|, |z(\overrightarrow{p3}) - z(\overrightarrow{p4})|\}, \quad (3.4.7)$$

where points $\overrightarrow{p1}$ to $\overrightarrow{p4}$ are points along the path and $z(\vec{p})$ is a measured distance at point $\vec{p}$. A threshold parameter is defined in order to differentiate between weak and strong edges. The threshold defines a value of the physical distance difference above which an edge is considered to be a physical object boundary. The filter mask weights are computed using formula 3.4.8, which incorporates a sigmoid function. The sigmoid function is often used when there is a need to make soft decisions. Here it provides a soft decision whether an edge is an object boundary or not:

$$w = 1 - \frac{1}{1 + e^{-k \cdot (|g_{max}| - th)}}, \quad (3.4.8)$$

where $|g_{max}|$ is the maximum distance gradient value along a path, *w* is the corresponding mask weight value and *th* is the distance difference threshold. In order for the filter to have unity gain, each filter mask is normalised prior to filtration, so that the sum of magnitudes of all weights is equal to one.

Figure 3.4.8 shows two examples of computed filter masks along with the underlying distance map which was used to compute them. In this example the filter mask has a size of 7x7 samples. A brighter colour represents a closer point in the distance map and a higher weight value in the filter mask. All weight values are non-negative by definition. The sample in the centre represents a currently filtered distance map sample.

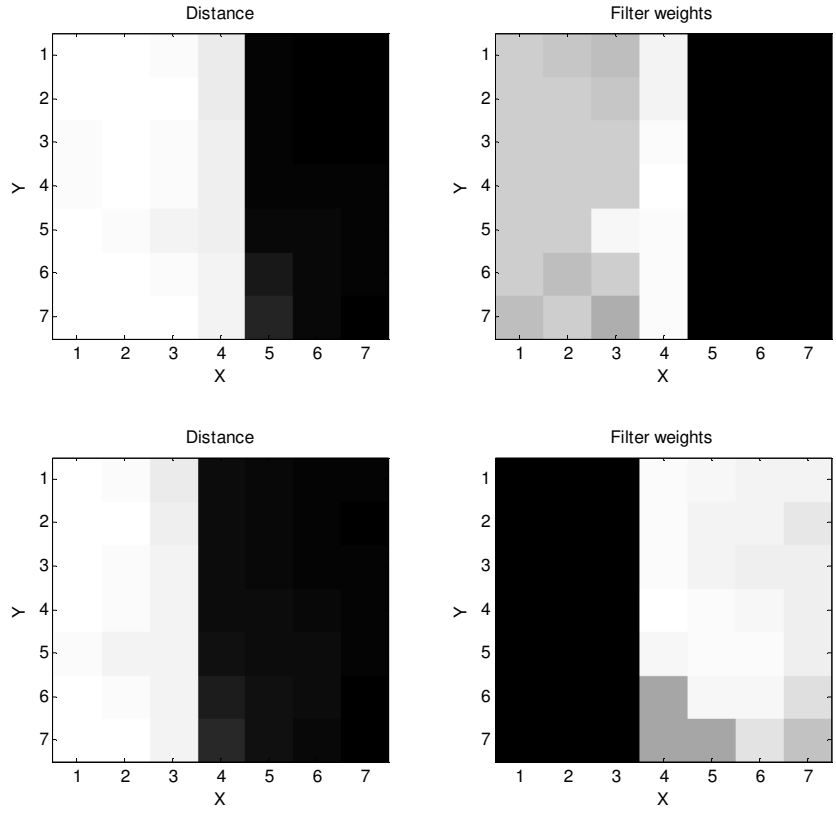



Figure 3.4.8 – Two examples of a distance map and spatial bilateral filter mask weights.

Despite the adaptation of the filter mask to the local gradient features of the distance map, sharp edges may still get blurred. As a result of the experiments, the author proposes to vary the filter strength according to the local gradient features of each pixel. The proposed method includes a linear interpolation between the original and filtered distance map according to equation 3.4.9:

$$z = w \cdot z_{org} + (1 - w) \cdot z_{flt}\,, \tag{3.4.9}$$

where $z$ is the resulting distance value, $z_{flt}$ is the value computed using the filter, $z_{org}$ is the original distance and $w$ is the interpolation factor.

Parameter $w$ is computed using a local gradient feature according to equation 3.4.8. The equation is identical as for weights in the filter mask but with one important exception, i.e. the threshold value is different. Setting the threshold for the interpolation to a higher value than for the filter mask weights allows to reduce the filtration strength at the sharp edges. The author's experiments have proved that the method provides better sharp edge preservation but is not required for acceptable noise reduction results.

### 3.4.3 Experimental results

Figures 3.4.9 and 3.4.10 show a single frame from a test video sequence recorded by a ToF camera. The left image represents the intensity image; the right image represents the distance map. Brighter values indicate closer objects. The distance map in Figure 3.4.9 was not filtered. The distance map in Figure 3.4.10 was filtered by the algorithm proposed by the author.

Figure 3.4.9 – Example intensity image and raw distance map.

Figure 3.4.10 – Example intensity image and filtered distance map.

The noise in the distance map is very noticeable in Figure 3.4.9.A significant amount of noise is present on all smooth surfaces, such as the floor and room's walls. What cannot be shown in Figure 3.4.9 is the fact that the noise is different for each captured frame and it changes in time.

The filtered distance map in Figure 3.4.10 contains much less spatial noise. The objects' surfaces are smooth while their edges are sharp. The scene background is also stationary in time.

Figures 3.4.11 and 3.4.12 show re-projections of a raw and a filtered distance map into the 3D space, respectively. The re-projection was done according to the pinhole camera model with the use of intrinsic parameters of the ToF camera. The $z$ axis direction is determined by the optical axis of the camera.

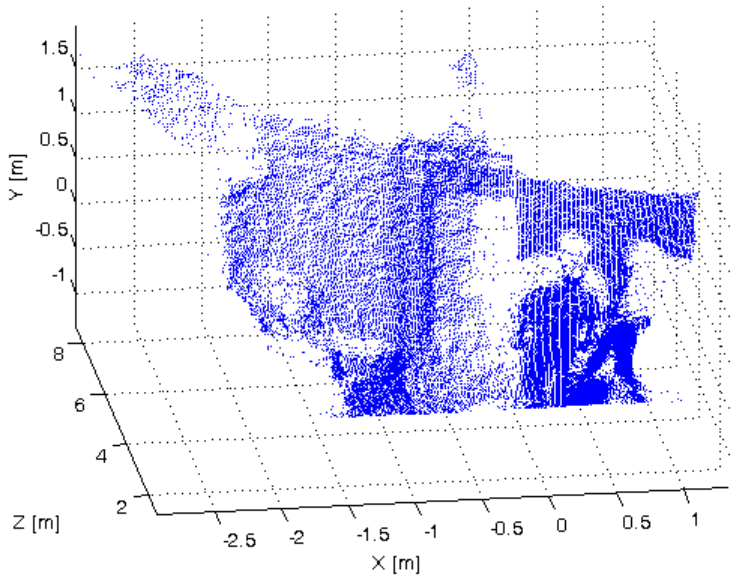



Figure 3.4.11– 3D scatter plot of a reconstructed scene using a raw captured depth map.

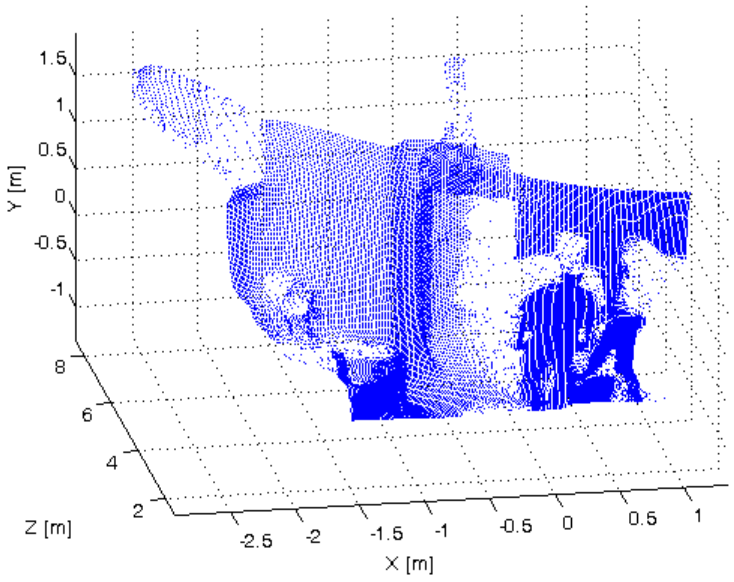



Figure 3.4.12 – 3D scatter plot of a partially reconstructed scene using the filtered depth map.

The reconstructed scene in Figure 3.4.11 exhibits a large amount of noise. Surfaces that are supposed to be flat are extremely irregular. It is difficult to determine the room's shape from the reconstructed data. On the other hand, the amount of noise in the scene as reconstructed using the filtered data is significantly reduced while the sharp object edges are well preserved. Noise reduction can be observed especially on flat surfaces.

## 3.5 Conclusions

Raw data from a ToF camera cannot be used directly for video and depth fusion. A depth map obtained by a ToF camera contains a large amount of spatial and temporal noise. It also needs a geometrical correction and distance measurement calibration.

The author has proposed three major pre-processing steps:

- Geometrical correction
- Distance calibration
- Noise reduction

A raw depth map needs a **geometrical correction** which transforms the distance measured by the camera to the representation commonly used for depth maps. The necessary correction procedure is usually provided by the camera's manufacturer. However, the parameters used are embedded in the correction algorithm. Details of the algorithm are also usually unknown to the end user. This poses a difficulty as the camera parameters are required for further data processing; therefore, the author proposed to perform the correction externally by using independently estimated intrinsic parameters. This way the parameters are known and the corrected data can be used for fusion with video data.

Another important issue is distance measurement calibration. Modern ToF cameras are usually calibrated internally so there is no need for any additional calibration; however, in cases when a camera is not calibrated or it is not calibrated for a particular modulation frequency, external calibration is required. The author proposed a **distance measurement calibration** technique which does not require any other reference distance measuring device than the ToF camera itself. The proposed method assumes a linear relation between the ToF measured distance and the distance that can be computed using Zhang's algorithm [Zhang_01]. It requires a calibration pattern and known intrinsic parameters of the camera. The method can be used to calibrate distance measurements or for verification of a ToF camera calibration.

Data provided by a ToF camera exhibits a large amount of noise. The noise has a destructive effect on virtual view synthesis that is going to be performed after depth and video data fusion. The depth map noise will affect pixel positions during the view synthesis which, in turn, will lead to a distorted image. The author proposed a **noise reduction algorithm** that allows to significantly reduce the amount of noise while preserving key depth map features such as sharp edges and smooth surfaces. The proposed algorithm is based on a motion-adaptive temporal IIR filter and a spatial bilateral FIR filter. In order to accurately detect motion, the intensity image is used instead of the depth map. The intensity image provides necessary texture features that are not present in a depth map. The proposed method allows to reduce noise in depth maps while keeping sharp edges intact and introducing no motion blur effects.

# 4. Synchronisation of video cameras and depth cameras

## 4.1 Introduction

In a multi-camera system, synchronisation between all cameras is mandatory. This synchronisation is responsible for maintaining a constant and equal frame rate among all of the cameras. It is also crucial that image acquisition take place at the same time instant in all of the cameras.

Incorrect camera synchronisation leads to erroneous depth map estimation, and causes moving objects to appear in different parts of the images for each camera. Changes in the objects' positions are wrongfully interpreted as disparity which, in turn, causes errors in depth estimation. Moreover, camera synchronisation is also very important in a multi-camera system with video and depth cameras. In order to conduct correct fusion of video and depth data, both the video and depth sequences need to be synchronised.

In this chapter the author will address camera synchronisation-related issues in multi-camera systems. The multi-camera systems that are taken under consideration consist of video cameras and depth cameras. Both types of cameras may require a different synchronisation signal but need to be synchronised using the same, reference clock source. The author addresses the problems regarding **conversion between synchronisation signals of different types** in order to make synchronisation between video and depth cameras possible. Furthermore, the author describes his design of a **hardware synchronization signal conversion module** which realises the concepts described in the following chapter. The detailed description of the device design is in annex A.

## 4.2 Camera synchronisation methods

Different types of cameras require different methods of external synchronisation. The author considers professional TV production cameras that are used in TV studios along with ToF depth cameras.

Television video cameras use a synchronisation signal which has the form of a blank video frame transmitted according to a particular video standard (e.g. according to SMPTE 274M [SMPTE274M]). This type of synchronisation signal is called a **"Genlock"** signal (an abbreviation for "generator locking") [Kovacs_01]. The Genlock signal carries information about complete video frame timing. The information includes video line delimiters and video frame and/or field delimiters. A video camera synchronises its internal clock generator to the incoming Genlock signal. A field or frame is delimited by a vertical blanking interval, which is a period of time during which no active video data is transmitted.

On the other hand, industrial cameras (such as the ToF depth cameras considered here) require a trigger pulse to begin the frame acquisition process. The **Trigger** signal has the form of a short pulse

that occurs once per video frame. The Trigger signal does not carry precise video frame timing information. The occurrence of a trigger pulse instructs the camera only to capture a frame. A trigger-synchronised camera does not synchronise its internal clock to the signal; it operates using its independent clock source.

Cameras that do not provide the means for external synchronisation cannot be used in a synchronous multi-camera system. These types of cameras operate using only their internal clocks.

The major difference between the Genlock and Trigger signal lies in the meaning of the timing information they carry. The Genlock signal carries precise timing information for a whole video frame, i.e. it contains video line timing information. The Genlock signal is continuous; it is used to synchronise the internal clock of a camera which, in turn, is used to synchronise all internal video signal processing along with the data stream output.

On the other hand, the Trigger signal does not provide any frame timing details. A single Trigger pulse is only an instruction for a camera to start frame capture. A sequence of trigger pulses does not need to be continuous nor does it have a fixed frequency. This allows a camera to capture frames with irregular intervals. A Trigger-synchronised camera cannot derive its internal clock frequency from it; therefore, the camera operates using its internal clock generator.

The difference between operation of a Genlock-synchronised camera and a Trigger-synchronised camera is shown in Figure 4.2.1:

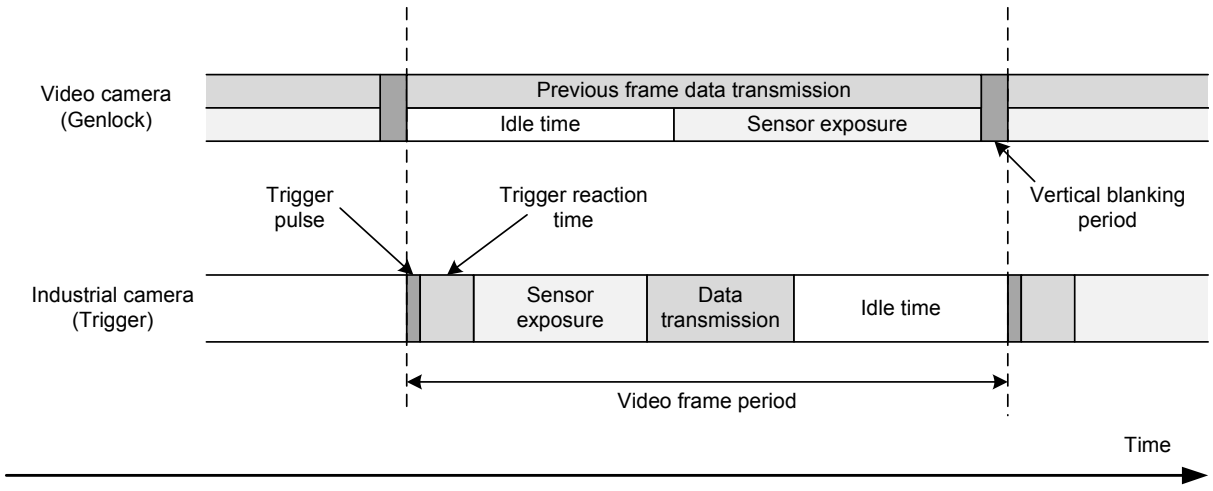



Figure 4.2.1 – Comparison of typical Genlock and Trigger synchronisation signal timing.

It must be noted that a Genlock-synchronised camera performs the frame acquisition and data transmission tasks simultaneously. The camera transmits data from a previous frame during the current frame exposition. This is not true for an industrial camera as the data transmission interface is not synchronised with a Genlock signal. The example timing, as shown in Figure 4.2.1, describes a camera which operates sequentially, i.e. data is sent directly after the sensor exposition interval.

## 4.3 Significance of synchronisation

In this chapter the author will focus on the influence of camera synchronisation on depth estimation in multi-camera systems. The author will provide a mathematical description of disparity estimation for a moving point in 3D space. This description will allow to assess how the motion of a point influences its disparity estimate when the multi-camera system is synchronous and when it is not. Moreover, the author conducted an experiment in order to prove the significance of camera synchronisation for depth estimation.

During the depth map estimation process, the disparity between corresponding points in two images is estimated. In a linear multi-camera system the disparity value is directly related to the distance between a point and the camera set in 3D space. For a synchronous multi-view sequence, each multi-view frame represents the scene at the same moment in time but seen from different positions in 3D space by each camera. When the sequence is not synchronous, each view may represent the scene at an arbitrary time instant. The lack of synchronisation does not manifest itself when the scene is stationary; however, for moving objects a problem emerges. Disparities, estimated by a depth estimation algorithm, do not necessary conform to actual object distances, i.e. they are a combination of object distance and its motion influence.

Let $\vec{q}$ be a moving point in 3D space whose motion is described by a constant 3D motion vector $\vec{v}$. The position of the point in 3D space at time instant *t* is denoted as $\overrightarrow{q^t}$. An image of the point is sampled by a parallel stereoscopic camera system. The cameras are placed next to each other along the *x* axis of the global coordinate system. Their optical axes are both parallel to the *z* axis. The situation is shown in Figure 4.3.1.

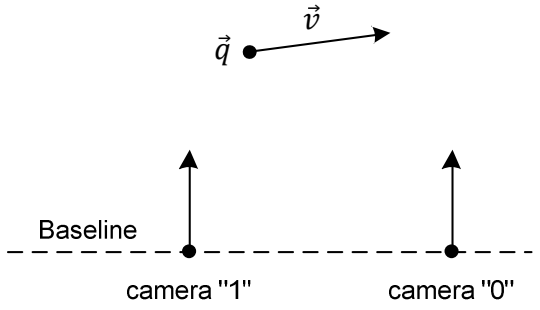



Figure 4.3.1 – Illustration of the stereo camera system under consideration.

The projection of point $\vec{q}$ at a time instant *t* onto the *i*-th camera image can be described by equation 4.3.1:

$$\vec{p}_i^t = \mathbf{K}_i \cdot [\mathbf{R}_i | \mathbf{T}_i] \cdot \vec{q}^t \,, \tag{4.3.1}$$

where $\overrightarrow{p_i^t}$ is the position of the projection of point $\overrightarrow{q^t}$, $\mathbf{K}_i$ is the *i*-th camera intrinsic matrix and $\mathbf{R}_i$ and $\mathbf{T}_i$ are the rotation matrix and translation vector for the *i*-th camera. For a parallel multi-camera system, the translation vectors contain only the horizontal translation component along the *x* axis as the cameras are placed next to each other. Moreover, as the optical axes are parallel, the rotation matrices are equal to the 3x3 identity matrix. Assuming that the cameras are identical, their intrinsic matrices $\mathbf{K}_i$ are also equal. The projection equation can then be simplified to:

$$\vec{p}_i^t = \mathbf{K} \cdot [\mathbf{I}|\mathbf{T}_i] \cdot \vec{q}^t \,. \tag{4.3.2}$$

After substituting the corresponding matrices and point coordinates, the equation takes the following form:

$$s_i^t \cdot \begin{bmatrix} p_{ix}^t \\ p_{iy}^t \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & t_{ix} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} q_x^t \\ q_y^t \\ q_z^t \end{bmatrix}, \tag{4.3.3}$$

where $q_x^t$, $q_y^t$, $q_z^t$ are the coordinates of point $\vec{q}$ in a 3D space, *fx* and *fy* are the focal lengths of projections defined by the pinhole camera model, $c_x$ and $c_y$ are the principal point coordinates with respect to the image frame, $t_{ix}$ is the horizontal translation of the *i*-th camera, $p_{ix}^t$, $p_{iy}^t$ are the coordinates of projection of point $\vec{q}$ and $s_i^t$ is a scaling factor for the *i*-th camera. The scaling factor corresponds to the distance between the *i*-th camera and the observed point. As the cameras are placed on the *xy* plane and oriented toward the *z* axis direction, the scaling factor $s_i^t$ is equal to the observed point *z* coordinate, hence $s_i^t = q_z^t$.

In a stereoscopic camera system the cameras are placed side-by-side horizontally. In that case disparity is defined as the difference of the horizontal coordinates of projections of point *q* onto each image plane of a camera. Assuming that the cameras are labelled “0” and “1”, the disparity can be written as:

$$d = p_{0x} - p_{1x} \,, \tag{4.3.4}$$

where $d$ is the disparity expressed in spatial sampling periods and $p_{0x}$ and $p_{0y}$ are $x$ coordinates of projections of point $\vec{q}$ on camera "0" and "1", respectively. The disparity value can be either positive or negative, depending on the ordering of the cameras. The $x$ coordinate of each projection can be derived from Equation 4.3.3:

$$p_{ix}^t = \frac{f_x}{q_z^t} \cdot (q_x^t + t_{ix}) + c_x \,. \tag{4.3.5}$$

Finally, for the synchronous stereoscopic system the disparity can be expressed as follows:

$$d = \frac{f_x}{q_z^t} \cdot [(q_x^t - q_x^t) + (t_{0x} - t_{1x})] \,, \tag{4.3.6}$$

The disparity value is reciprocal to the distance between the point and the camera set. An additional factor is the relative translation of the cameras, i.e. the baseline. Other point coordinates than $z$ do not influence the disparity.

If the system is not synchronous, Equation 4.3.6 no longer applies as the observed position of the moving point $\vec{q}$ is different for each camera. Let us assume that camera 0 samples the scene image at time instant $t_0$ while camera 1 does so at time instant $t_1$. Equation 4.3.6 now takes the following form:

$$d = f_x \cdot \left[ \frac{q_x^{t0} + t_{0x}}{q_z^{t0}} - \frac{q_x^{t1} + t_{1x}}{q_z^{t1}} \right]. \tag{4.3.7}$$

Assuming that the motion of point $\vec{q}$ is constant and described by a vector $\vec{v}$, we can substitute $q_x^{t0} + v_x \cdot \Delta t$ for $q_x^{t1}$ and $q_z^{t0} + v_z \cdot \Delta t$ for $q_z^{t1}$. This results in the following Equation:

$$d = f_x \cdot \left[ \frac{q_x^{t0} + t_{0x}}{q_z^{t0}} - \frac{(q_x^{t0} + v_x \cdot \Delta t) + t_{1x}}{(q_z^{t0} + v_z \cdot \Delta t)} \right], \tag{4.3.8}$$

where $\Delta t$ is the time difference between frame acquisition of the two unsynchronised video cameras. According to equation 4.3.8, the relation between disparity $d$ and the distance of the observed point $\vec{q}$

is no longer reciprocal. The disparity value is strongly affected by the motion of the point and no longer represents the distance to an object.

In the following tables, several numerical examples are shown of camera synchronisation influence on depth estimation. Table 4.3.1 presents the parameters of an example stereoscopic system. Table 4.3.2 shows the disparity values estimated for different objects moving with different speeds for a synchronous and a non-synchronous system. It is assumed that the delay between frame acquisitions of two cameras in the non-synchronous system may reach **20ms**, which is half of the frame period for a 25 fps camera.

Table 4.3.1 – Parameters of an example stereoscopic system.

| Parameter: | Value: |
|---|---|
| Camera resolution [px] | 1920x1080 |
| Image sensor size [mm] | 5.76 mm x 3.24 mm (3.0 µm spatial sampling period) |
| Lens focal length [mm] | 6 mm |
| System baseline [m] | 0.25 m |

Table 4.3.2 – Disparities estimated for a moving object for a synchronous and non-synchronous stereoscopic system.

| Object | Distance | Lateral speed | Disparity [px] (synchronous system) | Disparity [px] (non-synchronous system) | Disparity estimation error [px] |
|---|---|---|---|---|---|
| A walking man | 10m | 5.0 km/h | 50.00 | 55.50 | 5.5 |
| A cyclist | 10m | 15.0 km/h | 50.00 | 66.60 | 16.6 |
| A moving car | 10m | 30.0 km/h | 50.00 | 83.3 | 33.3 |

It should be noted that the disparity is not affected by a vertical motion of the object, provided that there is also no motion in the *z* direction. However, as the object position on the image changes, the depth estimation algorithm may not be able to find the correct correspondence between appropriate points of the object, as the search is usually limited to the horizontal direction only.

***Experimental results***

In order to test the influence of camera synchronisation on the state-of-the-art depth map estimation, the author conducted an experiment using the "Poznan Street" reference multi-view sequence [CFP]. The original sequence contains 9 views; depth maps are available for cameras 3,4 and 5. For the considered multi-view sequence, two depth maps were estimated for camera 4 using a camera pair formed by cameras 3 and 4. The software used for depth map estimation was DERS 5.1 reference software [DERS] with a configuration file as specified in the original publication for the

sequence with some minor changes [Domanski_15]. The semi-automatic depth estimation mode was changed to a fully automatic mode. In order to estimate depth using only two views, the centre camera was set to camera “4” while the left and right cameras were both set to camera “3”.

The following set of images, as shown in Figure 4.3.2, presents depth maps estimated for a synchronous and simulated non-synchronous version of the sequence. The depth map shown on the left was estimated for the synchronous sequence. The depth map on the right was estimated using an identical parameter set, but the colour data for camera “4” was shifted one frame backward in time with respect to the camera “3”.

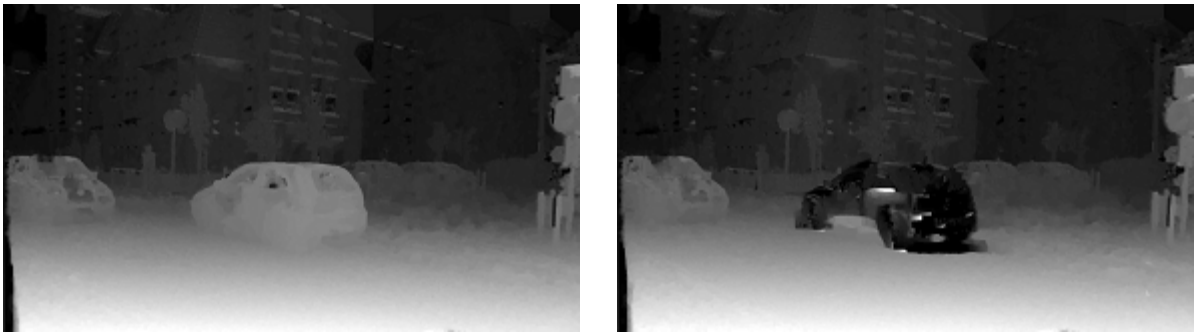


Figure 4.3.2 – Depth maps for camera “4” of the “Poznan Street” sequence as estimated for a synchronous system (left image) and for a simulated non-synchronous system (right image).

Depth maps obtained for the non-synchronous camera pair exhibit false disparities on moving objects (the car), while the background remains unaffected. This phenomenon is visible on the moving car as it changes its horizontal position in time within the image frame. Depth values in that region of the image are different than they are on the correct depth map for the synchronous system.

## 4.3 Conversion between different synchronisation signals

There is very little information available in the literature regarding conversion between different camera synchronisation signals. There are sources dealing with general issues regarding frequency conversion, e.g. Calbaza et al. proposed the implementation of a digital phase locked loop (PLL) for application in Genlock-synchronised systems [Calbaza_01]. Unfortunately, the solution proposed by Calbaza et al. does not address the problem of conversion between different synchronisation signals. The author did not find any literature directly devoted to this matter.

### 4.3.1 Conversion from a trigger signal to a Genlock signal

The method requires the derivation of a high frequency reference clock signal (used later for Genlock signal generation) from a trigger signal which as the form of periodic pulses [Calbaza_01].

Unfortunately, this approach is very sensitive to input signal jitter. The term "jitter" relates to a phenomenon that can be observed as short-term periodic signal frequency deviation from its desired

long-term frequency. The problem with jitter is that when the frequency of a signal is being multiplied, its jitter is also being multiplied by the same factor. Typically, trigger pulses occur once per video frame (a frequency of tens of Hz), and the Genlock signal requires a reference clock of a frequency of tens of MHz. The apparent range of the required multiplication factor makes the reference clock very sensitive to any jitter present in the trigger signal.

Jitter present in the Genlock signal is very much undesired since a synchronised camera uses it to derive its own internal clock frequency. As a result, the camera may not be able to lock its internal clock source to the Genlock signal correctly.

### 4.3.2 Conversion from a Genlock signal to a trigger signal

The approach based on the derivation of a trigger signal from a Genlock signal is less prone to jitter than the previous method. There is no requirement regarding frequency multiplication. The frequency of pulses in a trigger signal is equal to the frame rate, which is significantly lower than the clock frequency embedded in a Genlock signal. Frequency division reduces the jitter, hence the frequency of the derived trigger signal is more stable in time.

A trigger pulse occurs once per frame. The conversion from a Genlock to a trigger signal requires proper detection of frame/field delimiters in the Genlock signal. In a progressive scan system there is a single vertical blanking period for a frame. In an interlaced system there are two vertical blanking periods for each field. Figures 4.3.1 and 4.3.2 summarise the structures of a video frame in a progressive and interlaced scan system.

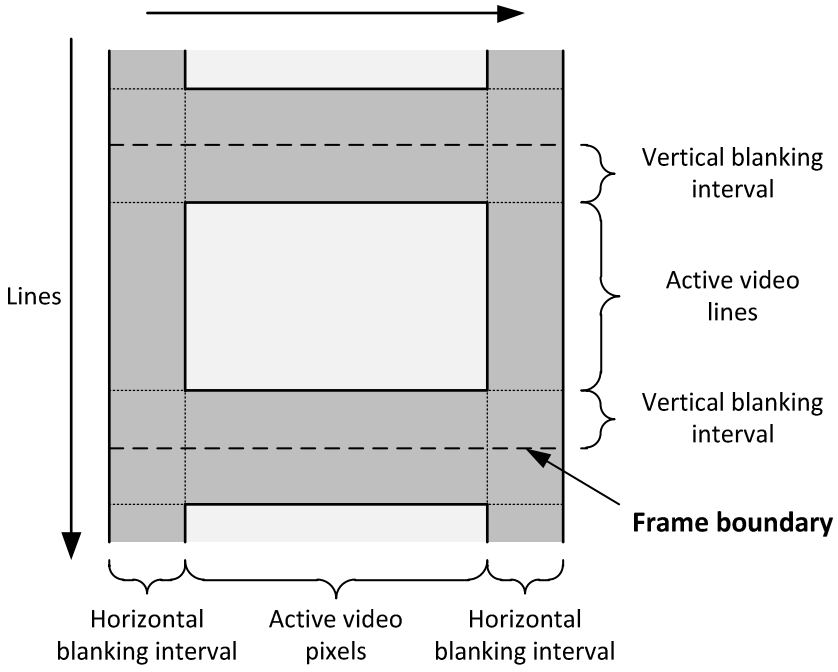



Figure 4.3.1 – Structure of a progressive video frame.

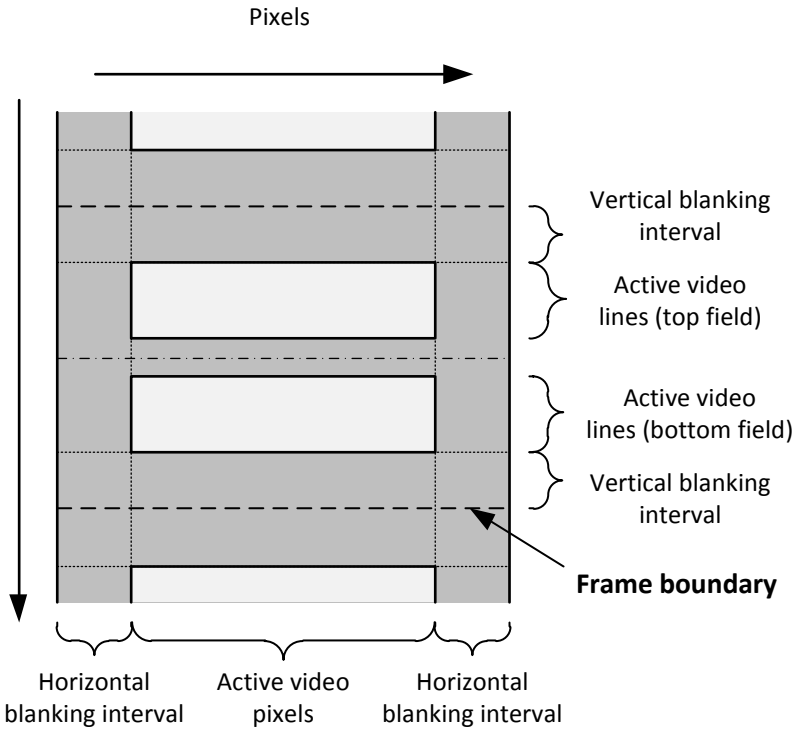



Figure 4.3.2. – Structure of an interlaced video frame.

A trigger pulse should be issued for every vertical blanking period which is a frame (not field) boundary.

### 4.3.3 Proposed solution

The author proposes to **derive the trigger signal from the Genlock signal**. This method is less prone to clock jitter and provides more flexibility as the output trigger signal does not need to comply with any strict timing constraints required by a particular video standard [SMPTE274M].

The conversion method, as proposed by the author, is based on issuing a single trigger pulse after the frame delimiting vertical blanking period that is present in the Genlock signal. The proposed method allows to introduce a fixed delay between the frame delimiter and the trigger pulse. The delay is required to compensate for the difference between the actual image capture time instants in the Genlock and trigger-synchronised cameras. The delay is also needed to compensate for the trigger reaction time of a trigger-synchronised camera.

The proposed method can operate for both progressive scan systems and interlaced scan systems. Figure 4.3.3 illustrates the principle of operation of the proposed conversion method.

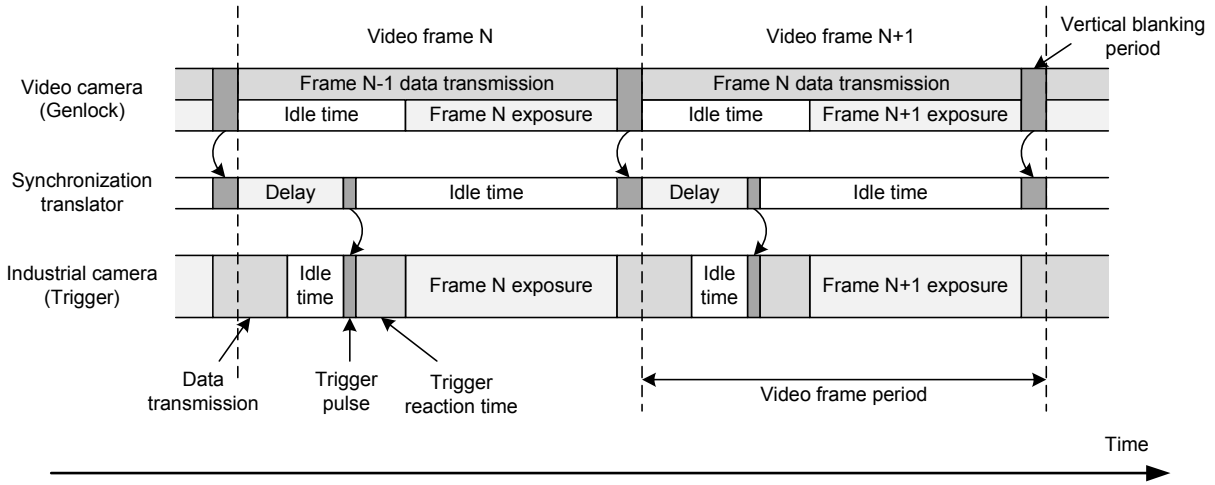



Figure 4.3.3 – Timing relation for a trigger signal derived from a Genlock signal used in the proposed solution.

The proposed method can provide a periodic trigger signal with a very stable frequency as no clock frequency multiplication occurs.

Derivation of the trigger pulses from a Genlock signal also provides the means of triggering the frame capture only at particular frames and for interleaved triggering of multiple cameras. Interleaved triggering is important when a camera (especially a ToF depth camera) has a longer exposure time than the frame rate period. Multiple ToF cameras may be triggered sequentially, and each of them will then provide a depth sequence with a lower frame rate than video cameras. All of these depth sequences may be combined via the video and depth fusion method that is described in this dissertation. This will allow to obtain a high quality depth sequence for a video camera while taking advantage of the longer exposure time of each individual ToF camera. This is not possible in the opposite approach, as the Genlock signal must be continuous and comply with the appropriate video standard which, in turn, enforces a particular and fixed frame rate.

The conversion between a Genlock signal and a trigger signal requires a hardware device dedicated to this task. The device must be able to correctly decode an input Genlock signal and to generate output trigger pulses according to the available synchronisation information. **The author proposes a design of such a synchronisation translation device.** The proposed design is based on a field programmable gate array (FPGA) chip which is suitable for high frequency signal processing. The block diagram of the architecture of the proposed device is shown in Figure 4.3.4.

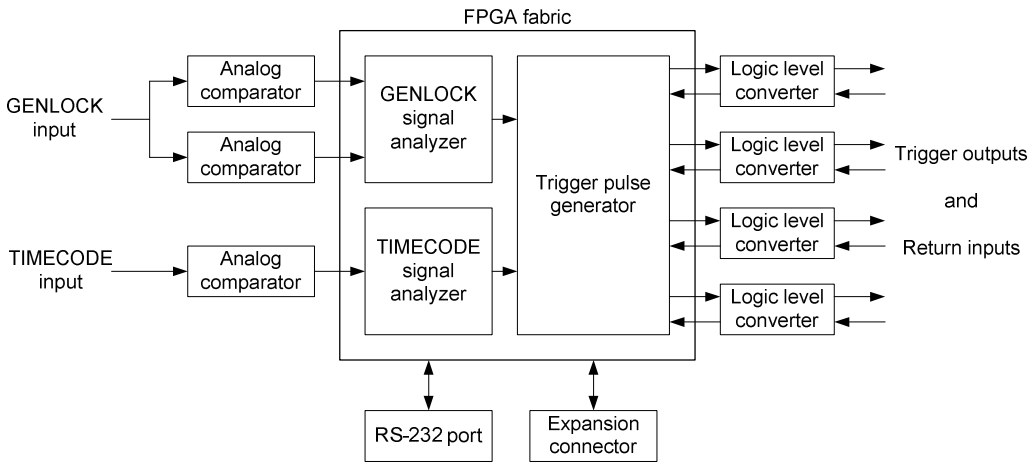



Figure 4.3.4 – Block diagram of the synchronisation signal conversion device proposed by the author.

The device consists of an FPGA chip with additional input and output signal buffers and communication interfaces. All of the synchronisation information processing takes place inside the FPGA chip. The device can perform the following functions:

- conversion from a Genlock signal to a trigger signal
- programmable delay between the Genlock vertical blanking interval and output trigger pulse
- independent triggering of up to four cameras
- programmable triggering of cameras only for selected video frames according to the input timecode signal.

The two input analog comparators are required to properly decode the tri-level synchronisation pulse which is present in the Genlock signal for a Full-HD video [SMPTE274M]. The third analog comparator is used for the timecode signal input. The logic level converters provide signal conditioning for the output trigger pulses.

Detailed information about the proposed design can be found in annex A.

## 4.4 Conclusions

Synchronisation in a multi-camera acquisition system is crucial for any processing which incorporates inter-view image relations. Lack of synchronisation in a multi-view video sequence causes improper depth estimation as disparities between neighbouring views are caused not only by different camera positions but are also a result of object motion. This leads to an estimation of invalid depth maps that does not reflect the distance between the camera and the scene.

A synchronisation converter is a dedicated device which is able to translate the synchronisation information between different synchronisation signals such as the Genlock signal and the trigger signal. The device is required for a multi-camera system that consists of different types of cameras

using different synchronisation mechanisms (video cameras and ToF cameras). The synchronisation converter must be aware of the time delay between the synchronisation information delivered to each camera and its actual sensor exposure time in order to properly synchronise these time periods. However, the author has found that for some cameras the delay can have a random value and therefore is unpredictable, which makes accurate synchronisation impossible to achieve. Fortunately, when the range of delay is relatively small as compared to the overall sensor exposure time and video frame rate, the unwanted effects are negligible and the system may be considered as fully synchronous.

In this chapter the author proposed a method of derivation of a trigger synchronisation signal from an input Genlock signal. The proposed method allows to synchronise sensor exposure timing between different types of cameras.

Moreover, the author proposed the architecture of a synchronisation signal conversion device based on an FPGA chip. The details of the design can be found in Annex A. The author managed to manufacture such a device himself. The device was used in all of the multi-camera system acquisition-related experiments performed by the author.

# 5. Fusion of video and depth data

## 5.1 Introduction

In this chapter the author presents the method he proposes for the fusion of video and depth data. The proposed method can be divided into two major steps:

- transformation of depth data into a video camera space followed by aggregation of depth data from multiple depth cameras,
- fusion of video and depth data.

In the first step, data from one or more depth cameras is transformed into a video camera space. This step is necessary as both cameras cannot be positioned in the same point in a 3D space. The transformation compensates for the difference in their view points and directions.

During the second step, the transformed depth data is fused with the video data. The fusion is done via an augmented depth estimation algorithm which accepts additional depth cues provided by the depth camera(s). This allows to take advantage of both stereo matching and depth measurement. The algorithm uses depth confidence data in order to make a decision whether to use captured depth or stereo correspondence.

A block diagram of the algorithm proposed by the author is shown in Figure 5.1.1.

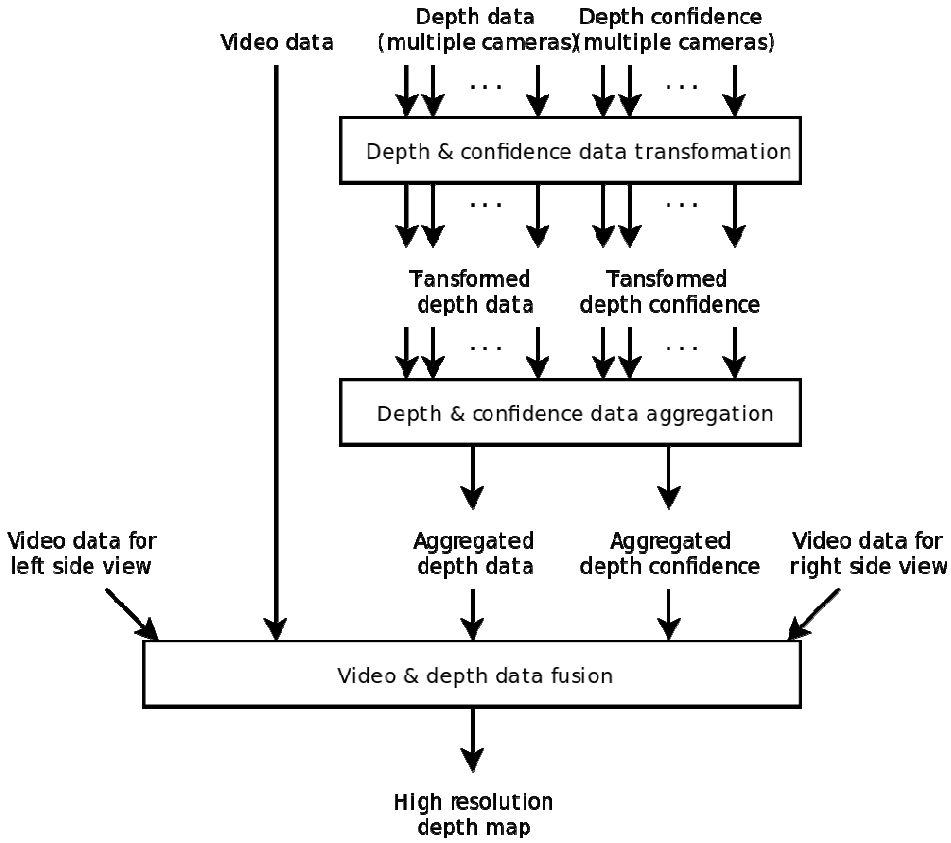



Figure 5.1.1. – Block diagram of the proposed video and depth fusion algorithm.

The confidence data is derived from measured amplitude data. It provides information about a particular distance measurement certainty. The confidence is required by the fusion process as it allows to make the decision whether to use ToF measurement or stereo correspondence information.

Because the proposed fusion algorithm takes advantage of stereo correspondence, more than one video view is required. The left and right side views are necessary to provide these correspondences. These views are videos captured by additional video cameras located left and right of the centre camera for which the depth is being fused.

## 5.2 Geometrical transformation and aggregation of depth data

It is physically impossible to place video and depth cameras exactly at the same location in 3D space. There are systems that use a mirror rig to simulate such a condition [Simanek_01], but despite all efforts it is impossible to align them accurately enough that their images will perfectly overlap. Therefore, **a transformation is required in order to transform depth data into a video camera space**. For such a transformation, both the intrinsic and extrinsic parameters of both cameras need to be known. What is more important is that depth information for the source camera image is required. Since depth cameras provide depth information by their nature, the depth data that they provide can be transformed to a video camera space without the need for prior depth estimation.

### 5.3.1 Transformation of depth data

***Problem statement***

Depth data from a depth camera needs to be transformed to a video camera space. An obvious method would be to project all of the depth pixels into the 3D space and then project them back onto the video camera space according to the DIBR technique [DIBR]. However, the large discrepancy between the resolutions of both cameras poses a problem. When low-resolution depth data is transformed into a high-resolution image, the result is a sparse depth map. Sparse depth map samples cannot be used for interpolation of a dense map. The reason for this is the occlusion effect. Due to the change in view point and its direction, depth samples that define objects with different distances may be interleaved in the target image space. This makes direct interpolation of the transformed sparse depth data impossible as there is no way to define the correct depth value in such areas. Figure 5.3.1 illustrates the problem.

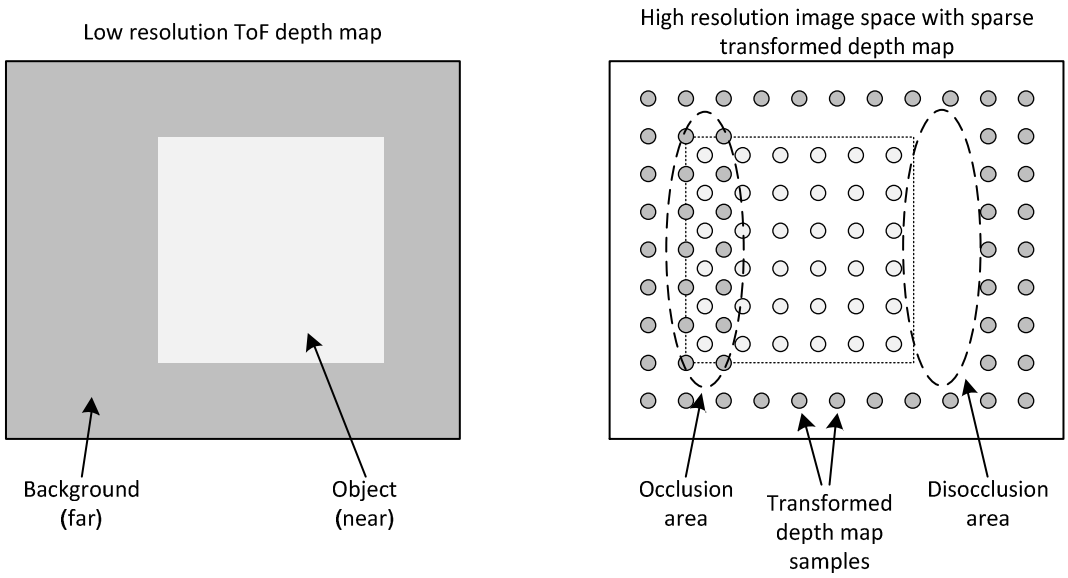



Figure 5.3.1 – Illustration of the low-resolution depth map transformation into a high-resolution image space.

***Solutions known from the literature***

There are several solutions known from the literature that address the problem. Lee et al. presented a method of depth and image data fusion using a direct per pixel transformation of depth samples into a high-resolution image [Lee_02]. An image segmentation technique was introduced which incorporated colour features and is based on the mean-shift algorithm [Comaniciu_01]. In the described method, each segment is assigned an initial depth value which corresponds to a transformed depth sample. As a result, an initial depth map is created. The initial depth map is then processed further. To overcome the occlusion problem, as described in this chapter, each segment is assigned not one but multiple initial depth values. Every possible initial value is taken into account during the processing.

A similar technique was used by Kang et al. [Kang_03], who proposed to directly transform low-resolution depth data into a high-resolution segmented image. The image is segmented using colour features. Each image segment is assigned multiple depth candidates. The depth of a candidate is computed by averaging the transformed depth samples over a local neighbourhood. The size of the local neighbourhood is chosen adaptively according to the size of the disocclusions present after depth data transformation. The final depth value for each segment is calculated by weighting all of the candidates according to their distance from the centroid of a segment.

Eichhardt et al. propose an algorithm for depth map upsampling which takes advantage of a color “guide image” [Eichhardt_01]. Unfortunately, the author cannot incorporate this approach as in his work as the algorithm introduced by Eichhardt et al. does not provide means for occlusion and disocclusion handling which is critical for the author’s approach.

Plank et al. introduce a depth and video data fusion method which also incorporates depth map upsampling using a color guide image [Plank_01]. The depth map is upsampled after being transformed to the video camera space. However, in their approach Plank et al. use ToF and colour sensor which are very close together. The author was unable to place cameras that he used that close together hence the method is not suitable for his system.

All of the techniques as described in the literature are based on depth transformation followed by depth up-sampling. This inevitably leads to erroneous depth in disoccluded areas. The depth samples in the disoccluded areas should be marked as unknown instead of having an interpolated depth value. Because the transformed depth map is sparse, there is no way to reliably determine which areas are actually disoccluded nor to determine their shape.

***Proposed solution***

The author proposes a different approach based on the **construction of a triangular mesh using 3D coordinates of the captured depth map samples projected onto the 3D space**. The 3D mesh can then be projected back onto an image plane. The projected mesh will maintain its structure, thus allowing interpolation of depth, data associated with its vertices, with full occlusion and disocclusion handling.

The input data for the mesh construction is a set of points in 3D space. These points form the vertices of the mesh. Each vertex can be assigned additional attributes, such as confidence of the distance measurement.

A basic primitive of a 3D mesh is a triangle formed by three vertices. The most widely known technique for the construction of triangular meshes is the Delaunay triangulation algorithm [Su_01]. The algorithm allows to create a triangular mesh from an unstructured point set. Unfortunately, there are several reasons why the author chose not to use it. First, the result of Delaunay triangulation depends on point coordinates and on the order in which the points are added to the mesh. That would lead to completely different triangulations for consecutive temporal frames of depth map sequence. The second reason is that the point set is structured which allows to use much simpler, less computational expensive approach.

Due to the rectangular structure of the input depth data structure (2D array), the author proposes to construct a mesh using quadrilaterals instead of triangles. Each quadrilateral of the mesh is formed by four neighbouring points of a depth map. As it is not possible in general to have four co-planar 3D points, each quadrilateral must be composed of a minimum of two triangles. The author proposes to construct quadrilaterals using four triangles according to Figure 5.3.2.

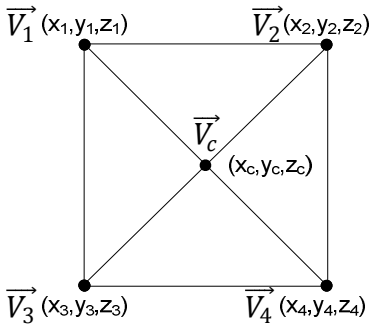



Figure 5.3.2 – Illustration of the quadrilateral construction scheme proposed by the author.

A quad is described by four corner vertices $\overrightarrow{V_1}$, $\overrightarrow{V_2}$, $\overrightarrow{V_3}$, $\overrightarrow{V_4}$ and the centre vertex $\overrightarrow{V_c}$. The centre vertex parameters (coordinates, colour, etc.) are an average taken from among all the other vertices.

In order to transform depth data into a video camera space, the triangular mesh is projected onto the camera image plane. The projected mesh maintains its structure, which allows to perform interpolation of the destination pixels using their coordinates and depths.

The interpolation is done independently for each triangle of a quadrilateral in the 2D space of the destination depth map. For each destination pixel, its barycentric coordinates [Coxeter_01] are computed within the triangle that the pixel belongs to. The barycentric coordinates of a point $\bar{p}$, inside a triangle with vertices $\bar{a}$, $\bar{b}$ and $\bar{c}$ are computed according to the following equations [Coxeter_01]:

$$u = \frac{(\bar{b}_y - \bar{c}_y)\cdot(\bar{p}_x - \bar{c}_x) + (\bar{c}_x - \bar{b}_x)\cdot(\bar{p}_y - \bar{c}_y)}{(\bar{b}_y - \bar{c}_y)\cdot(\bar{a}_x - \bar{c}_x) + (\bar{c}_x - \bar{b}_x)\cdot(\bar{a}_y - \bar{c}_y)}, \tag{5.3.1}$$

$$v = \frac{(\bar{b}_y - \bar{a}_y)\cdot(\bar{p}_x - \bar{c}_x) + (\bar{a}_x - \bar{c}_x)\cdot(\bar{p}_y - \bar{c}_y)}{(\bar{b}_y - \bar{c}_y)\cdot(\bar{a}_x - \bar{c}_x) + (\bar{c}_x - \bar{b}_x)\cdot(\bar{a}_y - \bar{c}_y)}, \tag{5.3.2}$$

$$w = 1 - u - v \text{ „} \tag{5.3.3}$$

where *u*, *v* and *w* are the barycentric coordinates of point $\bar{p}$. Point $\bar{p}$ actually belongs to the triangle when all the barycentric coordinates belong to the <0.0, 1.0> interval and are dimensionless.

The barycentric coordinates are then used as weights for interpolation of the depth data using known values associated with triangle vertices. The interpolation formula is given by equation 5.3.4:

$$d(\bar{p}) = u \cdot d(\bar{a}) + v \cdot d(\bar{b}) + w \cdot d(\bar{c}) \text{ „} \tag{5.3.4}$$

where $d(\bar{p})$ is the interpolated depth at point $\bar{p}$.

Figure 5.3.3 shows an example depth map captured by a ToF camera after preprocessing along with its projection onto a Full-HD video camera image space using the technique proposed by the author. The source ToF camera is located on the right of the target video camera at the same height.

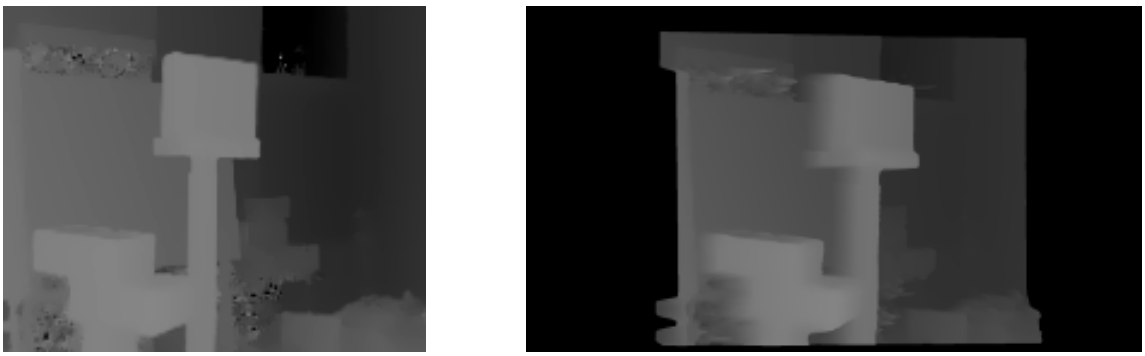


Figure 5.3.3 – Illustration of the depth map transformation using the proposed technique. The left image shows an original depth map while the right image shows the transformed depth map. The black colour on the transformed depth map indicates an unknown depth value.

The transformed depth data does not cover the whole image area. This is due to the different field of views of both cameras in conjunction with their relative positions in space. Areas where there is no depth data available are marked as an unknown depth.

As the mesh represents a continuous surface, a problem arises in the disoccluded areas, i.e. parts of the scene that are visible by the target video camera and are not visible by the source ToF camera. In such areas the rendered surface should not be continuous. The author proposes to handle mesh surface discontinuities by removing quadrilaterals that fulfil certain conditions.

The author proposed to remove quadrilaterals which extent along the *z* axis is greater than a threshold value *th*. The *z* axis extent of a quadrilateral with vertices $\overrightarrow{V_1}$, $\overrightarrow{V_2}$, $\overrightarrow{V_3}$, $\overrightarrow{V_4}$ is defined by the equation 5.3.5:

$$Z_{extent} = max\left(\vec{V}_{1z}, \vec{V}_{2z}, \vec{V}_{3z}, \vec{V}_{4z}\right) - min\left(\vec{V}_{1z}, \vec{V}_{2z}, \vec{V}_{3z}, \vec{V}_{4z}\right), \tag{5.3.5}$$

where $Z_{extent}$ is the range occupied by the quadrilateral in *Z* dimension. The author proposes to remove quadrilaterals for which the $Z_{extent}$ value exceeds the threshold parameter *th*. The method is illustrated in Figure 5.3.4.

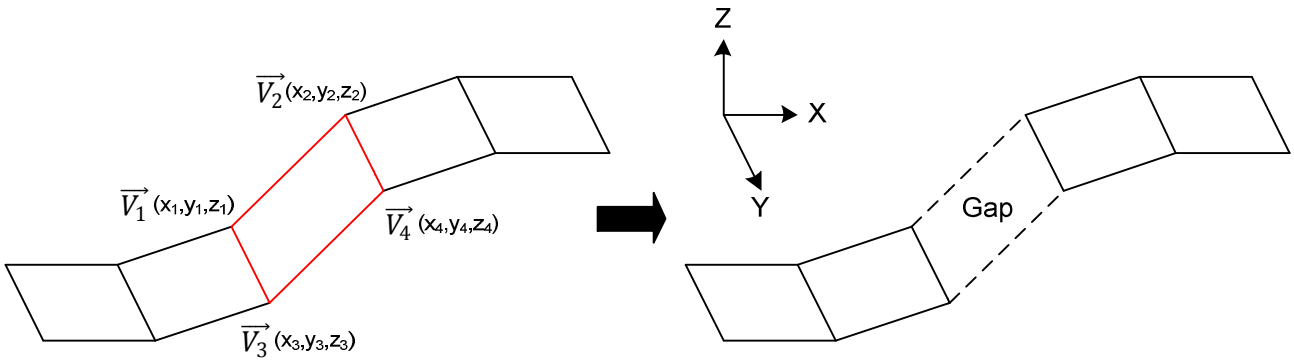



Figure 5.3.4 – Illustration of disocclusion handling in the mesh construction.

The value of $Z_{extent}$ for the quadrilateral marked in red exceeds the threshold; hence it is removed from the mesh. Figure 5.3.5 shows a transformed depth map using the proposed technique. The disoccluded areas are now discontinuous, as they are supposed to be.

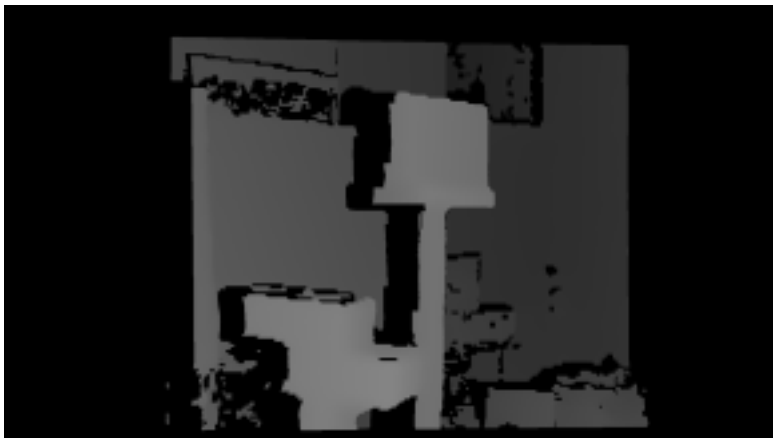


Figure 5.3.5 – Illustration of the rendered mesh of a 3D scene with disocclusion handling. The black colour indicates an unknown depth value.

The choice of the threshold parameter *th* is arbitrary however, its value should not be lower than the distance measurement resolution of the ToF camera. Basing on parameters of Mesa Imaging SR4000 ToF camera used by the author (which are summarised in table 2.6.1) [SR4000] the author proposes to set the value of *th* to 10cm.

***Conclusions***

A depth map transformation using mesh construction and projection provides an accurate means of depth information transfer between low-resolution ToF cameras and high-resolution video cameras. There are no ambiguities during interpolation as the depth samples are not transformed independently (as is done in the DIBR technique) but within a structure that keeps their mutual relations. This

ensures that the depth samples which represent two objects at different distances will not be scattered over the same area of the destination depth map.

The proposed method does not depend on colour information as the other various techniques described in the literature do. Because the structure of the mesh is known there is no need for any further decision to make as to which area of the destination image is to be assigned with a particular depth value.

### 5.3.2 Aggregation of data from multiple depth cameras

***Problem statement***

When a multi-camera system has more than one depth camera, **data from all the depth cameras can be aggregated**, which yields a wider field of view (more scene coverage) and better depth measurement accuracy. However, the most important benefit of multiple depth cameras is the ability to create a full and disocclusion-free 3D scene representation; therefore, in this chapter the author presents his method of aggregation of data from multiple depth cameras.

Disocclusion occurs when a part of the scene is visible by a target video camera and is not visible by a source depth camera. This situation is illustrated in Figure 5.3.6. Lest us suppose that there are two depth cameras oriented toward the same direction with a video camera between them. The scene contains a single object located in front of a flat background. Each individual depth camera cannot see the part of the scene behind the object which is visible to the video camera. As a result, each depth camera provides an incomplete 3D scene representation.

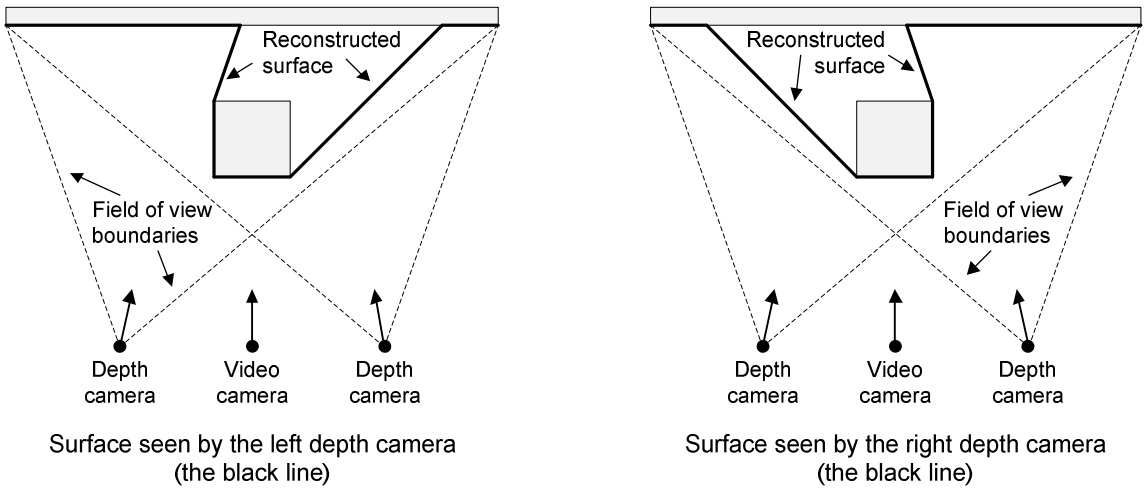



Figure 5.3.6 – Illustration of scene fragments that are visible by each depth camera (black thick line). The Figure shows the scene and the camera setup as viewed from above.

Fortunately, in the specific camera positions as shown in Figure 5.3.6 there are no parts of the scene that are covered by the field of view of the video camera and are not covered by the field of

view of any depth camera. This is illustrated in Figure 5.3.7. In such a situation a complete depth map of the scene can be generated for the video camera. The representation will still not fully resemble the scene as there are still areas that are not visible by any of the depth cameras. However, this is not important since for correct video and depth data fusion only parts of the scene seen by the video camera are required to have known depth information.

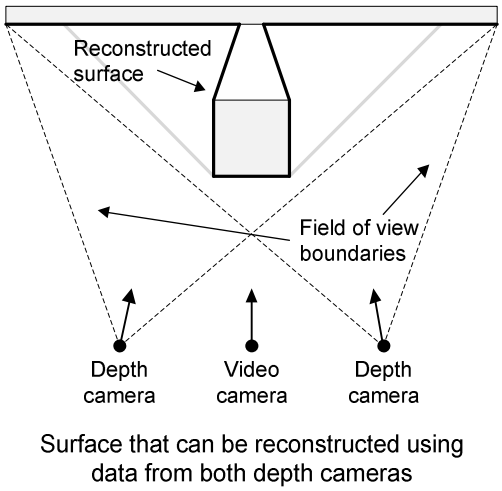



Figure 5.3.7 – Illustration of scene fragments that can be reconstructed using data from both depth cameras (black thick line).

The question arises as how to aggregate depth information in areas that are visible to more than one depth camera. From the theoretical point of view the measured distance values should be identical; however, due to various factors such as light reflections, inaccurate camera parameters or depth camera calibration these values may differ.

***Proposed solution***

The author proposes to aggregate data from multiple depth cameras by taking the farthest depth among all the available depth maps at every spatial location. In areas where there is only a single valid measurement, its value should be taken directly.

Taking the farthest depth allows to resolve problems with disocclusions. When a disocclusion happens, a part of the unknown background that was covered by an object should be revealed. The background is always farther than the object and it will always be farther than the depth that can be interpolated using data at the disocclusion area boundaries. By taking the farthest depth during aggregation the known background data from the other depth camera will always take precedence over the interpolated depth data.

A block diagram of the data flow is presented in Figure 5.3.8. The first stage is transformation of a depth map into a video camera space according to the algorithm described in chapter 5.3.1. During the second stage all of the transformed depth maps are aggregated according to Equation 5.3.1.

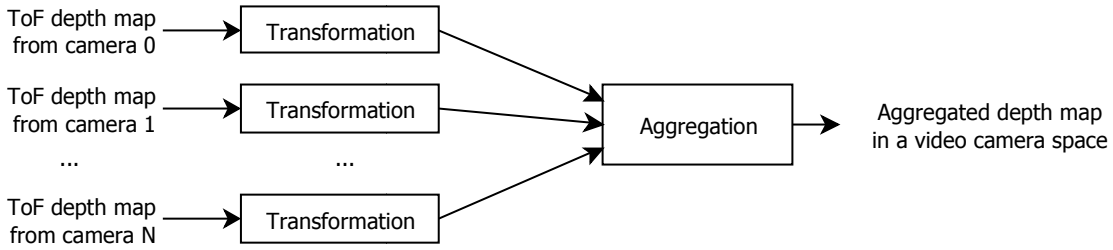



Figure 5.3.8 – Block diagram of ToF depth map aggregation.

Measured distance values are used for aggregation instead of depth values. The value of each sample in the aggregated depth map is computed according to the following equation:

$$z = max(z_1, z_2, \dots, z_N)\,, \tag{5.3.6}$$

where $z_k$ is the distance value of the depth sample for the $k$-th transformed input depth map and $z$ is the aggregated distance value. $N$ is the number of input depth maps.

The illustration in Figure 5.3.8 shows transformed depth maps for the video camera using data from the depth cameras. The cameras are placed in the space according to Figure 5.3.7. The depth maps were obtained using the mesh rendering technique as described in chapter 5.3.1. The left-hand image shows a depth map obtained using only data from the left depth camera, while the right-hand image shows a similar situation for the right depth camera. Bright pixels indicate close objects while dark pixels indicate far objects. The black areas indicate unknown depth values. The disocclusion effect manifests itself as the "shadow" cast by the foreground objects onto the background objects.

Figure 5.3.8. – Depth maps obtained using a single depth camera. Left image – the depth camera was located to the left of the video camera; right image – the depth camera was located to the right of the video camera.

Figure 5.3.9 shows a depth map after aggregation according to the method proposed by the author. A large number of areas where the depth value was unknown due to the use of a single depth camera were filled with data from the other camera. Also, the fused depth map data covered a larger area of the image as multiple depth cameras provide a much wider joint field of view. The remaining areas with an unknown depth value indicate parts of the scene which are not visible by any depth camera.

Figure 5.3.9. – Aggregated depth map which is the result of joining two depth maps that were obtained from individual depth cameras.

***Conclusions***

The proposed method of depth data aggregation allows to merge data from multiple depth cameras without much computational overhead. The method is pixel-based and does not make use of the scene structure whose analysis requires complex algorithms. Moreover, the method proves its reliability even when disoccluded areas were not properly identified during transformation of depth data to a video camera space. In this case taking the farthest distance measurement allows to properly reconstruct the missing information. When interpolating a surface over a disoccluded area using data from one depth camera, the resulting surface will always be located closer to the video camera than the invisible disoccluded background.

## 5.4 Fusion of video and depth data

During the fusion of video and depth data step, the transformed low spatial resolution depth data is merged with high resolution video data. Literature provides several approaches to this problem.

Evangelidis et al. propose a method based on hierarchical maximum a posteriori optimization [Evangelidis_01]. In the approach, the fusion is obtained by solving a series of local energy minimization hierarchically, by growing sparse initial disparities obtained from the depth data. The idea is similar to proposal of the author of this dissertation (described in this chapter), however it incorporates only local optimization.

Plank et al. [Plank_01] propose an algorithm which effectively describes a solution to transformation of data from a depth sensor to a video camera space. The transformation is followed by

image guided up-sampling of the depth data. Plank et al. claim that their method is suitable for sensors with resolution of 288x256 for depth and 640x480 for video. Unfortunately, the author uses sensors that exhibit much larger discrepancy in resolution, therefore more elaborate technique is required.

In order to fuse ToF depth data with video camera image data, **the author proposes to modify the DERS depth map estimation algorithm** based on global energy minimisation [DERS]. The incorporation of ToF is performed via modification of the cost model so that both stereo correspondences and ToF measurements are used.

### 5.4.1 DERS depth map estimation algorithm

The DERS depth map estimation algorithm is based on a global energy minimisation which is performed using the Graph Cuts algorithm [Boykov_01]. The global energy minimisation is preceded by initial cost estimation using stereo matching. The algorithm requires three images (left, centre and right) instead of two. This allows to handle occlusions that are inevitable for a stereo camera setup. The depth map is estimated for the central view using stereo correspondence of the centre to left and centre to right image.

The DERS algorithm can operate in a fully automatic mode (video analysis only) or in a semi-automatic mode. The semi-automatic mode allows to manually provide additional depth cues that can be used by the global optimisation algorithm.

The algorithm is developed and maintained by the MPEG group. It is implemented with all of its variations in the DERS (Depth Estimation Reference Software) [DERS].

The algorithm operates on a rectangular grid of pixels. The resolution of the grid is equal to the resolution of the estimated depth map. A global energy (cost) function is defined for each spatial position of the depth map. The function has the following form:

$$E(f) = \sum_{\forall p_i \in P} D\big(p_i, f(p_i)\big) + \sum_{\forall p_i, p_j \in Q} V\Big(p_i, p_j, f(p_i), f\big(p_j\big)\Big), \tag{5.4.1}$$

where $E$ is the energy function, $P$ is the set of all pixels and $Q$ is the set of all pixel pairs.

Function $f$ defines the relation between an actual estimated depth or disparity value and its corresponding numerical label used by the optimisation algorithm. This is necessary as depth/disparity values are in general fractional numbers, hence they cannot be used for indexing directly. The mapping is required as the number of possible disparity values under consideration is finite.

The cost function $D$ defines the cost of having a particular label for each pixel $p_i$. It informs us how well a certain label "fits" into that pixel, hence it is often known as the fitting cost function. The

higher the cost, the less likely a pixel is to have a particular label. The function is three-dimensional as it depends on the spatial position of a pixel and its label.

The second cost function $V$ defines the cost of having a label $f(p_i)$ for pixel $p_i$ under the condition that pixel $p_j$ has a label $f(p_j)$. Essentially, it defines how good two image pixels fit each other with their given labels. Function $V$ is known as the cross cost function. Lower cost values favour that a given label pair be assigned to a specific pixel pair. For simplification of the optimisation algorithm and to reduce its computational complexity, only the influence of neighbouring pixels is taken into account, hence the set $Q$ contains only pairs of neighbouring pixels. In general, all possible pixel pairs should be taken into account.

The DERS algorithm requires three views (left, centre and right) instead of two for occlusion handling [DERS]. For a stereo image pair there are always parts of the scene which are visible by one camera and not visible by the second one. In a three-camera setup there is no place which is visible to the centre camera and not visible by any of the side cameras. The depth map is estimated for the centre camera whereas the other two cameras provide information about disparity cues.

The depth estimation algorithm relies on certain similarity metrics that can be derived from image features. These metrics constitute the cost function $D$ used in the global optimisation algorithm. The most common cost model is the SAD metric computed for image blocks. These image blocks are formed by taking the neighbourhood of a given pixel. The SAD metric is computed for a block in the reference (centre) image and its correspondent in one of the side images for a given disparity value. Different metrics can be used, such as SSD.

As there are two stereo pairs for three input views, the final value of the cost function is computed by taking the minimum among the costs computed using centre-to-left and centre-to-right block matching. This results in taking the best match for each block from either the left or right side view.

On the other hand, the cross-cost function $V$ is modelled using the following equation:

$$V\left(p_i, p_j, f(p_i), f\left(p_j\right)\right) = \lambda \cdot \left|f(p_i) - f\left(p_j\right)\right|, \tag{5.4.2}$$

where $\lambda$ is a smoothness coefficient, which is used to control the smoothness of the depth map by changing the amount of additional cost (penalty) assigned to neighbouring pixels with different labels. The lower the penalty, the smoother the depth map, as the optimisation algorithm tends to choose labels that correspond to similar depth values. In the original algorithm the function does not depend on any image features nor on absolute pixel coordinates in the estimated depth map.

In the fully automatic mode, cost data is computed using image features only. There are two modifications that allow for semi-automatic operation by providing additional manually entered depth cues [Tanimoto_01]. Semi-automatic operation is allowed by providing a manually drawn depth map. Such a depth map can be sparse. It is assumed in the algorithm that each manually provided depth value is certain and therefore it propagates through the global optimisation procedure unchanged. An edge map can be provided along with the manual depth map. The edge map is a binary map that defines the edges of the colour image. Unfortunately, the edges can be specified only as binary, i.e. an edge that is either present or absent dramatically degrades the flexibility of the mode.

The semi-automatic depth estimation mode is not suitable for incorporation of data from depth cameras as the author intended at the beginning of his research. In this chapter the author provides his solution to the problem that is based on a modification of cost function models.

### 5.4.2 Modification of the fitting cost model

The author proposes to incorporate depth data, as provided by one or more depth cameras, into the state-of-the-art depth map estimation algorithm. The idea is **to incorporate additional depth cues into the cost model prior to the global optimisation stage**, which remains unchanged.

The fitting cost function $D$ will now depend on the stereo correspondence features and also on data provided by the depth camera, i.e the measured distance and the confidence of this measurement.

The cost data that comes from stereo correspondence is meant to provide depth cues in areas where image features are reliable for disparity estimation and for some reason the depth camera measurement exhibits low quality (confidence). The stereo correspondence features are also to be used for image areas where there is no depth camera data available. The depth data should be preferred over stereo correspondence everywhere where its confidence is high. This ensures the correct choice of the final depth value if the stereo correspondence is not reliable.

The fitting cost function depends on the spatial coordinates of a pixel and on the label of the candidate disparity value. Generally, a fitting cost function should have a single minimum. The minimum indicates a disparity that corresponds to the best stereo match for that pixel. The following two figures show an example image and the fitting cost functions for given spatial pixel locations.

Figure 5.4.1 shows the luminance of a single frame from the “Poznan Street” sequence from camera 3. There are three characteristic points that are marked, and each one indicates the centre of a 3x3 pixel block. Figure 5.4.2 illustrates the fitting cost curves for these points. The cost was computed using the SAD metric for luminance only. The disparity values indicate disparities between images of camera 3 and camera 4.


Figure 5.4.1 – Single frame from the “Poznan Street” sequence with marked points.

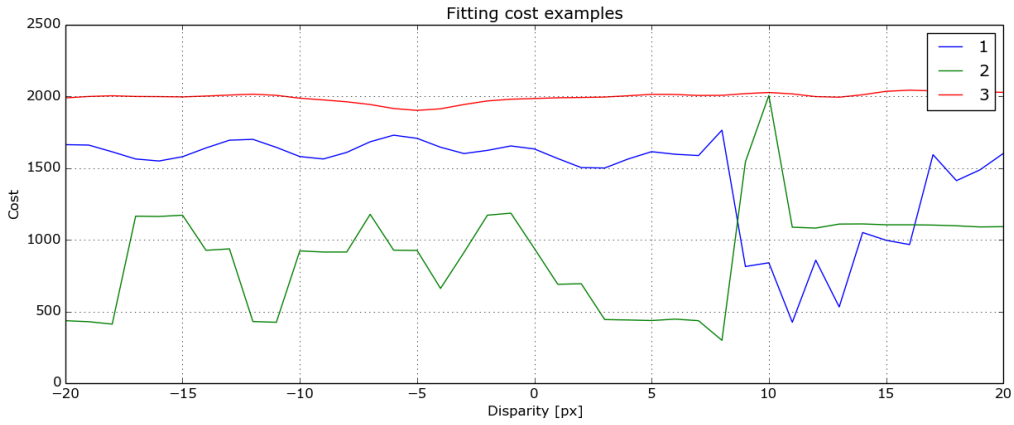



Figure 5.4.2 – Plot of fitting cost curves for points marked in Figure 5.4.1.

Unfortunately, a depth camera does not provide a set of disparities along with their fitting costs. Instead it provides a single depth value for each pixel along with information as to how confident the measurement is. Therefore **the author proposes to use a cost modelling function** that will be used to compute costs for all considered disparity values. The proposed function is parameterized so its shape depends on the single depth value and its confidence provided by the ToF camera.

***Fitting cost functions***

**The author proposes a set of cost functions** that will be used for modeling of the fitting cost for data obtained from a depth camera. Each of the proposed functions has a set of features that are similar to the fitting cost function computed using stereo matching. The fitting cost function must have a single global minimum for a given disparity value that is supposed to be optimal for a given pixel.

The function should rise monotonically along with the increasing difference between the given disparity and the one for which the minimum exists. The greater the difference between the given depth and argument, the higher the cost of selecting that value for the optimal choice.

The author introduces a base cost function that is equal to the absolute difference of a given best disparity value and the disparity considered by the optimisation algorithm. The base cost function is defined by the formula:

$$g(d) = |d - d_0|\,, \tag{5.4.3}$$

where $g$ is the base cost function, $d$ is a possible considered depth value and $d_0$ is the depth provided by the depth camera. The base cost function $g$ is then used to derive the actual cost function used for modelling of the fitting cost term $D$.

The author suggests a set of fitting cost functions that are to be evaluated for their performance in the depth map estimation process. These functions are denoted as $G(d)$:

- Linear slope

  The linear cost function model is given by the equation:

  $$G(d) = a \cdot g(d)\,, \tag{5.4.4}$$

  where a is the coefficient that controls the slope of the cost increase.

- Quadratic slope

The quadratic cost modelling function rises along with the squared difference of disparities. The function is given by the equation:

$$G(d) = a \cdot g(d)^2\,, \tag{5.4.5}$$

where a is the control parameter.

- Inverted Gaussian shape

The third model uses the inverted Gaussian function given by the following formula:

$$G(d) = 1 - e^{\frac{-g(d)^2}{2 \cdot \sigma^2}}\,, \tag{5.4.6}$$

where the parameter σ controls the Gaussian curve shape.

For each proposed cost model the value of 0 indicates the minimal cost and the value of one indicates the maximum possible cost. For the linear and quadratic model, the value of the function $G$ is

clipped so that it is always lesser or equal to one. The Gaussian function is always lesser than or equal to one by its nature.

The following figures show the plots of example realisations of the proposed cost model functions for $d_0$=25 and a disparity range from 0 to 50 pixels.

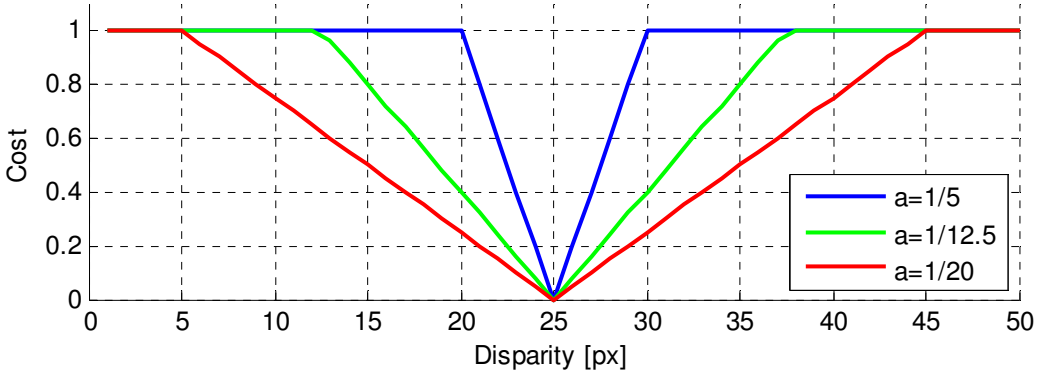



Figure 5.4.3. – Three example realisations of the fitting cost function defined by formula 5.4.4.

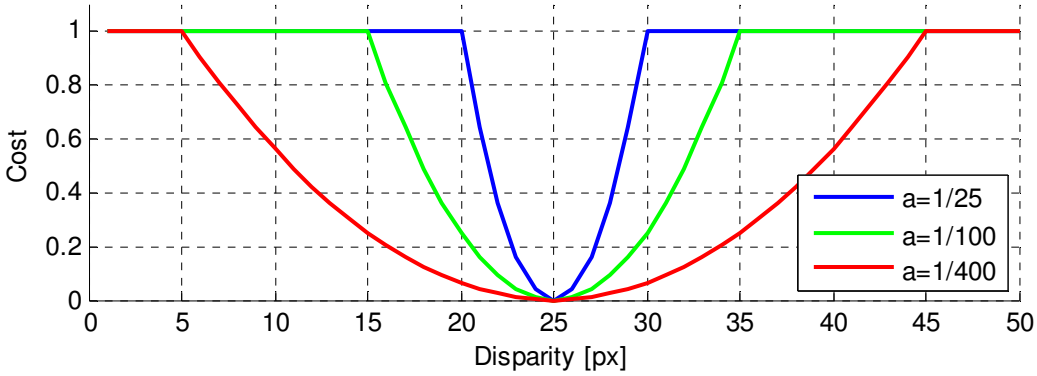



Figure 5.4.4. – Three exemple realisations of the fitting cost function defined by formula 5.4.5.

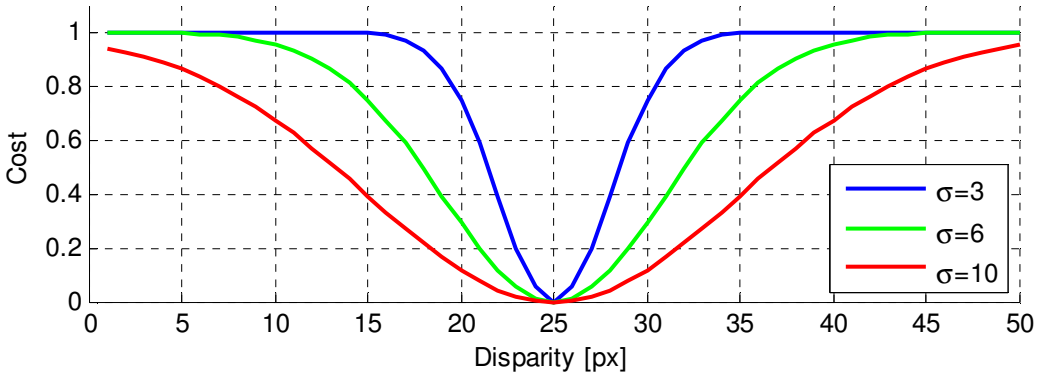



Figure 5.4.5. – Three exemple realisations of the fitting cost function defined by formula 5.4.6.

The author also proposes to use a step cost model. The step function has a constant value of zero for a specific range of depth values and a value of one for the others. The cost model function is given by formula 5.4.7:

$$G(d) = \begin{cases} 0 & for \quad |d - d_0| < b \\ 1 & otherwise \end{cases}, \tag{5.4.7}$$

where $b$ is the control parameter that controls the width of the zero cost range. Figure 5.4.5 shows the plots of three example realisations of the proposed step cost model function.

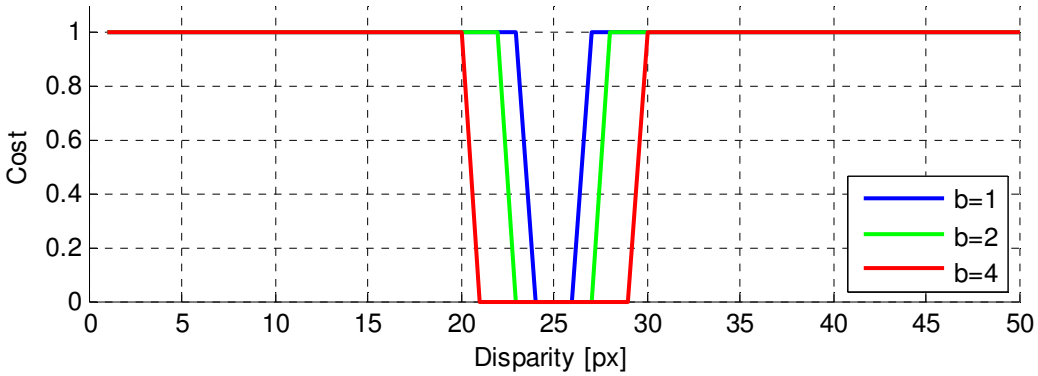



Figure 5.4.6 – Three example realisations of the step cost function defined by formula 5.4.7.

The reasoning behind the choice of function shape is that the depth value as provided by the depth camera may not be the actual correct depth but can be close to the correct one. The step cost function allows for a range of possible depth values to be treated by the processing algorithm as equally probable. In this case, possibly correct depth information from the neighbourhood of a pixel can be propagated without any additional cost penalty.

For disparities that are very different from the assumed optimum given by the depth camera it makes no difference which one will be chosen as all of them are invalid. The correct disparity is usually close to the depth camera measurement. This raises the following question: Does the fitting cost function have to increase along with the disparity difference or is there a point where the disparity difference is so large that a maximum cost value should be selected?

The author proposes a modification to the previously described cost function set which is given by the following formula:

$$G'^{(d)} = \begin{cases} G(d) & for \quad |d - d_0| < b \\ 1 & otherwise \end{cases}, \tag{5.4.8}$$

where $b$ is the factor that controls the maximum disparity difference. Beyond the boundary of $\langle -b|+b\rangle$ the function assumes the maximum possible cost value.

The next three figures illustrate the cost models as defined by Formulas 5.4.4, 5.4.5 and 5.4.6 when using the modified cost function $G'$. Each figure shows three example functions for different settings of the maximum disparity difference.

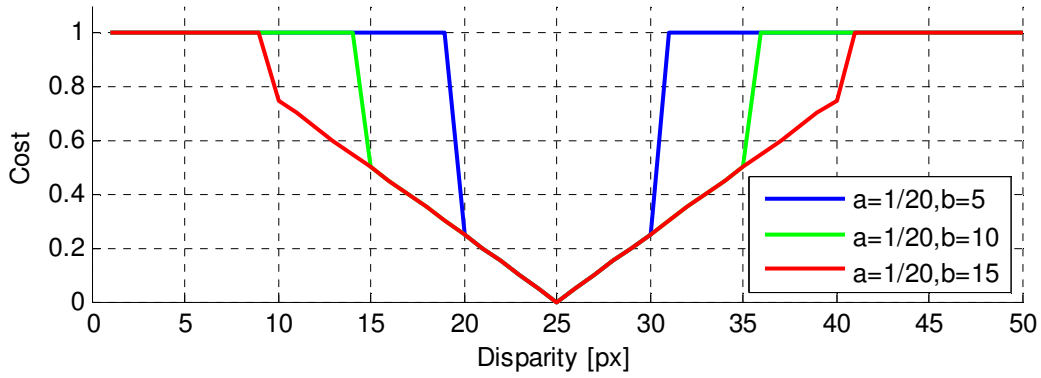



Figure 5.4.7 – Three exemple realisations of the fitting cost function defined by formula 5.4.4 after application of formula 5.4.8.

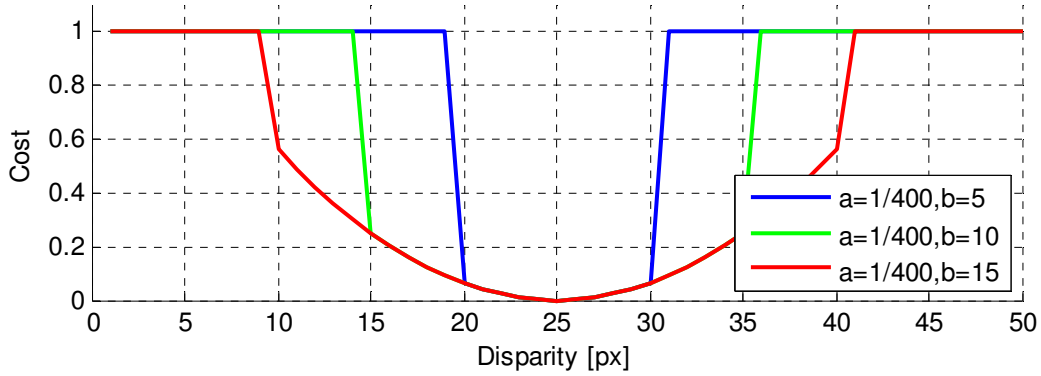



Figure 5.4.8. – Three exemple realisations of the fitting cost function defined by formula 5.4.5 after application of formula 5.4.8.

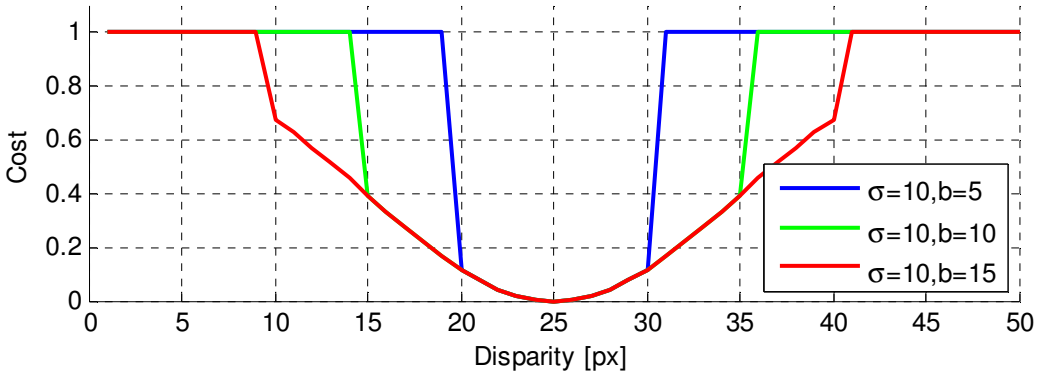



Figure 5.4.9. – Three exemple realisations of the fitting cost function defined by formula 5.4.6 after application of formula 5.4.8.

All of the proposed fitting cost modelling functions allow to introduce depth information that comes from a depth camera into the global optimisation algorithm. Together with the confidence information that is used to decide whether a proposed cost model function or stereo correspondence is better, the global optimisation algorithm can take advantage of both data sources.

***Incorporation of the distance measurement confidence***

Each of the proposed cost modelling functions $G$ takes a value of 0 for the depth value $d_0$ provided by the depth camera. However, it is not always true that this is the best depth value for a given pixel location. Many external factors may influence individual depth measurement. In order to differentiate between certain and uncertain depth measurement it is necessary to take advantage of the measurement confidence value provided by the depth camera.

The author proposes to **use confidence data for weighting** between the proposed cost model function $G$ and the original cost function $D$ derived from stereo correspondence. The greater the confidence value, the more the final cost function should resemble the cost model, so the depth camera measurements are accurate. For a low confidence value a cost model that uses image features should be implemented because the depth camera cannot provide reliable data.

The author proposes two methods of incorporation of confidence data into the fitting cost model:

- linear interpolation between the cost model and the stereo matching cost,
- linear interpolation of two factors. The first is equal to the product of the cost model and the stereo matching cost while the second one is equal to the stereo matching cost itself.

The first proposed method takes advantage of linear interpolation between the fitting cost given by one of the proposed models and the fitting cost computed using the image features of a given stereo pair. formula 5.4.9 formalises the definition of the final fitting cost function D.

$$D(d) = c \cdot G(d) + (1 - c) \cdot D_{SAD}(d)\,, \tag{5.4.9}$$

where D(d) is the final fitting cost function, G(d) is the cost model and DSAD(d) is the cost computed using stereo matching. Factor c is the confidence value; it ranges from 0 (completely unreliable distance measurement) to 1 (a certain measurement).

The linear interpolation between the stereo correspondence-derived cost and the depth camera cost model allows to fluently blend between these two functions. The final fitting cost model function D(d) takes a shape that resembles the cost model G(d) for high confidence values and the stereo correspondence-derived cost model $D_{SAD}(d)$ for low confidence values.

The second weighting method also uses linear interpolation but interpolates between the stereo correspondence cost $D_{SAD}(d)$ for low confidence and the same cost multiplied by an appropriate cost model G(d) for high confidence values. The final fitting cost D(d) is given by the following formula:

$$D(d) = c \cdot D_{SAD}(d) \cdot G(d) + (1 - c) \cdot D_{SAD}(d)\,. \tag{5.4.10}$$

Multiplication of the stereo correspondence cost and the cost modelling function provides the means of shaping the original cost $D_{SAD}(d)$. The shaped cost function exhibits features of both the stereo correspondence and depth camera cost models. Weighting between the original and shaped cost function allows to control the amount of shaping influence.

The second cost weighting method allows to resolve ambiguities in situations where the cost function derived from the stereo correspondence features exhibits more than one local minimum. Multiplying such a function by the depth camera fitting cost model allows to reduce the cost values for depth closer than that measured by the camera. The correct local minimum may then be chosen by the global optimisation algorithm.

The cost function values obtained by SAD or similar metric computation may exceed the range that a cost modelling function can have. Before the weighting procedure, the stereo correspondence cost function must be scaled in order to match that range. The scaling must be uniform for all pixels in order for the costs to be comparable. The author proposes to scale the stereo correspondence cost function to the range from zero to one according to the maximal theoretical cost value that a particular

metric (SAD, SSD, etc.) can take. This value depends on the size of the block used for SAD computation and on the image colour sample's precision.

### 5.4.3 Modification of the cross cost model

The author also proposes to **incorporate image edge features into the cross cost model** *V*. The idea behind this is to penalise situations when two neighbouring pixels have similar depth values and there is an edge in between them. The output depth map will contain sharp edges as the global optimisation algorithm will likely select different depths for such pixel pairs. For edgeless areas, the depth map will remain smooth.

Edges in colour images are likely to be physical object boundaries. Unfortunately, in most cases this assumption is not true, as the presence of an edge does not necessarily indicate an object boundary but might be, for example, an indication of a texture feature. On the other hand, two different objects of the same colour that cover each other will not produce an edge on the colour image.

The modified cross cost model is defined by formula 5.4.11.

$$V\left(p_i, p_j, f(p_i), f(p_j)\right) = w(p_i, p_j) \cdot \lambda \cdot \left|f(p_i) - f(p_j)\right|. \tag{5.4.11}$$

The $w(p_i,p_j)$ is an additional weighting term given by the following formula (for an 8-bit luminance image):

$$w(p_i, p_j) = \frac{1}{1 + k \cdot \left|\frac{I(p_i) - I(p_j)}{255}\right|}, \tag{5.4.12}$$

where $I(p_i)$ and $I(p_j)$ are the luminance values of the colour image for points $p_i$ and $p_j$, respectively. The value of $w$ is equal to one if no edge is present. With the increase of edge strength, the value of $w$ decreases asymptotically to zero. Parameter $k$ controls the rate of the decrease, hence the strength of influence of an edge on the final cross cost value.

***Experimental results***

The following figures show an image and the depth maps estimated using an original and modified cross cost function. The data comes from the Middlebury test data set from year 2006 [Scharstein_01] [Scharstein_02] [Scharstein_03][Hirschmuller_01].

Figure 5.4.10. – Colour image for camera 1 of the "Aloe" multi-view image from the Middlebury test data set.

The left image of Figure 5.4.11 shows the estimated depth map for one of the views available using an original, unmodified automatic depth estimation algorithm (with Graph Cuts). Some parts of the flower blend into the background. On the other hand, the right image shows a depth map estimated with the cross cost function as proposed by the author. All parts of the flower now have the correct depth.

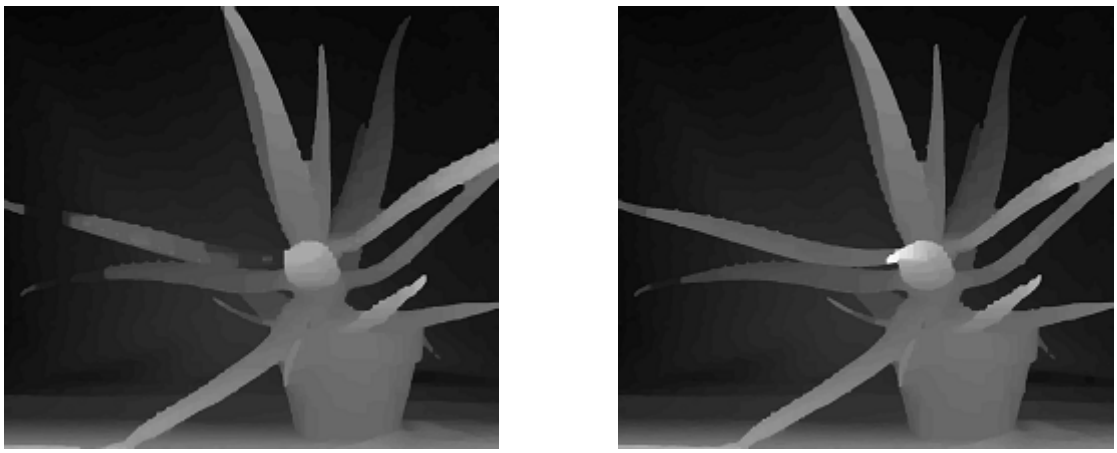


Figure 5.4.11. – Estimated depth map for camera 0. The depth map on the left was estimated with the original cross cost function while the right depth map was estimated using the proposed image edge incorporation technique.

The proposed modification is not directly related to the incorporation of depth camera data into the depth map estimation process and can be used independently. The experiments show that in both cases the method leads to an improvement of depth map quality – more details of the flower shape are now included in the depth map.

## 5.5 Video plus depth multi-view data sets

The author was unable to find any public test multi-view sequences or images data set that contained data from video and one or more depth cameras along with their parameters. Due to the highly experimental nature of such mixed video and depth multi-camera systems, there are only a few laboratories in the world that have them. ToF cameras are rather expensive, which also limits their popularity. The author managed to find certain publications in which the authors described their mixed video plus depth acquisition systems [Kang_02]. Unfortunately, the sequences captured by the described systems were not available.

### 5.5.1 Video plus depth multi-view acquisition system constructed by the author

The laboratory of the Chair of Multimedia Telecommunication and Microelectronics at Poznan University of Technology is equipped with 11 Full-HD video cameras (Canon XH-G1 [XH-G1]) and three Mesa Imaging ToF cameras (SR4000 [SR4000]) that can be used to construct a mixed video plus depth multi-view acquisition system. The author managed to assemble several variations of such a system and to capture some multi-view plus depth data that was used to evaluate the proposed depth estimation algorithm. For the parameters of the Mesa Imaging SR4000 ToF camera please refer to the table 2.6.1. The table 5.5.1 summarises parameters of the Canon XH-G1 camera used:

Table 5.5.1 – Summary of relevant parameters of the Canon XH-G1 video camera.

| Parameter name | Parameter value |
|---|---|
| Resolution | 1920 x 1080 (interpolated using 3x 1440x1080 CCD sensors, one for each colour channel) |
| Sensor size | 1/3 inch |
| Lens focal length | Adjustable 4.5 mm to 90 mm (the author used 4.5mm setting) |
| Frame rate | 25 Hz in progressive frame scan mode |
| Color bit depth | 8-bit |
| External synchronization | Genlock and timecode signal input |

The author has introduced a naming convention for video plus depth sequences recorded at Poznan University of Technology. Each sequence is named according to the following formula:

"*n*T+*m*D"

where *n* denotes the number of video cameras and *m* denotes the number of ToF depth cameras.

Several camera arrangements were evaluated which contain different numbers of video and depth cameras. Each one exhibits various features and advantages. The most basic camera placement that allows to estimate a depth map without occlusion is the 3T+2D (three video cameras plus two

depth cameras). An illustration of such camera placement is shown in Figure 5.5.1. In such a system an occlusion-free depth map can be estimated for the central video camera "1" using colour information from video cameras "0" and "2". The placement of depth cameras at the sides ensures that disocclusions caused by data transformation from one depth camera to camera "1" space can be fully filled with data from the other camera.

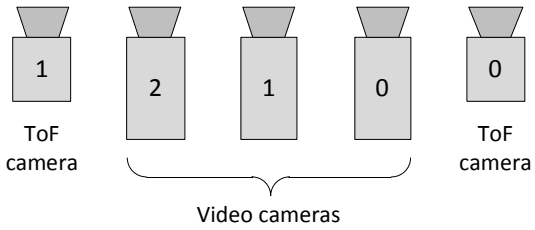



Figure 5.5.1 – 3T+2D video plus depth acquisition system camera placement.

Unfortunately, only one depth map can be estimated in the 3T+2D system. This is not sufficient for virtual view synthesis since in order to avoid disocclusions two depth maps with associated images are required.

Adding more video cameras helps to solve the problem. By adding one or more video cameras, the estimation of two depth maps is possible. The author proposes a system with five video cameras and two depth cameras, i.e. a 5T+2D system. An illustration of the camera placement is shown in Figure 5.5.2 – the number of video cameras is increased by two. This allows for an occlusion-free depth map to be estimated for cameras "1", "2" and "3". Actually, the author suggests to compute the depth map for cameras "1" and "3" and to use camera "2" as a reference for virtual view quality assessment.

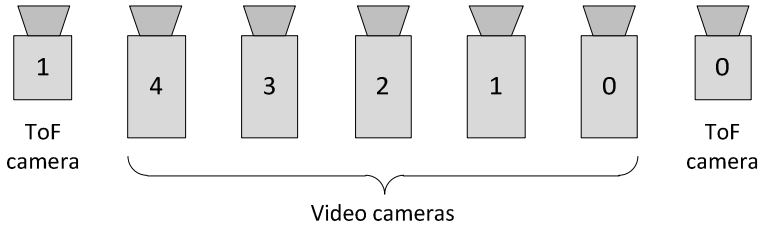



Figure 5.5.2 – 5T+2D video plus depth acquisition system camera placement.

The drawback of such camera placement is the increased distance between one depth camera and the video camera which the depth map is estimated for, e.g. the distance between camera "3" and depth camera "0" is much greater than for depth camera "0". Further placement of a depth camera causes larger disocclusion areas in the transformed depth map. What is more, inaccuracies of depth measurement and camera parameter estimation have greater influence on the depth map transformation.

A 6T+3D system can be constructed by placing the third depth camera in the centre of the system and adding one more video camera. The system basically consists of two 3T+2D systems that share one depth camera. The camera placement is shown in Figure 5.5.3. In such a system an occlusion-free depth map can be estimated for cameras "1" and "4" by using colour information from the neighbouring video cameras. For each depth map, data from the closest two depth cameras can be used.

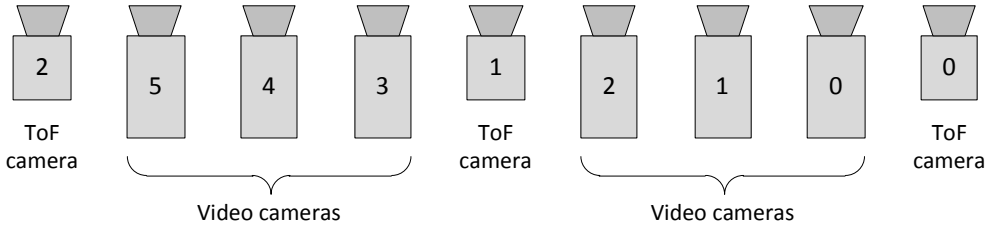



Figure 5.5.3 – 6T+3D video plus depth acquisition system camera placement.

Due to technical reasons, the author was unable to construct a system where, for example, the depth cameras were located above or below the video cameras. Camera placement on a single line enforces the large length of the whole system. This, in turn, poses some requirements for the scene setup, particularly regarding object distances from the system. A large camera spread requires that objects be located farther from the system. On the other hand, the quality of the depth map captured by ToF cameras decreases along with the distance as more intensive light is required to illuminate objects that are farther away. Also, the field of view plays a significant role. A greater distance from the system requires that the fields of view of all the cameras overlap by a sufficient amount so that the multi-view processing can make sense.

### 5.5.2 Data set created by the author

For the final evaluation of all the proposed modifications to the depth map estimation algorithm the author managed to create his own test data set that contains images captured by the mixed video plus depth multi-camera acquisition system. The author also managed to create a data set that contains images captured by multiple acquisition systems. Each system exhibits a different number and layout of video and depth cameras.

Despite all efforts related to synchronisation between video and depth cameras, the operation principle of ToF cameras used ruled out the recording of video sequences that contain motion. The fact that a ToF camera performs background lighting suppression causes artifacts on the moving objects' edges. These artifacts manifest themselves as incorrect distance measurements and strongly influence video and ToF depth data fusion. As a result, the obtained depth maps are of poor quality as the edges are their most important features; therefore, the author chose to use only a single temporal frame to evaluate all of the proposed algorithms.

The test data captured by the 3T+2D multi-camera setup represents various arrangements of flat calibration boards and cardboard boxes. Both types of objects are made of paper, which makes their surface non-glossy. This, in turn, helps to achieve accurate ToF measurements as there are no reflections in the object's surfaces. The black and white rectangles on the calibration boards help to assess the proposed depth estimation performance for a tiled texture. It is also useful to determine the influence of object reflectivity on ToF measurement quality. A total of three object arrangements were recorded, both containing similar setups. The captured multi-view images were named "Boards_01", "Boards_02" and "Boards_03".

Data obtained using the 5T+2D multi-camera system represents a different situation. The scene consists of an office workplace with multiple monitors and other small objects placed on the desk. There are also large cardboards stacked on top of each other. The scene contains objects with various reflectivity and glossiness which strongly affects the ToF measurements. The image was named "Office".

The 6T+3D mixed multi-camera system was used to capture an image of a flower in a pot situated in front of a green curtain that served as a uniform background. The background has relatively low glossiness, which aids the ToF measurements, yet the flower's leaves and pedestal which it is standing on have high glossiness, which causes impairments in the distance measurements.

Table 5.5.2 summarises the data set created by the author. For details containing sample images of colour and depth data, please refer to annex C.

Table 5.5.2. – Summary of test the data set as created by the author.

| **Name** | **System type** | **Description** |
|---|---|---|
| Boards_01 | 3T+2D | Various setups of flat paper calibration boards and cardboard boxes. |
| Boards_02 | 3T+2D | |
| Boards_03 | 3T+2D | |
| Office | 5T+2D | Office workplace with many LCD monitors and various objects on the desk. |
| Flower | 6T+3D | A flower in a pot standing on a pedestal in front of a uniform non-glossy green background. |

### 5.5.3 Data from an existing data set

Despite the ability to capture mixed video plus depth multi-view sequences, such an acquisition system does not provide any ground-truth depth data which might be used for the proposed fusion algorithm evaluation without additional lighting equipment. Fortunately, there are publicly available multi-view images that provide ground-truth depth data for certain views. In order to evaluate the proposed depth estimation method, the author used a data set provided by Middelbury College which

provides ground-truth depth data [Scharstein_01] [Scharstein_02] [Scharstein_03][Hirschmuller_01]. Unfortunately, the sequences provided contain only still multi-view frames. This is due to the limitation of the ground-truth depth data acquisition technique.

The Middlebury stereo datasets contain a series of still multi-view images of various scene setups captured by a linear acquisition system. There are up to seven views provided for each scene. The image vertical resolution has a constant value of 1100 lines, while the horizontal resolution is different for each multi-view image and ranges from 1330 to 1390 pixels. The provided images are rectified and have lens distortion removed. Among all the views, two of them are provided with ground-truth disparity data that was obtained using the structured lighting technique [Scharstein_02]. The structured lighting-based method allowed to acquire the distance information using the same camera setup as that used to acquire the image data. Therefore, no data transformation between camera spaces was required. Ground-truth data images contain gaps, i.e. areas where the true disparity is unknown. The gaps are caused by the limitations of the structured light acquisition technique.

The author used the ground-truth depth information to simulate a multi-view system with ToF cameras. These cameras provided data with QCIF resolution which is roughly 1/8th of the Full-HD resolution. In order to mimic data provided by a ToF camera, the author decimated the ground-truth data to 1/8th of its original size. A ToF camera also provides a confidence map, which is an image where each pixel is associated with a value that defines the certainty of the measured distance. For the confidence emulation the author used the gap information from the ground-truth data. Gap areas have a confidence value of zero while other areas have a maximum value of one. The intrinsic matrix of each camera associated with the ground-truth data needed to be scaled by a factor of 1/8th in order to match the new resolution. The extrinsic parameters remained unchanged. The simulated mixed acquisition system consists of seven video cameras and two ToF cameras which are placed exactly at the same locations as their corresponding video cameras. Figure 5.5.4 shows the placement of the cameras in the simulated multi-view acquisition system.

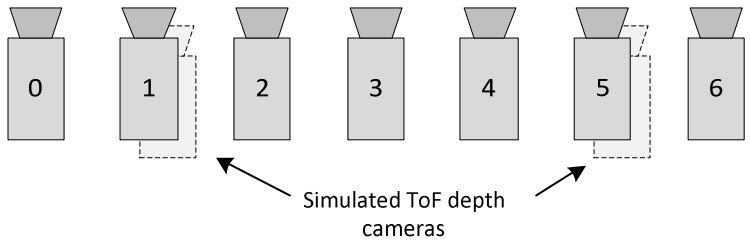



Figure 5.5.4 – Simulated multi-view plus depth acquisition system created using the Middlebury data set.

A total of 27 multi-view images from the 2005 and 2006 Middlebury data set were used for an evaluation of the algorithm proposed by the author. All of the images were cropped to 1232x1104 in

order to keep the resolution constant between all of the images. The camera intrinsic parameters were modified accordingly.

## 5.6 Evaluation of the proposed depth map estimation algorithm

### 5.6.1 Methodology

There are mainly two methods of comparison of depth map quality:

- Virtual view synthesis quality
- Direct depth map quality comparison with ground-truth data

The author of this dissertation used both of those methods to provide an evidence that depth maps, estimated by his proposed algorithm, are of higher quality than when using state-of-the-art depth estimation.

**The author did not take the liberty to perform subjective quality assessment of virtual views.** The reason for this is that there were no significant visual differences between virtual views synthesized using depth maps obtained by different methods. Moreover, the goal of video and depth fusion is to provide better quality depth maps in terms of scene representation which in some cases may not improve virtual view quality. For detailed explanation please refer to the next section of this chapter.

***Depth quality assessment through virtual view quality***

A widely recognised method of depth map quality assessment is virtual view synthesis quality. The idea is to perform a virtual view synthesis using two colour images and two depth maps. The virtual view position in 3D space corresponds to the position of a physical camera in a multi-camera system which was used to capture the images. The virtual view and the view from the physical camera, i.e. the reference view, are compared using an image similarity metric, usually the PSNR [Huynh-Thu_01] or SSIM [Wang_01] metric. In most cases the PSNR metric is used as it is better recognised than the SSIM and its properties are well known.

Depth map quality assessment via virtual view synthesis is an appropriate method for comparison of various automatic or semi-automatic depth estimation algorithms. It allows to compare algorithms that take advantage of stereo correspondence features. **Unfortunately, the method is not appropriate for depth maps estimated using data provided by depth cameras.**

The main reason is that the PSNR, SSIM and most of objective image quality measures exhibit poor correlation with subjective quality. Especially, when used for virtual view assessment, where image distortions/artefacts are of different kind than i.e. for image compression.

In their work Tikanmaki et al. [Tikanmaki_01] describe experiments regarding bitrate allocation for video plus depth compression. The decoded video plus depth data quality is then assessed through

virtual view quality using PSNR and VSSIM [Wang_01] (a version of SSIM adapted for video). Objective measures are compared with subjective quality assessment results. The conclusion is that the correlation between those two measures is poor.

Also Banitalebi-Dehkordi et al. present in their publication a comparative study of 3D video quality metrics [Dehkordi_01]. They also propose an introduction of a new one. In the paper, a several comparisons are presented between an objective quality measure versus subjective quality measure. For subjective tests the Mean Opinion Score (MOS) was used. The conclusion is that the correlation between an objective measure and MOS is not high enough to use it for virtual view quality assessment.

The second reason why the virtual view PSNR is not a good measure of depth map quality is that **a depth map estimated automatically does not necessarily represent physical distance.** It rather represents a globally optimal correspondence between two or more images in the sense defined by the similarity metric used for block/segment matching during its estimation. The problem is an analogous to motion estimation in video compression. The motion vector does not necessarily represent physical motion. Such a depth map may be used to generate a good quality virtual view as the view synthesis process is based on image pixel relocation according to the supplied depth map. Such a virtual view may still be similar to a reference view in the sense of a similarity metric such as PSNR, **but this does not mean that the depth map is an accurate representation of the 3D scene.**

Depth maps, estimated using data from depth cameras, may be very different from those estimated using only stereo correspondence features. A depth camera provides physical distance measurements which in some cases may not correspond to distances derived from stereo correspondence. This means that an automatically estimated depth map may provide a virtual view that is more similar to the reference view than the method that uses data from a depth camera. **This makes depth quality assessment via virtual view synthesis inadequate in such situations. Despite this fact the author presented a depth quality comparison using this method because of its wide recognition.**

***Depth quality assessment through comparison with ground-truth data***

Another set of depth quality assessment methods which does not use virtual view synthesis is based on **comparison with ground-truth data** [Scharstein_03]. A ground-truth depth map represents the physical distance to objects in the scene. The advantage of such methods is that they assess the quality of the depth map itself rather than the image which was created using that depth map. Unfortunately, obtaining ground-truth depth data for registered multi-view images requires an additional distance measurement technique which also has finite accuracy and may introduce errors to the ground-truth data.

The author was unable to provide ground-truth depth data for the test data he had created due to technical reasons. Instead, the author used the Middlebury data set that contains both colour and ground-truth depth data [Scharstein_01] [Scharstein_02] [Scharstein_03] [Hirschmuller_01] Because the Middlebury data set does not contain ToF depth camera data, the author used the ground-truth data provided to simulate the ToF cameras. For additional details, please refer to chapter 5.5.3.

The author has used the following images from the Middlebury data set: Aloe, Art, Baby1, Baby2, Baby3, Books, Bowling1, Bowling2, Cloth1, Cloth2, Cloth3, Cloth4, Dolls, Flowerpots, Lampshade1, Lampshade2, Laundry, Midd1, Midd2, Moebius, Monopoly, Plastic, Reindeer, Rocks1, Rocks2, Wood1 and Wood2.

The author used both virtual view synthesis and ground-truth-based depth quality assessment methods to compare depth maps created by his modified depth estimation algorithm and the original un-modified one.

For quality assessment via virtual view synthesis, two colour images and two depth maps were used. The virtual views were synthesised using the state-of-the-art view synthesis algorithm [Mori_01]. The algorithm is maintained by the MPEG group and is implemented in the View Synthesis Reference Software (VSRS) [VSRS]. Each output virtual view is compared with the reference using the PSNR metric. The PSNR is computed for luminance and chrominance separately.

Quality assessment for depth maps that have their corresponding ground-truth data was done using a selected method from the literature [Scharstein_03]. The percentage metrics were used as they provide the clearest representation of the results. Table 5.6.1 summarises the metrics that were used.

Table 5.6.1. – Metrics used for depth map evaluation versus ground-truth data.

| **Metric name** | **Symbol** | **Description** |
|---|---|---|
| Bad pixels - all | $B$ | percentage of bad pixels |
| Bad pixels - nonocc | $B_{\bar{o}}$ | percentage of bad pixels in non-occluded regions |
| Bad pixels - occ | $B_o$ | percentage of bad pixels in occluded regions |
| Bad pixels - textured | $B_T$ | percentage of bad pixels in textured regions |
| Bad pixels - textureless | $B_{\bar{T}}$ | percentage of bad pixels in textureless regions |
| Bad pixels - discontinous | $B_D$ | percentage of bad pixels near depth discontinuities |

The term “Bad pixels” corresponds to pixels where the disparity differs from the ground truth more than a certain threshold. All parameters, including the threshold, that are required to compute these metrics were set as specified in the original publication. Because depth maps are estimated using three views instead of two, no occlusion-related issues exist; therefore, metrics $B_{\acute{o}}$ and $B_o$ are meaningless.

### 5.6.2 Experiment conditions

Each test multi-view image was rectified using transformations estimated by algorithms as proposed by the author which are described in chapter 2.5. Depth camera data was processed by methods described in chapter 3.0. Table 5.6.2 summarises the various important parameters used in the algorithms proposed by the author.

Table 5.6.2. – Important parameters related to video and depth data rectification.

| **Parameter description** | **Value** |
|---|---|
| ToF infrared noise variance vs. intensity approximation polynomial coefficients (see table 3.4.1 in chapter 3.4.1) | $a_0$=8.04e1, $a_1$=2.16e-2 |
| ToF distance discontinuity threshold (see chapter 5.2 and equation 5.3.5) | 10.0 cm |

A set of depth maps was created for each test image. Each set contains depth maps computed for two chosen cameras in order to be able to perform the virtual view synthesis. For the 3T+2D camera system the depth could be estimated only for one camera and the synthesis was performed using a single view only. Table 5.6.3 summarises the video and depth camera indices used for depth map estimation.

Table 5.6.3. – Video and depth cameras used for depth map estimation.

| **System type** | **Left depth** | **Right depth** |
|---|---|---|
| 3T+2D | Depth for view: 1<br>Video cameras: 0,1, 2<br>Depth cameras: 0,1 | N/A |
| 5T+2D | Depth for view: 3<br>Video cameras: 2,3, 4<br>Depth cameras: 0,1 | Depth for view: 2<br>Video cameras: 1,2, 3<br>Depth cameras: 0,1 |
| 6T+3D | Depth for view: 4<br>Video cameras: 3,4, 5<br>Depth cameras: 1,2 | Depth for view: 1<br>Video cameras: 0,1, 2<br>Depth cameras: 0,1 |
| Middlebury data set (closest depth cameras) | Depth for view: 1<br>Video cameras: 0,1, 2<br>Depth cameras: 0 | Depth for view: 5<br>Video cameras: 4,5, 6<br>Depth cameras: 1 |
| Middlebury data set (farthest depth cameras) | Depth for view: 1<br>Video cameras: 0,1, 2<br>Depth cameras: 1 | Depth for view: 5<br>Video cameras: 4,5, 6<br>Depth cameras: 0 |

In each setup the depth cameras closest to the target camera were chosen in order to minimise possible distortions caused by the depth data transformation.

For the Middlebury data set, the simulated ToF camera positions correspond directly to the positions of video cameras which the depth is estimated for. The use of such data would reduce the problem of depth and image fusion to depth map up-scaling in which the transformation between camera spaces is neglected. In order not to omit issues related to data transformation, the author performed the experiment using ground-truth data from view 1 as simulated ToF data for view 5 and vice versa. This way the depth data needs to be transformed and the transformation between camera spaces is not omitted.

A number of depth maps were computed for each image set. The estimation was performed using an original algorithm from the DERS software as well as with the author's various modifications turned on and off. Table 5.6.4 shows the most important parameters supplied to the DERS software which remain constant for all of the experiment cases.

Table 5.6.4. – Depth map estimation parameters used for the DERS software.

| **Parameter** | **Value** |
|---|---|
| Precision | Half-pixel |
| Matching method | Block matching |
| Matching block size | 3x3 pixels |
| Horizontal search range (disparity range) | Variable, camera placement and scene setup dependent |
| Vertical search range | ±1 pixels |
| Smoothing coefficient | 1.0 |
| Sub-pixel interpolation filter | MPEG-4 AVC 6-tap |

The whole experiment can be divided into four cases which are summarised in Table 5.6.5. The different cases correspond to the cross-cost model modification turned on and off and the fitting cost weighting method (Formulas 5.4.9 and 5.4.10). For each case the same set of fitting cost model parameters was used in order to make the results comparable between one another.

Table 5.6.5. – Summary of experiment cases.

| **Test case** | **Parameters** |
|---|---|
| Case 1 | • Original, unmodified cross cost function<br>• Fitting cost weighting according to formula 5.4.9 |
| Case 2 | • Original, unmodified cross cost function<br>• Fitting cost weighting according to formula 5.4.10 |
| Case 3 | • Modified cross-cost function that takes colour image edges into account<br>• Fitting cost weighting according to formula 5.4.9 |
| Case 4 | • Modified cross-cost function that takes colour image edges into account<br>• Fitting cost weighting according to formula 5.4.10 |

Finally, for each multi-view image and each depth map pair a virtual view was synthesised using original colour images and estimated depth maps.

For the 3T+2D camera system the virtual view synthesis was performed using only a single view and a depth map from the centre camera. There is no other camera with available depth map to be used as a second data source for view synthesis. A similar situation pertains to the 6T+3D data set, where two virtual views for cameras 2 and 3 were synthesised using the same source left and right view and depth map.

Virtual view synthesis with only one source camera will lead to disocclusions in the virtual view image. The author proposed to synthesise two virtual views that would correspond to the positions of cameras 0 and 2. Both virtual views will have disocclusions in different areas. The PSNR metric was computed for each view only for the non-disoccluded areas. Both PSNR values were averaged.

Table 5.6.6 summarises the virtual view synthesis camera indices that were used. For the synthesis cases where there are more than one variant, the PSNRs of the virtual views were averaged.

Table 5.6.6. – View indices used for virtual view synthesis.

| **System type** | **Left source view** | **Target view** | **Right source view** |
|---|---|---|---|
| 3T+2D (variant 1) | 1 | 0 | 1 |
| 3T+2D (variant 2) | 1 | 2 | 1 |
| 5T+2D | 3 | 2 | 1 |
| 6T+3D (variant 1) | 4 | 2 | 1 |
| 6T+3D (variant 2) | 4 | 3 | 1 |
| Middlebury data set | 1 | 3 | 5 |

For the test data from the Middlebury data set which contain ground-truth depth data, depth quality assessment using ground-truth data was performed. The depth comparison metrics that were used are summarised in Table 5.6.1 of the previous chapter. All metrics were computed as indicated in the original publication [Scharstein_03]. Because the depth maps were estimated using three views which eliminated occlusion problems, all occlusion-related metrics are not meaningful and therefore are not shown among the other metrics.

### 5.6.3 Virtual view synthesis results

The following sub-chapter presents the PSNR values of the virtual views. The virtual views were generated using parameters according to Table 5.6.6. For data sets with more than one variant (multiple virtual views for each depth map case), the PSNR is an average of all of them.

***Results for data sets created by the author***

Tables 5.6.7 and 5.6.8 present the luminance PSNR values for virtual views of the data set created by the author for experiment cases 1 and 2. The original cross-cost model was used with no edge information incorporation. Table 5.6.9 shows the results obtained using cost weighting according to formula 5.4.9, while Table 5.6.10 shows the results when weighting according to formula 5.4.10 – experiment cases 3 and 4.

Table 5.6.7. – Virtual view PSNR for depth maps estimated using the proposed algorithm. Settings for Case 1.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Boards_01 | 26.15 | 24.54 | 24.60 | 24.67 | 24.44 | 24.47 | 24.48 | 24.48 | 24.54 | 24.59 | 24.47 | 24.51 | 24.55 |
| Boards_02 | 26.47 | 24.89 | 24.93 | 25.03 | 24.78 | 24.82 | 24.87 | 24.85 | 24.94 | 25.07 | 24.84 | 24.91 | 24.98 |
| Boards_03 | 26.42 | 25.33 | 25.22 | 25.45 | 25.39 | 25.34 | 25.29 | 25.36 | 25.37 | 25.28 | 25.34 | 25.33 | 25.32 |
| Boards_04 | 23.48 | 23.21 | 23.23 | 23.02 | 23.14 | 23.16 | 23.20 | 23.23 | 23.23 | 23.14 | 23.18 | 23.23 | 23.23 |
| Flower | 31.66 | 27.23 | 26.86 | 27.02 | 26.60 | 26.60 | 26.67 | 26.63 | 26.79 | 26.92 | 26.63 | 26.76 | 26.82 |
| Office | 32.34 | 30.21 | 30.37 | 30.82 | 30.14 | 30.23 | 30.32 | 30.26 | 30.47 | 30.57 | 30.24 | 30.44 | 30.48 |
| **Average** | **27.75** | **25.90** | **25.87** | **26.00** | **25.75** | **25.77** | **25.80** | **25.80** | **25.89** | **25.93** | **25.78** | **25.86** | **25.90** |

Table 5.6.8. – Virtual view PSNR for depth maps estimated using the proposed algorithm. Settings for Case 2.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Boards_01 | 26.15 | 26.06 | 26.09 | 26.11 | 26.06 | 26.14 | 26.11 | 26.08 | 26.13 | 26.16 | 26.09 | 26.12 | 26.15 |
| Boards_02 | 26.47 | 26.38 | 26.38 | 26.40 | 26.39 | 26.35 | 26.32 | 26.38 | 26.32 | 26.47 | 26.39 | 26.32 | 26.43 |
| Boards_03 | 26.42 | 26.05 | 25.98 | 26.11 | 26.04 | 26.13 | 26.16 | 26.12 | 26.13 | 26.15 | 26.11 | 26.12 | 26.15 |
| Boards_04 | 23.48 | 23.32 | 23.37 | 23.22 | 23.34 | 23.25 | 23.24 | 23.32 | 23.32 | 23.32 | 23.32 | 23.27 | 23.36 |
| Flower | 31.66 | 31.31 | 31.17 | 31.21 | 31.17 | 31.18 | 31.14 | 31.17 | 31.16 | 31.18 | 31.17 | 31.14 | 31.15 |
| Office | 32.34 | 32.68 | 32.66 | 32.49 | 32.67 | 32.62 | 32.63 | 32.62 | 32.58 | 32.56 | 32.61 | 32.61 | 32.59 |
| **Average** | **27.75** | **27.63** | **27.61** | **27.59** | **27.61** | **27.61** | **27.60** | **27.62** | **27.60** | **27.64** | **27.61** | **27.59** | **27.64** |

Table 5.6.9. – Virtual view PSNR for depth maps estimated using the proposed algorithm. Settings for Case 3.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Boards_01 | 25.99 | 24.54 | 24.62 | 24.74 | 24.49 | 24.55 | 24.58 | 24.58 | 24.65 | 24.73 | 24.56 | 24.62 | 24.70 |
| Boards_02 | 25.64 | 24.66 | 24.73 | 24.79 | 24.63 | 24.64 | 24.67 | 24.69 | 24.76 | 24.82 | 24.68 | 24.74 | 24.78 |
| Boards_03 | 26.23 | 25.25 | 25.12 | 25.23 | 25.17 | 25.11 | 25.11 | 25.10 | 25.14 | 25.15 | 25.13 | 25.14 | 25.15 |
| Boards_04 | 23.23 | 22.88 | 22.93 | 22.80 | 22.81 | 22.84 | 22.89 | 22.89 | 22.96 | 22.94 | 22.88 | 22.95 | 22.96 |
| Flower | 32.22 | 27.43 | 26.93 | 27.18 | 26.74 | 26.67 | 26.78 | 26.68 | 26.96 | 27.16 | 26.67 | 26.85 | 27.00 |
| Office | 32.87 | 30.61 | 30.81 | 31.21 | 30.52 | 30.66 | 30.77 | 30.65 | 30.93 | 31.15 | 30.63 | 30.81 | 30.97 |
| **Average** | **27.70** | **25.90** | **25.86** | **25.99** | **25.73** | **25.75** | **25.80** | **25.76** | **25.90** | **25.99** | **25.76** | **25.85** | **25.93** |

Table 5.6.10. – Virtual view PSNR for depth maps estimated using the proposed algorithm. Settings for Case 4.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Boards_01 | 25.99 | 25.98 | 26.11 | 26.11 | 26.05 | 26.10 | 26.08 | 26.10 | 26.11 | 26.11 | 26.10 | 26.11 | 26.09 |
| Boards_02 | 25.64 | 25.50 | 25.56 | 25.59 | 25.53 | 25.58 | 25.55 | 25.57 | 25.57 | 25.54 | 25.58 | 25.58 | 25.56 |
| Boards_03 | 26.23 | 25.67 | 25.72 | 25.88 | 25.70 | 25.93 | 25.88 | 25.83 | 25.90 | 25.90 | 25.82 | 25.93 | 25.87 |
| Boards_04 | 23.23 | 23.18 | 23.20 | 23.09 | 23.23 | 23.10 | 23.11 | 23.14 | 23.11 | 23.09 | 23.17 | 23.11 | 23.09 |
| Flower | 32.22 | 32.14 | 32.11 | 31.93 | 32.09 | 32.12 | 31.90 | 32.05 | 31.90 | 31.88 | 32.07 | 31.89 | 31.85 |
| Office | 32.87 | 33.21 | 33.07 | 32.92 | 33.18 | 33.11 | 33.09 | 33.12 | 32.95 | 32.91 | 33.13 | 32.97 | 32.95 |
| **Average** | **27.70** | **27.61** | **27.63** | **27.59** | **27.63** | **27.66** | **27.60** | **27.64** | **27.59** | **27.57** | **27.65** | **27.60** | **27.57** |

**Unfortunately, none of the experiments yielded virtual view PSNR value improvement.** There are many reasons for the lack of depth map quality increase measured via the PSNR.

First, **the PSNR does not correspond to the subjective quality of an image**, especially when used to measure virtual view quality [Tikanmaki_01] [Dehkordi_01]. The view synthesis process performs image pixel relocation which, depending on the image content, may not introduce significant subjective image distortions. On the other hand, **the PSNR metric is very sensitive to per-pixel colour differences, which cause its value to drop even when not visible to the viewer.**

The ToF depth camera used here exhibited much lower resolution than the target depth map which, when combined with the need for data transformation from one camera space to another, yields inaccurate scene representation. The author also managed to observe severe distortions in the distance measurements caused by light reflections from glossy and even non-glossy surfaces. The multi-path light wave propagation falsifies distance measurements, which distorts the scene model even more.

***Results for the Middlebury data set***

The Middlebury data set allowed the author to overcome some of the limitations of the ToF technology in order to prove the concept of the proposed depth estimation algorithm modifications. Instead of real ToF cameras, the ground-truth data was used to simulate their presence. In order not to simplify the depth and video data fusion to depth up-scaling, the simulated ToF cameras used were located far away from the camera that the depth maps were estimated for (see Table 5.6.3).

Table 5.6.11 shows the average PSNRs for virtual views computed among all of the Middlebury data set (27 test multi-view images). For detailed results of each individual image, please refer to annex D.

Table 5.6.11. – Average PSNR values for virtual views.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Case 1 | 36.51 | 36.04 | 36.25 | 36.24 | 34.95 | 35.22 | 35.41 | 36.02 | 36.26 | 36.34 | 35.54 | 36.24 | 36.32 |
| Case 2 | 36.51 | 36.55 | 36.53 | 36.52 | 36.55 | 36.53 | 36.53 | 36.52 | 36.53 | 36.50 | 36.53 | 36.52 | 36.51 |
| Case 3 | 36.58 | 36.06 | 36.24 | 36.27 | 34.95 | 35.20 | 35.41 | 36.02 | 36.25 | 36.34 | 35.54 | 36.23 | 36.30 |
| Case 4 | 36.58 | 36.60 | 36.59 | 36.56 | 36.60 | 36.57 | 36.57 | 36.57 | 36.57 | 36.55 | 36.57 | 36.57 | 36.56 |

With the distance measurement error eliminated (ground-truth data used) **there are experiment cases that yield PSNR gains** with respect to automatically estimated depth maps. Unfortunately, the low-resolution-simulated ToF camera data still does not provide sufficiently precise additional information to significantly improve the estimated depth maps' quality. The low-resolution difference and the need to transform depth data from one camera space to another is still a major issue that impairs the depth estimation process.

### 5.6.4 Comparison with ground-truth depth maps

The following four Tables, 5.6.12, 5.6.13, 5.6.14 and 5.6.15, show the comparison results of estimated depth maps with ground-truth depth maps provided for the data set. All metrics were computed as indicated in the original publication [Scharstein_03]. The tables summarise the averages taken over the whole test data set. The results for individual images are shown in annex D.

Table 5.6.12. – Percentage of bad pixels in the depth map for various pixel classes according to Table 5.6.1. Results for Case 1.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| $B$ | 9.63 | 4.47 | 6.09 | 8.95 | 4.58 | 4.46 | 4.33 | 4.36 | 4.58 | 6.00 | 4.42 | 4.27 | 4.63 |
| $B_{\bar{T}}$ | 11.19 | 4.37 | 6.58 | 10.19 | 4.36 | 4.31 | 4.23 | 4.27 | 4.79 | 6.60 | 4.30 | 4.28 | 4.88 |
| $B_T$ | 4.57 | 5.31 | 5.16 | 4.93 | 5.74 | 5.44 | 5.19 | 5.21 | 4.74 | 4.61 | 5.32 | 4.86 | 4.68 |
| $B_D$ | 21.61 | 18.84 | 20.27 | 21.80 | 19.66 | 19.24 | 18.76 | 18.68 | 18.78 | 19.47 | 18.95 | 18.13 | 18.76 |

Table 5.6.13. – Percentage of bad pixels in the depth map for various pixel classes according to Table 5.6.1. Results for Case 2.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| $B$ | 9.63 | 9.63 | 9.68 | 9.74 | 9.48 | 9.60 | 9.75 | 9.68 | 9.64 | 9.89 | 9.54 | 9.62 | 9.87 |
| $B_{\bar{T}}$ | 11.19 | 11.19 | 11.24 | 11.33 | 10.98 | 11.18 | 11.33 | 11.26 | 11.23 | 11.42 | 11.10 | 11.20 | 11.41 |
| $B_T$ | 4.57 | 4.39 | 4.40 | 4.44 | 4.38 | 4.43 | 4.46 | 4.41 | 4.46 | 4.68 | 4.41 | 4.45 | 4.66 |
| $B_D$ | 21.61 | 20.54 | 20.67 | 20.80 | 20.70 | 20.74 | 20.76 | 20.72 | 20.85 | 20.90 | 20.66 | 20.77 | 20.91 |

Table 5.6.14. – Percentage of bad pixels in the depth map for various pixel classes according to Table 5.6.1. Results for Case 3.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| $B$ | 8.91 | 4.35 | 6.02 | 8.63 | 4.43 | 4.31 | 4.19 | 4.25 | 4.49 | 5.90 | 4.28 | 4.17 | 4.55 |
| $B_{\bar{T}}$ | 10.37 | 4.27 | 6.55 | 9.82 | 4.23 | 4.18 | 4.10 | 4.18 | 4.72 | 6.54 | 4.18 | 4.21 | 4.84 |
| $B_T$ | 4.08 | 5.08 | 4.95 | 4.67 | 5.48 | 5.18 | 4.93 | 4.96 | 4.50 | 4.38 | 5.08 | 4.63 | 4.44 |
| $B_D$ | 18.82 | 18.14 | 19.74 | 20.83 | 18.88 | 18.43 | 17.95 | 18.02 | 18.14 | 18.95 | 18.20 | 17.56 | 18.21 |

Table 5.6.15. – Percentage of bad pixels in the depth map for various pixel classes according to Table 5.6.1. Results for Case 4.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| $B$ | 8.91 | 8.89 | 8.95 | 9.01 | 8.92 | 9.02 | 9.05 | 8.99 | 9.07 | 9.28 | 8.98 | 9.07 | 9.10 |
| $B_{\bar{T}}$ | 10.37 | 10.35 | 10.43 | 10.47 | 10.38 | 10.50 | 10.53 | 10.47 | 10.54 | 10.75 | 10.46 | 10.55 | 10.58 |
| $B_T$ | 4.08 | 3.96 | 4.01 | 4.05 | 3.98 | 4.03 | 4.04 | 4.01 | 4.06 | 4.16 | 4.01 | 4.05 | 4.08 |
| $B_D$ | 18.82 | 18.55 | 18.74 | 18.83 | 18.71 | 18.80 | 18.86 | 18.78 | 18.90 | 19.03 | 18.79 | 18.85 | 18.96 |

**The average results of the depth maps' comparison show an improvement when using the ToF data and stereo correspondence instead of stereo correspondence only.** As was expected, the strongest improvement can be observed in the textureless areas ($B_{\bar{T}}$) of the multi-view images. This is due to the fact that stereo correspondence features cannot be reliably estimated for textureless regions. On the other hand, in textured regions the improvement is minimal or even non-existent. This might be caused by the difference between the optimal disparity derived from the stereo correspondence features and the transformed depth data from the simulated ToF camera(s). Minimal improvement can also be observed for the depth discontinuity pixel class. The lack of improvement is caused by the need for up-scaling depth camera data which does not provide depth information with sufficient spatial resolution.

Nevertheless, the results show that the idea behind video and ToF depth data fusion is correct. A better quality depth map can be estimated, but sufficient quality ToF data is required in order to accurately represent the 3D model of the scene.

## 5.7 Conclusions

Despite his best efforts the author was unable to provide ground-truth depth data for video plus depth data sets created by him. These are 3T+2D, 5T+2D and 6T+3D. No other depth acquisition technique than stereo correspondence matching and time-of-flight was available to him at the time of research which took place in years 2011 – 2012. Therefore for those data sets only comparison using virtual view PSNR was possible.

**The ToF depth and video data fusion algorithm as proposed by the author allows to obtain more accurate depth maps.** Better accuracy can be achieved in terms of conformance to ground-truth depth data (tables 5.6.12 – 5.6.16). Unfortunately it cannot be achieved when comparing using virtual view PSNR (tables 5.6.7 – 5.6.11). It must be noted that the PSNR of a virtual view is not the best estimate of depth map accuracy in terms of 3D scene representation correctness.

**The depth accuracy improvement, in terms of reduction of bad pixel ratio, can be observed mostly in textureless areas** (as defined in [Scharstein_03]). In those areas there are no reliable stereo correspondence features to be used. The missing information is filled by data from depth camera(s) which provide actual distance measurements. There is no significant decrease of bad pixel ratio in other areas as additional depth camera data does not carry more information than stereo correspondence.

Among the four test cases, the greatest improvement can be seen in those in which the fitting cost is weighted according to the equation 5.4.9 instead of 5.4.10. Those are Case 1 and Case 3. The equation 5.4.9 describes a linear transition between stereo correspondence fitting cost and a cost model proposed by the author which performed best. For cases 2 and 4 the equation 5.4.10 is used which effectively define a method of changing shape of the fitting cost function. The second proposed method of fitting cost weighting turned out not to be as effective as the first one.

According to results summarised in tables 5.6.12 – 5.6.16, the fitting cost model used does not influence the final outcome. There is an exception for the step model (equation 5.4.7) – if the parameter $b$ is greater than few disparity units then the fitting cost becomes equal for too wide range of disparity values around the measured one. This leads to ambiguity and, in turn, to loss of estimated depth map accuracy. For other proposed cost models the only important feature is that they have a single minimum for the measured distance value. Parameters that change steepness of the curve influence the final depth map in marginal way.

Unfortunately, a real-world ToF camera (according to today's technology) provides inaccurate distance measurements due to multi-path light wave propagation. Each pixel of a ToF camera sensor registers reflected light that comes not only from a direct reflection (camera – object – camera) but also from multiple reflections (camera – object 1– object 2 – camera). The ToF technology would work reliably only for objects that do not disperse light but rather reflect it only in the direction that the light comes from. The ToF distance measurement errors cause object edge relocation when transforming depth data from the ToF camera to a video camera space. This, in turn, impairs the data fusion process as correct edge information is crucial for depth map and virtual view synthesis quality.

On the other hand, a depth map estimated using stereo correspondence features only will yield a better virtual view PSNR than a depth map estimated using stereo correspondence fused with ToF measurements. This is due to a fact that a depth estimation algorithm performs global optimization of a cost defined in terms of images similarity which is later being assessed using the PSNR metric. The problem is analogous to motion estimation used in video compression. A motion vector does not necessarily correspond to a physical object motion but rather defines a best block match. A similar problem exists for depth maps which can be treated as a horizontal motion field for a two-frame video sequence where the first frame was captured in one camera location and the second in the other camera location.

An inconsistency may arise when trying to fuse stereo correspondence information with information from a ToF camera measurement. The virtual view PSNR metric represents per-pixel similarity between the virtual and the reference image, which does not actually take depth map quality into account.

Despite all the problems and difficulties that the author of this dissertation had to overcome, the thesis T1: "Relatively simple design of a system for hybrid depth acquisition with the use of time-of-flight depth cameras and video cameras allows to obtain higher quality depth maps as compared to systems based on either time-of-flight cameras or video analysis only." has been proven.

# 6. Depth map refinement via inter-view consistency improvement

## 6.1 Introduction

Modern multi-view compression algorithms that are used for multi-view depth compression rely on inter-view consistency of the depth maps. Inter-view consistency defines the amount of consensus of a 3D scene representation of multiple depth maps. The consistency causes a large redundancy in information carried by the depth maps; therefore, it is much desired. Unfortunately, depth maps are estimated independently, which causes them to be non-consistent.

In this chapter the author presents his algorithm that allows to improve the inter-view consistency of depth maps without the need for their re-estimation, which would require significant computational effort.

## 6.2 Multi-view video compression

Multi-view video compression takes advantage of inter-view similarities in a multi-view sequence. This is done in the same way as for temporal similarities in a single-view video compression. In a multi-view video codec, the inter-prediction mechanism is extended so it can use not only temporal dependencies but also spatial dependencies.

Figure 6.2.1 illustrates an example inter-prediction scheme for a three-view multi-view video sequence. This is not the only possible scheme, as different schemes that incorporate, for example, joint temporal and spatial prediction are also allowed.

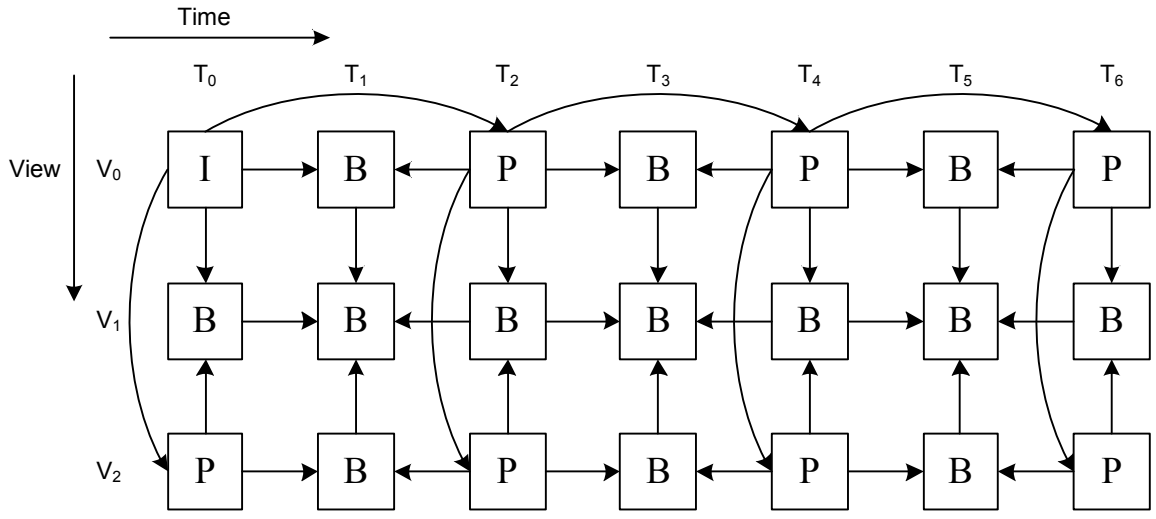



Figure 6.2.1 – Example of an inter-prediction scheme for a multi-view video codec.

The multi-view compression mechanism was introduced first in the H.264/AVC codec [AVC_01] as an extension to the original standard. Annex H extends the H.264/AVC standard by

allowing compression of multiple views. The H.264/AVC Annex H is known as the MVC video codec [AVC_02]. It should be noted that the MVC codec is not meant for the coding of depth maps; it treats depth maps as monochromatic images and neglects their unique features.

The Annex I of the H.264/AVC standard introduces a new depth-aware extension to the MVC known as the MVC+D [AVC_03]. The MVC+D codec uses different compression tools for colour data and depth data. This allows to exploit the differences in their characteristics, which was not possible in the MVC codec.

The Annex J extends the MVC+D even further by allowing it to take advantage of the full 3D scene representation provided by a video plus depth multi-view sequence [AVC_04]. New compression tools such as View Synthesis Prediction (VSP) and Depth-based Motion Vector Prediction (D-MVP) use depth maps for derivation of prediction data from other views. The codec is named 3D-AVC.

A similar multi-view extension as for the H.264/AVC codec exists for the most recent compression technology, i.e. the HEVC codec [HEVC_01]. The extension, known as MV-HEVC, is basically the same to the HEVC codec as the MVC extension is to the AVC codec. The MV-HEVC codec extends the inter-prediction by allowing it to derive data from other spatial views [HEVC_02].

Depth maps can be compressed with the same or lower spatial resolution as colour images. Research shows that decreasing depth resolution does not significantly impair virtual view synthesis [Klim_01][Klim_02].

MVC compression efficiency is highly related to image similarities across views. The more similar neighbouring views are, the more effective inter-view prediction can be. Multi-view sequence video frames are usually very similar across views, as the camera system captures images of the same scene. However, the same cannot be said for depth maps as these are estimated independently for each view, they may not contain such inter-view similarities.

## 6.3 Depth map inter-view consistency

Depth maps, estimated by an automatic depth estimation algorithm, usually exhibit low inter-view consistency. Figure 6.3.1 shows a typical depth inconsistency issue for the multi-view sequence "Newspaper" [CFP]. The effect can be visible in the central section of each depth map image.

Figure 6.3.1 – Example of an inter-view depth map inconsistency in the sequence "Newspaper" [CFP]. The cameras are 02, 04 and 06, beginning from the left-hand image.

Some parts of depth maps, especially the background behind the person in the centre, exhibit different depth values for each view. The reason why the depth maps may exhibit low inter-view consistency is the fact that they are estimated using different images from independent video cameras.

For a linear multi-camera system, the DERS depth map estimation algorithm [DERS] requires three neighbouring views in order to estimate a depth map for a single camera. The use of three views is necessary to avoid disocclusion effects. This means that depth estimation for the *i*-th camera uses completely independent stereo correspondence than estimation for cameras *i*-(1+*n*) and *i*+(1+*n*), where *n* is any natural number (excluding 0). This situation is shown in Figure 6.3.2:

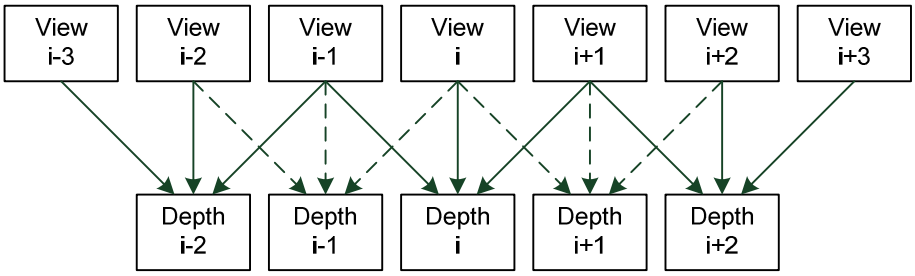



Figure 6.3.2 – Three-view depth map estimation scheme for a linear multi-camera system.

The independent estimation of depth maps for individual views causes them to be inconsistent. Different disparities may be chosen by the stereo matching algorithm for the same areas of different image pairs due to their unique local features or image noise. For the areas of the images where there are no reliable features present, a global optimisation algorithm is used which determines the most probable disparity value based on then neighbourhood of the area. Depending on the structure of each image, the assumed disparities may also be different as the neighbourhood is not the same.

The literature does not provide much information on inter-view depth map inconsistency. Some publications are related to improved depth map estimation algorithms aimed at reducing the inconsistency.

Lee et al. [Lee_01] proposed to estimate depth maps for three neighbouring views simultaneously by using the graph cuts algorithm [Boykov_01]. A modified cost function, proposed by the authors, takes advantage of intermediate disparity differences between views. The function adds additional penalty to the fitting cost term that is related to the inter-view disparity difference.

A similar approach was proposed by the researchers of the Fraunhoffer HHI Institute [Mueller_01]. The authors suggested an iterative algorithm for inter-view consistent depth map estimation. Initial depth maps are estimated for stereo pairs using conventional means. During each step the inter-view consistency is verified within and across stereo pairs. Consistent areas of depth maps are fused together while other areas are interpolated. There are no additional details on inter-view consistency-related issues in the publication.

As was mentioned earlier in this chapter, the inter-view inconsistency affects the performance of multi-view video compression algorithms; therefore, by improving the consistency the compression ratio of multi-view sequences can also be improved without any quality loss. The author proposes a novel algorithm in order to solve the problem of low inter-view depth map consistency. The algorithm is based on depth map postprocessing which allows to exchange depth information between depth maps. The information exchange allows to reach depth consensus in depth map areas where the inter-view consistency is low. The algorithm is iterative in nature, as during each step the depth information is changed according to a change computed using that information from the previous step. Special care is taken in order not to introduce distortions to the processed depth maps that may emerge during virtual view synthesis. At each step of the algorithm the possible influence of the introduced change on the synthesised image is evaluated and the depth map is corrected accordingly.

## 6.4 Inter-view depth inconsistency measure

The author defines an inter-view inconsistency measure that will be used in the proposed depth processing algorithm. The general idea behind the concept of the proposed measure is that it should reflect the differences in the 3D scene representations of each single depth map.

Each depth map of a multi-view sequence provides a partial 3D scene representation. Moreover, each depth map is defined in a different coordinate system related to each camera. In order to be able to compare these representations, each depth map needs to be transformed to a common coordinate system.

The proposed inter-view inconsistency measure is defined using an average variance of depths computed across all available views. Assuming that the input multi-view sequence has $N$ views with depth maps $D_1...D_n$, the author proposes to define a partial inconsistency measure $V_i$ associated with each $i$-th depth map. In order to compute $V_i$, all depth maps must be transformed into the common coordinate system of the $i$-th view. Let the transformed depth maps be denoted as $D_1'...D_n'$. The transformation is done using the DIBR algorithm [DIBR]. Then for each pixel a depth variance is

computed across all transformed depth pixels with the same spatial coordinates within the image frame. The value of $V_i$ is computed by averaging all the pixel variances according to equation 6.4.1.

$$V_i = \frac{1}{w \cdot h} \sum_{y=0}^{w} \sum_{x=0}^{h} var(D_1'(x,y), D_2'(x,y), \dots, D_N'(x,y)) , \tag{6.4.1}$$

where $w$ and $h$ are depth map dimensions and $var()$ determines variance function.

The higher the value of $V_i$, the more inconsistent the depth maps are. The process is identical for each view and the final inter-view inconsistency measure $V$ is computed by taking the arithmetic mean of all $V_i$ values from all available views according to the equation 6.4.2.

$$V = \frac{1}{N} \sum_{i=1}^{N} V_i \tag{6.4.2}$$

The proposed measure allows to reliably determine the inter-view inconsistency of depth maps in a multi-view sequence. The higher the variance, the higher the spread of depth values across all depth maps. It must be noted that the proposed measure reflects average discrepancies in distance representation; therefore, similar values of $V$ can be found for a sequence with a small area with large inconsistency as well as for a sequence with a large area with small inconsistency.

## 6.5 Proposed depth inter-view consistency improvement algorithm

The author proposes an innovative algorithm aimed at improving inter-view depth map consistency [Kurc_01]. The algorithm modifies depth maps by using information from all available views. Each temporal frame of the multi-view sequence is processed independently. The proposed algorithm does not provide any temporal consistency enhancement techniques as it is not the goal of its operation.

The algorithm is iterative. All depth maps are processed simultaneously in a single iteration. During each iteration the following steps are taken: transformation of depth maps into a common coordinate space, inter-view information exchange between all transformed depth maps, and depth value restoration. After each iteration, the inter-view consistency of the processed depth maps is assessed using the measure proposed by the author. A decision is taken whether to continue the processing towards further inter-view consistency improvement or whether to stop because no further improvement is possible.

A general block diagram of the proposed algorithm is shown in Figure 6.5.1. The diagram represents an information flow path for a single view. The scheme is identical for all views in the sequence.

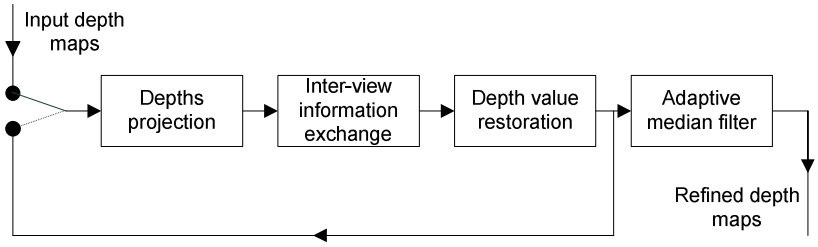



Figure 6.5.1 – General block diagram of information flow for a single view.

### 6.5.1. Iterative processing

The detailed block diagram, as shown in Figure 6.5.2, illustrates the data flow for a single iteration. It is assumed that the input multi-view sequence contains N views, and each *i*-th view is associated with a depth map $D_i$.

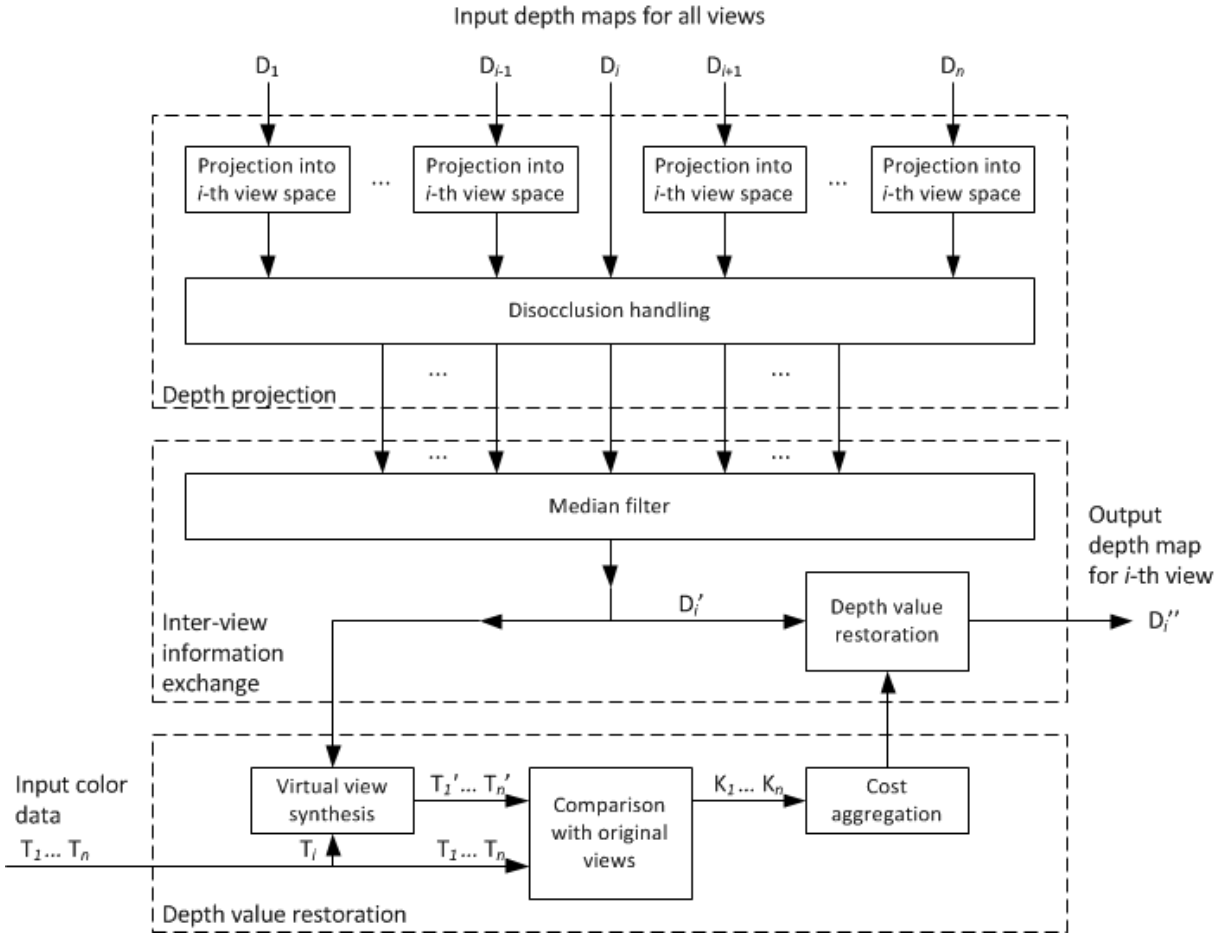



Figure 6.5.2 – Detailed block diagram of information flow in each iteration for *i-th* view.

A depth map for *i*-th view represents the distance defined in its camera space. In order for all depth maps from all cameras to be processed together, they must be transformed into a single camera space. Therefore the first step of each iteration is a transformation of each depth map to the *i*-th camera space. This is done via per pixel projection according to the DIBR algorithm [DIBR]. As a result, for

each *i*-th view of the input sequence there is a set of virtual depth maps defined in the coordinate system of the *i*-th view camera.

Virtual depth maps contain disocclusions, i.e. areas where no depth data is available. The author proposes to deal with them by selecting a minimum depth value for each disoccluded pixel among all the pixels of the same spatial coordinates but originating from other virtual depth maps. The reason why a minimum depth value is chosen is because a disocclusion occurs where a part of the background covered by a foreground object is revealed. The background is the farthest area in the scene, hence the choice is to select the minimum.

During the second stage of each iteration, information between the virtual depth maps is exchanged. The author proposes to use a weighted median filter [Bovik_01] which operates in the view domain independently for each pixel, for which the weight in the filter is proportional to the difference between the view index of that pixel and the currently processed view index *i*. The farther a pixel is located from the filtered pixel in the view domain, the fewer times its depth value is taken for median value computation.

As a result of the filtration, a new depth map $D_i'$ associated with the *i*-th view is created. The new depth map provides the scene representation which incorporates depth information from all available views. It is defined in the coordinate system of the *i*-th camera.

The third stage of each iteration is depth quality assessment. Differences introduced to depth map $D_i'$ may result in virtual view quality loss when that depth map is used for virtual view synthesis. To reduce possible distortions introduced by median filtration, depth map $D_i'$ is modified in the following manner. Depth map $D_i'$ and the corresponding input image $T_i$ are used to create a new set of virtual images by projecting input image$T_i$ onto every *i*-th view position. The virtual images are then compared to the original input images$T_1...T_n$ and a similarity measure $K_i$ for each pixel of the virtual images is computed. Disoccluded areas are not taken into account. Finally, an overall image similarity measure $K$ is computed by taking for each pixel the worst value among all the similarity measures $K_1...K_n$.

The author chose to use the SSIM (Structural Similarity) measure for virtual image quality assessment [Wang_01] instead of the PSNR [Huynh-Thu_01], which is widely used for image distortion analysis. The research shows that SSIM is more correlated with subjective image quality than PSNR [Wang_02]. The worse the similarity, the lower the value of SSIM, hence the minimum value among$K_1...K_n$ is taken.

For depth map $D_i$' areas where the value of $K$ falls below a given threshold, the depth value is restored to the value from the previous iteration. As a result a new depth map $D_i''$ is created. Trivial depth value restoration causes spatial discontinuities in the depth map; therefore, instead of taking the

previous depth value directly an arithmetic mean of values from the current and previous iteration is taken. This ensures that the output depth map is smooth.

The depth value restoration is done according to the following equation:

$$D_i'' = \begin{cases} \frac{D_i' + \widetilde{D}_i''}{2} & if\ K < th \\ D_i' & otherwise \end{cases}, \tag{6.5.1}$$

where $\widetilde{D}_i''$ is a value of $D_i$ from the previous iteration and *th* is a predefined threshold.

All three processing stages are repeated until no further improvement in inter-view consistency is observed. At the end of each iteration the inter-view inconsistency measure $V_i$ is computed and compared to the value from the previous iteration. If the difference is less than a specified threshold (meaning that no further improvement can be done), the processing is terminated.

### 6.5.2 Postprocessing

The processed depth map $D_i$" may still contain some local spatial discontinuities, mostly single-pixel-sized. These originate as artefacts of the DIBR virtual view synthesis process. In order to remove them, the author proposes to use an adaptive median filter.

The goal is to remove single, isolated pixels that are significantly different from their neighbourhood. The proposed filter is adaptive in such a way that a decision is taken for each pixel as to whether its depth is a result of a distortion or not. The decision is taken according to the following formula 6.5.2:

$$s(x,y) = \begin{cases} 1 & if\ \left|Average\big(\Omega(x,y)\big) - d(x,y)\right| > th \\ 0 & otherwise \end{cases}, \tag{6.5.2}$$

where $s(x,y)$ is a binary indicator whether a pixel requires filtering or not, *th* is a given threshold, $d(x,y)$ is the depth value at spatial coordinates $x,y$ and $\Omega(x,y)$ is the 3-by-3 pixel neighbourhood at spatial position $x,y$. equation 6.5.3 defines a filtered depth value of a pixel.

$$d_f(x,y) = \begin{cases} Median\big(\Omega(x,y)\big) & if\ s(x,y) = 1 \\ d(x,y) & otherwise \end{cases}. \tag{6.5.3}$$

The $d_f(x,y)$ is the filtered depth at spatial coordinates $x,y$.

Care must be taken when choosing the threshold parameter *th*, as choosing too low a threshold will cause correct pixels to be filtered while choosing too high a threshold will result in skipping the filtration of distorted pixels. The filter operates in the local spatial neighbourhood only and works independently for each view.

### 6.5.3 Conclusions

In some cases the spatial resolution of the provided depth maps may not be sufficient for the necessary filtration procedures of the proposed algorithm. The reason is that during the virtual view synthesis, computed spatial pixel coordinates are in general not integers. The rounding process may cause those two points to overlap, which causes an occlusion effect. A solution to this problem is an increase in the spatial resolution of the destination virtual depth map; therefore, the author proposes to represent the depth maps with half-pixel or even quarter-pixel resolution for the purpose of processing. The sub-pixel representation of a depth data requires dedicated up-conversion and down-conversion mechanisms to be used. The up-conversion is done during the virtual view synthesis process and the down-conversion is performed using traditional low-pass filtration followed by spatial decimation.

It is possible that a multi-view sequence contains depth maps that represent different distance ranges. This happens when the parameters Z-Near and Z-Far of the near and far clipping planes are different for each view. In this case, prior to processing, the depth maps need to be normalised to a common distance representation range. Another approach is possible where all depth maps are normalised to the range of a destination depth map during the virtual view synthesis process. The author chose the latter approach as it does not require global re-normalisation of all the depth maps of the sequence.

## 6.5 Inter-view inconsistency reduction

The author conducted an experiment to prove the proposed algorithm's ability to remove inter-view depth inconsistencies. The experiment was aimed at evaluating the level of inter-view inconsistency in the original depth maps and in depth maps processed by the proposed algorithm.

A representative set of multi-view sequences was used [CFP] for the evaluation. The author of this dissertation has also contributed to creation of this test material [Domański_15]. Each sequence contains three views with depth maps. Most of the sequences are provided with semi-automatically estimated depth maps which exhibit inter-view inconsistencies. Two sequences are synthetic and therefore provide ground-truth depth maps for each view. These depth maps are consistent in nature but were processed in order to verify the behaviour of the algorithm in such a case. Table 6.5.1 summarises the multi-view sequences used in the experiment.

Table 6.5.1. – Summary of the test sequences [CFP]

| Sequence name | Resolution | Frame rate | Length (frames) | Depth map type | Z-Near and Z-Far parameters |
|---|---|---|---|---|---|
| Balloons | 1024x768 | 30 Hz | 300 | semi-automatic | constant |
| Undo Dancer | 1920x1088 | 25 Hz | 250 | ground truth | constant |
| GT Fly | 1920x1088 | 25 Hz | 250 | ground truth | variable, different between cameras |
| Kendo | 1024x768 | 30 Hz | 300 | semi-automatic | constant |
| Lovebird1 | 1024x768 | 30 Hz | 240 | semi-automatic | constant, different between cameras |
| Newspaper | 1024x768 | 30 Hz | 300 | semi-automatic | constant |
| Poznan Hall2 | 1920x1088 | 25 Hz | 200 | semi-automatic | constant |
| Poznan Street | 1920x1088 | 25 Hz | 250 | semi-automatic | constant |

Each sequence was processed by the proposed algorithm. The spatial resolution enhancement was set to quarter-pixel in the horizontal dimension only. This is due to the fact that all sequences were captured using a linear camera setup. The minimum allowed inter-view inconsistency change between consecutive iterations was set to 0.05. The iterative process terminates when the change is less than the preset threshold. Table 6.5.2 summarises the inter-view inconsistency measure for each sequence before and after processing by the proposed algorithm.

Table 6.5.2. – Average inter-view inconsistency for each sequence before and after processing by the proposed algorithm.

| Sequence name | Inconsistency (original depth maps) | Inconsistency (processed depth maps) | Number of iterations |
|---|---|---|---|
| Balloons | 48.15 | 1.78 | 13 |
| Undo Dancer | 0.73 | 0.56 | 3 |
| GT Fly | 3.09 | 1.70 | 3 |
| Kendo | 251.82 | 1.34 | 11 |
| Lovebird1 | 21.14 | 2.22 | 8 |
| Newspaper | 155.77 | 2.34 | 15 |
| Poznan Hall2 | 6.96 | 0.33 | 7 |
| Poznan Street | 5.20 | 0.43 | 9 |

The proposed algorithm is able to significantly reduce the inter-view inconsistency for all tested sequences. The amount of reduction varies depending on the type of sequence and the scene structure.

The improvement is insignificant for synthetic sequences that have ground-truth depth maps. This is due to the fact that their depth maps were consistent at the beginning of processing. The non-zero inconsistency value is caused by finite resolution of the depth map representation which uses only 8 bits per pixel.

The difference in Z-Near and Z-Far parameters is also the cause of higher inter-view inconsistency. In such a case for each view different values of depth represent different physical distances. The granularity of physical distance representation may also vary between the views.

The greatest inter-view consistency improvement can be achieved for natural multi-view sequences with automatically or semi-automatically estimated depth maps. As was mentioned before in this chapter, depth maps in such sequences are estimated independently, which leads to inter-view inconsistencies.

The experiment showed that the inconsistency drops significantly within the first few iterations. Figure 6.5.1 shows plots that present the inter-view inconsistency change versus the number of algorithm iterations for each multi-view sequence.

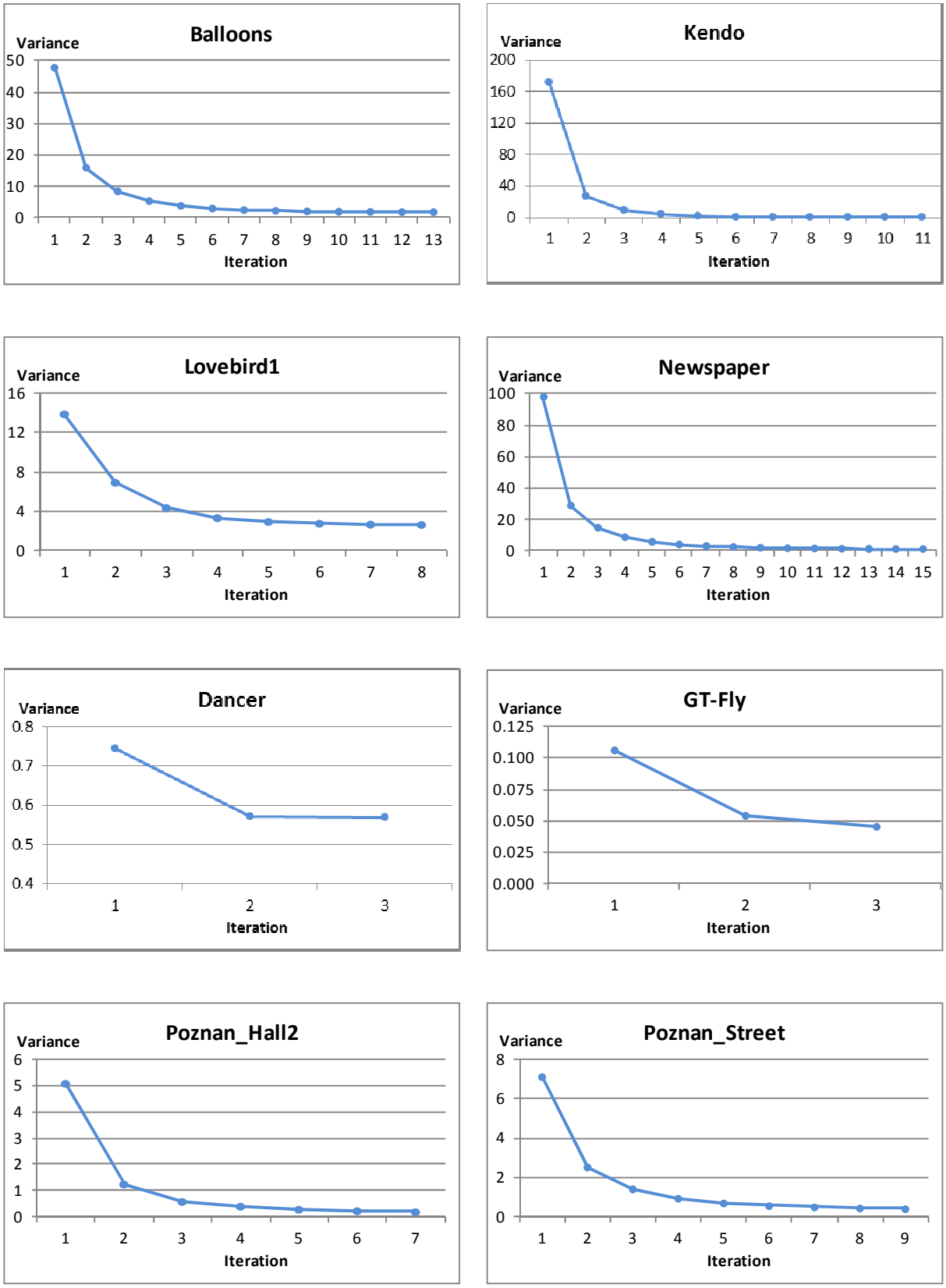



Figure 6.5.1 – Plots describing inter-view inconsistency during each iteration.

As the algorithm operates on each temporal frame independently, the plots show the inconsistency change for a single temporal frame only. In this case it is the first temporal frame of each sequence.

The plots prove that the algorithm reaches convergence for different kinds of multi-view sequences. The tested sequence set is considered as a representative sample of multi-view content by the MPEG group. It is therefore safe to assume that the algorithm is convergent. The number of iterations required to reach the termination condition depends highly on the original sequence inter-view consistency. The higher the consistency, the lower the number of iterations is required. For synthetic depth maps (sequences Dancer and GT-Fly) the algorithm stops on the third iteration and the consistency improvement is negligible. On the other hand, for sequences with poor depth map consistency (such as Newspaper) it took 15 iterations to reach the termination condition.

Nevertheless, the proposed algorithm is a useful tool that allows to improve the quality of depth maps for multi-view processing such as virtual view synthesis and multi-view-oriented video compression.

## 6.6 Impact on multi-view video compression

The main goal of the proposed depth map processing algorithm is to increase multi-view compression efficiency. The author conducted an experiment in order to assess the compression ratio improvement that the proposed algorithm can give in conjunction with state-of-the-art multi-view video compression techniques. For this purpose, three of the most recent multi-view compression techniques were used: the MVC+D and 3D-AVC codecs based on H.264/AVC compression technology and the 3D-HEVC based on the most recent single view HEVC codec.

The experiment was performed using a reference multi-view sequence set [CFP] according to the common test conditions for multi-view compression evaluation provided by the MPEG group [CTC]. The common test conditions specify that the PSNR measure [PSNR] should be used for compressed video quality assessment. The PSNR is used to assess the quality of compressed video as well as virtual views, synthesised using compressed video and depth maps. The document also specifies the parameters for virtual view synthesis to be performed using reconstructed video and depth maps.

Each test sequence was compressed using a set of four quality settings as provided in the common test conditions recommendation [CTC]. For each quality setting, the decompressed video frame was compared with the reference using the PSNR metric. The decompressed image along with the decompressed depth map were used to generate a virtual view, which was then compared to the reference also using the PSNR.

The PSNR values are used to obtain the rate-distortion (RD) curve points. Each RD point comprises a PSNR value and a bitrate value that corresponds to a part of the bitstream used to encode a particular kind of piece of information. There are two RD curves for each compression scenario:

- video PSNR vs. video data bitrate
- virtual view PSNR vs. overall data stream bitrate

Each quality setting is also provided with a reference anchor RD curve point according to [CTC]. These references are used for comparison with the data points obtained in the experiment.

Bitrate gains and PSNR gains, obtained by application of the proposed depth map refinement algorithm, were evaluated using the Bjontegaard metric [Bjontegaard_01] with piecewise cubic interpolation. The Bjontegaard metric is based on computation of the surface area between two RD curves. Each curve is interpolated using four data points.

Bitrate changes are expressed as percentages of the bitrate reduction. Negative values indicate reduction, while positive values indicate an increase of the bitrate. PSNR changes are expressed as absolute PSNR differences in decibels. Positive values indicate an increase of quality; negative values indicate its decrease. Video data coding bitrate and PSNR gains were computed using bitrates of the video data components of the bitstream versus average PSNRs of reconstructed video frames. Synthesised view gains were computed using the overall bitrate versus average PSNR of virtual views.

The plots of RD curves used for bitrate and PSNR gain estimation are shown in annex B of this dissertation. Due to negligible changes in the bitrate and PSNR for video data coding, only plots of virtual view PSNR versus overall bitrate are shown here.

### 6.6.1 Compression using H.264/AVC-based codecs

The 3D-ATM reference software [3D-ATM] was used for the H.264/AVC-based compression. The 3D-ATM software provides two modes of operation, i.e. “High profile” for MVC+D compression and “Enhanced High Profile” for 3D-AVC compression. In this experiment, depth maps processed by the proposed inter-view consistency improving algorithm were used as summarised in table 6.5.2.

Table 6.6.1 summarises the results for MVC+D codec compression and Table 6.6.2 for 3D-HEVC codec compression. Columns labelled “Coded video frames” summarise change in bitrate allocated for video data only versus video PSNR change. Columns labelled “Synthesided views” depict change in bitrate allocated for depth data versus virtual view PSNR change which was synthesized using decoded depth maps.

Table 6.6.1 – Bitrate and PSNR gains for video and virtual views for the MVC+D codec with the proposed depth map inter-view consistency improvement algorithm.

| | Coded video frames | | Synthesised views | |
|---|---|---|---|---|
| **Sequence** | **Video ΔBR [%]** | **Video ΔPSNR [dB]** | **Depth ΔBR [%]** | **Virtual view ΔPSNR [dB]** |
| Poznan Hall 2 | 0.00% | 0.00 | -2.68% | 0.10 |
| Poznan Street | 0.00% | 0.00 | -6.22% | 0.22 |
| Undo Dancer | 0.00% | 0.00 | -3.61% | 0.12 |
| GT Fly | 0.00% | 0.00 | -2.97% | 0.12 |
| Kendo | 0.00% | 0.00 | -7.84% | 0.39 |
| Balloons | 0.00% | 0.00 | -3.96% | 0.20 |
| Newspaper | 0.00% | 0.00 | -8.05% | 0.32 |
| **Average** | **0.00%** | **0.00** | **-5.05%** | **0.21** |

Table 6.6.2 – Bitrate and PSNR gains for video data and virtual views for the 3D-AVC codec with the proposed depth map inter-view consistency improvement algorithm.

| | Coded video frames | | Synthesised views | |
|---|---|---|---|---|
| **Sequence** | **Video ΔBR [%]** | **Video ΔPSNR [dB]** | **Depth ΔBR [%]** | **Virtual view ΔPSNR [dB]** |
| Poznan Hall 2 | 0.02% | 0.00 | -2.90% | 0.06 |
| Poznan Street | 0.02% | 0.00 | -7.93% | 0.22 |
| Undo Dancer | -0.02% | 0.00 | -1.08% | 0.04 |
| GT Fly | -0.03% | 0.00 | -5.53% | 0.16 |
| Kendo | -0.03% | 0.00 | -7.39% | 0.29 |
| Balloons | -0.01% | 0.00 | -3.42% | 0.14 |
| Newspaper | -0.01% | 0.00 | -9.84% | 0.36 |
| **Average** | **-0.01%** | **0.00** | **-5.44%** | **0.18** |

For the MVC+D codec there is no change in video data compression performance between using original and processed depth maps. Only depth map compression is affected. This is due to the fact that the video data and depth maps are coded independently. For the 3D-AVC codec there is a slight bitrate gain that ranges up to 0.03%. This is because the 3D-AVC codec may derive control information from encoded depth maps for video data compression.

The depth map compression results show a significant decrease in the bitrate of the depth map bitstream components. Moreover, there is an increase in the PSNR of virtual views. This is an expected result as the proposed depth map processing algorithm increases their inter-view consistency. This, in turn, allows the codec to exploit their inter-view similarities.

### 6.6.2 Compression using the HEVC-based codec

For the purpose of 3D-HEVC compression, HTM reference software version 5.0 was used which implements the full codec [3D-HTM].

Table 6.6.3 summarises the results for the 3D-HEVC state-of-the-art compression technique. Column labelled "Coded video frames" summarise change in bitrate allocated for video data only versus video PSNR change. Column labelled "Synthesided views" depict change in bitrate allocated for depth data versus virtual view PSNR change which was synthesized using decoded depth maps.

Table 6.6.3 – Bitrate and PSNR gains for video data and virtual views for the 3D-HEVC codec with the proposed depth map inter-view consistency improvement algorithm.

| | **Coded video frames** | | **Synthesised views** | |
|---|---|---|---|---|
| **Sequence** | **Video ΔBR [%]** | **Video ΔPSNR [dB]** | **Depth ΔBR [%]** | **Virtual view ΔPSNR [dB]** |
| Poznan Hall 2 | -0.01% | 0.00 | -3.37% | 0.11 |
| Poznan Street | 0.08% | 0.00 | -6.66% | 0.22 |
| Undo Dancer | -0.03% | 0.00 | -3.99% | 0.13 |
| GT Fly | -0.16% | 0.01 | -3.00% | 0.11 |
| Kendo | 0.08% | 0.00 | -8.80% | 0.39 |
| Balloons | 0.11% | -0.01 | -4.39% | 0.20 |
| Newspaper | -0.21% | 0.01 | -8.03% | 0.31 |
| **Average** | **-0.02%** | **0.00** | **-5.46%** | **0.21** |

Similarly, as for the AVC-based codecs, the bitrate reduction is observed for depth map compression. The proposed algorithm allows to reduce the bitrate of a multi-view sequence by over 5% when measured over average virtual view quality. The amount of gain depends on the initial quality of the depth maps of the input sequence. The more consistent the depth was before processing, the less bitrate reduction is observed.

## 6.7 Conclusions

The proposed algorithm allows to significantly reduce inter-view inconsistency of the depth maps. This inconsistency reduction can be observed for different kinds of multi-view sequences with different scenes and depth map structures. The proposed algorithm was tested over a representative multi-view sequence set provided by the MPEG group [CFP]. The algorithm was evaluated using multi-view sequences with three views; however, the proposed method is not limited to three views, thus a larger number of views can be used.

The major goal of improving depth map inter-view consistency is to increase the compression ratio of a multi-view-oriented video codec. State-of-the-art multi-view compression techniques use depth maps for inter-view prediction mechanisms and virtual view synthesis. The consistency of depth maps is very important. The application of the proposed algorithm to depth maps prior to their

compression yields an over 5% reduction of the compressed bitstream size. This reduction is achieved while maintaining constant or even increased quality of virtual views synthesised from coded video data and depth maps.

The depth map processing method allows to improve their quality without the need for their re-estimation. This, in turn, leads to computing time reduction as simultaneous depth map estimation for all views would be extremely time consuming.

Therefore, the thesis T2: “Improvement of inter-view depth map consistency that increases depth map quality and increases the compression efficiency of compression algorithms which exploit inter-view relations of depth maps” has been proven.

# 7. Conclusions

## 7.1 Original achievements of the dissertation

### 7.1.1 Inter-view consistency improvement of depth maps

Independent estimation of depth maps for neighbouring views unavoidably causes them to be inconsistent. The reason is that each depth map is estimated using stereo correspondences from different camera pair or triple [DERS].

In order not to estimate multiple depth maps simultaneously, which would be time- and memory-consuming, the author proposed a depth processing algorithm. The algorithm allows to eliminate inter-view inconsistencies by exchanging information between them. The details of the algorithm are described in chapter 6.

Processed depth maps exhibit better inter-view consistency, which allows them to be encoded more efficiently using a multi-view codec. Compression experiments using state-of-the-art multi-view codecs such as **MVC**, **MVC+D**, **3D-AVC** and **3D-HEVC** have shown that the bitrate can be reduced by an **average of 5%** when compared to compression of original depth maps. What is more important is that the quality of compressed video data, encoded using depth information, is not negatively affected by the depth map processing.

The proposed algorithm became a part of the 3D video compression technology proposed by Poznań University of Technology for the MPEG competition in 2011 [Domanski_01][Domanski_13]. The compression technology was assessed as the **second best in the world** in terms of multi-view compression efficiency.

### 7.1.2 Depth estimation augmented by data from ToF depth camera(s)

Another major achievement of the author as presented in this dissertation is a method of fusion of depth information from ToF camera(s) with video information from video camera(s).

The fusion process allows to incorporate additional depth cues into one of the state-of-the-art depth map estimation algorithms [DERS]. These additional cues allow the estimated depth map to be more accurate and more consistent with actual physical distances. Video data and ToF depth fusion is done before the global optimisation stage of the depth estimation algorithm. Therefore, the global optimisation algorithm can choose the best disparity for each pixel by using stereo correspondence information together with depth measurements.

The author managed to develop a method which allows to aggregate distance data from more than one ToF camera. The use of multiple ToF cameras yields better distance measurement accuracy as well as wider scene coverage.

The modification of the depth estimation algorithm along with the method of aggregation of data from multiple ToF cameras is described in chapter 5 of this dissertation.

### 7.1.3 Other important achievements

***Calibration of multi-camera systems with video and ToF depth camera(s):***

The author conducted a large amount of work regarding camera calibration and image rectification. Many modifications were introduced to existing camera calibration algorithms so that they can operate on depth camera data and take advantage of depth features. The most important contributions are: **a modification of the camera distribution line direction** that compensates the common rotation of all cameras in the system (chapter 2.5.2) and **the estimation of depth camera extrinsic parameters** using depth (chapter 2.6).

***ToF depth noise reduction***

Distance data provided by a ToF camera is noisy. Presence of a spatial and temporal noise has a negative effect on 3D scene representation. **Therefore the author has proposed a ToF data denoising algorithm.** The algorithm proposed by the author takes advantage of both intensity data and distance data in order to perform **spatial bilateral filtration** and **motion adaptive temporal filtration.** As a result, temporal depth fluctuations are reduced, which improves the temporal consistency of the depth map sequence without motion blur and object edge distortion effects. The proposed technique is fully described in chapter 3.4.

***Synchronisation of video cameras and depth cameras***

In a multi-view camera system, all cameras are required to capture frames synchronously. Unfortunately, both video and depth cameras use different kinds of synchronisation signals, which makes it impossible to synchronise them directly.

The author thoroughly investigated the problem of conversion between the Genlock signal that is used for video cameras and the Trigger signal that is used for depth cameras. As a result, the author has proposed a **method of conversion from a Genlock to a Trigger signal** which allows a single Genlock signal generator to synchronise both video and depth cameras. The details of this synchronisation are presented in chapter 4.

Moreover, the author managed to **design and build a hardware device that implements the proposed algorithm**. The device converts an input Genlock signal to an output Trigger signal for depth cameras. Detailed information on the design of this device is shown in annex A of this dissertation.

## 7.2 Future work

Currently, depth acquisition technology provides many limitations which do not allow all of the proposed algorithms to reach their full potential.

Time-of-Flight imaging technology requires fast image sensors with built-in phase measurement sub-circuits. Due to their complexity they cannot reach high resolutions as can typical modern video sensors. ToF cameras produce a significant amount of noise at frame rates close to those used for video acquisition. This noise is mostly the result of short integration time of noisy data that comes from the sensor.

On the other hand, distance acquisition technologies such as structured lighting also require many improvements in order to provide high quality depth data, e.g. Microsoft Kinect uses an infra-red laser projector that projects a special dot pattern which is then observed by a camera and its distortions are translated into distance data. The problem is that the pattern has finite resolution which is much lower than the camera and which makes that technology non-suitable for measurement of small objects.

Other structured lighting techniques that consist of a video projector and a video camera provide very accurate distance measurement, although they are slow, as many patterns need to be projected consecutively in order to measure the distance at full resolution. Most of these systems operate in a visible light range, which makes them non-suitable for multi-camera systems as they will interfere with video acquisition. There are infra-red based structured lighting systems, which would not interfere with video cameras, but the problem of slow acquisition still remains unsolved. ToF sensors are much faster.

The author hopes that this technology will develop in the future and that high-resolution, fast and accurate depth measurements using ToF cameras will be possible.

# Annex A - Implementation of the synchronisation signal conversion device

The main function of the proposed design is conversion from an input Genlock synchronisation signal to an output periodic trigger pulse signal. The Genlock frame boundary is defined by a series of vertical blanking pulses which the proposed device is intended to detect. Due to the specific relation between actual sensor exposure times in synchronised cameras, the output trigger signal must have the possibility of being delayed about a preset time value. The proposed multi-camera system may contain more than one depth camera. As it is difficult to have many ToF cameras measuring distance simultaneously due to possible interference among them, each camera may have its own, independent trigger output which allows to implement interleaved triggering in a sequence. Interleaved triggering also makes it possible to extend the integration time of the camera beyond the period of a single video frame. The effective frame rate for each depth camera may be lower than for video cameras.

Because depth cameras do not support timecodes, it is impossible to determine the exact frame correspondence between video and depth frames. There is no possibility of overcoming this ambiguity other than by applying an external light stimulus to all cameras. However, this problem may be reduced by starting the triggering of depth cameras on particular video frames. In order to do this the timecode information must be known by the synchronisation device. The proposed design incorporates an external timecode input which allows to input this information [SMPTE12].

In order to be able to verify the correctness of synchronisation between video and depth cameras, the synchronization converter needs to be equipped with a light-emitting device which is visible to both types of cameras. The light-emitting device must also be able to change its displaying pattern in time. The way of pattern change must allow to determine the sensor exposure moment by observing the pattern image provided by the camera.

The need for real-time signal processing and meeting the strict timing constraints implies that the device must be based on a combinatorial and sequential logic design rather than on a program executed by a microcontroller. The design is also required to be flexible and reprogrammable, which disqualifies an approach based on connecting specialised (but fixed-function) integrated circuits. The choice was made to use the field programmable gate array (FPGA) fabric which is very flexible and allows to test and implement different logic designs. The use of an FPGA integrated circuit also simplifies the electrical circuit design and the physical board layout thanks to fully configurable input and output ports.

## Functional design

The design can be divided into several functional blocks. The input signal conditioners for Genlock and timecode signals are used to convert input voltage levels to levels accepted by the FPGA chip. All processing blocks are implemented inside the FPGA fabric. Output trigger pulses are

converted to 5v TTL logic signals accepted by depth cameras. Possible return signals from synchronised cameras are converted back to 3.3v logic levels. The RS-232 [RS232] port provides the means of control and debugging. The complete block diagram is shown in Figure A.1.

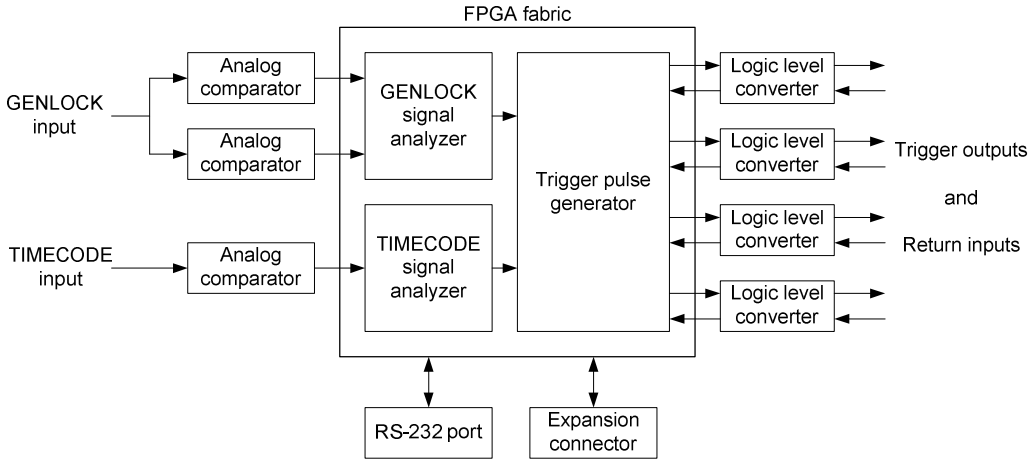



Figure A.1 – Block diagram of the synchronisation device proposed by the author.

The Genlock signal is a tri-level signal. The voltage on the line may assume three possible voltage levels: zero, positive and negative. To represent three states using binary signals, at least two digital signals are required. Two input comparators are responsible for detection of positive and negative voltage on the line. The output comparison decision signals are connected to the FPGA chip. The timecode signal is a bi-level signal where the positive and negative voltages define the state of the line; hence only one comparator is needed.

For FPGA logic implementation the author used the Xilinx Spartan-3 FPGA. The chip type is XC3S200 contained in a TQFP144 case. According to the Xilinx device datasheet it has 200,000 system gates and 2160 slices. This is more than enough for the implementation of Genlock and timecode analysers and trigger pulse generators. The maximum logic operating frequency may be as high as 100MHz but it decreases rapidly along with the increase in logic design complexity to about 50MHz. A frequency of 50MHz is not enough for Genlock signal generation (as its base frequency for 1080p50 video is 74.25MHz), but an analysis of the signal requires only approximate pulse width detection which can be done using logic running at 50MHz.

The Genlock and timecode signals are decoded inside the FPGA fabric. The Genlock signal decoder is responsible for extraction of horizontal, vertical and frame synchronisation signals. The decoder measures incoming pulse lengths and returns the information to the finite state machine which identifies the appropriate sequence and generates output frame synchronisation signals. A similar process is done for the timecode signal. The timecode information is extracted and stored in memory registers. Because the Genlock and timecode signals are supplied to the device via two independent inputs, they need to be synchronised. As the Genlock signal is the reference one, the

timecode is synchronised to the Genlock frame boundary provided by the synchronisation analyser. Both signals are synchronous.

The correct combination of frame synchronisation pulse and desired timecode value triggers an output pulse generator. The generator is designed in such a way that it responds to an input pulse with an output pulse of constant length. The output pulse length is chosen to meet the synchronised camera requirements.

Output trigger signals generated by the FPGA chip need to have appropriate voltage levels to be recognised by depth cameras. The conversion is performed by four independent channel logic-level translators which convert input logic levels to output 5v TTL levels accepted by cameras. According to information from the datasheets of particular cameras, the camera may provide a return signal which informs about the finished frame capture process. There are additional four inputs capable of interpreting 5v logic level signals for these signals. The voltage level conversion is performed by a simple resistive divider.

The experimental nature of the described device and the need for its further development imply the use of additional electrical connections to the FPGA chip. For development purposes the device is equipped with a RS-232 [RS232] serial port. The RS232 standard is widely used in industrial devices thanks to its simplicity. The intended use is communication between the synchronisation device and an external control device such as a PC computer running the appropriate application. The expansion slot provides a means to access most of the unused FPGA chip input/output pins. Some of these pins are used to communicate with the light-emitting device, which is used, in turn, to verify the synchronisation accuracy.

The light-emitting device has a modular construction. Each module is equipped with eight infrared and eight visible light-emitting diodes (LEDs) arranged in a dual row fashion. Each type of diode is meant to be visible by a different type of camera. Visible LEDs can be registered by a video camera while infrared diodes deliberately distort the ToF camera measurements which, in turn, can be detected in the image. Each LED section is driven by an 8-bit shift register equipped with constant current LED drivers. Each module has an independent shift register for each LED kind. The block diagram of a single LED module is shown in Figure A.2.

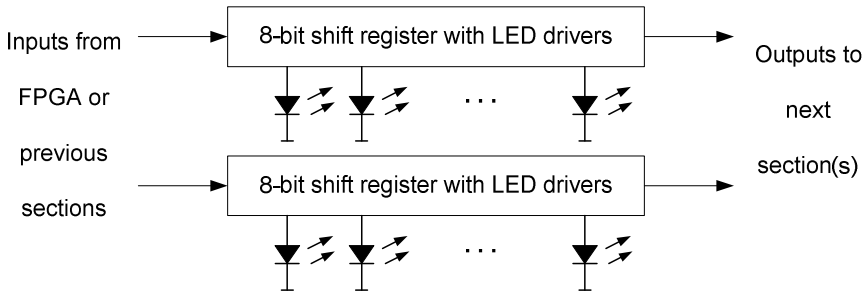



Figure A.2 – Block diagram of the light-emitting module.

The use of shift registers allows for modular construction. Output from one register can be connected to the input from the other from the next section. The registers form a dual chain which allows to display a different pattern by the infrared and visible diodes.

## Electrical design

The electrical design of the synchronisation device can be divided into several functional blocks:

Power supply

Input signal conditioner

FPGA chip and non-volatile configuration memory

Output signal conditioner

The electrical schematic of the device is divided into seven parts in order to maintain clarity of the documentation. Figures A.5 to A.11 show the schematic diagrams of the synchronisation translation device. These figures are located at the end of this chapter.

### *Power supply*

The power supply block provides voltage stabilisation for all the voltage supply rails needed in the design. These rails are:

**+3.3V** used for power supply to all digital chips, especially for their I/O connections

**+2.5V** FPGA chip internal clock management modules and configuration

**+1.2V** FPGA fabric (core)

**-5V** negative power supply for the Genlock input analog comparator

A detailed power block schematic is shown in the "Power" schematic sheet. The synchronisation module is powered from a single DC +5V power supply. The 3.3V power rail is supplied by the LM2673 switching mode, step-down regulator. The regulator chip is equipped with a soft start subcircuit and output current limiter. It realises a typical step down converter which stores energy in the inductor's magnetic field. Control and feedback are provided by the chip itself, while the rest of the necessary components are connected externally. The output voltage is set by a resistive voltage divider which provides the feedback voltage for the control chip. Input and output decoupling capacitors provide voltage ripple rejection.

The main power supply for the FPGA chip is generated by a dual switching mode regulator built using the LTC3417 chip. The chip contains two synchronous switching mode step-down regulators capable of delivering a 1.4A and 800mA output current. The regulators can operate at a frequency up to 4MHz, which allows the use of smaller inductors that, in turn, decreases their physical size. The

1.4A regulator is configured to power a 1.2V supply rail and the 800mA regulator to power a 2.5V supply rail. Both supply rails power the LED diodes for visual inspection of power status.

Correct Genlock signal interpretation requires interpretation of the negative voltages that are present in the signal. In order to do this the input signal is DC coupled to the analog comparator. Detection of negative voltages requires symmetrical power supply of the comparator. The positive +5V voltage is taken from the input power connector directly. The negative voltage is generated by a charge pump-based voltage inverter realised using the ICL7660 chip. Charge pump voltage converters are not able to deliver high power to the load when compared to switching mode. Because the input comparator is the only chip which requires a negative power supply and due to its low power requirements, the solution with the charge pump is sufficient in this design.

### *Input signal conditioner*

All input electrical signals must be properly conditioned to meet the FPGA chip's electrical characteristics. The conditioning is performed by a set of analog comparators. Each comparator determines the sign of the input voltages' difference which is encoded at an appropriate logic level driving the corresponding FPGA input.

The Genlock signal (also known as the tri-level sync signal) contains three voltage levels which are interpreted by a dual comparator built on the LT1715 chip. Due to the presence of very short pulses, the comparator needs to have a fast reaction time to carry the input pulses through. The selected LT1715 chip has a reaction time of 4ns and is able to toggle its output with a frequency of 150 MHz, which is more than enough for the Genlock signal. The input signal is DC coupled to the negative input of comparator A and the positive input of comparator B. The 75Ω resistor provides input impedance matching and termination. Complementary inputs of comparators are connected to the resistive divider, which provides positive and negative +/- 100mV reference voltage. According to [SMPTE240M], the synchronisation signal amplitude is about 300mV. The 100mV threshold allows to detect the attenuated signal while still preventing false detections of unavoidable noise. Comparator A is responsible for detection of a positive voltage above 100mV, while comparator B detects negative voltage below -100mV. The comparator outputs are connected directly to the FPGA chip.

According to [SMPTE12], the longitudinal timecode signal is a bi-level signal with a maximum amplitude of 2V peak-to-peak and the shortest pulse of 40µs. These characteristics allow for the construction of a signal conditioner using the inexpensive and widely available dual operational amplifier LM258. The operational amplifier A works as an analog comparator with hysteresis. Amplifier B is not used. The input signal is AC coupled through a capacitor to the negative input of amplifier A. The positive input is connected to a resistive voltage divider which provides the reference voltage, and to the output through a resistor to form a hysteresis loop. AC coupling of the input signal eliminates the need for a symmetrical power supply. The operational amplifier chip is powered from

the +5V rail. The output of the amplifier can be connected to the FPGA chip directly because of the output stage's voltage drop which prevents the voltage from reaching the value of the power supply.

### *FPGA chip and non-volatile configuration memory*

The Xilinx XC3S200 chip connections are shown on schematic sheets "S3_Power", "S3_Banks" and "S3_Config". The chip is powered from 3.3V, 2.5V and 1.2V supply rails. All I/O banks are powered from the 3.3V rail. Each power pin of the chip is provided with its own decoupling capacitor located close to it.

The FPGA fabric is programmed via the JTAG interface [JTAG]. There are two JTAG connectors on the PCB board, and each of them supports one of the Xilinx programming cables directly. The JTAG signals are connected through series of 100Ω resistors to protect the chip's programming pins in case excessive voltage is applied. The FPGA chip is equipped only with a volatile configuration memory; therefore, an additional device is needed to store the configuration permanently. The configuration data stream is stored inside the Xilinx XCF02S flash chip, which is a dedicated FPGA configuration storage. Both the FPGA and the flash chip are connected to the JTAG interface. JTAG allows multiple devices to be connected in a serial chain, where the data and commands pass through all the devices. This allows to have only a single programming connector for multiple devices.

The configuration may be loaded into an FPGA chip directly or into the flash chip. Data transfer from the flash chip to the FPGA occurs through a set of dedicated connections, apart from the JTAG interface. Whether the configuration is to be loaded from flash memory or not is configured by a jumper located on the PCB board. The design is equipped with a program request button which triggers data transfer from the flash to the FPGA configuration memory. There is also a LED diode which indicates successful configuration.

The master clock for all FPGA modules is generated by an external crystal generator. There is a place on the PCB board for a secondary generator which is currently not used but allows further development of the module.

In order to provide other configuration than the FPGA configuration itself, a set of user switches and push buttons is added to the design. There are 8 independent switches and 4 momentary push buttons connected to the FPGA chip. These are used in the design for the control of synchronisation signal processing. Additionally, there are four general purpose LEDs which currently serve as trigger output activity indicators.

### *Output signal conditioner*

The output signal conditioner realises logic level translation from the 3.3V to 5V standard. Detailed information can be found in the "Outputs" schematic sheet. The design is based on the octal

logic level translator chip 74LVC4245 with dual supply voltages for each side of the conversion. Only four converters are used, and the unused converter's inputs are connected to the ground. Output signals are connected through a series of 100Ω resistors for short circuit protection. Optional return signal input converters are realised using simple resistive dividers.

### *LED Emitting device*

Each LED module uses two shift registers with built-in LED drivers. A pair of registers allows to drive infrared and visible LEDs independently. The registers share a common clock signal while the data inputs remain separated. The module is powered from the FPGA board from the 3.3v supply rail. The power, clock signal and outputs of the last bit of each register are available on the output connector for driving the next LED module in the chain.

## Hardware design

The PCB board of the design contains two signal layers. The copper thickness is 35µm and the overall board thickness is 1.5mm. Most of the components are surface mounted with an exception for the connectors, which are mounted through-hole in order to provide better mechanical durability. All passive components (resistors and capacitors) are standard 0603-sized SMD components [IPC-SM-782][IPC-7351]. The 0603 size is small enough to allow a very compact design and large enough to still allow manual soldering. The components are placed on the top side of the board with some minor exceptions for those which did not fit.

All of the electrical signal PCB traces were routed manually. The top layer is dedicated to power supply traces and all signal traces while the bottom layer is mostly covered by the ground plane. Signal traces which could not be routed on the top layer are routed on the bottom layer. The idea is to keep the ground plane as continuous as possible in order to provide an uninterrupted return path for signal return currents. Signal traces are 10mil wide, which is sufficient to carry relatively small signal currents. On the other hand, power supply traces need to be much wider to carry large supply currents. Most power traces are 50mil thick, which has proved to be sufficient.

The light-emitting device was manufactured as four identical independent boards, each one containing a single shift register and eight pairs of LEDs. Due to their simplicity the boards were designed as single layer only. Separation of the light-emitting device from the rest of the synchronisation module allows it to be mobile ;hence it is easier to place it in front of the camera system.

Figues A.3 and A.4 show photographs of the assembled synchronisation translation device and the light-emitting device with four LED modules.

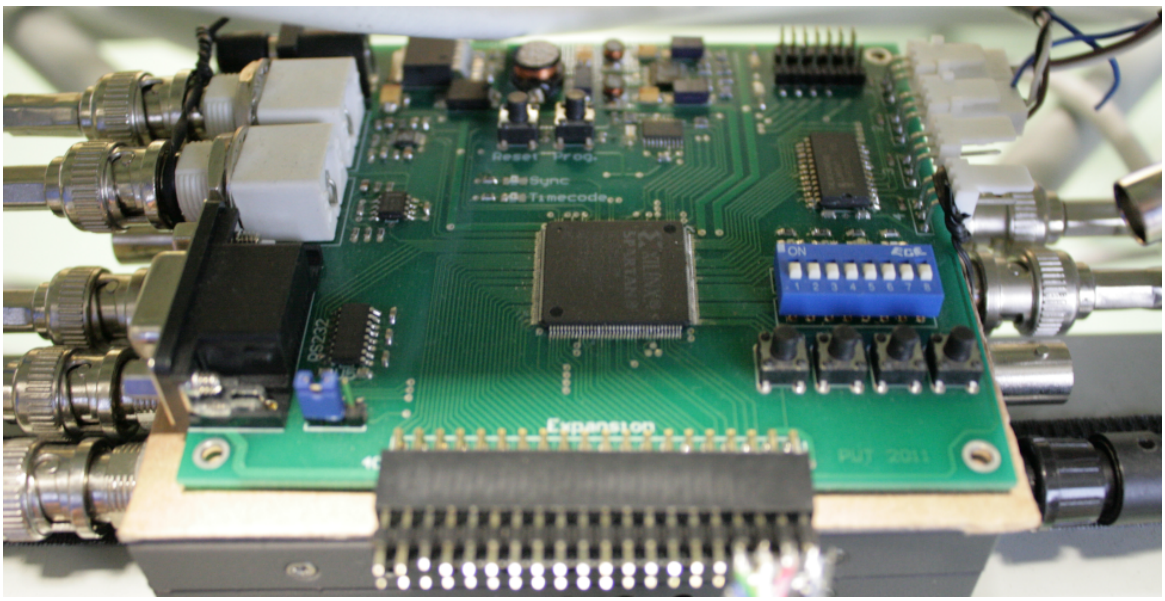



Figure A.3 – Image of the author's synchronisation translation module.

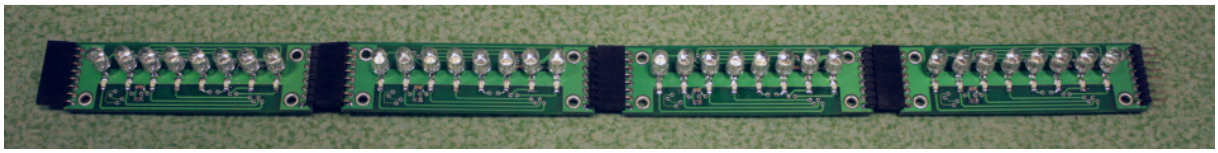


Figure A.4 – Image of the author's light-emitting device used for camera synchronisation testing.

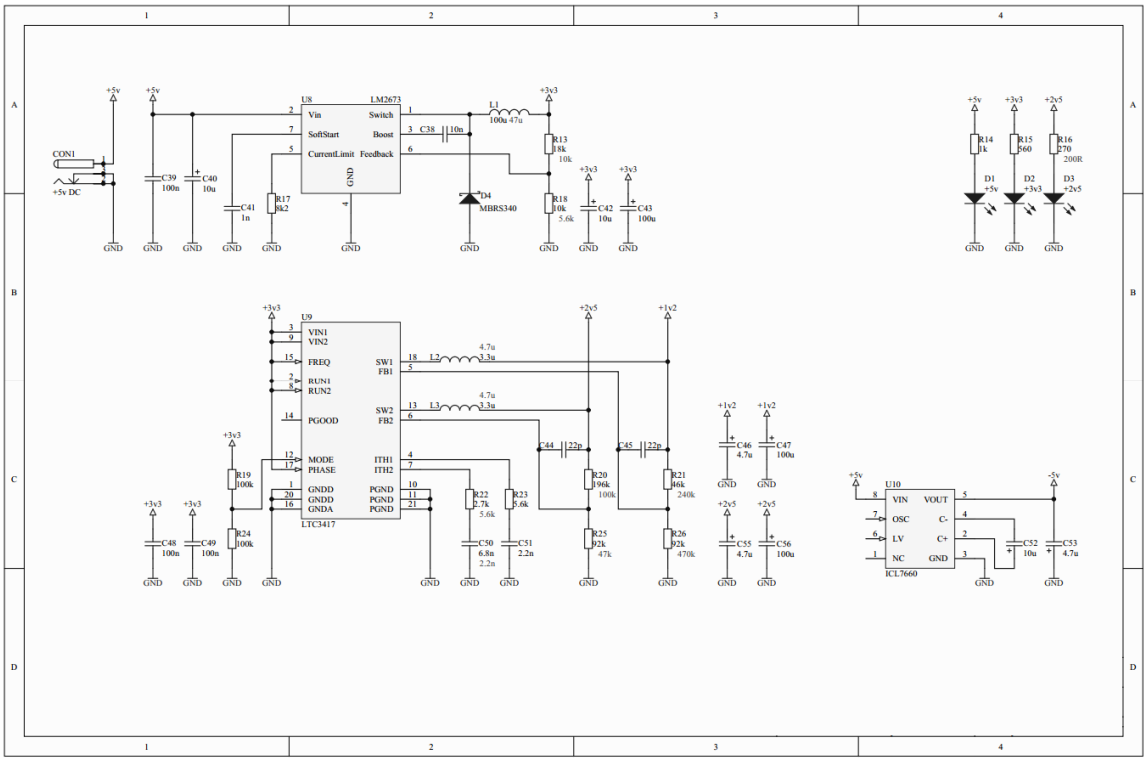



Figure A.5. – Schematic of the power supply of the device.

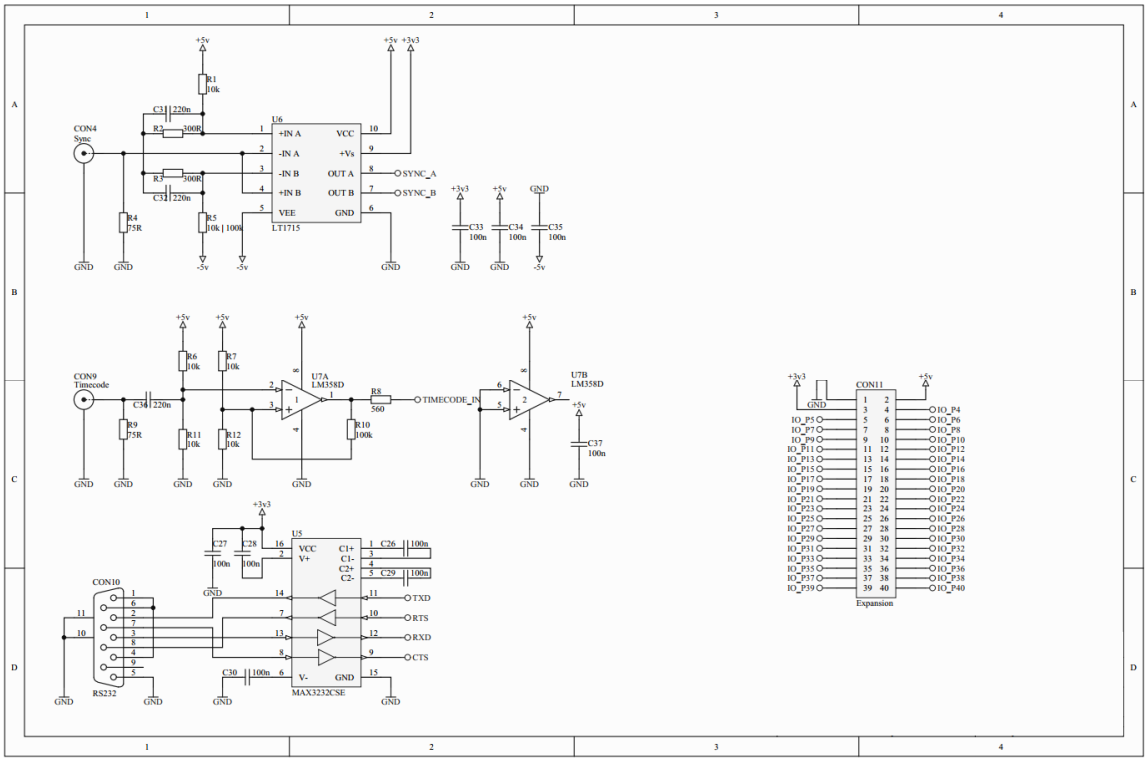



Figure A.6. – Schematic of the input signal conditioners, RS-232 interface and expansion connector.

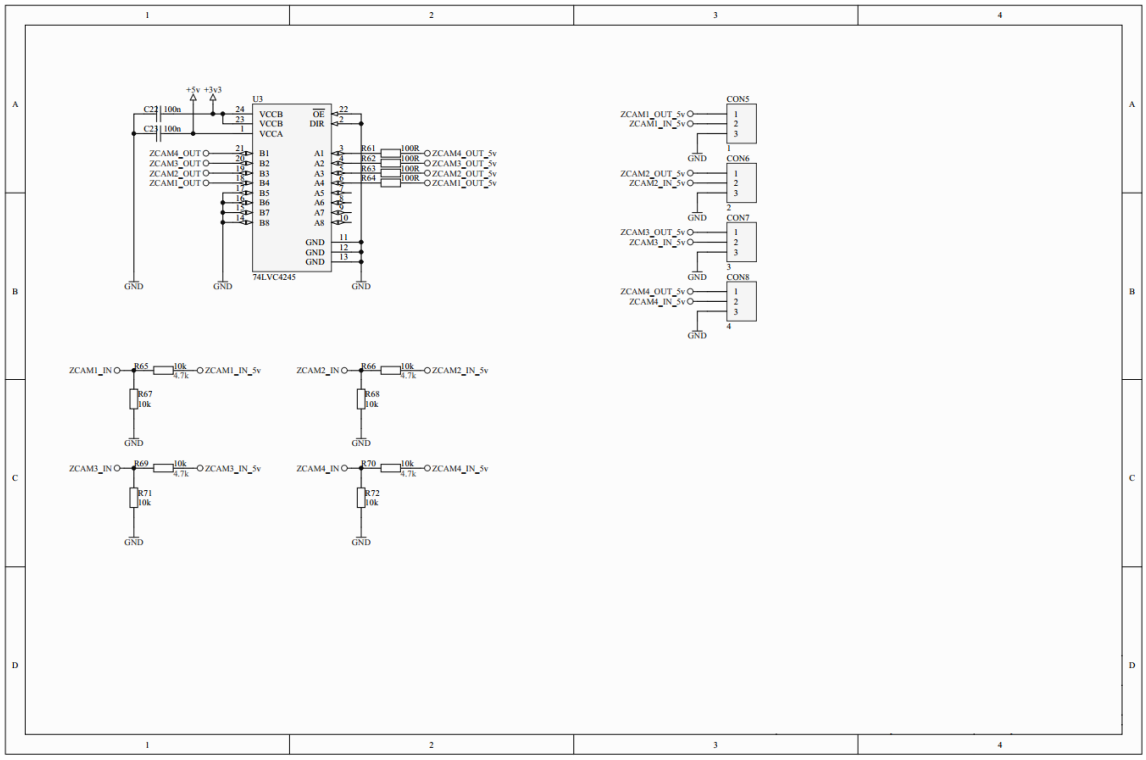



Figure A.7. – Schematic of the output and input signal conditioner for the trigger confirmation signal.

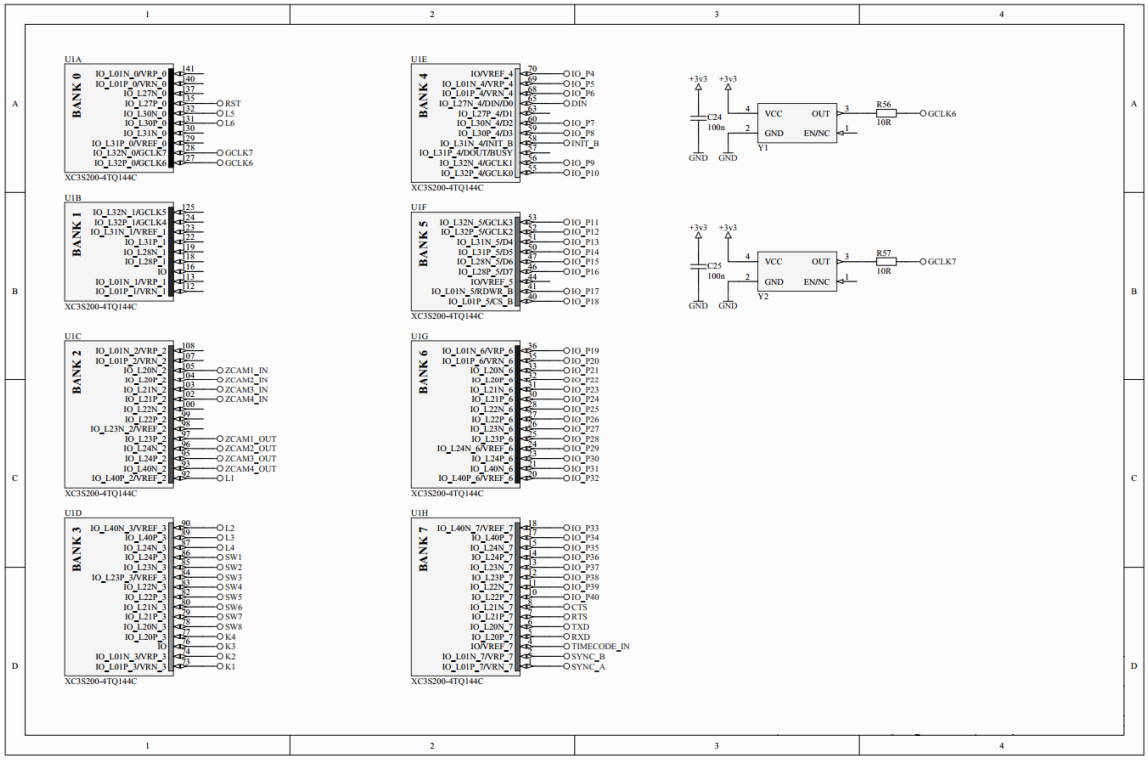



Figure A.8. – Schematic of the clock generators and FPGA input/output connections.

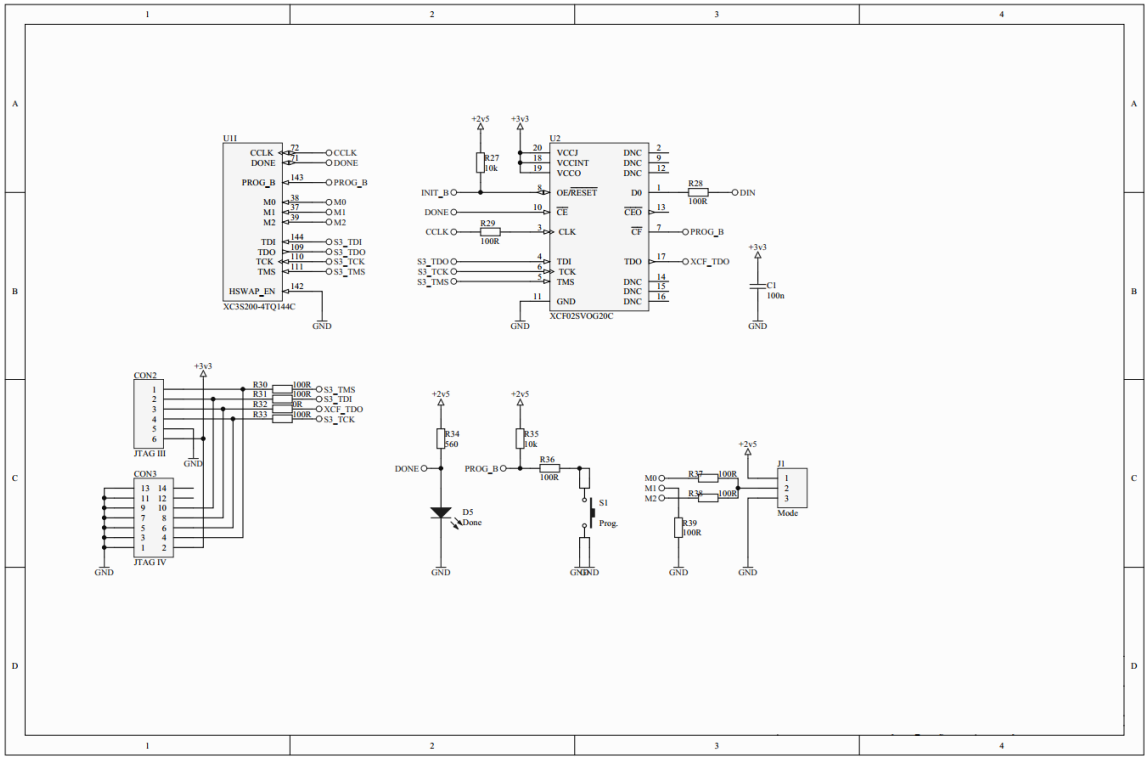



Figure A.9. – Schematic of the FPGA configuration block.

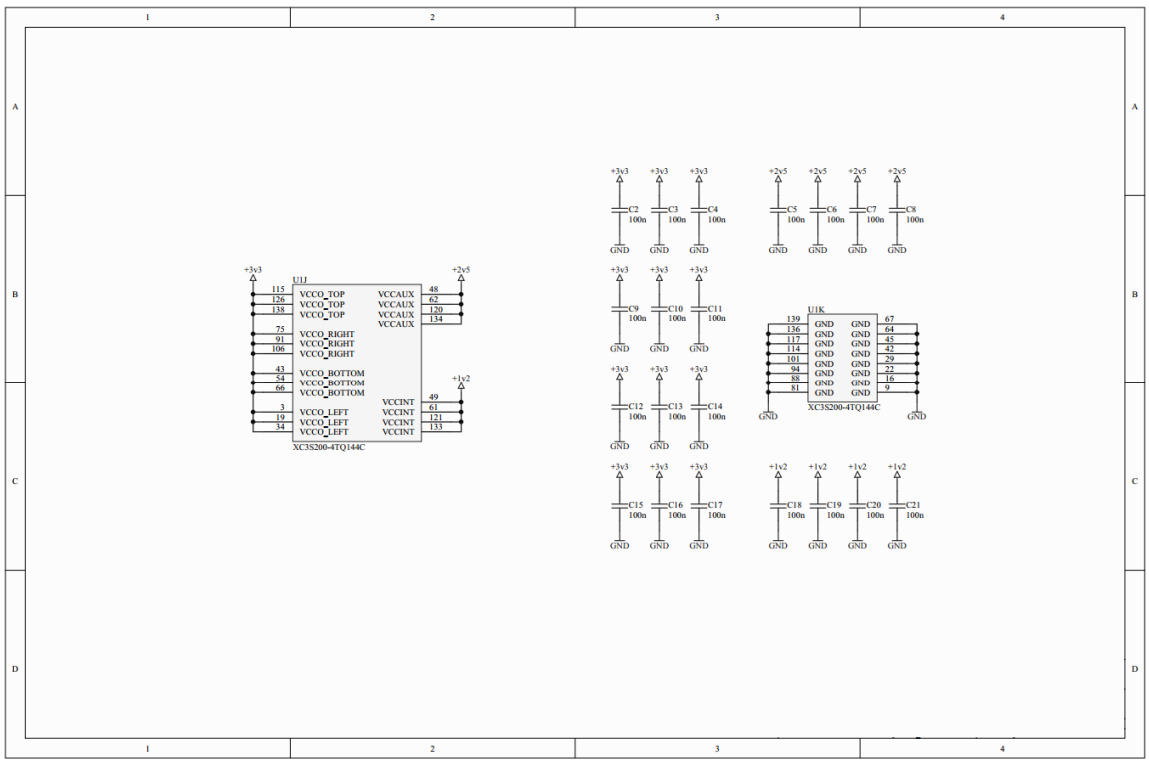



Figure A.10. – Schematic of the FPGA power connectors.

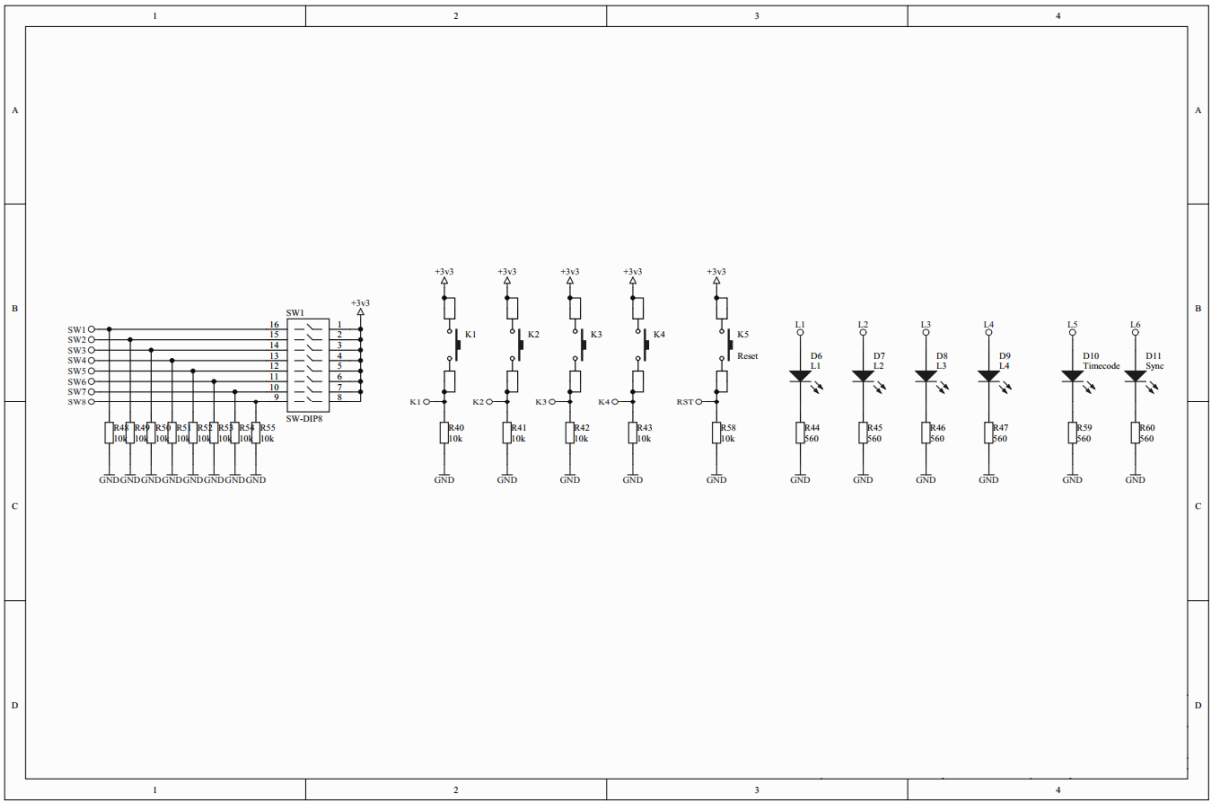
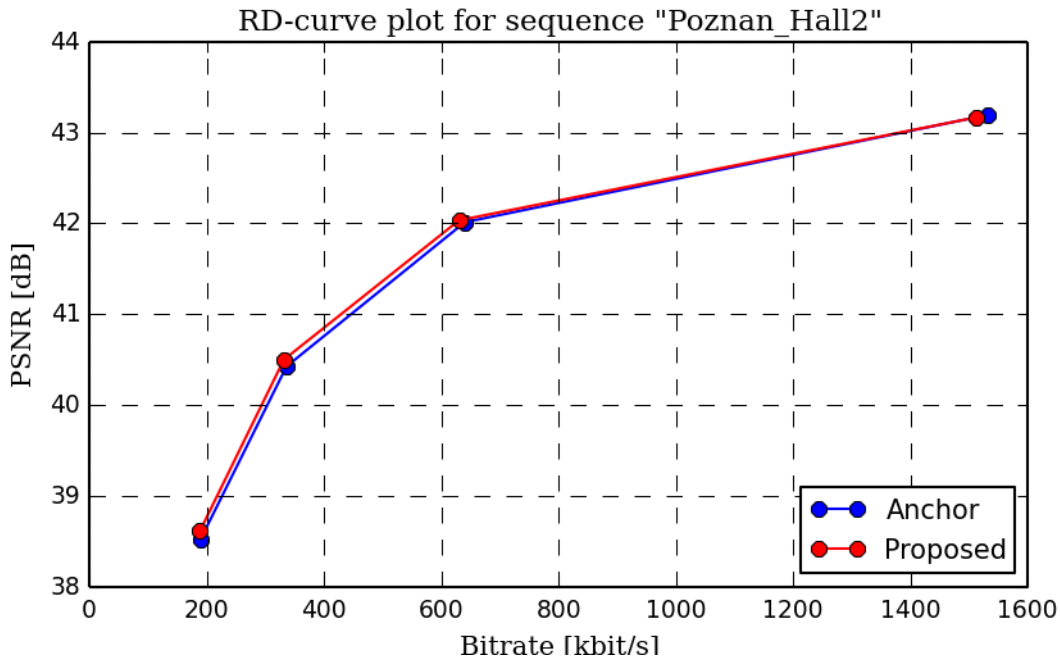


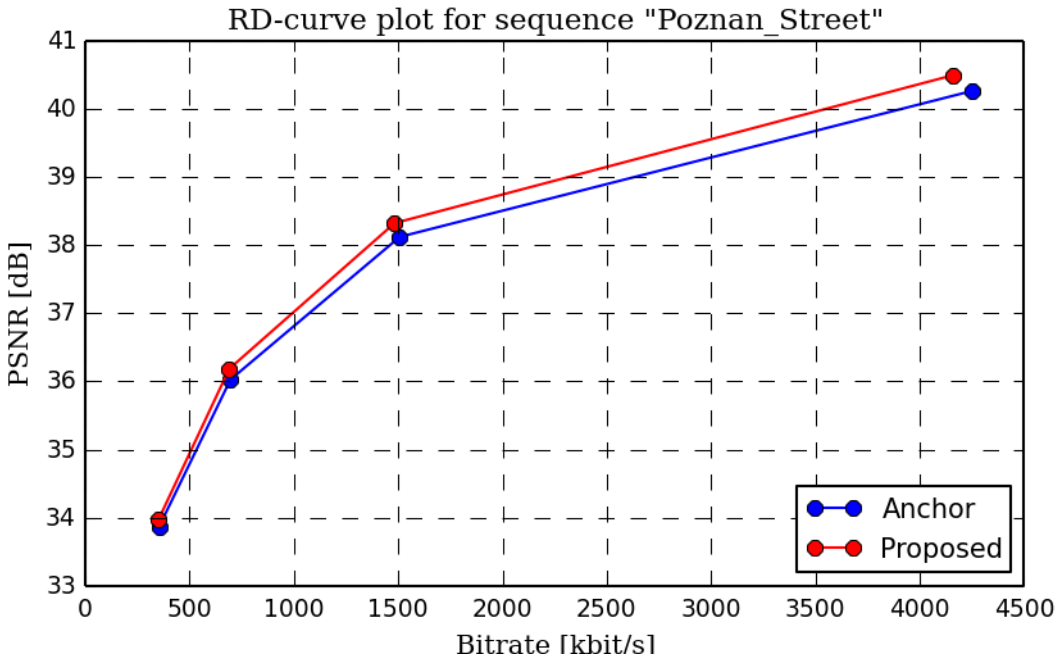

Figure A.11. – Schematic of the user interface elements.

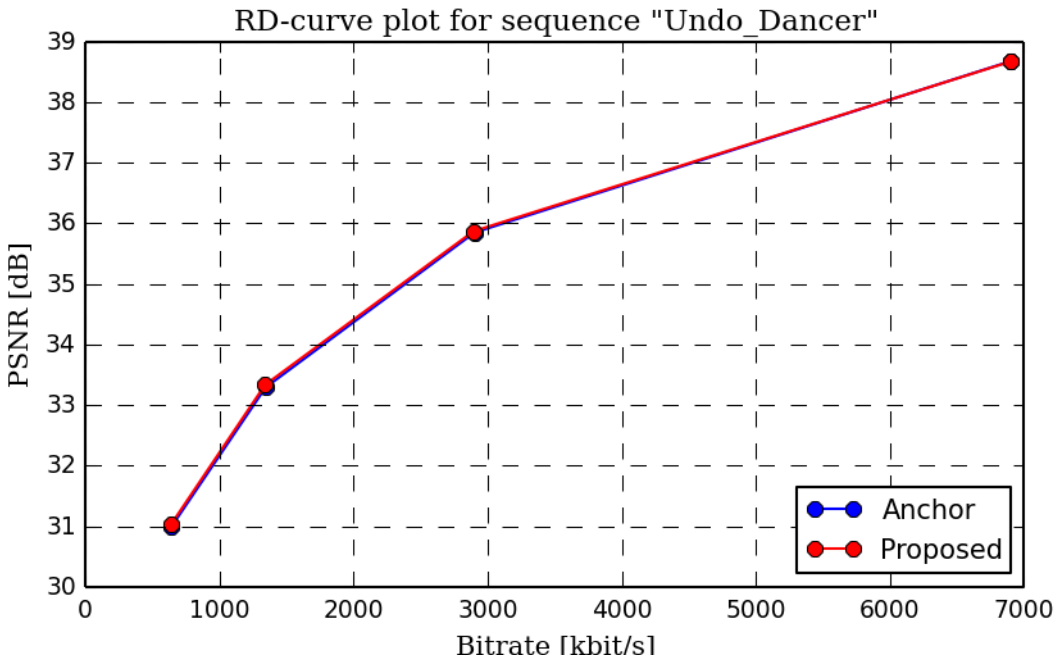
# Annex B – RD multi-view compression curves for the depth map inter-view consistency improvement algorithm proposed by the author

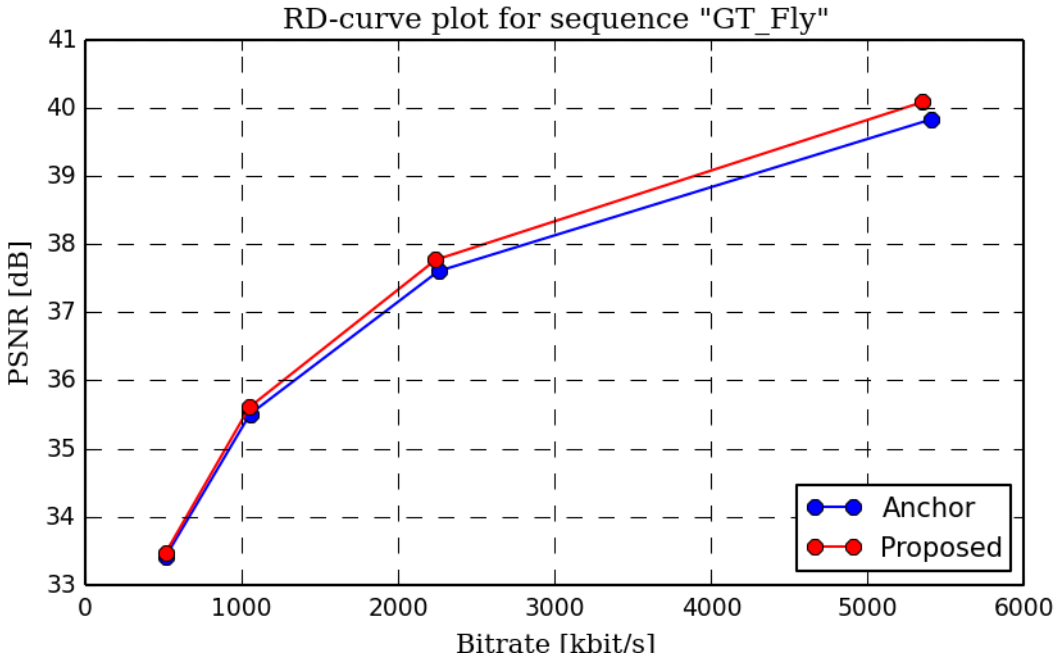
The rate-distortion curve plots shown in this Annex provide detailed information about the behaviour of a multi-view codec in conjunction with the proposed algorithm. For more details regarding the algorithm, experimental conditions and multi-view sequences, please refer to chapter 6

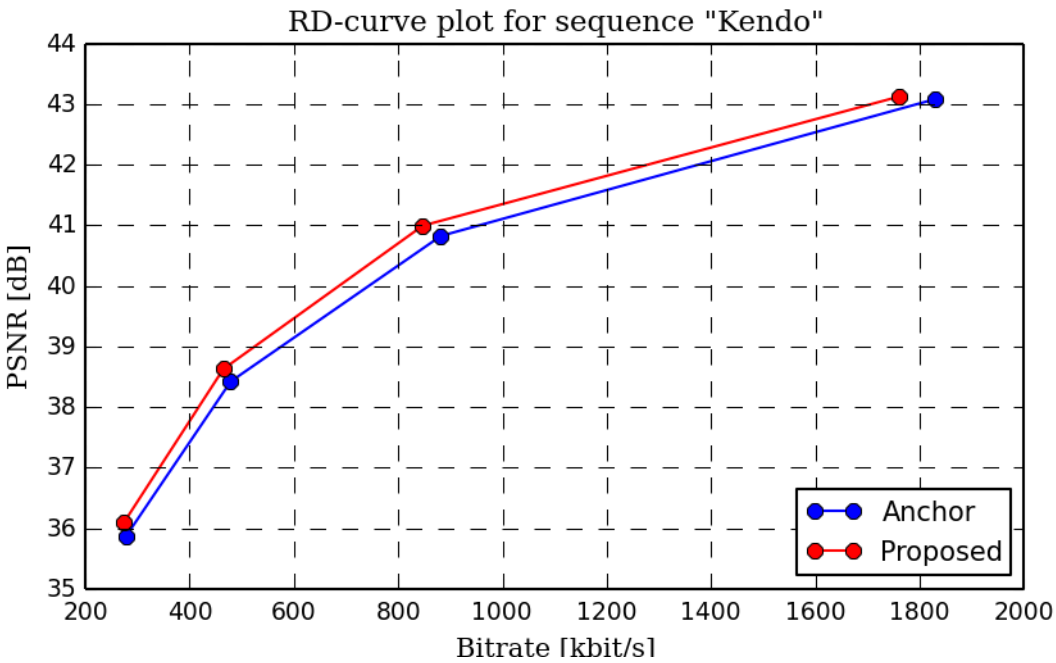
## B.1 RD curve plots for average virtual view quality versus overall sequence bitrate for the 3D-HEVC codec.

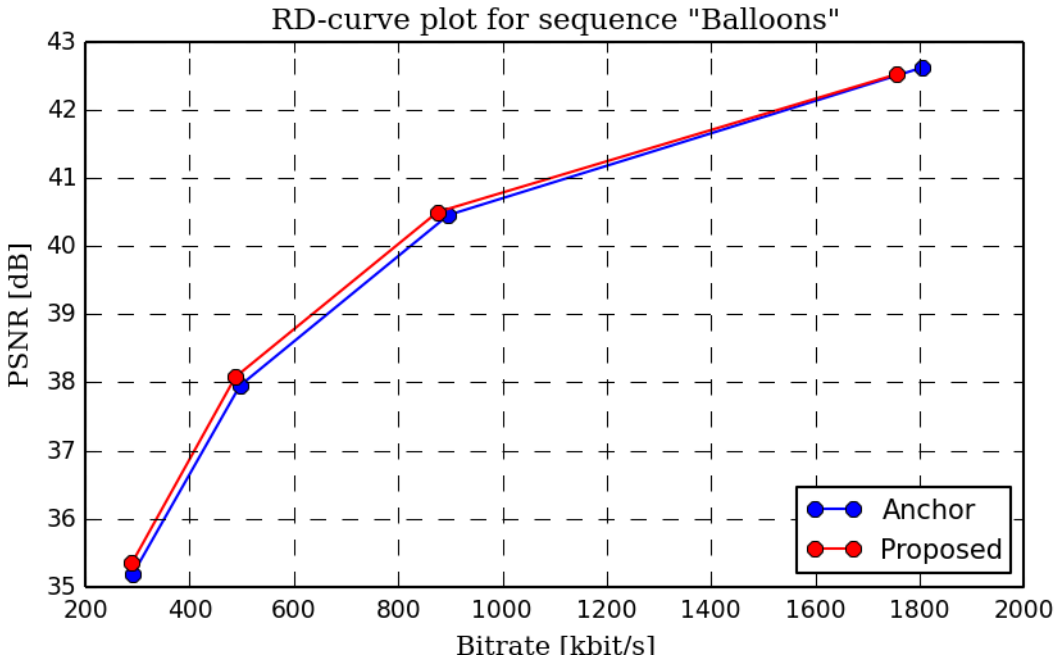


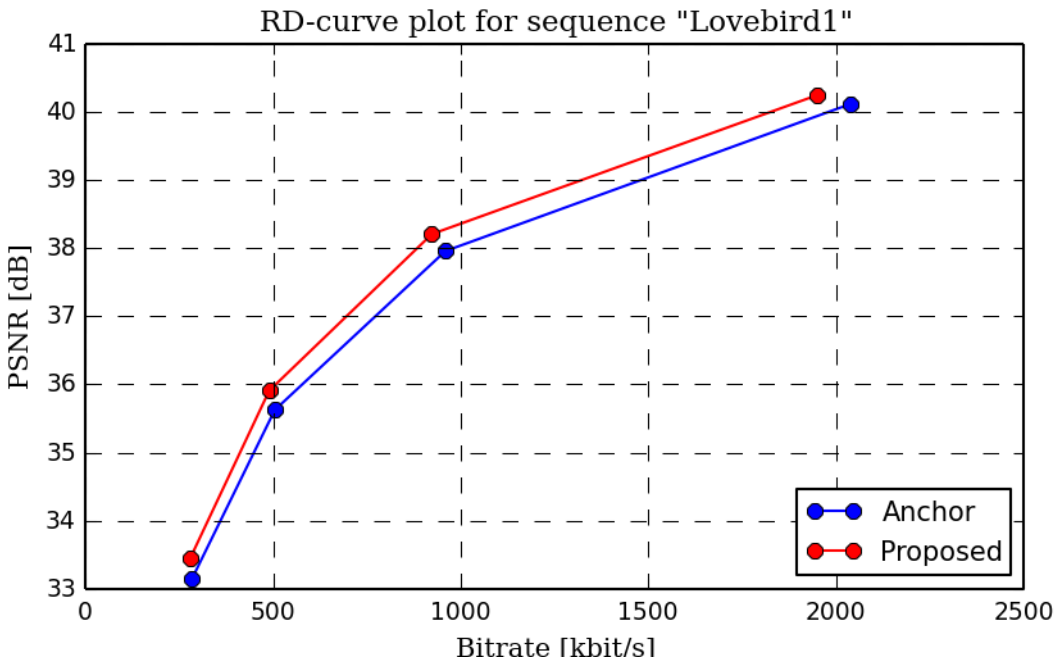




RD-curve plot for sequence "Undo_Dancer"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

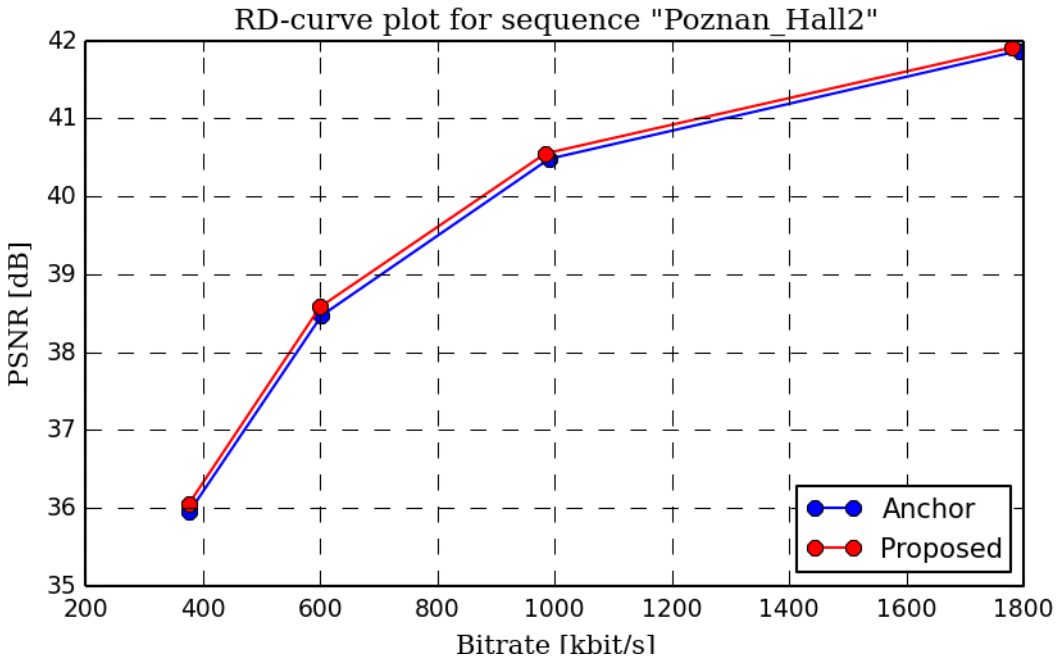

RD-curve plot for sequence "GT_Fly"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

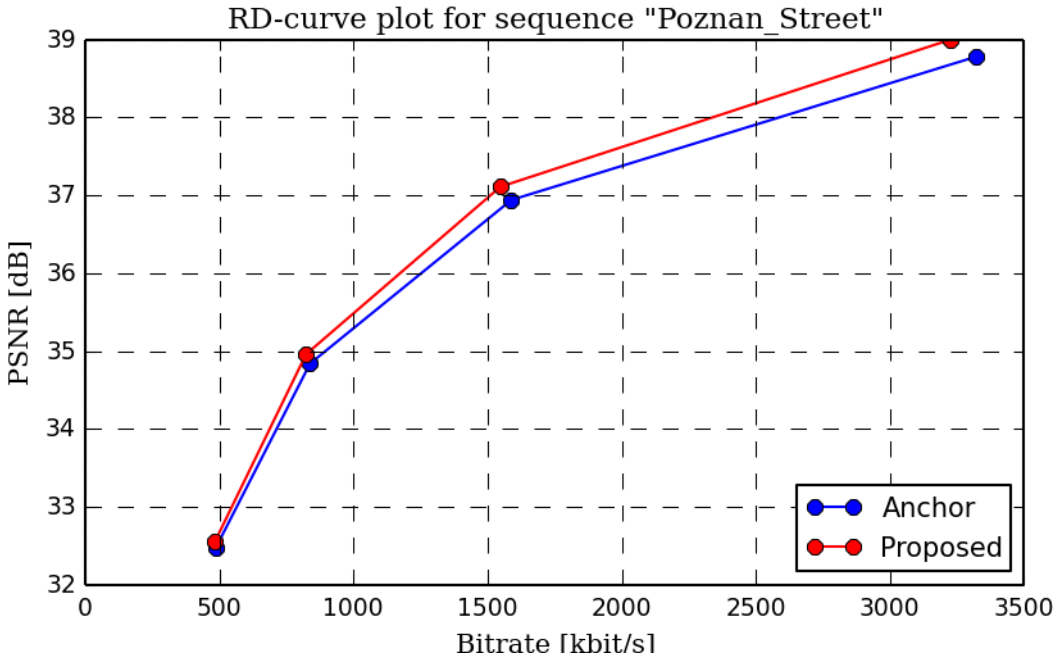

RD-curve plot for sequence "Kendo"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

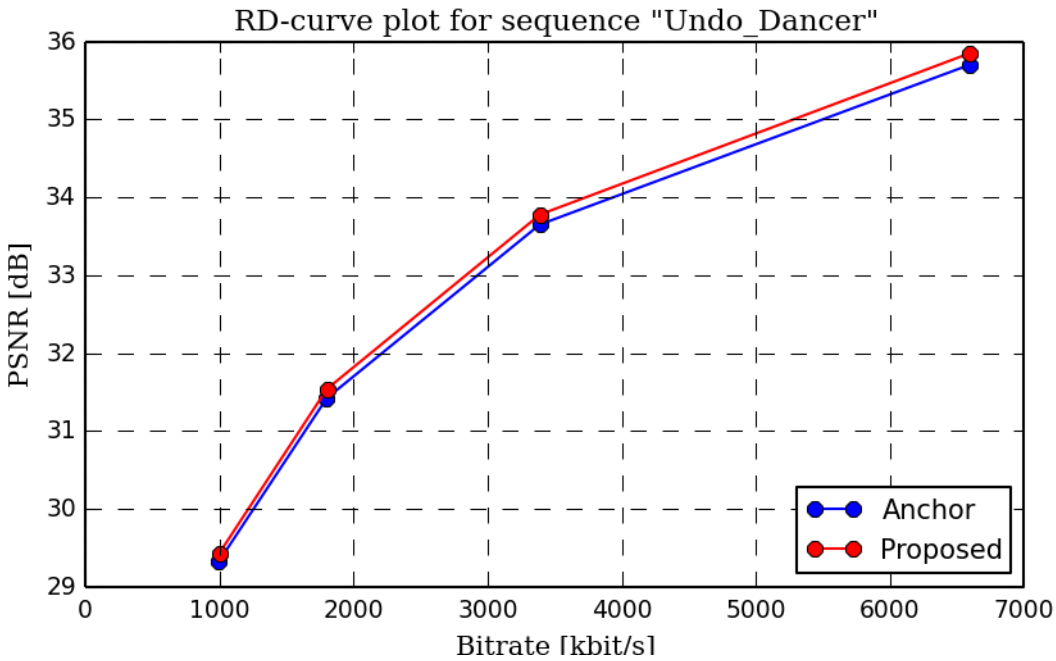

RD-curve plot for sequence "Balloons"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

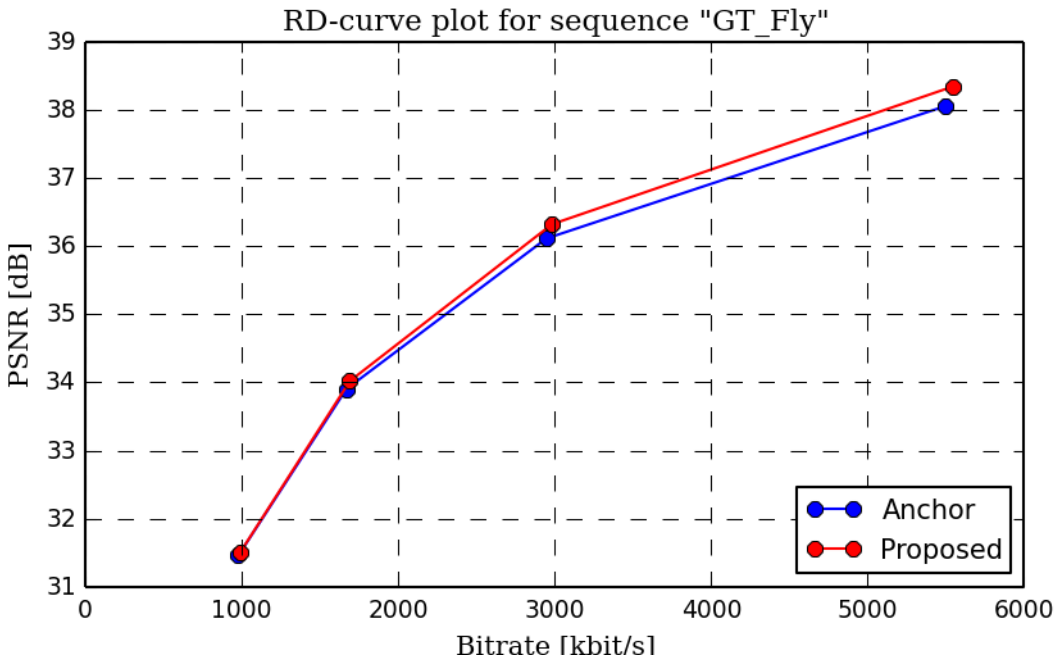

RD-curve plot for sequence "Lovebird1"
PSNR [dB]
Bitrate [kbit/s]
41
40
39
38
37
36
35
34
33
0
500
1000
1500
2000
2500
Anchor
Proposed

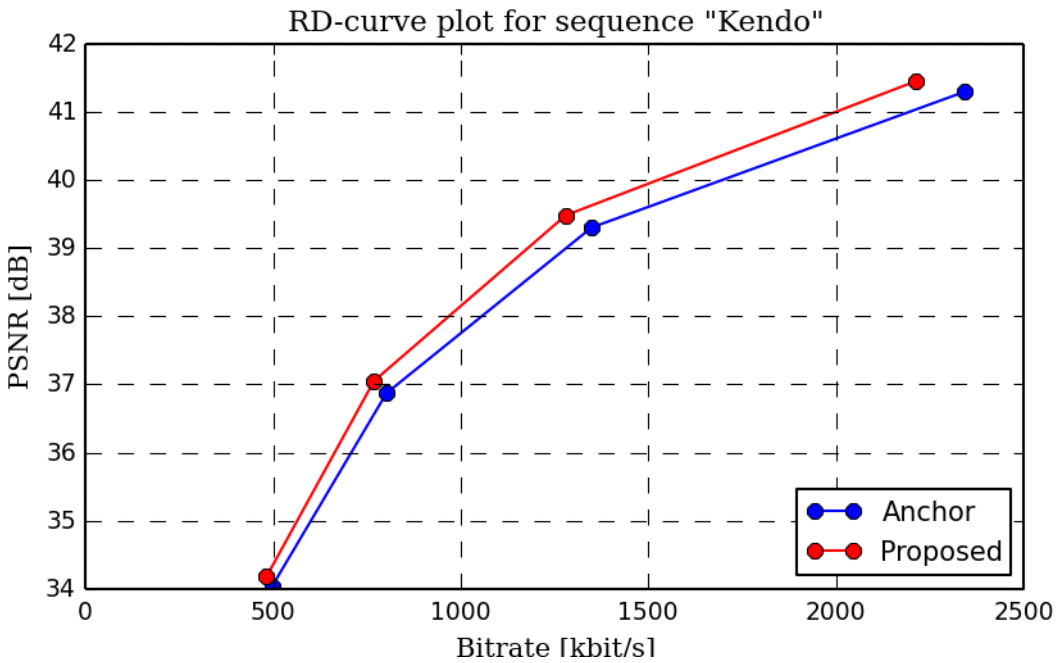
**B.2 RD curve plots for average virtual view quality versus overall sequence bitstream rate for the MVC+D codec.**

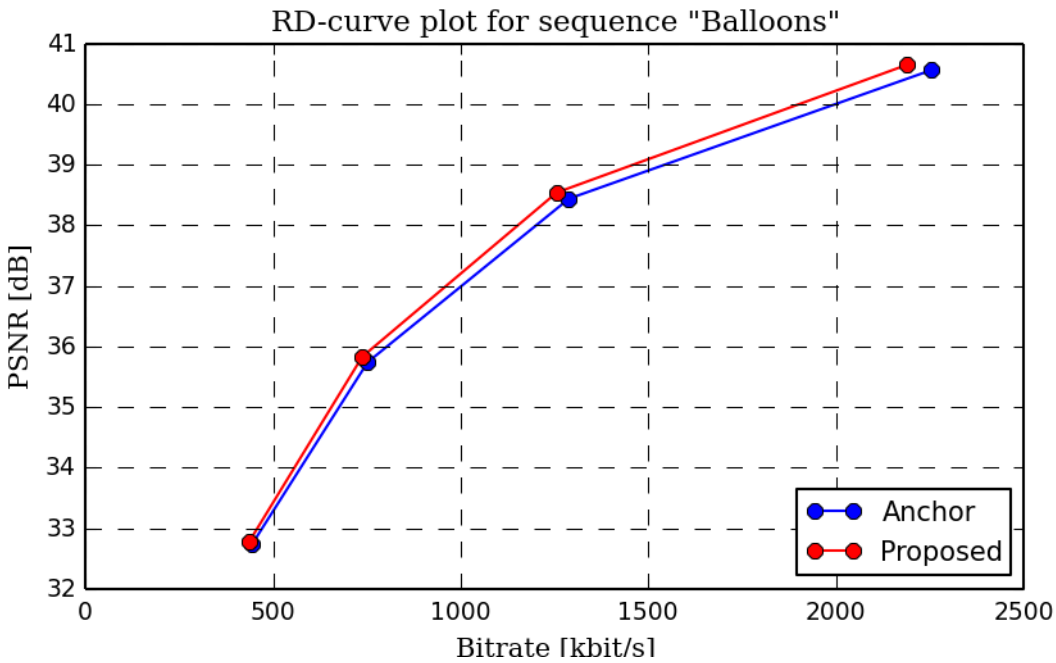


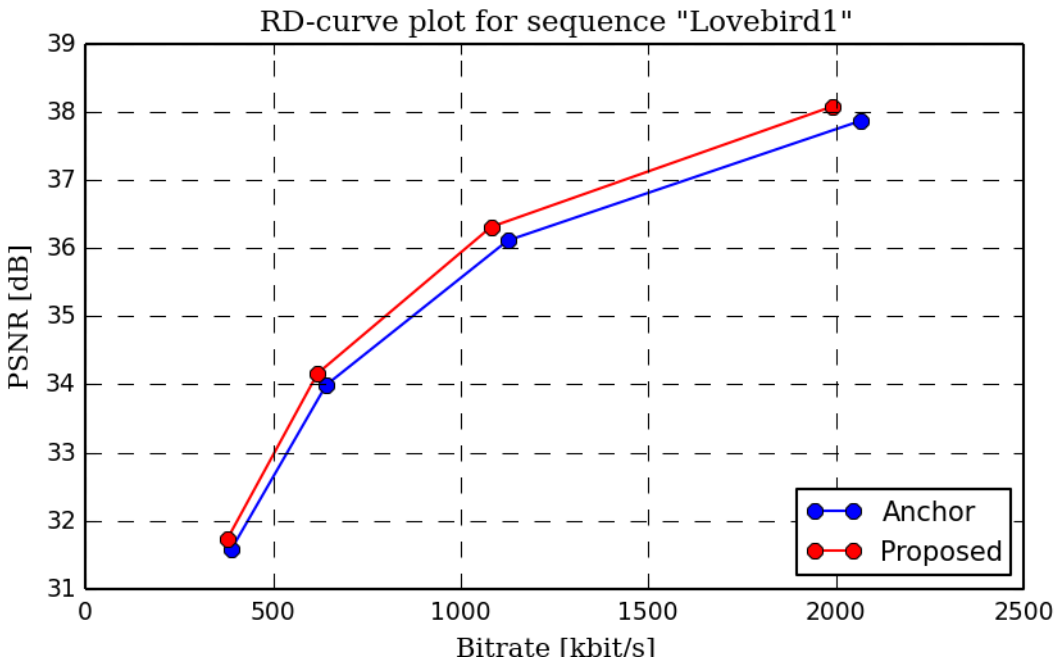




RD-curve plot for sequence "Undo_Dancer"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

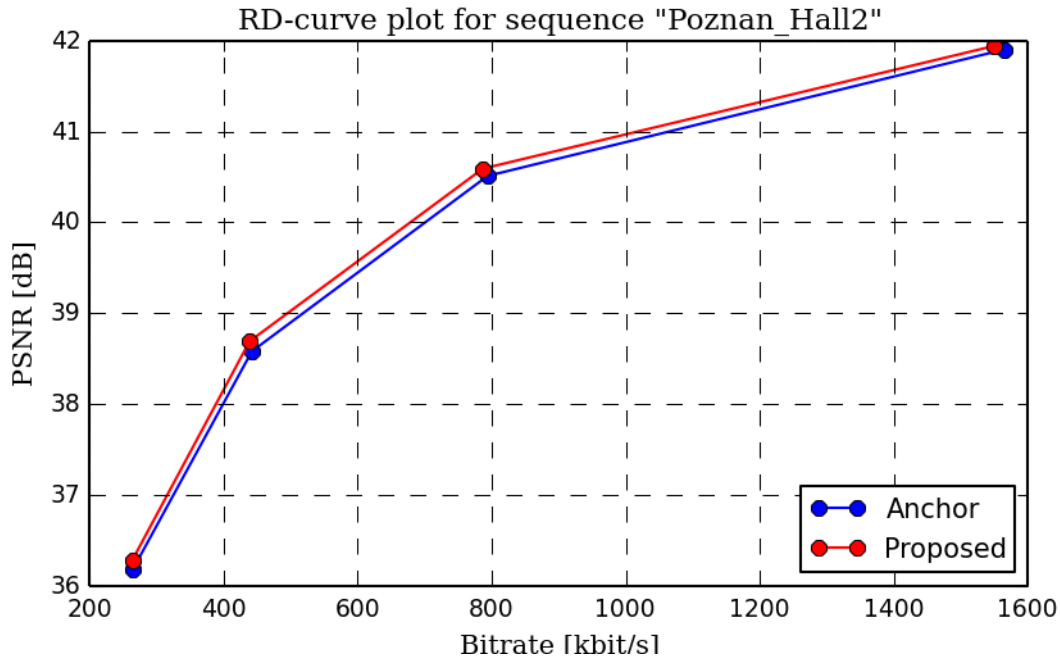

RD-curve plot for sequence "GT_Fly"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

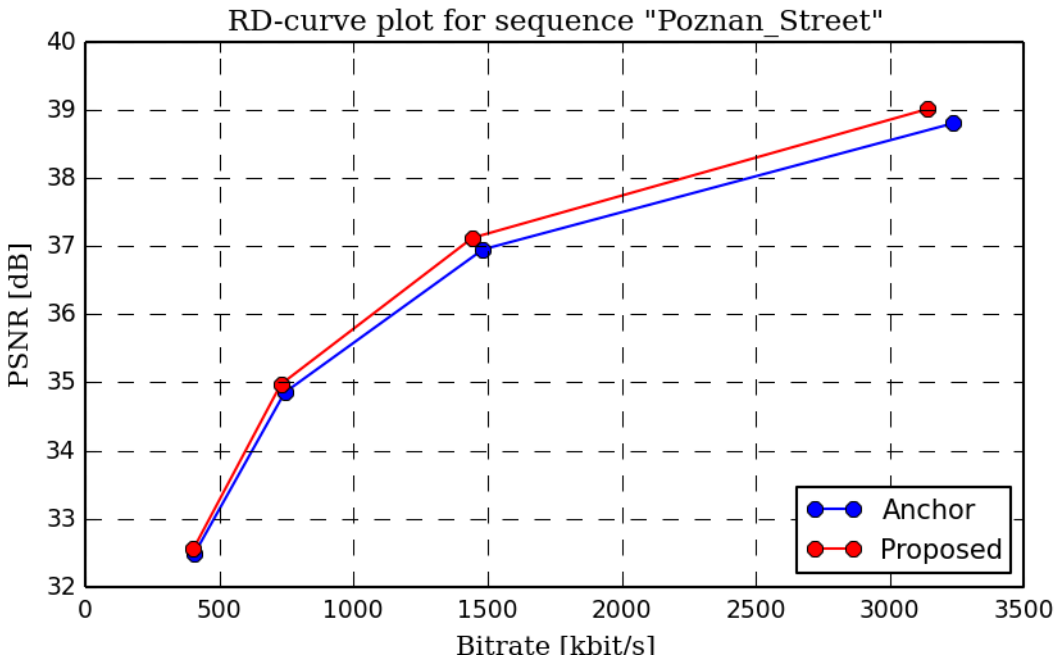

RD-curve plot for sequence "Kendo"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

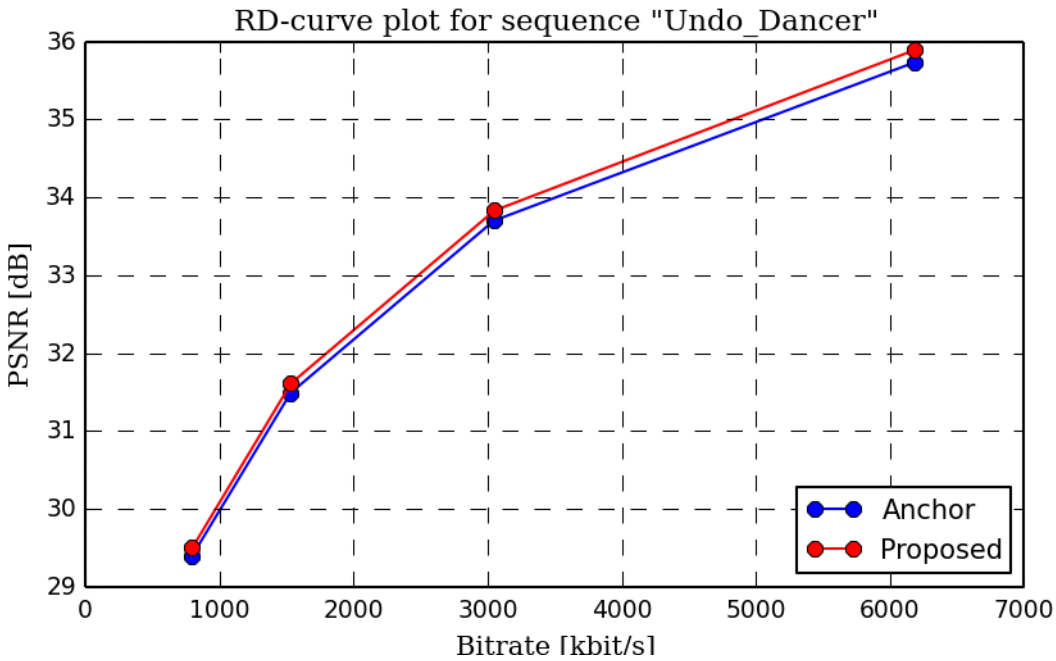

RD-curve plot for sequence "Balloons"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed

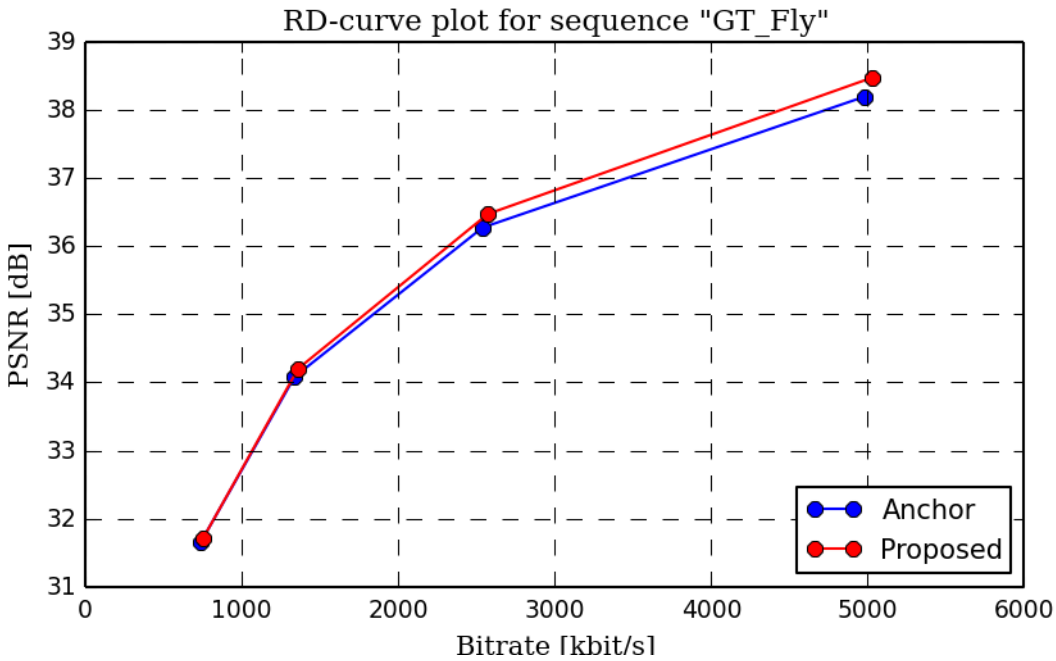

RD-curve plot for sequence "Lovebird1"
PSNR [dB]
Bitrate [kbit/s]
39
38
37
36
35
34
33
32
31
0
500
1000
1500
2000
2500
Anchor
Proposed

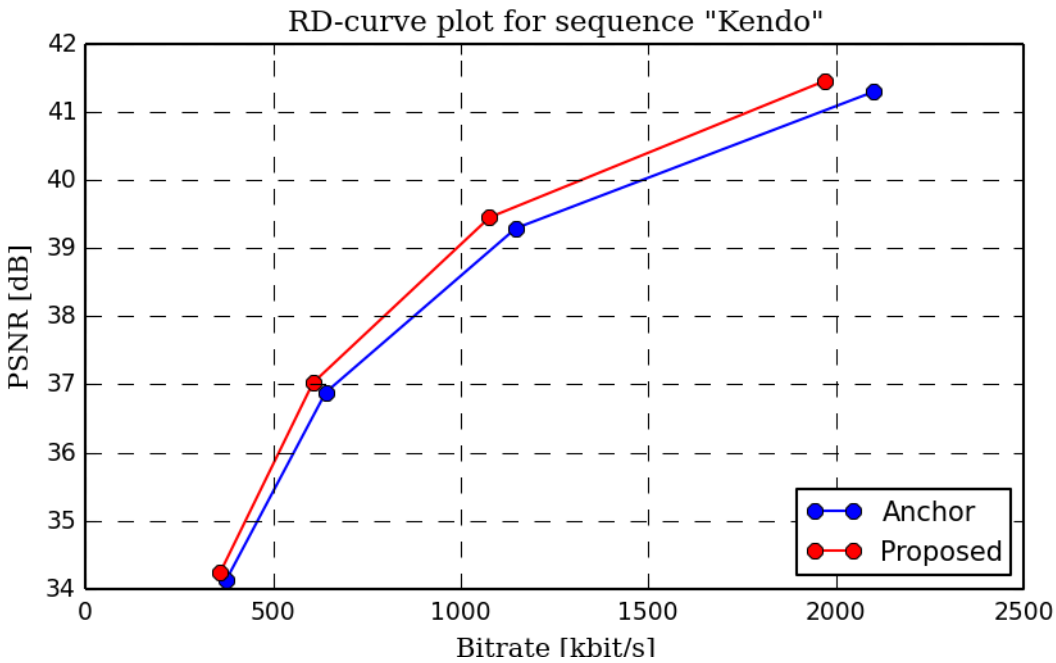
**B.3 RD curve plots for average virtual view quality versus overall sequence bitstream rate for the 3D-AVC codec.**

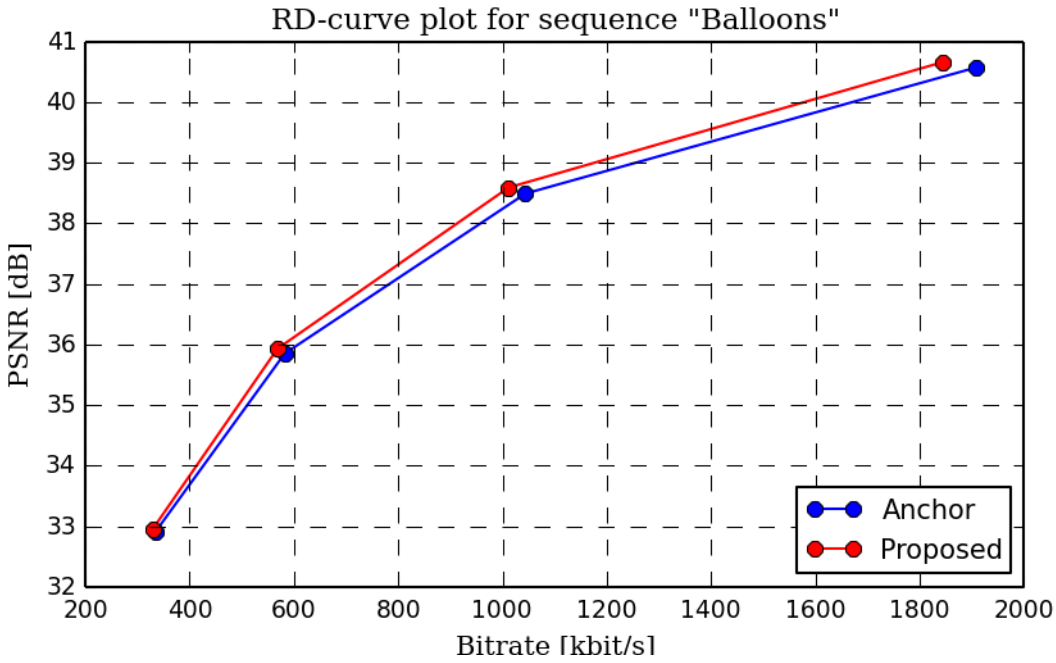


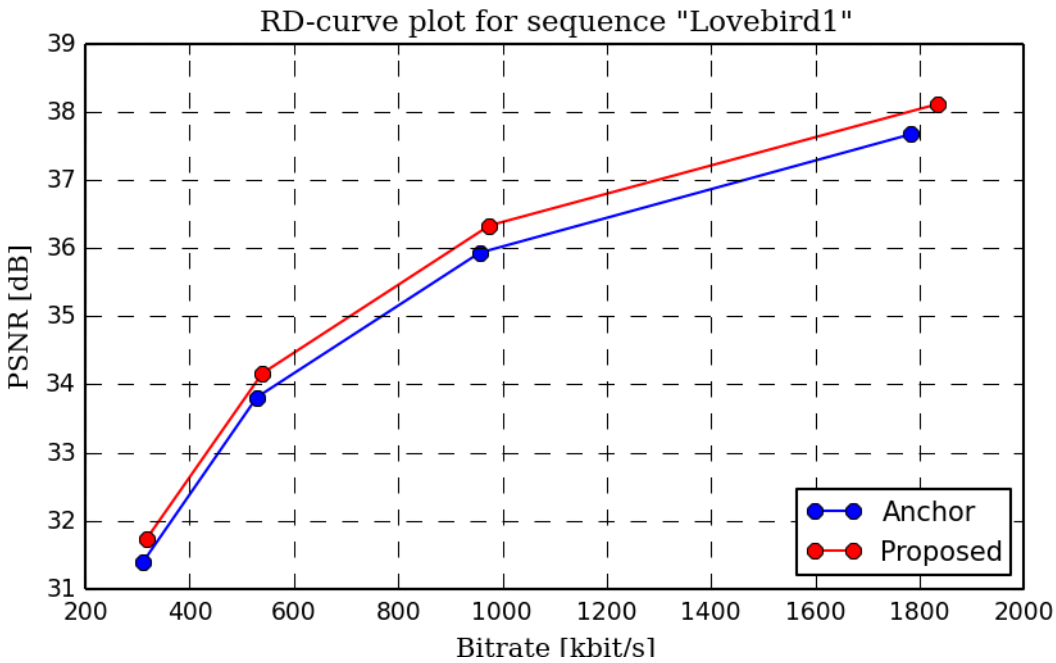




RD-curve plot for sequence "Undo_Dancer"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed
36
35
34
33
32
31
30
29
0
1000
2000
3000
4000
5000
6000
7000


RD-curve plot for sequence "GT_Fly"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed
39
38
37
36
35
34
33
32
31
0
1000
2000
3000
4000
5000
6000


RD-curve plot for sequence "Kendo"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed
42
41
40
39
38
37
36
35
34
0
500
1000
1500
2000
2500


RD-curve plot for sequence "Balloons"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed
41
40
39
38
37
36
35
34
33
32
200
400
600
800
1000
1200
1400
1600
1800
2000


RD-curve plot for sequence "Lovebird1"
PSNR [dB]
Bitrate [kbit/s]
Anchor
Proposed
39
38
37
36
35
34
33
32
31
200
400
600
800
1000
1200
1400
1600
1800
2000

# Annex C – Sample video frames and ToF depth maps used to evaluate the algorithms proposed by the author.

Sample video frames and depth maps captured by ToF cameras that come from the test data set created by the author. Each of the presented images was rectified in order to remove lens distortion and camera misalignment. No further processing was applied.

The presented video frames and depth maps were used to evaluate the video and distance data fusion algorithm that was described in detail in chapter 5.

## C.1 Data from the 3T+2D camera system

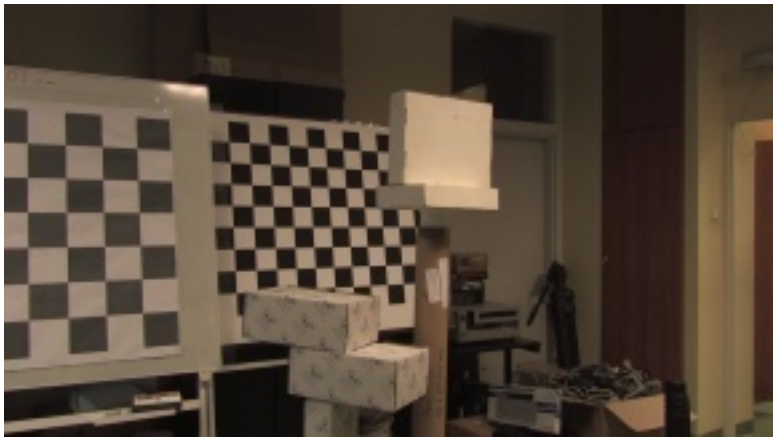


Figure C.1.1. – Image for camera 02 of the “Boards_01” test data set.

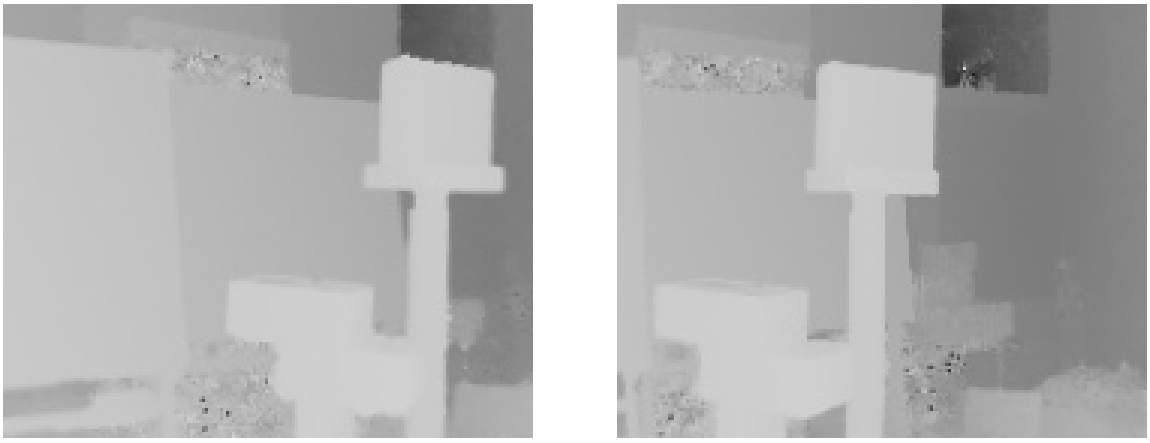


Figure C.1.2. – Depth maps from ToF cameras for the “Boards_01” test data set.

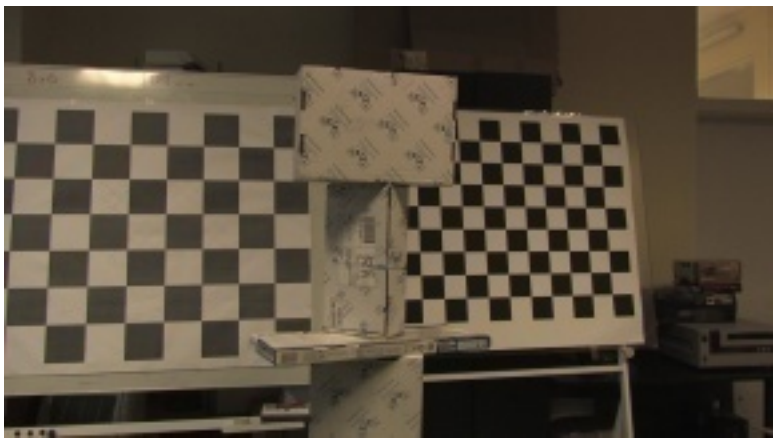


Figure C.1.3. – Image for camera 02 of the “Boards_02” test data set.

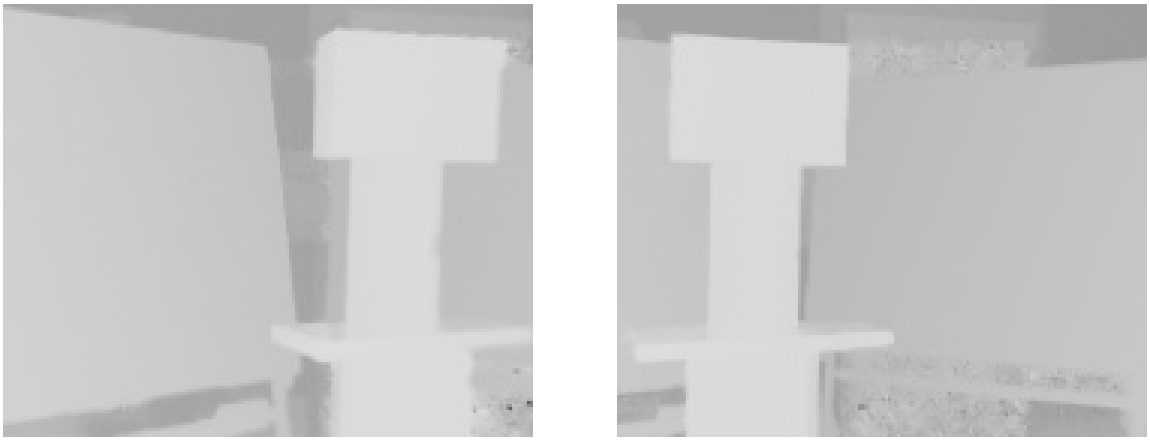


Figure C.1.4. – Depth maps from ToF cameras for the “Boards_02” test data set.

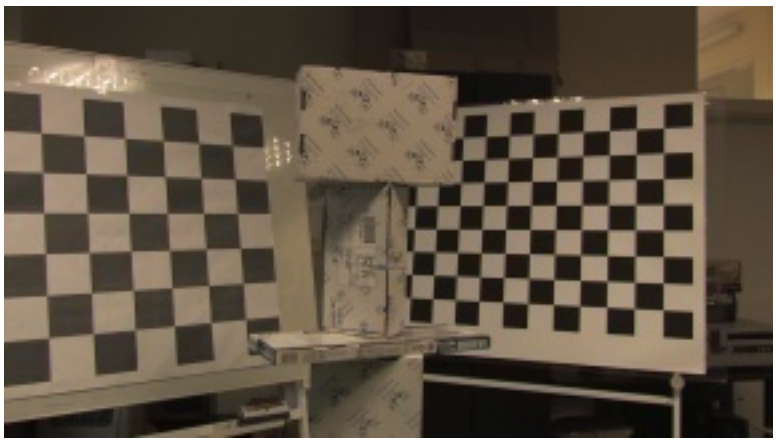


Figure C.1.5. – Image for camera 02 of the “Boards_03” test data set.

Figure C.1.6. – Depth maps from ToF cameras for the “Boards_03” test data set.

## C.2 Data from the 5T+2D camera system

Figure C.2.1. – Image for camera 03 of the “Office” test data set.

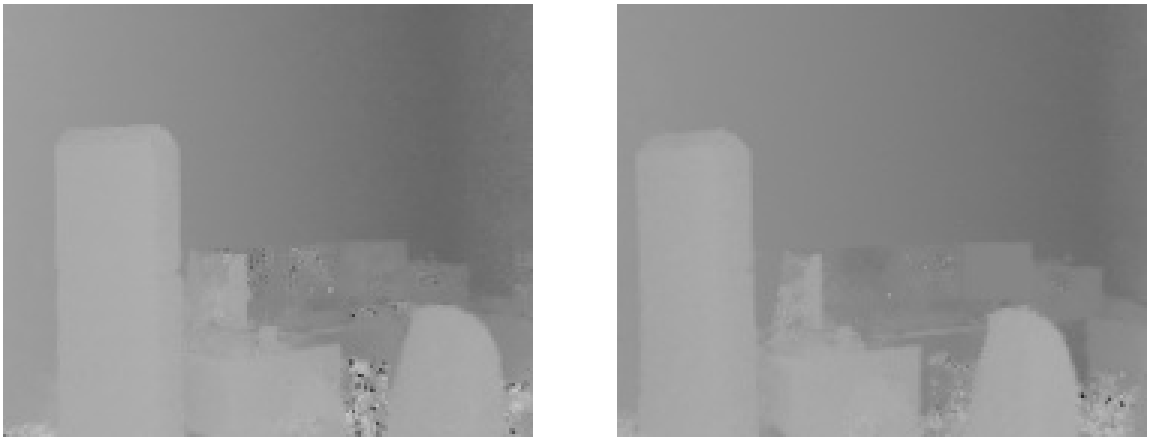


Figure C.2.2. – Depth maps from ToF cameras for the “Office” test data set.

## C.3 Data from the 6T+3D camera system

Figure C.3.1. – Image for camera 03 of the “Flower” test data set.

Figure C.3.2. – Depth maps from ToF cameras for the “Flower” test data set.

# Annex D – View synthesis quality and results of comparison between estimated depth maps and ground-truth depth maps

This Annex contains detailed virtual view PSNR values and bad-pixel percentage ratios for experiments described in chapter 5.

Sub-section D.1 contains the detailed results of virtual view synthesis for the Middlebury data set. Tables D.1.1 to D.1.4 provide the virtual view luminance PSNR values for each of the test cases. The average PSNR value taken over the whole data set is also shown in the last row of each table.

Sub-section D.2 presents the bad-pixel percentage ratio. Tables D.2.1 to D.2.4 show the results for case 1 for individual pixel classes: all pixels, textureless pixels, textured pixels and pixels near depth discontinuities. Tables D.2.5-D.2.8, D.2.9-D.2.11 and D.2.12-D.2.16 show the results for experiment cases 2,3 and 4, respectively. The last row of each table contains the average value computed over all test multi-view images.

## D.1 View synthesis quality for estimated depth maps

Table D.1.1. – Virtual view PSNR for depth maps estimated using the proposed algorithm, experiment case 1. Original cross-cost function, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 32.34 | 30.80 | 30.51 | 30.18 | 29.84 | 29.71 | 29.89 | 30.71 | 30.31 | 30.72 | 30.17 | 30.52 | 30.44 |
| Art | 29.64 | 28.64 | 28.93 | 29.17 | 28.64 | 28.90 | 29.03 | 29.01 | 29.19 | 29.33 | 29.00 | 29.21 | 29.28 |
| Baby1 | 39.08 | 38.53 | 38.45 | 38.28 | 36.56 | 36.60 | 36.66 | 38.32 | 38.30 | 38.30 | 36.94 | 38.38 | 38.36 |
| Baby2 | 38.97 | 38.85 | 38.84 | 38.64 | 37.96 | 38.25 | 38.32 | 38.85 | 38.87 | 38.79 | 38.40 | 38.96 | 38.84 |
| Baby3 | 40.51 | 39.85 | 40.04 | 40.20 | 37.84 | 38.03 | 38.28 | 39.83 | 40.02 | 40.29 | 38.53 | 40.10 | 40.19 |
| Books | 33.86 | 33.65 | 34.02 | 33.88 | 32.25 | 32.64 | 33.03 | 33.75 | 34.11 | 34.18 | 33.30 | 34.05 | 34.17 |
| Bowling1 | 34.22 | 35.71 | 35.78 | 33.92 | 33.87 | 33.92 | 34.06 | 33.86 | 34.19 | 34.29 | 33.88 | 34.17 | 34.33 |
| Bowling2 | 37.09 | 37.12 | 37.23 | 37.33 | 36.26 | 36.42 | 36.54 | 37.08 | 37.21 | 37.45 | 36.74 | 37.13 | 37.24 |
| Cloth1 | 39.20 | 39.08 | 39.17 | 39.17 | 36.83 | 36.87 | 36.95 | 39.12 | 39.17 | 39.17 | 37.28 | 39.16 | 39.17 |
| Cloth2 | 39.36 | 38.33 | 38.75 | 39.06 | 37.27 | 37.57 | 37.81 | 38.49 | 38.91 | 39.03 | 37.90 | 38.76 | 38.96 |
| Cloth3 | 36.18 | 35.96 | 36.15 | 36.17 | 33.81 | 33.97 | 34.12 | 36.04 | 36.18 | 36.22 | 34.87 | 36.13 | 36.20 |
| Cloth4 | 35.40 | 34.62 | 35.05 | 35.01 | 33.39 | 33.79 | 33.91 | 34.91 | 35.07 | 35.06 | 34.22 | 35.05 | 35.10 |
| Dolls | 33.80 | 32.77 | 33.26 | 33.54 | 31.80 | 32.39 | 32.70 | 33.05 | 33.53 | 33.73 | 32.80 | 33.41 | 33.62 |
| Flowerpots | 36.12 | 35.86 | 36.01 | 36.38 | 35.77 | 36.39 | 36.51 | 35.99 | 36.77 | 36.84 | 36.00 | 36.58 | 36.80 |
| Lampshade1 | 39.65 | 38.87 | 39.15 | 38.81 | 38.34 | 38.53 | 38.57 | 39.00 | 38.95 | 38.91 | 38.69 | 39.04 | 39.10 |
| Lampshade2 | 38.98 | 38.46 | 38.62 | 38.41 | 37.64 | 37.68 | 37.84 | 38.23 | 38.40 | 38.54 | 37.94 | 38.50 | 38.60 |
| Laundry | 33.87 | 32.29 | 32.92 | 33.39 | 31.47 | 32.01 | 32.50 | 32.64 | 33.32 | 33.54 | 32.50 | 33.21 | 33.48 |
| Midd1 | 34.21 | 33.88 | 33.94 | 34.02 | 33.37 | 33.64 | 33.77 | 34.02 | 33.96 | 33.96 | 33.87 | 34.00 | 33.97 |
| Midd2 | 34.36 | 34.15 | 34.18 | 34.27 | 32.37 | 33.01 | 33.36 | 34.02 | 34.27 | 34.24 | 33.79 | 34.24 | 34.28 |
| Moebius | 34.56 | 32.81 | 33.55 | 34.25 | 31.67 | 32.25 | 32.79 | 33.25 | 34.16 | 34.34 | 32.77 | 33.94 | 34.27 |
| Monopoly | 32.81 | 32.70 | 32.71 | 32.78 | 30.81 | 31.70 | 32.02 | 32.56 | 32.87 | 32.84 | 32.27 | 32.80 | 32.90 |
| Plastic | 42.64 | 42.24 | 42.48 | 42.07 | 40.74 | 40.86 | 41.11 | 41.87 | 41.88 | 41.88 | 41.32 | 42.04 | 41.74 |
| Reindeer | 30.16 | 30.41 | 30.72 | 31.04 | 29.78 | 30.03 | 30.29 | 30.50 | 31.04 | 31.07 | 30.36 | 30.83 | 31.05 |
| Rocks1 | 38.78 | 38.09 | 38.39 | 38.55 | 37.67 | 37.83 | 37.93 | 38.18 | 38.47 | 38.55 | 38.01 | 38.39 | 38.50 |
| Rocks2 | 40.53 | 40.28 | 40.44 | 40.49 | 39.36 | 39.49 | 39.58 | 40.27 | 40.39 | 40.43 | 39.57 | 40.37 | 40.44 |
| Wood1 | 40.09 | 40.49 | 40.67 | 40.71 | 40.03 | 40.03 | 40.05 | 40.49 | 40.71 | 40.75 | 40.08 | 40.69 | 40.72 |
| Wood2 | 39.38 | 38.71 | 38.77 | 38.82 | 38.26 | 38.40 | 38.50 | 38.60 | 38.79 | 38.81 | 38.46 | 38.76 | 38.80 |
| **Average** | **36.51** | **36.04** | **36.25** | **36.24** | **34.95** | **35.22** | **35.41** | **36.02** | **36.26** | **36.34** | **35.54** | **36.24** | **36.32** |

Table D.1.2. – Virtual view PSNR for depth maps estimated using the proposed algorithm, experiment case 2. Original cross-cost function, weighting mode according to formula 5.4.10.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 32.34 | 32.19 | 31.79 | 31.72 | 32.16 | 31.72 | 31.69 | 31.68 | 31.73 | 31.57 | 31.77 | 31.69 | 31.59 |
| Art | 29.64 | 29.75 | 29.74 | 29.71 | 29.74 | 29.73 | 29.71 | 29.71 | 29.74 | 29.63 | 29.73 | 29.72 | 29.71 |
| Baby1 | 39.08 | 39.13 | 39.11 | 39.10 | 39.11 | 39.11 | 39.10 | 39.10 | 39.10 | 39.08 | 39.10 | 39.11 | 39.08 |
| Baby2 | 38.97 | 38.72 | 38.89 | 38.78 | 38.91 | 38.75 | 38.81 | 38.81 | 38.87 | 38.81 | 38.87 | 38.78 | 38.77 |
| Baby3 | 40.51 | 40.38 | 40.36 | 40.37 | 40.36 | 40.36 | 40.37 | 40.34 | 40.36 | 40.34 | 40.36 | 40.36 | 40.34 |
| Books | 33.86 | 33.85 | 33.84 | 33.82 | 33.85 | 33.85 | 33.84 | 33.80 | 33.95 | 33.81 | 33.95 | 33.84 | 33.80 |
| Bowling1 | 34.22 | 35.73 | 35.77 | 35.77 | 35.76 | 35.77 | 35.82 | 35.81 | 35.77 | 35.80 | 35.77 | 35.82 | 35.81 |
| Bowling2 | 37.09 | 36.93 | 36.91 | 36.98 | 36.85 | 36.97 | 37.11 | 36.97 | 36.99 | 37.06 | 36.91 | 36.96 | 37.12 |
| Cloth1 | 39.20 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 | 39.17 |
| Cloth2 | 39.36 | 39.31 | 39.31 | 39.32 | 39.31 | 39.31 | 39.32 | 39.32 | 39.31 | 39.33 | 39.32 | 39.33 | 39.31 |
| Cloth3 | 36.18 | 36.21 | 36.21 | 36.20 | 36.22 | 36.20 | 36.21 | 36.20 | 36.21 | 36.19 | 36.21 | 36.21 | 36.20 |
| Cloth4 | 35.40 | 35.39 | 35.40 | 35.38 | 35.40 | 35.38 | 35.38 | 35.37 | 35.38 | 35.39 | 35.39 | 35.38 | 35.39 |
| Dolls | 33.80 | 33.76 | 33.76 | 33.75 | 33.77 | 33.76 | 33.76 | 33.75 | 33.76 | 33.75 | 33.76 | 33.75 | 33.75 |
| Flowerpots | 36.12 | 35.76 | 35.76 | 35.77 | 35.76 | 35.77 | 35.77 | 35.77 | 35.77 | 35.77 | 35.77 | 35.77 | 35.77 |
| Lampshade1 | 39.65 | 39.93 | 39.93 | 39.93 | 39.96 | 39.99 | 39.89 | 39.94 | 39.94 | 39.71 | 39.94 | 39.90 | 39.92 |
| Lampshade2 | 38.98 | 39.21 | 39.23 | 39.18 | 39.23 | 39.18 | 39.08 | 39.12 | 39.24 | 39.04 | 39.24 | 39.13 | 39.08 |
| Laundry | 33.87 | 33.85 | 33.86 | 33.80 | 33.86 | 33.85 | 33.79 | 33.78 | 33.82 | 33.41 | 33.82 | 33.80 | 33.48 |
| Midd1 | 34.21 | 34.19 | 34.18 | 34.19 | 34.18 | 34.17 | 34.18 | 34.20 | 34.17 | 34.20 | 34.18 | 34.19 | 34.20 |
| Midd2 | 34.36 | 34.40 | 34.33 | 34.36 | 34.33 | 34.36 | 34.36 | 34.37 | 34.33 | 34.36 | 34.33 | 34.36 | 34.36 |
| Moebius | 34.56 | 34.48 | 34.42 | 34.39 | 34.43 | 34.38 | 34.44 | 34.43 | 34.38 | 34.44 | 34.37 | 34.43 | 34.44 |
| Monopoly | 32.81 | 32.95 | 32.80 | 32.81 | 32.90 | 32.84 | 32.84 | 32.81 | 32.83 | 32.81 | 32.84 | 32.82 | 32.81 |
| Plastic | 42.64 | 42.60 | 42.59 | 42.57 | 42.60 | 42.59 | 42.56 | 42.59 | 42.58 | 42.57 | 42.59 | 42.58 | 42.58 |
| Reindeer | 30.16 | 30.31 | 30.31 | 30.31 | 30.32 | 30.28 | 30.28 | 30.28 | 30.31 | 30.28 | 30.31 | 30.28 | 30.28 |
| Rocks1 | 38.78 | 38.78 | 38.78 | 38.77 | 38.78 | 38.77 | 38.77 | 38.77 | 38.78 | 38.77 | 38.78 | 38.78 | 38.77 |
| Rocks2 | 40.53 | 40.50 | 40.49 | 40.50 | 40.49 | 40.47 | 40.50 | 40.49 | 40.50 | 40.50 | 40.49 | 40.50 | 40.50 |
| Wood1 | 40.09 | 39.95 | 40.05 | 40.05 | 40.06 | 40.09 | 40.10 | 40.08 | 40.05 | 40.26 | 40.05 | 40.08 | 40.08 |
| Wood2 | 39.38 | 39.38 | 39.39 | 39.38 | 39.38 | 39.39 | 39.39 | 39.39 | 39.38 | 39.39 | 39.38 | 39.38 | 39.40 |
| **Average** | **36.51** | **36.55** | **36.53** | **36.52** | **36.55** | **36.53** | **36.53** | **36.52** | **36.53** | **36.50** | **36.53** | **36.52** | **36.51** |

Table D.1.3. – Virtual view PSNR for depth maps estimated using the proposed algorithm, experiment case 3. Modified cross-cost function that takes image edges into account, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 33.47 | 31.56 | 30.93 | 30.81 | 30.23 | 30.20 | 30.73 | 31.14 | 30.85 | 31.69 | 30.67 | 31.08 | 31.29 |
| Art | 29.68 | 28.80 | 29.15 | 29.27 | 28.73 | 28.86 | 29.04 | 28.94 | 29.34 | 29.53 | 28.88 | 29.18 | 29.44 |
| Baby1 | 39.24 | 38.66 | 38.53 | 38.37 | 36.66 | 36.66 | 36.73 | 38.38 | 38.55 | 38.40 | 37.02 | 38.59 | 38.49 |
| Baby2 | 38.90 | 38.83 | 38.79 | 38.76 | 37.93 | 38.21 | 38.29 | 38.82 | 38.84 | 38.74 | 38.38 | 38.89 | 38.86 |
| Baby3 | 40.56 | 39.79 | 40.03 | 40.24 | 37.86 | 38.12 | 38.31 | 39.89 | 40.17 | 40.29 | 38.54 | 40.21 | 40.26 |
| Books | 33.82 | 33.61 | 33.92 | 33.78 | 32.22 | 32.59 | 33.01 | 33.62 | 34.05 | 34.16 | 33.26 | 33.94 | 34.11 |
| Bowling1 | 33.44 | 35.32 | 35.06 | 33.44 | 33.30 | 33.42 | 33.48 | 33.49 | 33.74 | 33.69 | 33.60 | 33.68 | 33.68 |
| Bowling2 | 37.24 | 37.17 | 37.32 | 37.31 | 36.26 | 36.45 | 36.52 | 37.08 | 37.19 | 37.34 | 36.73 | 37.12 | 37.17 |
| Cloth1 | 39.22 | 39.09 | 39.18 | 39.18 | 36.84 | 36.88 | 36.96 | 39.13 | 39.18 | 39.18 | 37.29 | 39.18 | 39.19 |
| Cloth2 | 39.44 | 38.40 | 38.80 | 39.17 | 37.32 | 37.63 | 37.85 | 38.55 | 38.96 | 39.13 | 37.96 | 38.79 | 39.05 |
| Cloth3 | 36.16 | 35.80 | 35.97 | 36.00 | 33.71 | 33.87 | 34.02 | 35.87 | 36.01 | 36.03 | 34.74 | 35.98 | 36.02 |
| Cloth4 | 35.43 | 34.65 | 35.08 | 35.16 | 33.43 | 33.80 | 33.97 | 34.94 | 35.16 | 35.04 | 34.25 | 35.18 | 35.17 |
| Dolls | 33.90 | 32.89 | 33.45 | 33.61 | 31.90 | 32.44 | 32.75 | 33.11 | 33.55 | 33.81 | 32.86 | 33.45 | 33.69 |
| Flowerpots | 36.36 | 36.15 | 36.28 | 36.44 | 36.08 | 36.23 | 36.29 | 36.28 | 36.26 | 36.34 | 36.27 | 36.37 | 36.27 |
| Lampshade1 | 39.57 | 38.73 | 38.98 | 38.84 | 38.23 | 38.43 | 38.57 | 38.93 | 38.86 | 38.72 | 38.59 | 38.94 | 38.88 |
| Lampshade2 | 39.28 | 38.62 | 38.76 | 38.85 | 37.80 | 37.88 | 38.05 | 38.45 | 38.58 | 38.64 | 38.13 | 38.78 | 38.78 |
| Laundry | 33.92 | 32.27 | 32.91 | 33.56 | 31.47 | 32.05 | 32.51 | 32.61 | 33.41 | 33.67 | 32.45 | 33.28 | 33.60 |
| Midd1 | 34.02 | 33.78 | 33.91 | 33.95 | 33.31 | 33.56 | 33.68 | 33.90 | 33.88 | 33.89 | 33.78 | 33.93 | 33.90 |
| Midd2 | 34.40 | 34.07 | 34.14 | 34.23 | 32.34 | 32.99 | 33.31 | 33.97 | 34.30 | 34.23 | 33.74 | 34.21 | 34.28 |
| Moebius | 34.60 | 32.85 | 33.55 | 34.22 | 31.68 | 32.22 | 32.75 | 33.31 | 34.09 | 34.30 | 32.79 | 33.90 | 34.22 |
| Monopoly | 32.86 | 32.77 | 32.76 | 32.86 | 30.85 | 31.72 | 32.04 | 32.61 | 32.91 | 32.93 | 32.31 | 32.80 | 32.94 |
| Plastic | 42.65 | 42.27 | 42.48 | 42.11 | 40.76 | 40.86 | 41.11 | 41.88 | 41.89 | 41.89 | 41.32 | 42.06 | 41.74 |
| Reindeer | 30.53 | 30.33 | 30.65 | 30.94 | 29.69 | 29.99 | 30.22 | 30.39 | 31.03 | 31.13 | 30.35 | 30.78 | 31.03 |
| Rocks1 | 38.53 | 37.82 | 38.00 | 38.19 | 37.38 | 37.54 | 37.67 | 37.84 | 38.08 | 38.24 | 37.70 | 38.01 | 38.14 |
| Rocks2 | 40.56 | 40.27 | 40.47 | 40.57 | 39.35 | 39.48 | 39.57 | 40.28 | 40.53 | 40.53 | 39.57 | 40.50 | 40.57 |
| Wood1 | 40.53 | 40.45 | 40.60 | 40.63 | 39.99 | 39.97 | 40.08 | 40.41 | 40.63 | 40.65 | 40.01 | 40.61 | 40.64 |
| Wood2 | 39.28 | 38.67 | 38.83 | 38.88 | 38.32 | 38.47 | 38.56 | 38.66 | 38.82 | 38.88 | 38.49 | 38.78 | 38.79 |
| **Average** | **36.58** | **36.06** | **36.24** | **36.27** | **34.95** | **35.20** | **35.41** | **36.02** | **36.25** | **36.34** | **35.54** | **36.23** | **36.30** |

Table D.1.4. – Virtual view PSNR for depth maps estimated using the proposed algorithm, experiment case 4. Modified cross-cost function that takes image edges into account, weighting mode according to formula 5.4.10.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 33.47 | 33.38 | 33.36 | 33.31 | 33.39 | 33.31 | 33.29 | 33.29 | 33.32 | 33.25 | 33.33 | 33.31 | 33.30 |
| Art | 29.68 | 29.69 | 29.68 | 29.69 | 29.68 | 29.67 | 29.68 | 29.67 | 29.69 | 29.69 | 29.69 | 29.67 | 29.67 |
| Baby1 | 39.24 | 39.24 | 39.23 | 39.23 | 39.23 | 39.24 | 39.22 | 39.23 | 39.24 | 39.20 | 39.24 | 39.22 | 39.21 |
| Baby2 | 38.90 | 38.77 | 38.77 | 38.83 | 38.78 | 38.84 | 38.78 | 38.81 | 38.82 | 38.75 | 38.84 | 38.81 | 38.78 |
| Baby3 | 40.56 | 40.47 | 40.43 | 40.39 | 40.44 | 40.43 | 40.47 | 40.48 | 40.42 | 40.42 | 40.43 | 40.45 | 40.48 |
| Books | 33.82 | 33.81 | 33.83 | 33.81 | 33.83 | 33.81 | 33.81 | 33.80 | 33.92 | 33.79 | 33.83 | 33.82 | 33.77 |
| Bowling1 | 33.44 | 33.94 | 33.95 | 33.94 | 33.95 | 33.95 | 33.95 | 33.94 | 33.95 | 33.94 | 33.95 | 33.95 | 33.95 |
| Bowling2 | 37.24 | 37.11 | 37.17 | 37.13 | 37.11 | 37.17 | 37.17 | 37.21 | 37.19 | 37.20 | 37.17 | 37.17 | 37.16 |
| Cloth1 | 39.22 | 39.19 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 | 39.18 |
| Cloth2 | 39.44 | 39.42 | 39.42 | 39.41 | 39.44 | 39.44 | 39.44 | 39.44 | 39.42 | 39.42 | 39.41 | 39.44 | 39.43 |
| Cloth3 | 36.16 | 36.05 | 36.04 | 36.03 | 36.05 | 36.04 | 36.05 | 36.04 | 36.04 | 36.05 | 36.04 | 36.04 | 36.06 |
| Cloth4 | 35.43 | 35.46 | 35.46 | 35.44 | 35.46 | 35.46 | 35.44 | 35.44 | 35.46 | 35.42 | 35.46 | 35.45 | 35.45 |
| Dolls | 33.90 | 33.92 | 33.87 | 33.87 | 33.88 | 33.88 | 33.87 | 33.87 | 33.87 | 33.88 | 33.87 | 33.87 | 33.87 |
| Flowerpots | 36.36 | 36.41 | 36.40 | 35.78 | 36.41 | 35.78 | 35.79 | 35.78 | 35.75 | 35.78 | 35.76 | 35.79 | 35.78 |
| Lampshade1 | 39.57 | 39.70 | 39.71 | 39.70 | 39.73 | 39.70 | 39.78 | 39.79 | 39.72 | 39.71 | 39.72 | 39.79 | 39.77 |
| Lampshade2 | 39.28 | 39.57 | 39.58 | 39.51 | 39.63 | 39.51 | 39.46 | 39.48 | 39.59 | 39.41 | 39.58 | 39.48 | 39.47 |
| Laundry | 33.92 | 33.91 | 33.90 | 33.92 | 33.90 | 33.93 | 33.93 | 33.85 | 33.91 | 33.84 | 33.91 | 33.94 | 33.85 |
| Midd1 | 34.02 | 34.05 | 34.04 | 34.04 | 34.05 | 34.02 | 34.03 | 34.04 | 34.04 | 34.04 | 34.03 | 34.04 | 34.05 |
| Midd2 | 34.40 | 34.43 | 34.37 | 34.37 | 34.37 | 34.38 | 34.39 | 34.39 | 34.38 | 34.38 | 34.38 | 34.39 | 34.39 |
| Moebius | 34.60 | 34.41 | 34.37 | 34.42 | 34.38 | 34.43 | 34.43 | 34.44 | 34.33 | 34.44 | 34.32 | 34.42 | 34.43 |
| Monopoly | 32.86 | 33.02 | 32.86 | 32.82 | 32.95 | 32.90 | 32.87 | 32.86 | 32.88 | 32.85 | 32.88 | 32.87 | 32.85 |
| Plastic | 42.65 | 42.60 | 42.62 | 42.63 | 42.63 | 42.63 | 42.62 | 42.62 | 42.64 | 42.61 | 42.62 | 42.62 | 42.63 |
| Reindeer | 30.53 | 30.71 | 30.67 | 30.70 | 30.70 | 30.72 | 30.70 | 30.71 | 30.70 | 30.63 | 30.70 | 30.71 | 30.70 |
| Rocks1 | 38.53 | 38.48 | 38.48 | 38.44 | 38.48 | 38.47 | 38.44 | 38.43 | 38.48 | 38.43 | 38.48 | 38.44 | 38.43 |
| Rocks2 | 40.56 | 40.58 | 40.56 | 40.56 | 40.57 | 40.56 | 40.56 | 40.56 | 40.56 | 40.56 | 40.56 | 40.56 | 40.56 |
| Wood1 | 40.53 | 40.59 | 40.59 | 40.59 | 40.58 | 40.60 | 40.60 | 40.58 | 40.58 | 40.59 | 40.58 | 40.60 | 40.59 |
| Wood2 | 39.28 | 39.33 | 39.33 | 39.33 | 39.33 | 39.33 | 39.34 | 39.33 | 39.33 | 39.32 | 39.33 | 39.33 | 39.33 |
| **Average** | **36.58** | **36.60** | **36.59** | **36.56** | **36.60** | **36.57** | **36.57** | **36.57** | **36.57** | **36.55** | **36.57** | **36.57** | **36.56** |

## D.2 Results of comparison between estimated depth maps and ground-truth

Table D.2.1. – Percentage of bad pixels in estimated depth maps (all pixels included), experiment case 1.

Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 5.64 | 4.93 | 5.71 | 7.20 | 5.68 | 5.62 | 5.42 | 4.97 | 4.93 | 5.79 | 5.45 | 4.78 | 5.18 |
| Art | 9.91 | 10.37 | 11.35 | 11.44 | 10.94 | 10.73 | 10.10 | 10.75 | 10.49 | 10.18 | 10.70 | 10.51 | 10.11 |
| Baby1 | 5.72 | 2.72 | 3.70 | 6.37 | 2.63 | 2.62 | 2.60 | 2.64 | 2.79 | 3.93 | 2.63 | 2.72 | 2.80 |
| Baby2 | 4.07 | 2.40 | 3.34 | 4.90 | 2.26 | 2.20 | 2.20 | 2.32 | 2.59 | 3.61 | 2.26 | 2.37 | 2.69 |
| Baby3 | 7.25 | 3.37 | 4.99 | 6.61 | 3.57 | 3.36 | 3.23 | 3.38 | 3.05 | 4.69 | 3.38 | 3.17 | 3.08 |
| Books | 9.84 | 4.58 | 7.75 | 9.27 | 4.39 | 4.39 | 4.37 | 4.54 | 6.12 | 7.53 | 4.43 | 4.66 | 6.48 |
| Bowling1 | 20.60 | 11.56 | 13.54 | 18.07 | 11.73 | 11.69 | 11.57 | 11.70 | 12.28 | 14.45 | 11.63 | 11.77 | 12.62 |
| Bowling2 | 23.99 | 3.59 | 9.55 | 21.39 | 3.21 | 3.20 | 3.19 | 3.44 | 4.38 | 10.43 | 3.25 | 3.85 | 4.81 |
| Cloth1 | 0.13 | 0.17 | 0.12 | 0.12 | 0.22 | 0.21 | 0.19 | 0.16 | 0.12 | 0.12 | 0.17 | 0.13 | 0.12 |
| Cloth2 | 2.18 | 3.24 | 3.16 | 2.68 | 3.53 | 3.37 | 3.19 | 3.20 | 2.77 | 2.45 | 3.26 | 2.82 | 2.65 |
| Cloth3 | 0.98 | 1.20 | 1.08 | 0.99 | 1.37 | 1.33 | 1.28 | 1.19 | 1.03 | 0.98 | 1.23 | 1.08 | 1.00 |
| Cloth4 | 1.20 | 1.64 | 1.47 | 1.24 | 1.86 | 1.73 | 1.65 | 1.58 | 1.28 | 1.22 | 1.60 | 1.37 | 1.26 |
| Dolls | 5.14 | 5.58 | 5.79 | 5.52 | 6.01 | 5.64 | 5.32 | 5.40 | 4.80 | 4.86 | 5.48 | 4.84 | 4.75 |
| Flowerpots | 20.63 | 4.20 | 9.53 | 19.95 | 4.16 | 4.09 | 4.05 | 4.10 | 5.62 | 10.45 | 4.07 | 4.22 | 5.85 |
| Lampshade1 | 9.88 | 5.07 | 7.71 | 10.52 | 5.26 | 5.04 | 4.94 | 5.04 | 5.56 | 7.83 | 5.02 | 4.79 | 5.75 |
| Lampshade2 | 9.76 | 3.48 | 7.47 | 16.31 | 3.60 | 3.49 | 3.35 | 3.48 | 3.28 | 7.53 | 3.50 | 3.32 | 3.24 |
| Laundry | 8.13 | 7.66 | 8.76 | 8.92 | 8.18 | 7.60 | 7.13 | 7.15 | 6.73 | 7.47 | 7.35 | 6.53 | 6.68 |
| Midd1 | 8.06 | 4.62 | 4.16 | 6.16 | 4.35 | 4.28 | 4.26 | 4.18 | 4.08 | 4.01 | 4.25 | 4.55 | 4.07 |
| Midd2 | 13.47 | 4.05 | 4.73 | 6.47 | 4.10 | 3.93 | 3.82 | 3.85 | 3.78 | 4.44 | 3.97 | 3.97 | 3.86 |
| Moebius | 9.84 | 6.61 | 8.45 | 10.66 | 7.42 | 6.99 | 6.61 | 6.43 | 6.52 | 7.54 | 6.78 | 5.80 | 6.50 |
| Monopoly | 14.93 | 3.98 | 3.56 | 3.58 | 3.90 | 3.83 | 3.79 | 3.79 | 3.44 | 3.33 | 3.85 | 3.29 | 3.01 |
| Plastic | 35.83 | 7.25 | 17.42 | 36.97 | 6.35 | 6.47 | 6.41 | 6.29 | 9.34 | 19.05 | 6.41 | 7.26 | 9.43 |
| Reindeer | 4.17 | 5.70 | 5.80 | 5.10 | 6.11 | 5.87 | 5.62 | 5.77 | 5.05 | 4.70 | 5.84 | 5.22 | 4.93 |
| Rocks1 | 3.62 | 3.26 | 3.73 | 3.52 | 3.42 | 3.34 | 3.28 | 3.23 | 3.50 | 3.50 | 3.29 | 3.09 | 3.53 |
| Rocks2 | 2.80 | 1.73 | 2.72 | 2.76 | 1.73 | 1.70 | 1.69 | 1.73 | 2.54 | 2.59 | 1.73 | 1.80 | 2.60 |
| Wood1 | 7.16 | 5.39 | 5.30 | 5.58 | 5.47 | 5.46 | 5.34 | 5.27 | 5.27 | 5.09 | 5.40 | 5.15 | 5.17 |
| Wood2 | 15.05 | 2.26 | 3.50 | 9.45 | 2.29 | 2.28 | 2.32 | 2.28 | 2.43 | 4.31 | 2.32 | 2.27 | 2.72 |
| **Average** | **9.63** | **4.47** | **6.09** | **8.95** | **4.58** | **4.46** | **4.33** | **4.36** | **4.58** | **6.00** | **4.42** | **4.27** | **4.63** |

Table D.2.2. – Percentage of bad pixels in estimated depth maps (only textureless pixels), experiment case 1. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 8.76 | 6.81 | 8.72 | 12.00 | 7.59 | 7.53 | 7.23 | 6.70 | 7.06 | 9.01 | 7.40 | 6.61 | 7.63 |
| Art | 9.84 | 9.35 | 10.87 | 11.36 | 9.88 | 9.82 | 9.26 | 9.92 | 10.12 | 9.95 | 9.80 | 9.93 | 9.72 |
| Baby1 | 7.10 | 2.99 | 4.31 | 8.01 | 2.86 | 2.85 | 2.82 | 2.87 | 3.06 | 4.63 | 2.86 | 2.98 | 3.07 |
| Baby2 | 4.75 | 2.49 | 3.72 | 5.83 | 2.29 | 2.24 | 2.25 | 2.39 | 2.77 | 4.12 | 2.32 | 2.48 | 2.90 |
| Baby3 | 8.21 | 3.10 | 5.16 | 7.32 | 3.28 | 3.11 | 3.01 | 3.17 | 2.81 | 4.93 | 3.14 | 2.95 | 2.86 |
| Books | 13.51 | 5.62 | 10.43 | 12.71 | 5.20 | 5.24 | 5.23 | 5.52 | 8.03 | 10.14 | 5.34 | 5.73 | 8.50 |
| Bowling1 | 21.22 | 10.53 | 12.81 | 18.17 | 10.63 | 10.63 | 10.54 | 10.68 | 11.41 | 13.95 | 10.58 | 10.81 | 11.80 |
| Bowling2 | 30.82 | 3.76 | 11.56 | 27.31 | 3.19 | 3.20 | 3.22 | 3.54 | 4.78 | 12.80 | 3.29 | 4.08 | 5.35 |
| Cloth1 | 0.15 | 0.15 | 0.14 | 0.14 | 0.17 | 0.17 | 0.17 | 0.16 | 0.13 | 0.13 | 0.16 | 0.14 | 0.13 |
| Cloth2 | 2.07 | 2.70 | 2.94 | 2.47 | 2.77 | 2.67 | 2.57 | 2.67 | 2.52 | 2.28 | 2.69 | 2.39 | 2.43 |
| Cloth3 | 0.92 | 1.10 | 1.00 | 0.92 | 1.28 | 1.30 | 1.26 | 1.14 | 1.02 | 0.93 | 1.20 | 1.05 | 0.97 |
| Cloth4 | 1.17 | 1.45 | 1.43 | 1.13 | 1.55 | 1.51 | 1.48 | 1.45 | 1.21 | 1.15 | 1.45 | 1.26 | 1.24 |
| Dolls | 5.28 | 4.64 | 5.52 | 5.67 | 4.80 | 4.64 | 4.53 | 4.55 | 4.34 | 4.75 | 4.59 | 4.17 | 4.41 |
| Flowerpots | 21.61 | 4.17 | 9.81 | 20.86 | 4.11 | 4.06 | 4.03 | 4.07 | 5.68 | 10.80 | 4.04 | 4.19 | 5.92 |
| Lampshade1 | 9.88 | 4.37 | 7.34 | 10.53 | 4.53 | 4.36 | 4.35 | 4.38 | 5.09 | 7.65 | 4.35 | 4.23 | 5.31 |
| Lampshade2 | 9.77 | 2.87 | 7.25 | 17.03 | 2.96 | 2.95 | 2.83 | 2.93 | 2.77 | 7.48 | 2.94 | 2.81 | 2.76 |
| Laundry | 8.18 | 5.51 | 8.23 | 8.99 | 5.64 | 5.35 | 5.10 | 5.19 | 5.57 | 6.99 | 5.31 | 4.99 | 5.63 |
| Midd1 | 10.03 | 4.72 | 4.20 | 7.20 | 4.09 | 4.10 | 4.12 | 4.14 | 4.21 | 4.16 | 4.10 | 4.79 | 4.21 |
| Midd2 | 17.75 | 4.19 | 5.26 | 7.78 | 4.02 | 3.93 | 3.81 | 3.90 | 4.00 | 4.97 | 3.97 | 4.21 | 4.13 |
| Moebius | 17.34 | 7.03 | 12.18 | 18.04 | 7.44 | 7.28 | 7.16 | 6.95 | 8.92 | 11.85 | 7.25 | 6.75 | 9.16 |
| Monopoly | 18.86 | 3.75 | 3.33 | 3.45 | 3.53 | 3.50 | 3.47 | 3.50 | 3.21 | 3.08 | 3.55 | 2.96 | 2.65 |
| Plastic | 37.22 | 7.20 | 17.82 | 38.35 | 6.23 | 6.37 | 6.33 | 6.21 | 9.38 | 19.57 | 6.33 | 7.23 | 9.48 |
| Reindeer | 4.09 | 4.99 | 5.43 | 4.82 | 5.27 | 5.24 | 5.17 | 5.11 | 4.82 | 4.67 | 5.18 | 4.80 | 4.75 |
| Rocks1 | 4.57 | 3.69 | 4.58 | 4.38 | 3.77 | 3.68 | 3.65 | 3.65 | 4.26 | 4.35 | 3.70 | 3.54 | 4.34 |
| Rocks2 | 3.65 | 1.93 | 3.45 | 3.57 | 1.87 | 1.84 | 1.84 | 1.93 | 3.22 | 3.32 | 1.92 | 2.06 | 3.32 |
| Wood1 | 9.37 | 6.78 | 6.71 | 7.10 | 6.77 | 6.77 | 6.62 | 6.60 | 6.68 | 6.42 | 6.70 | 6.47 | 6.54 |
| Wood2 | 16.05 | 1.99 | 3.36 | 9.89 | 2.01 | 2.01 | 2.06 | 2.02 | 2.20 | 4.26 | 2.06 | 2.03 | 2.51 |
| **Average** | **11.19** | **4.37** | **6.58** | **10.19** | **4.36** | **4.31** | **4.23** | **4.27** | **4.79** | **6.60** | **4.30** | **4.28** | **4.88** |

Table D.2.3. – Percentage of bad pixels in estimated depth maps (only textured pixels), experiment case 1.

Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 3.75 | 3.78 | 3.89 | 4.29 | 4.52 | 4.46 | 4.33 | 3.91 | 3.63 | 3.83 | 4.27 | 3.67 | 3.69 |
| Art | 10.05 | 12.39 | 12.29 | 11.60 | 13.05 | 12.53 | 11.78 | 12.38 | 11.23 | 10.64 | 12.48 | 11.68 | 10.88 |
| Baby1 | 2.29 | 2.07 | 2.21 | 2.31 | 2.05 | 2.05 | 2.05 | 2.07 | 2.12 | 2.18 | 2.07 | 2.08 | 2.13 |
| Baby2 | 2.19 | 2.15 | 2.33 | 2.34 | 2.18 | 2.10 | 2.06 | 2.13 | 2.08 | 2.19 | 2.12 | 2.08 | 2.10 |
| Baby3 | 3.84 | 4.35 | 4.41 | 4.15 | 4.58 | 4.23 | 4.01 | 4.16 | 3.93 | 3.85 | 4.25 | 3.93 | 3.85 |
| Books | 3.93 | 2.91 | 3.44 | 3.72 | 3.07 | 3.01 | 2.97 | 2.96 | 3.05 | 3.30 | 2.96 | 2.92 | 3.23 |
| Bowling1 | 17.21 | 17.25 | 17.59 | 17.59 | 17.76 | 17.52 | 17.25 | 17.34 | 17.06 | 17.18 | 17.42 | 17.06 | 17.10 |
| Bowling2 | 3.32 | 3.06 | 3.37 | 3.35 | 3.28 | 3.21 | 3.12 | 3.11 | 3.12 | 3.14 | 3.11 | 3.12 | 3.15 |
| Cloth1 | 0.12 | 0.18 | 0.12 | 0.12 | 0.25 | 0.23 | 0.20 | 0.17 | 0.12 | 0.12 | 0.17 | 0.13 | 0.12 |
| Cloth2 | 2.33 | 3.93 | 3.46 | 2.97 | 4.53 | 4.27 | 4.00 | 3.88 | 3.09 | 2.68 | 4.00 | 3.37 | 2.94 |
| Cloth3 | 1.00 | 1.22 | 1.09 | 1.01 | 1.40 | 1.33 | 1.28 | 1.21 | 1.04 | 0.99 | 1.24 | 1.09 | 1.01 |
| Cloth4 | 1.23 | 1.75 | 1.51 | 1.31 | 2.03 | 1.85 | 1.75 | 1.66 | 1.32 | 1.26 | 1.69 | 1.43 | 1.29 |
| Dolls | 4.97 | 6.69 | 6.11 | 5.34 | 7.45 | 6.82 | 6.26 | 6.41 | 5.34 | 5.00 | 6.55 | 5.65 | 5.16 |
| Flowerpots | 4.87 | 4.60 | 4.91 | 5.26 | 4.88 | 4.63 | 4.51 | 4.52 | 4.60 | 4.74 | 4.51 | 4.55 | 4.67 |
| Lampshade1 | 9.84 | 11.49 | 11.13 | 10.47 | 11.94 | 11.26 | 10.43 | 11.05 | 9.89 | 9.50 | 11.18 | 10.00 | 9.78 |
| Lampshade2 | 9.87 | 9.70 | 9.75 | 9.52 | 10.10 | 8.98 | 8.71 | 9.06 | 8.46 | 8.17 | 9.25 | 8.56 | 8.21 |
| Laundry | 8.04 | 11.77 | 9.78 | 8.78 | 13.03 | 11.91 | 11.01 | 10.89 | 8.94 | 8.40 | 11.24 | 9.48 | 8.67 |
| Midd1 | 3.62 | 4.38 | 4.06 | 3.84 | 4.94 | 4.69 | 4.56 | 4.27 | 3.80 | 3.67 | 4.59 | 4.00 | 3.76 |
| Midd2 | 3.46 | 3.71 | 3.49 | 3.40 | 4.29 | 3.92 | 3.83 | 3.74 | 3.26 | 3.19 | 3.98 | 3.42 | 3.24 |
| Moebius | 3.39 | 6.24 | 5.25 | 4.32 | 7.39 | 6.74 | 6.14 | 5.98 | 4.45 | 3.85 | 6.37 | 4.99 | 4.22 |
| Monopoly | 3.55 | 4.66 | 4.23 | 3.95 | 4.98 | 4.81 | 4.74 | 4.65 | 4.11 | 4.09 | 4.72 | 4.26 | 4.07 |
| Plastic | 6.79 | 7.85 | 8.37 | 8.11 | 8.62 | 8.37 | 7.90 | 7.78 | 8.01 | 7.92 | 7.87 | 7.69 | 8.09 |
| Reindeer | 4.31 | 6.66 | 6.33 | 5.50 | 7.26 | 6.74 | 6.27 | 6.68 | 5.40 | 4.80 | 6.74 | 5.82 | 5.21 |
| Rocks1 | 2.04 | 2.54 | 2.32 | 2.08 | 2.84 | 2.77 | 2.68 | 2.53 | 2.26 | 2.11 | 2.61 | 2.34 | 2.20 |
| Rocks2 | 1.23 | 1.36 | 1.39 | 1.28 | 1.47 | 1.44 | 1.42 | 1.36 | 1.30 | 1.25 | 1.39 | 1.32 | 1.29 |
| Wood1 | 1.42 | 1.77 | 1.63 | 1.61 | 2.06 | 2.05 | 2.02 | 1.79 | 1.61 | 1.61 | 2.00 | 1.71 | 1.60 |
| Wood2 | 4.71 | 5.03 | 4.90 | 4.78 | 5.07 | 4.97 | 4.91 | 4.93 | 4.84 | 4.81 | 4.96 | 4.78 | 4.80 |
| **Average** | **4.57** | **5.31** | **5.16** | **4.93** | **5.74** | **5.44** | **5.19** | **5.21** | **4.74** | **4.61** | **5.32** | **4.86** | **4.68** |

Table D.2.4. – Percentage of bad pixels in estimated depth maps (only pixels near depth discontinuities), experiment case 1. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 17.49 | 17.28 | 18.99 | 21.87 | 18.65 | 18.47 | 17.92 | 17.28 | 17.39 | 18.88 | 18.00 | 16.83 | 17.98 |
| Art | 23.34 | 25.56 | 27.29 | 26.86 | 26.49 | 25.95 | 25.06 | 25.81 | 25.41 | 24.74 | 25.91 | 25.63 | 24.76 |
| Baby1 | 24.64 | 22.53 | 24.66 | 27.98 | 22.09 | 22.09 | 21.97 | 22.19 | 23.47 | 25.28 | 22.21 | 22.76 | 23.55 |
| Baby2 | 19.49 | 17.21 | 19.46 | 22.20 | 16.79 | 16.48 | 16.48 | 17.13 | 17.65 | 19.38 | 16.83 | 17.17 | 18.02 |
| Baby3 | 18.68 | 17.02 | 18.85 | 20.54 | 17.72 | 16.82 | 16.18 | 16.78 | 16.51 | 17.40 | 16.85 | 16.19 | 16.62 |
| Books | 24.43 | 18.09 | 21.98 | 24.24 | 18.56 | 18.43 | 18.28 | 18.15 | 20.02 | 20.92 | 18.29 | 18.46 | 20.58 |
| Bowling1 | 41.64 | 37.43 | 38.30 | 40.67 | 38.59 | 38.39 | 37.72 | 38.74 | 37.73 | 37.93 | 38.06 | 37.67 | 37.41 |
| Bowling2 | 33.68 | 21.11 | 26.36 | 35.62 | 20.38 | 20.22 | 20.12 | 21.16 | 24.71 | 28.55 | 20.31 | 22.63 | 26.08 |
| Cloth1 | 3.87 | 4.75 | 3.62 | 3.60 | 6.11 | 5.73 | 5.18 | 4.58 | 3.57 | 3.58 | 4.60 | 3.76 | 3.57 |
| Cloth2 | 10.36 | 13.62 | 13.38 | 12.06 | 14.75 | 14.25 | 13.81 | 13.54 | 12.34 | 11.48 | 13.81 | 12.40 | 12.12 |
| Cloth3 | 6.52 | 7.98 | 7.13 | 6.55 | 9.06 | 8.72 | 8.40 | 7.92 | 6.82 | 6.46 | 8.17 | 7.14 | 6.64 |
| Cloth4 | 8.77 | 11.63 | 10.55 | 9.21 | 13.26 | 12.46 | 12.08 | 11.36 | 9.38 | 8.96 | 11.44 | 10.00 | 9.26 |
| Dolls | 13.59 | 16.20 | 16.12 | 14.58 | 17.32 | 16.39 | 15.63 | 15.84 | 14.44 | 13.55 | 16.03 | 14.55 | 14.20 |
| Flowerpots | 34.75 | 20.68 | 28.23 | 35.42 | 20.94 | 20.54 | 20.32 | 20.41 | 25.23 | 30.03 | 20.44 | 21.05 | 26.22 |
| Lampshade1 | 28.39 | 24.19 | 26.48 | 29.05 | 24.78 | 24.33 | 24.15 | 23.95 | 24.61 | 26.93 | 24.04 | 23.56 | 24.98 |
| Lampshade2 | 24.15 | 22.45 | 23.37 | 24.41 | 22.98 | 22.90 | 22.32 | 22.78 | 22.10 | 21.80 | 22.87 | 22.54 | 21.69 |
| Laundry | 18.42 | 23.03 | 21.70 | 19.87 | 24.66 | 23.05 | 21.59 | 21.50 | 19.48 | 18.65 | 22.22 | 19.65 | 19.00 |
| Midd1 | 22.57 | 18.34 | 17.99 | 19.34 | 19.27 | 18.90 | 18.71 | 17.88 | 17.35 | 17.24 | 18.65 | 18.05 | 17.29 |
| Midd2 | 29.98 | 19.56 | 20.15 | 22.83 | 20.56 | 19.68 | 19.36 | 19.07 | 18.14 | 18.51 | 19.75 | 18.83 | 18.35 |
| Moebius | 18.07 | 19.66 | 20.06 | 20.18 | 22.41 | 21.14 | 20.07 | 19.26 | 17.11 | 17.58 | 20.37 | 17.40 | 16.78 |
| Monopoly | 23.13 | 22.91 | 19.65 | 18.54 | 22.72 | 22.27 | 22.04 | 21.88 | 19.31 | 18.07 | 22.36 | 20.11 | 18.34 |
| Plastic | 42.88 | 25.22 | 34.25 | 48.76 | 26.09 | 26.87 | 26.54 | 25.28 | 28.95 | 39.28 | 26.40 | 26.24 | 30.12 |
| Reindeer | 18.23 | 22.43 | 22.71 | 20.76 | 23.68 | 23.13 | 22.52 | 22.64 | 20.96 | 19.76 | 22.86 | 21.40 | 20.67 |
| Rocks1 | 13.74 | 13.59 | 14.89 | 13.48 | 14.48 | 14.17 | 13.86 | 13.50 | 13.93 | 13.49 | 13.84 | 12.64 | 14.00 |
| Rocks2 | 16.12 | 11.17 | 15.58 | 15.64 | 11.16 | 11.01 | 10.98 | 11.19 | 14.62 | 14.83 | 11.18 | 11.60 | 14.93 |
| Wood1 | 29.13 | 18.15 | 19.45 | 20.87 | 19.50 | 19.47 | 17.64 | 17.41 | 18.78 | 16.73 | 18.67 | 14.91 | 17.14 |
| Wood2 | 17.40 | 16.94 | 16.13 | 13.57 | 17.80 | 17.65 | 17.55 | 17.10 | 16.97 | 15.77 | 17.64 | 16.38 | 16.20 |
| **Average** | **21.61** | **18.84** | **20.27** | **21.80** | **19.66** | **19.24** | **18.76** | **18.68** | **18.78** | **19.47** | **18.95** | **18.13** | **18.76** |

Table D.2.5. – Percentage of bad pixels in estimated depth maps (all pixels included), experiment case 2.

Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 5.64 | 5.52 | 5.63 | 5.94 | 5.37 | 5.90 | 5.95 | 5.86 | 6.19 | 6.24 | 5.69 | 6.00 | 6.22 |
| Art | 9.91 | 8.52 | 8.71 | 8.75 | 8.67 | 8.76 | 8.78 | 8.74 | 8.78 | 8.76 | 8.74 | 8.75 | 8.76 |
| Baby1 | 5.72 | 5.74 | 5.78 | 5.79 | 5.76 | 5.79 | 5.83 | 5.81 | 5.84 | 5.85 | 5.81 | 5.81 | 5.83 |
| Baby2 | 4.07 | 4.34 | 4.15 | 4.33 | 4.14 | 4.38 | 4.35 | 4.12 | 4.63 | 4.34 | 4.16 | 4.48 | 4.38 |
| Baby3 | 7.25 | 7.12 | 7.14 | 7.15 | 7.14 | 7.17 | 7.19 | 7.11 | 7.21 | 7.20 | 7.13 | 7.18 | 7.20 |
| Books | 9.84 | 9.49 | 9.31 | 9.73 | 9.33 | 9.61 | 9.69 | 9.52 | 9.83 | 9.81 | 9.51 | 9.71 | 9.84 |
| Bowling1 | 20.60 | 20.62 | 20.72 | 20.78 | 20.69 | 20.77 | 20.64 | 20.77 | 20.64 | 20.63 | 20.77 | 20.77 | 20.64 |
| Bowling2 | 23.99 | 24.78 | 24.75 | 24.67 | 19.97 | 24.64 | 24.51 | 24.56 | 24.50 | 24.49 | 24.63 | 24.60 | 24.46 |
| Cloth1 | 0.13 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 |
| Cloth2 | 2.18 | 2.06 | 2.07 | 2.08 | 2.06 | 2.08 | 2.08 | 2.07 | 2.08 | 2.09 | 2.07 | 2.09 | 2.09 |
| Cloth3 | 0.98 | 0.93 | 0.95 | 0.95 | 0.94 | 0.94 | 0.94 | 0.94 | 0.95 | 0.95 | 0.94 | 0.94 | 0.95 |
| Cloth4 | 1.20 | 1.06 | 1.06 | 1.07 | 1.08 | 1.08 | 1.08 | 1.07 | 1.08 | 1.09 | 1.08 | 1.09 | 1.09 |
| Dolls | 5.14 | 5.15 | 5.20 | 5.20 | 5.18 | 5.18 | 5.26 | 5.17 | 5.27 | 5.27 | 5.17 | 5.21 | 5.26 |
| Flowerpots | 20.63 | 20.77 | 20.75 | 20.80 | 20.77 | 20.74 | 20.74 | 20.81 | 20.75 | 20.75 | 20.81 | 20.61 | 20.75 |
| Lampshade1 | 9.88 | 8.94 | 8.83 | 8.95 | 8.95 | 9.00 | 8.95 | 8.96 | 8.93 | 10.66 | 8.96 | 8.98 | 10.29 |
| Lampshade2 | 9.76 | 9.91 | 9.93 | 10.64 | 9.90 | 10.63 | 11.16 | 9.92 | 10.96 | 11.12 | 9.92 | 10.96 | 11.09 |
| Laundry | 8.13 | 7.30 | 7.54 | 7.71 | 7.35 | 7.64 | 7.72 | 7.46 | 7.71 | 9.43 | 7.46 | 7.68 | 9.37 |
| Midd1 | 8.06 | 7.58 | 7.60 | 7.65 | 7.59 | 7.59 | 7.59 | 7.65 | 7.61 | 7.60 | 7.64 | 7.58 | 7.61 |
| Midd2 | 13.47 | 12.68 | 12.71 | 12.71 | 12.70 | 12.69 | 12.74 | 12.71 | 12.72 | 12.76 | 12.71 | 12.73 | 12.77 |
| Moebius | 9.84 | 9.75 | 9.74 | 9.78 | 9.77 | 9.80 | 9.74 | 9.77 | 9.81 | 9.83 | 9.78 | 9.75 | 9.81 |
| Monopoly | 14.93 | 15.19 | 15.36 | 15.43 | 15.62 | 15.44 | 15.51 | 15.42 | 15.43 | 15.35 | 15.03 | 15.43 | 15.32 |
| Plastic | 35.83 | 40.07 | 40.83 | 40.45 | 40.48 | 36.80 | 40.32 | 40.50 | 36.86 | 40.27 | 37.17 | 36.90 | 40.31 |
| Reindeer | 4.17 | 4.26 | 4.31 | 4.29 | 4.27 | 4.29 | 4.30 | 4.29 | 4.31 | 4.32 | 4.27 | 4.29 | 4.30 |
| Rocks1 | 3.62 | 3.39 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 | 3.40 |
| Rocks2 | 2.80 | 2.72 | 2.73 | 2.73 | 2.72 | 2.74 | 2.73 | 2.73 | 2.73 | 2.74 | 2.73 | 2.74 | 2.73 |
| Wood1 | 7.16 | 7.13 | 7.14 | 7.14 | 7.13 | 7.14 | 7.14 | 7.14 | 7.14 | 7.06 | 7.14 | 7.14 | 7.15 |
| Wood2 | 15.05 | 14.85 | 14.86 | 14.87 | 14.86 | 14.86 | 14.87 | 14.88 | 14.87 | 14.87 | 14.88 | 14.88 | 14.87 |
| **Average** | **9.63** | **9.63** | **9.68** | **9.74** | **9.48** | **9.60** | **9.75** | **9.68** | **9.64** | **9.89** | **9.54** | **9.62** | **9.87** |

Table D.2.6. – Percentage of bad pixels in estimated depth maps (textureless pixels only), experiment case 2. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 8.76 | 8.80 | 8.96 | 9.59 | 8.42 | 9.51 | 9.60 | 9.43 | 10.11 | 10.20 | 9.05 | 9.72 | 10.20 |
| Art | 9.84 | 8.05 | 8.30 | 8.35 | 8.24 | 8.36 | 8.38 | 8.33 | 8.37 | 8.34 | 8.33 | 8.34 | 8.35 |
| Baby1 | 7.10 | 7.17 | 7.22 | 7.23 | 7.20 | 7.23 | 7.30 | 7.26 | 7.30 | 7.31 | 7.26 | 7.26 | 7.29 |
| Baby2 | 4.75 | 5.14 | 4.88 | 5.13 | 4.87 | 5.19 | 5.14 | 4.84 | 5.50 | 5.13 | 4.90 | 5.31 | 5.19 |
| Baby3 | 8.21 | 8.06 | 8.08 | 8.08 | 8.07 | 8.11 | 8.13 | 8.04 | 8.15 | 8.14 | 8.06 | 8.12 | 8.15 |
| Books | 13.51 | 13.02 | 12.83 | 13.34 | 12.86 | 13.16 | 13.27 | 13.07 | 13.48 | 13.45 | 13.06 | 13.31 | 13.49 |
| Bowling1 | 21.22 | 21.33 | 21.44 | 21.51 | 21.42 | 21.50 | 21.35 | 21.50 | 21.35 | 21.32 | 21.50 | 21.49 | 21.34 |
| Bowling2 | 30.82 | 31.92 | 31.87 | 31.76 | 25.51 | 31.73 | 31.56 | 31.62 | 31.55 | 31.53 | 31.72 | 31.67 | 31.49 |
| Cloth1 | 0.15 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 |
| Cloth2 | 2.07 | 1.93 | 1.93 | 1.94 | 1.93 | 1.96 | 1.95 | 1.93 | 1.95 | 1.96 | 1.93 | 1.95 | 1.96 |
| Cloth3 | 0.92 | 0.80 | 0.83 | 0.82 | 0.81 | 0.83 | 0.82 | 0.82 | 0.82 | 0.83 | 0.82 | 0.82 | 0.82 |
| Cloth4 | 1.17 | 0.94 | 0.94 | 0.95 | 0.98 | 0.98 | 0.98 | 0.95 | 0.98 | 0.99 | 0.99 | 0.99 | 0.99 |
| Dolls | 5.28 | 5.40 | 5.46 | 5.43 | 5.43 | 5.41 | 5.54 | 5.39 | 5.56 | 5.55 | 5.39 | 5.46 | 5.54 |
| Flowerpots | 21.61 | 21.75 | 21.73 | 21.78 | 21.75 | 21.72 | 21.71 | 21.79 | 21.72 | 21.73 | 21.79 | 21.58 | 21.73 |
| Lampshade1 | 9.88 | 8.91 | 8.77 | 8.90 | 8.92 | 8.96 | 8.90 | 8.92 | 8.89 | 10.75 | 8.92 | 8.94 | 10.39 |
| Lampshade2 | 9.77 | 9.97 | 9.99 | 10.75 | 9.97 | 10.75 | 11.29 | 9.97 | 11.09 | 11.26 | 9.97 | 11.10 | 11.22 |
| Laundry | 8.18 | 6.96 | 7.29 | 7.50 | 7.03 | 7.44 | 7.48 | 7.18 | 7.51 | 7.48 | 7.17 | 7.47 | 7.44 |
| Midd1 | 10.03 | 9.41 | 9.42 | 9.47 | 9.42 | 9.40 | 9.39 | 9.49 | 9.40 | 9.39 | 9.49 | 9.38 | 9.40 |
| Midd2 | 17.75 | 16.80 | 16.86 | 16.86 | 16.85 | 16.83 | 16.82 | 16.86 | 16.78 | 16.85 | 16.86 | 16.80 | 16.87 |
| Moebius | 17.34 | 17.25 | 17.24 | 17.30 | 17.29 | 17.36 | 17.21 | 17.28 | 17.34 | 17.35 | 17.31 | 17.22 | 17.34 |
| Monopoly | 18.86 | 19.27 | 19.50 | 19.60 | 19.85 | 19.62 | 19.71 | 19.59 | 19.60 | 19.49 | 19.06 | 19.61 | 19.46 |
| Plastic | 37.22 | 41.69 | 42.47 | 42.07 | 42.10 | 38.24 | 41.93 | 42.12 | 38.29 | 41.89 | 38.62 | 38.33 | 41.93 |
| Reindeer | 4.09 | 4.29 | 4.37 | 4.33 | 4.31 | 4.34 | 4.35 | 4.34 | 4.36 | 4.37 | 4.31 | 4.35 | 4.34 |
| Rocks1 | 4.57 | 4.23 | 4.25 | 4.25 | 4.25 | 4.25 | 4.25 | 4.25 | 4.25 | 4.25 | 4.25 | 4.24 | 4.25 |
| Rocks2 | 3.65 | 3.55 | 3.56 | 3.56 | 3.55 | 3.57 | 3.56 | 3.56 | 3.56 | 3.57 | 3.56 | 3.57 | 3.56 |
| Wood1 | 9.37 | 9.34 | 9.35 | 9.35 | 9.34 | 9.35 | 9.35 | 9.35 | 9.35 | 9.24 | 9.35 | 9.35 | 9.36 |
| Wood2 | 16.05 | 15.90 | 15.91 | 15.92 | 15.91 | 15.91 | 15.92 | 15.93 | 15.92 | 15.92 | 15.93 | 15.93 | 15.92 |
| **Average** | **11.19** | **11.19** | **11.24** | **11.33** | **10.98** | **11.18** | **11.33** | **11.26** | **11.23** | **11.42** | **11.10** | **11.20** | **11.41** |

Table D.2.7. – Percentage of bad pixels in estimated depth maps (textured pixels only), experiment case 2. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 3.75 | 3.54 | 3.61 | 3.73 | 3.52 | 3.71 | 3.73 | 3.69 | 3.80 | 3.83 | 3.65 | 3.74 | 3.81 |
| Art | 10.05 | 9.44 | 9.54 | 9.56 | 9.51 | 9.55 | 9.57 | 9.54 | 9.59 | 9.59 | 9.54 | 9.57 | 9.59 |
| Baby1 | 2.29 | 2.19 | 2.21 | 2.21 | 2.19 | 2.22 | 2.20 | 2.22 | 2.22 | 2.24 | 2.21 | 2.22 | 2.23 |
| Baby2 | 2.19 | 2.11 | 2.11 | 2.13 | 2.09 | 2.13 | 2.13 | 2.11 | 2.19 | 2.13 | 2.11 | 2.19 | 2.13 |
| Baby3 | 3.84 | 3.79 | 3.83 | 3.84 | 3.80 | 3.83 | 3.83 | 3.82 | 3.85 | 3.85 | 3.82 | 3.83 | 3.86 |
| Books | 3.93 | 3.78 | 3.64 | 3.91 | 3.62 | 3.88 | 3.91 | 3.80 | 3.94 | 3.93 | 3.79 | 3.91 | 3.95 |
| Bowling1 | 17.21 | 16.74 | 16.73 | 16.81 | 16.72 | 16.78 | 16.77 | 16.76 | 16.79 | 16.82 | 16.74 | 16.79 | 16.81 |
| Bowling2 | 3.32 | 3.18 | 3.17 | 3.19 | 3.12 | 3.18 | 3.19 | 3.18 | 3.19 | 3.19 | 3.17 | 3.19 | 3.20 |
| Cloth1 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 | 0.12 |
| Cloth2 | 2.33 | 2.24 | 2.25 | 2.26 | 2.24 | 2.25 | 2.25 | 2.25 | 2.26 | 2.26 | 2.25 | 2.26 | 2.26 |
| Cloth3 | 1.00 | 0.96 | 0.98 | 0.97 | 0.97 | 0.97 | 0.97 | 0.97 | 0.97 | 0.98 | 0.97 | 0.97 | 0.97 |
| Cloth4 | 1.23 | 1.13 | 1.14 | 1.15 | 1.14 | 1.14 | 1.15 | 1.14 | 1.15 | 1.15 | 1.14 | 1.15 | 1.15 |
| Dolls | 4.97 | 4.86 | 4.90 | 4.91 | 4.87 | 4.91 | 4.92 | 4.90 | 4.92 | 4.93 | 4.90 | 4.92 | 4.92 |
| Flowerpots | 4.87 | 4.95 | 4.95 | 4.95 | 4.95 | 4.95 | 4.95 | 4.95 | 4.95 | 4.95 | 4.95 | 4.94 | 4.95 |
| Lampshade1 | 9.84 | 9.28 | 9.33 | 9.35 | 9.27 | 9.34 | 9.36 | 9.35 | 9.35 | 9.90 | 9.31 | 9.37 | 9.48 |
| Lampshade2 | 9.87 | 9.58 | 9.62 | 9.74 | 9.57 | 9.73 | 10.11 | 9.61 | 9.90 | 10.04 | 9.61 | 9.90 | 10.05 |
| Laundry | 8.04 | 7.95 | 8.03 | 8.11 | 7.96 | 8.04 | 8.17 | 8.01 | 8.11 | 13.15 | 8.01 | 8.09 | 13.04 |
| Midd1 | 3.62 | 3.48 | 3.52 | 3.55 | 3.48 | 3.53 | 3.54 | 3.51 | 3.59 | 3.58 | 3.51 | 3.55 | 3.59 |
| Midd2 | 3.46 | 3.01 | 2.99 | 2.98 | 2.99 | 2.98 | 3.18 | 2.98 | 3.19 | 3.18 | 2.97 | 3.18 | 3.19 |
| Moebius | 3.39 | 3.30 | 3.29 | 3.31 | 3.29 | 3.30 | 3.32 | 3.30 | 3.34 | 3.36 | 3.29 | 3.33 | 3.34 |
| Monopoly | 3.55 | 3.35 | 3.37 | 3.33 | 3.35 | 3.31 | 3.31 | 3.34 | 3.31 | 3.32 | 3.32 | 3.31 | 3.30 |
| Plastic | 6.79 | 6.54 | 6.64 | 6.81 | 6.66 | 6.78 | 6.82 | 6.69 | 6.80 | 6.77 | 6.67 | 6.79 | 6.85 |
| Reindeer | 4.31 | 4.26 | 4.28 | 4.28 | 4.26 | 4.26 | 4.27 | 4.27 | 4.28 | 4.31 | 4.26 | 4.27 | 4.29 |
| Rocks1 | 2.04 | 1.99 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 |
| Rocks2 | 1.23 | 1.21 | 1.21 | 1.21 | 1.21 | 1.21 | 1.21 | 1.21 | 1.21 | 1.22 | 1.21 | 1.21 | 1.21 |
| Wood1 | 1.42 | 1.40 | 1.40 | 1.40 | 1.40 | 1.40 | 1.40 | 1.40 | 1.41 | 1.40 | 1.40 | 1.41 | 1.41 |
| Wood2 | 4.71 | 4.05 | 4.05 | 4.05 | 4.03 | 4.05 | 4.04 | 4.06 | 4.05 | 4.02 | 4.05 | 4.04 | 4.06 |
| **Average** | **4.57** | **4.39** | **4.40** | **4.44** | **4.38** | **4.43** | **4.46** | **4.41** | **4.46** | **4.68** | **4.41** | **4.45** | **4.66** |

Table D.2.8. – Percentage of bad pixels in estimated depth maps (only pixels near depth discontinuities), experiment case 2. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 17.49 | 16.77 | 17.09 | 17.85 | 16.45 | 17.77 | 17.90 | 17.67 | 18.42 | 18.58 | 17.19 | 17.99 | 18.52 |
| Art | 23.34 | 20.70 | 21.01 | 21.12 | 20.95 | 21.14 | 21.20 | 21.06 | 21.20 | 21.18 | 21.06 | 21.14 | 21.18 |
| Baby1 | 24.64 | 23.59 | 23.86 | 23.97 | 23.76 | 23.96 | 24.17 | 23.97 | 24.31 | 24.41 | 23.95 | 24.00 | 24.25 |
| Baby2 | 19.49 | 19.48 | 19.26 | 19.31 | 19.18 | 19.71 | 19.36 | 18.84 | 20.14 | 19.29 | 19.14 | 19.51 | 19.47 |
| Baby3 | 18.68 | 17.49 | 17.63 | 17.63 | 17.58 | 17.57 | 17.67 | 17.46 | 17.73 | 17.73 | 17.56 | 17.67 | 17.77 |
| Books | 24.43 | 23.10 | 23.25 | 24.02 | 23.13 | 23.70 | 23.79 | 23.86 | 24.19 | 24.08 | 23.71 | 24.01 | 24.01 |
| Bowling1 | 41.64 | 42.73 | 43.02 | 43.10 | 42.84 | 43.06 | 42.78 | 43.05 | 42.77 | 42.85 | 43.05 | 43.06 | 42.82 |
| Bowling2 | 33.68 | 33.41 | 33.34 | 33.55 | 35.26 | 33.44 | 32.68 | 33.31 | 33.26 | 33.15 | 33.30 | 33.17 | 33.14 |
| Cloth1 | 3.87 | 3.56 | 3.55 | 3.55 | 3.56 | 3.59 | 3.58 | 3.57 | 3.59 | 3.59 | 3.58 | 3.56 | 3.59 |
| Cloth2 | 10.36 | 9.71 | 9.75 | 9.80 | 9.70 | 9.83 | 9.82 | 9.75 | 9.83 | 9.87 | 9.74 | 9.84 | 9.87 |
| Cloth3 | 6.52 | 6.16 | 6.29 | 6.26 | 6.20 | 6.24 | 6.23 | 6.23 | 6.25 | 6.27 | 6.23 | 6.25 | 6.26 |
| Cloth4 | 8.77 | 7.80 | 7.83 | 7.88 | 7.89 | 7.92 | 7.95 | 7.85 | 7.95 | 7.97 | 7.93 | 7.96 | 7.97 |
| Dolls | 13.59 | 13.01 | 13.13 | 13.21 | 13.07 | 13.22 | 13.28 | 13.14 | 13.28 | 13.29 | 13.15 | 13.28 | 13.27 |
| Flowerpots | 34.75 | 35.44 | 35.49 | 35.67 | 35.46 | 35.27 | 35.15 | 35.73 | 35.24 | 35.25 | 35.71 | 35.03 | 35.26 |
| Lampshade1 | 28.39 | 24.77 | 24.79 | 24.90 | 24.71 | 24.90 | 24.88 | 24.72 | 24.93 | 26.81 | 24.72 | 24.97 | 26.10 |
| Lampshade2 | 24.15 | 21.34 | 21.49 | 21.59 | 21.31 | 21.40 | 21.79 | 21.41 | 21.52 | 21.63 | 21.40 | 21.54 | 21.79 |
| Laundry | 18.42 | 16.60 | 17.03 | 17.19 | 16.77 | 17.02 | 17.24 | 16.90 | 17.20 | 18.30 | 16.88 | 17.13 | 18.12 |
| Midd1 | 22.57 | 20.43 | 20.49 | 20.65 | 20.40 | 20.57 | 20.58 | 20.64 | 20.68 | 20.62 | 20.64 | 20.60 | 20.66 |
| Midd2 | 29.98 | 27.89 | 28.00 | 27.94 | 28.02 | 27.87 | 27.87 | 27.97 | 27.73 | 27.97 | 27.96 | 27.79 | 27.99 |
| Moebius | 18.07 | 17.72 | 17.68 | 17.76 | 17.70 | 17.74 | 17.62 | 17.67 | 17.86 | 17.94 | 17.67 | 17.63 | 17.86 |
| Monopoly | 23.13 | 22.43 | 22.73 | 22.91 | 23.45 | 22.79 | 22.94 | 22.89 | 22.78 | 22.67 | 21.83 | 22.84 | 22.40 |
| Plastic | 42.88 | 41.57 | 41.90 | 42.44 | 42.20 | 42.04 | 42.84 | 42.16 | 42.67 | 42.33 | 41.88 | 42.46 | 42.89 |
| Reindeer | 18.23 | 18.22 | 18.46 | 18.41 | 18.35 | 18.38 | 18.42 | 18.44 | 18.48 | 18.57 | 18.35 | 18.40 | 18.45 |
| Rocks1 | 13.74 | 12.71 | 12.76 | 12.77 | 12.79 | 12.79 | 12.77 | 12.77 | 12.77 | 12.77 | 12.79 | 12.75 | 12.77 |
| Rocks2 | 16.12 | 15.35 | 15.39 | 15.40 | 15.35 | 15.44 | 15.37 | 15.39 | 15.38 | 15.43 | 15.39 | 15.44 | 15.39 |
| Wood1 | 29.13 | 28.74 | 28.82 | 28.82 | 28.77 | 28.79 | 28.85 | 28.81 | 28.86 | 27.85 | 28.81 | 28.85 | 28.90 |
| Wood2 | 17.40 | 13.95 | 14.11 | 14.02 | 14.06 | 13.96 | 13.89 | 14.19 | 13.85 | 13.77 | 14.18 | 13.96 | 13.82 |
| **Average** | **21.61** | **20.54** | **20.67** | **20.80** | **20.70** | **20.74** | **20.76** | **20.72** | **20.85** | **20.90** | **20.66** | **20.77** | **20.91** |

Table D.2.9. – Percentage of bad pixels in estimated depth maps (all pixels included), experiment case 3.

Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 3.87 | 4.63 | 5.56 | 6.21 | 5.31 | 5.25 | 5.04 | 4.70 | 4.54 | 5.34 | 5.11 | 4.48 | 4.79 |
| Art | 8.51 | 9.54 | 10.40 | 10.51 | 10.05 | 9.95 | 9.32 | 10.04 | 9.65 | 9.19 | 9.93 | 9.64 | 9.36 |
| Baby1 | 5.36 | 2.62 | 3.64 | 6.25 | 2.45 | 2.44 | 2.42 | 2.53 | 2.67 | 4.00 | 2.47 | 2.61 | 2.74 |
| Baby2 | 3.99 | 2.34 | 3.51 | 4.33 | 2.27 | 2.22 | 2.21 | 2.33 | 2.61 | 3.77 | 2.28 | 2.38 | 2.71 |
| Baby3 | 6.80 | 3.36 | 5.04 | 6.50 | 3.50 | 3.27 | 3.15 | 3.35 | 2.99 | 4.77 | 3.32 | 3.13 | 3.02 |
| Books | 9.90 | 4.50 | 8.06 | 9.29 | 4.26 | 4.25 | 4.23 | 4.44 | 6.19 | 7.65 | 4.28 | 4.55 | 6.66 |
| Bowling1 | 18.62 | 11.41 | 13.37 | 17.76 | 11.54 | 11.46 | 11.37 | 11.56 | 12.03 | 14.23 | 11.43 | 11.57 | 12.44 |
| Bowling2 | 23.01 | 3.49 | 9.50 | 20.38 | 3.07 | 3.05 | 3.03 | 3.29 | 4.36 | 10.38 | 3.09 | 3.77 | 4.82 |
| Cloth1 | 0.14 | 0.19 | 0.14 | 0.14 | 0.24 | 0.23 | 0.21 | 0.18 | 0.14 | 0.14 | 0.18 | 0.15 | 0.14 |
| Cloth2 | 1.98 | 3.21 | 3.13 | 2.64 | 3.50 | 3.34 | 3.16 | 3.16 | 2.73 | 2.42 | 3.23 | 2.79 | 2.62 |
| Cloth3 | 0.89 | 1.18 | 1.07 | 0.98 | 1.36 | 1.31 | 1.26 | 1.18 | 1.02 | 0.96 | 1.22 | 1.07 | 1.00 |
| Cloth4 | 1.06 | 1.61 | 1.44 | 1.18 | 1.83 | 1.69 | 1.60 | 1.54 | 1.23 | 1.17 | 1.57 | 1.32 | 1.21 |
| Dolls | 4.83 | 5.56 | 5.87 | 5.45 | 5.97 | 5.60 | 5.29 | 5.38 | 4.79 | 4.87 | 5.46 | 4.83 | 4.79 |
| Flowerpots | 19.55 | 4.20 | 9.52 | 19.83 | 4.18 | 4.12 | 4.07 | 4.14 | 5.63 | 10.39 | 4.09 | 4.25 | 5.89 |
| Lampshade1 | 8.71 | 4.99 | 7.72 | 10.30 | 5.06 | 4.92 | 4.82 | 4.83 | 5.51 | 7.52 | 4.88 | 4.75 | 5.70 |
| Lampshade2 | 8.20 | 3.18 | 7.15 | 15.85 | 3.26 | 3.17 | 3.08 | 3.14 | 3.02 | 7.31 | 3.19 | 3.08 | 2.98 |
| Laundry | 7.17 | 7.47 | 8.42 | 8.39 | 8.01 | 7.43 | 6.97 | 6.96 | 6.20 | 7.14 | 7.15 | 6.20 | 6.16 |
| Midd1 | 7.32 | 4.59 | 4.41 | 6.24 | 4.16 | 4.08 | 4.04 | 4.08 | 4.27 | 4.38 | 4.07 | 4.53 | 4.32 |
| Midd2 | 12.92 | 4.03 | 5.03 | 6.08 | 3.90 | 3.73 | 3.67 | 3.76 | 3.86 | 4.74 | 3.83 | 4.07 | 4.02 |
| Moebius | 9.40 | 6.49 | 8.33 | 10.44 | 7.26 | 6.87 | 6.49 | 6.39 | 6.51 | 7.20 | 6.70 | 5.78 | 6.40 |
| Monopoly | 14.54 | 3.48 | 3.20 | 3.16 | 3.32 | 3.27 | 3.23 | 3.28 | 3.00 | 2.93 | 3.27 | 2.80 | 2.57 |
| Plastic | 33.24 | 7.25 | 17.35 | 34.69 | 6.38 | 6.40 | 6.37 | 6.36 | 9.38 | 18.94 | 6.37 | 7.29 | 9.42 |
| Reindeer | 3.93 | 5.62 | 5.75 | 5.09 | 5.99 | 5.73 | 5.50 | 5.68 | 5.04 | 4.69 | 5.74 | 5.18 | 4.93 |
| Rocks1 | 3.45 | 3.25 | 3.73 | 3.48 | 3.42 | 3.34 | 3.28 | 3.22 | 3.48 | 3.48 | 3.28 | 3.08 | 3.50 |
| Rocks2 | 2.68 | 1.72 | 2.71 | 2.75 | 1.72 | 1.69 | 1.68 | 1.72 | 2.54 | 2.58 | 1.72 | 1.78 | 2.60 |
| Wood1 | 6.38 | 5.58 | 5.77 | 6.10 | 5.64 | 5.62 | 5.49 | 5.49 | 5.65 | 5.72 | 5.58 | 5.51 | 5.66 |
| Wood2 | 14.13 | 2.00 | 2.70 | 9.03 | 2.04 | 2.02 | 2.06 | 2.02 | 2.14 | 3.51 | 2.06 | 2.02 | 2.44 |
| **Average** | **8.91** | **4.35** | **6.02** | **8.63** | **4.43** | **4.31** | **4.19** | **4.25** | **4.49** | **5.90** | **4.28** | **4.17** | **4.55** |

Table D.2.10. – Percentage of bad pixels in estimated depth maps (textureless pixels only), experiment case 3. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 5.39 | 6.34 | 8.51 | 9.86 | 7.06 | 7.00 | 6.66 | 6.33 | 6.40 | 8.27 | 6.88 | 6.11 | 6.98 |
| Art | 8.59 | 8.70 | 10.00 | 10.55 | 9.15 | 9.19 | 8.63 | 9.42 | 9.39 | 9.03 | 9.21 | 9.15 | 9.12 |
| Baby1 | 6.68 | 2.87 | 4.23 | 7.86 | 2.64 | 2.63 | 2.60 | 2.74 | 2.92 | 4.76 | 2.66 | 2.85 | 3.01 |
| Baby2 | 4.71 | 2.45 | 3.97 | 5.13 | 2.33 | 2.29 | 2.29 | 2.43 | 2.85 | 4.37 | 2.37 | 2.53 | 2.98 |
| Baby3 | 7.71 | 3.13 | 5.25 | 7.21 | 3.25 | 3.05 | 2.96 | 3.17 | 2.76 | 5.09 | 3.09 | 2.94 | 2.83 |
| Books | 13.67 | 5.52 | 10.90 | 12.73 | 5.06 | 5.08 | 5.07 | 5.39 | 8.18 | 10.35 | 5.18 | 5.59 | 8.83 |
| Bowling1 | 18.90 | 10.35 | 12.60 | 17.79 | 10.40 | 10.37 | 10.31 | 10.50 | 11.13 | 13.69 | 10.33 | 10.58 | 11.59 |
| Bowling2 | 29.60 | 3.64 | 11.50 | 25.99 | 3.02 | 3.03 | 3.03 | 3.38 | 4.77 | 12.73 | 3.11 | 4.00 | 5.37 |
| Cloth1 | 0.16 | 0.18 | 0.16 | 0.16 | 0.19 | 0.19 | 0.19 | 0.18 | 0.16 | 0.16 | 0.18 | 0.16 | 0.16 |
| Cloth2 | 1.88 | 2.70 | 2.93 | 2.45 | 2.76 | 2.67 | 2.57 | 2.67 | 2.51 | 2.27 | 2.68 | 2.39 | 2.43 |
| Cloth3 | 0.79 | 1.09 | 1.00 | 0.93 | 1.27 | 1.28 | 1.25 | 1.12 | 1.01 | 0.93 | 1.20 | 1.05 | 0.97 |
| Cloth4 | 0.98 | 1.43 | 1.40 | 1.07 | 1.54 | 1.50 | 1.45 | 1.43 | 1.17 | 1.10 | 1.43 | 1.23 | 1.18 |
| Dolls | 4.98 | 4.66 | 5.71 | 5.65 | 4.78 | 4.62 | 4.51 | 4.57 | 4.39 | 4.82 | 4.58 | 4.19 | 4.54 |
| Flowerpots | 20.46 | 4.16 | 9.80 | 20.74 | 4.13 | 4.08 | 4.04 | 4.11 | 5.69 | 10.74 | 4.06 | 4.23 | 5.96 |
| Lampshade1 | 8.75 | 4.34 | 7.40 | 10.36 | 4.38 | 4.29 | 4.27 | 4.23 | 5.09 | 7.35 | 4.26 | 4.23 | 5.30 |
| Lampshade2 | 8.33 | 2.64 | 7.01 | 16.65 | 2.69 | 2.71 | 2.63 | 2.65 | 2.58 | 7.33 | 2.69 | 2.64 | 2.58 |
| Laundry | 6.98 | 5.36 | 7.86 | 8.38 | 5.54 | 5.25 | 5.01 | 5.02 | 4.97 | 6.67 | 5.14 | 4.64 | 5.04 |
| Midd1 | 9.03 | 4.72 | 4.58 | 7.31 | 3.87 | 3.87 | 3.87 | 4.03 | 4.51 | 4.72 | 3.90 | 4.81 | 4.61 |
| Midd2 | 17.05 | 4.15 | 5.68 | 7.26 | 3.79 | 3.70 | 3.65 | 3.81 | 4.16 | 5.43 | 3.79 | 4.37 | 4.40 |
| Moebius | 16.71 | 6.87 | 12.01 | 17.66 | 7.24 | 7.12 | 7.01 | 6.89 | 8.93 | 11.22 | 7.13 | 6.76 | 9.00 |
| Monopoly | 18.52 | 3.30 | 3.00 | 3.07 | 3.00 | 2.99 | 2.96 | 3.05 | 2.83 | 2.75 | 3.02 | 2.53 | 2.30 |
| Plastic | 34.50 | 7.21 | 17.76 | 35.96 | 6.27 | 6.30 | 6.29 | 6.29 | 9.42 | 19.45 | 6.29 | 7.25 | 9.46 |
| Reindeer | 3.98 | 5.00 | 5.54 | 4.94 | 5.22 | 5.18 | 5.13 | 5.10 | 4.95 | 4.81 | 5.15 | 4.84 | 4.90 |
| Rocks1 | 4.37 | 3.68 | 4.59 | 4.36 | 3.77 | 3.69 | 3.65 | 3.65 | 4.23 | 4.33 | 3.70 | 3.54 | 4.31 |
| Rocks2 | 3.51 | 1.92 | 3.44 | 3.55 | 1.87 | 1.84 | 1.84 | 1.92 | 3.22 | 3.32 | 1.91 | 2.05 | 3.32 |
| Wood1 | 8.40 | 7.11 | 7.43 | 7.88 | 7.07 | 7.05 | 6.89 | 6.97 | 7.26 | 7.35 | 7.02 | 7.02 | 7.29 |
| Wood2 | 15.22 | 1.84 | 2.61 | 9.56 | 1.88 | 1.87 | 1.92 | 1.87 | 2.01 | 3.51 | 1.92 | 1.89 | 2.33 |
| **Average** | **10.37** | **4.27** | **6.55** | **9.82** | **4.23** | **4.18** | **4.10** | **4.18** | **4.72** | **6.54** | **4.18** | **4.21** | **4.84** |

Table D.2.11. – Percentage of bad pixels in estimated depth maps (textured pixels only), experiment case 3. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 2.94 | 3.60 | 3.77 | 3.99 | 4.24 | 4.19 | 4.06 | 3.71 | 3.42 | 3.57 | 4.03 | 3.49 | 3.45 |
| Art | 8.35 | 11.20 | 11.18 | 10.44 | 11.85 | 11.45 | 10.68 | 11.26 | 10.16 | 9.53 | 11.35 | 10.62 | 9.83 |
| Baby1 | 2.08 | 1.99 | 2.17 | 2.26 | 1.97 | 1.97 | 1.97 | 2.00 | 2.05 | 2.12 | 2.00 | 2.01 | 2.07 |
| Baby2 | 2.00 | 2.03 | 2.25 | 2.13 | 2.11 | 2.02 | 1.97 | 2.04 | 1.95 | 2.11 | 2.04 | 1.98 | 1.97 |
| Baby3 | 3.55 | 4.20 | 4.29 | 4.02 | 4.43 | 4.06 | 3.84 | 4.02 | 3.79 | 3.68 | 4.11 | 3.80 | 3.71 |
| Books | 3.82 | 2.86 | 3.47 | 3.74 | 2.98 | 2.91 | 2.86 | 2.89 | 2.98 | 3.28 | 2.84 | 2.87 | 3.16 |
| Bowling1 | 17.09 | 17.25 | 17.61 | 17.66 | 17.81 | 17.47 | 17.20 | 17.40 | 17.00 | 17.18 | 17.46 | 17.01 | 17.10 |
| Bowling2 | 3.08 | 3.00 | 3.35 | 3.30 | 3.20 | 3.13 | 3.03 | 3.03 | 3.09 | 3.13 | 3.03 | 3.06 | 3.11 |
| Cloth1 | 0.12 | 0.19 | 0.13 | 0.13 | 0.26 | 0.24 | 0.21 | 0.18 | 0.13 | 0.13 | 0.18 | 0.14 | 0.13 |
| Cloth2 | 2.11 | 3.87 | 3.40 | 2.89 | 4.46 | 4.20 | 3.92 | 3.81 | 3.02 | 2.60 | 3.94 | 3.30 | 2.88 |
| Cloth3 | 0.92 | 1.21 | 1.08 | 0.99 | 1.39 | 1.32 | 1.27 | 1.19 | 1.02 | 0.97 | 1.23 | 1.07 | 1.00 |
| Cloth4 | 1.11 | 1.70 | 1.47 | 1.25 | 1.98 | 1.79 | 1.69 | 1.61 | 1.27 | 1.21 | 1.64 | 1.38 | 1.23 |
| Dolls | 4.66 | 6.64 | 6.05 | 5.21 | 7.37 | 6.77 | 6.21 | 6.35 | 5.28 | 4.92 | 6.50 | 5.59 | 5.09 |
| Flowerpots | 4.87 | 4.74 | 5.00 | 5.28 | 4.99 | 4.72 | 4.55 | 4.62 | 4.64 | 4.77 | 4.61 | 4.62 | 4.69 |
| Lampshade1 | 8.35 | 10.89 | 10.66 | 9.77 | 11.27 | 10.74 | 9.78 | 10.32 | 9.42 | 9.08 | 10.51 | 9.52 | 9.31 |
| Lampshade2 | 7.11 | 8.66 | 8.64 | 8.27 | 8.97 | 7.85 | 7.69 | 8.04 | 7.44 | 7.26 | 8.26 | 7.66 | 7.14 |
| Laundry | 7.54 | 11.52 | 9.50 | 8.41 | 12.75 | 11.59 | 10.74 | 10.65 | 8.57 | 8.04 | 10.97 | 9.18 | 8.30 |
| Midd1 | 3.49 | 4.30 | 4.02 | 3.86 | 4.81 | 4.54 | 4.41 | 4.19 | 3.73 | 3.61 | 4.46 | 3.92 | 3.68 |
| Midd2 | 3.25 | 3.73 | 3.51 | 3.31 | 4.17 | 3.79 | 3.70 | 3.64 | 3.16 | 3.12 | 3.91 | 3.35 | 3.14 |
| Moebius | 3.12 | 6.16 | 5.18 | 4.21 | 7.27 | 6.65 | 6.05 | 5.95 | 4.40 | 3.74 | 6.33 | 4.94 | 4.15 |
| Monopoly | 2.99 | 4.01 | 3.77 | 3.41 | 4.26 | 4.08 | 3.99 | 3.93 | 3.49 | 3.46 | 3.99 | 3.59 | 3.36 |
| Plastic | 6.67 | 7.77 | 8.31 | 7.98 | 8.52 | 8.31 | 7.83 | 7.71 | 7.97 | 7.86 | 7.79 | 7.65 | 8.05 |
| Reindeer | 3.90 | 6.47 | 6.07 | 5.32 | 7.03 | 6.51 | 6.04 | 6.48 | 5.22 | 4.58 | 6.54 | 5.66 | 5.03 |
| Rocks1 | 1.93 | 2.52 | 2.29 | 2.03 | 2.82 | 2.76 | 2.66 | 2.52 | 2.23 | 2.07 | 2.59 | 2.32 | 2.16 |
| Rocks2 | 1.18 | 1.35 | 1.39 | 1.28 | 1.46 | 1.42 | 1.40 | 1.35 | 1.30 | 1.24 | 1.37 | 1.31 | 1.29 |
| Wood1 | 1.15 | 1.61 | 1.46 | 1.45 | 1.88 | 1.86 | 1.84 | 1.61 | 1.45 | 1.45 | 1.82 | 1.55 | 1.44 |
| Wood2 | 2.88 | 3.58 | 3.57 | 3.43 | 3.64 | 3.54 | 3.47 | 3.51 | 3.48 | 3.44 | 3.54 | 3.37 | 3.53 |
| **Average** | **4.08** | **5.08** | **4.95** | **4.67** | **5.48** | **5.18** | **4.93** | **4.96** | **4.50** | **4.38** | **5.08** | **4.63** | **4.44** |

Table D.2.12. – Percentage of bad pixels in estimated depth maps (only pixels near depth discontinuities), experiment case 3. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 12.25 | 16.09 | 18.21 | 19.36 | 17.28 | 17.12 | 16.51 | 16.21 | 15.97 | 17.28 | 16.70 | 15.60 | 16.44 |
| Art | 19.32 | 22.85 | 24.50 | 24.18 | 23.86 | 23.63 | 22.67 | 23.47 | 23.07 | 22.13 | 23.59 | 23.20 | 22.59 |
| Baby1 | 21.31 | 21.87 | 24.38 | 28.02 | 20.77 | 20.72 | 20.59 | 21.46 | 22.66 | 25.80 | 21.05 | 21.92 | 23.01 |
| Baby2 | 18.13 | 16.66 | 20.06 | 20.35 | 16.82 | 16.52 | 16.46 | 17.09 | 17.58 | 19.82 | 16.87 | 17.14 | 18.04 |
| Baby3 | 16.79 | 16.67 | 18.54 | 20.28 | 17.34 | 16.31 | 15.73 | 16.38 | 15.97 | 16.89 | 16.43 | 15.70 | 16.15 |
| Books | 22.72 | 17.49 | 22.19 | 23.20 | 17.84 | 17.68 | 17.48 | 17.49 | 19.51 | 20.43 | 17.45 | 17.68 | 20.36 |
| Bowling1 | 37.69 | 36.36 | 36.88 | 38.73 | 37.08 | 36.69 | 36.22 | 37.67 | 35.93 | 36.35 | 36.53 | 36.22 | 36.08 |
| Bowling2 | 28.20 | 20.68 | 25.74 | 31.70 | 19.79 | 19.56 | 19.39 | 20.47 | 24.10 | 27.11 | 19.63 | 21.99 | 25.48 |
| Cloth1 | 3.92 | 5.21 | 4.10 | 4.05 | 6.56 | 6.23 | 5.69 | 5.03 | 4.07 | 4.08 | 5.12 | 4.26 | 4.06 |
| Cloth2 | 9.25 | 13.46 | 13.20 | 11.81 | 14.55 | 14.08 | 13.60 | 13.34 | 12.12 | 11.24 | 13.64 | 12.23 | 11.91 |
| Cloth3 | 5.91 | 7.88 | 7.08 | 6.48 | 8.98 | 8.63 | 8.33 | 7.81 | 6.73 | 6.38 | 8.08 | 7.07 | 6.59 |
| Cloth4 | 7.76 | 11.35 | 10.29 | 8.73 | 12.97 | 12.13 | 11.65 | 11.05 | 8.97 | 8.59 | 11.16 | 9.61 | 8.82 |
| Dolls | 12.48 | 16.09 | 15.99 | 14.14 | 17.16 | 16.25 | 15.50 | 15.70 | 14.29 | 13.35 | 15.91 | 14.42 | 14.07 |
| Flowerpots | 33.35 | 20.96 | 28.30 | 34.71 | 21.17 | 20.81 | 20.48 | 20.76 | 25.46 | 29.90 | 20.69 | 21.43 | 26.63 |
| Lampshade1 | 23.47 | 23.18 | 25.91 | 27.65 | 23.65 | 23.37 | 23.07 | 22.52 | 23.78 | 25.72 | 23.01 | 22.75 | 24.16 |
| Lampshade2 | 17.17 | 20.53 | 21.71 | 22.28 | 20.95 | 20.97 | 20.74 | 20.77 | 20.45 | 20.58 | 20.99 | 21.06 | 20.21 |
| Laundry | 15.93 | 22.39 | 20.05 | 18.23 | 24.03 | 22.34 | 20.97 | 20.80 | 17.52 | 17.10 | 21.54 | 18.43 | 17.00 |
| Midd1 | 18.61 | 17.26 | 17.12 | 17.97 | 18.10 | 17.62 | 17.35 | 16.72 | 16.39 | 16.67 | 17.42 | 16.98 | 16.40 |
| Midd2 | 27.09 | 19.17 | 21.12 | 22.43 | 19.51 | 18.69 | 18.48 | 18.43 | 18.11 | 19.74 | 18.89 | 18.93 | 18.76 |
| Moebius | 16.74 | 19.37 | 19.38 | 19.28 | 21.82 | 20.71 | 19.68 | 19.12 | 16.84 | 16.31 | 20.08 | 17.16 | 16.36 |
| Monopoly | 19.72 | 19.44 | 16.46 | 15.06 | 18.82 | 18.35 | 18.10 | 18.39 | 16.01 | 14.72 | 18.35 | 16.66 | 14.70 |
| Plastic | 39.47 | 24.86 | 34.16 | 47.56 | 26.24 | 26.41 | 26.07 | 25.65 | 29.27 | 39.09 | 26.00 | 26.00 | 29.83 |
| Reindeer | 16.86 | 21.91 | 22.10 | 20.30 | 23.03 | 22.41 | 21.86 | 22.09 | 20.61 | 19.35 | 22.30 | 20.92 | 20.44 |
| Rocks1 | 12.75 | 13.50 | 14.80 | 13.25 | 14.43 | 14.13 | 13.78 | 13.44 | 13.75 | 13.32 | 13.79 | 12.55 | 13.80 |
| Rocks2 | 15.46 | 11.14 | 15.53 | 15.57 | 11.12 | 10.95 | 10.90 | 11.14 | 14.60 | 14.81 | 11.12 | 11.52 | 14.92 |
| Wood1 | 24.64 | 20.49 | 23.98 | 26.73 | 21.73 | 21.46 | 19.56 | 20.39 | 23.38 | 23.40 | 21.14 | 19.90 | 22.75 |
| Wood2 | 11.18 | 12.99 | 11.29 | 10.27 | 14.16 | 13.84 | 13.72 | 13.15 | 12.60 | 11.54 | 13.80 | 12.70 | 12.07 |
| **Average** | **18.82** | **18.14** | **19.74** | **20.83** | **18.88** | **18.43** | **17.95** | **18.02** | **18.14** | **18.95** | **18.20** | **17.56** | **18.21** |

Table D.2.13. – Percentage of bad pixels in estimated depth maps (all pixels included), experiment case 4. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 3.87 | 4.07 | 4.16 | 4.34 | 4.09 | 4.29 | 4.33 | 4.30 | 4.35 | 4.53 | 4.26 | 4.38 | 4.39 |
| Art | 8.51 | 7.20 | 7.47 | 7.48 | 7.40 | 7.48 | 7.48 | 7.48 | 7.50 | 7.54 | 7.48 | 7.50 | 7.55 |
| Baby1 | 5.36 | 5.48 | 5.52 | 5.51 | 5.50 | 5.53 | 5.52 | 5.51 | 5.52 | 5.59 | 5.51 | 5.53 | 5.57 |
| Baby2 | 3.99 | 3.99 | 4.00 | 4.14 | 4.06 | 4.15 | 4.09 | 4.07 | 4.09 | 4.10 | 4.07 | 4.14 | 4.08 |
| Baby3 | 6.80 | 6.81 | 6.83 | 6.84 | 6.84 | 6.82 | 6.86 | 6.82 | 6.87 | 6.86 | 6.82 | 6.87 | 6.87 |
| Books | 9.90 | 9.51 | 9.21 | 9.60 | 9.27 | 9.61 | 9.62 | 9.44 | 9.61 | 9.63 | 9.43 | 9.59 | 9.70 |
| Bowling1 | 18.62 | 18.86 | 18.96 | 19.06 | 18.94 | 19.02 | 19.17 | 19.01 | 19.06 | 19.03 | 19.01 | 19.05 | 19.06 |
| Bowling2 | 23.01 | 23.68 | 23.80 | 23.15 | 23.60 | 23.75 | 23.82 | 23.80 | 23.72 | 23.76 | 23.78 | 23.88 | 23.71 |
| Cloth1 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 | 0.14 |
| Cloth2 | 1.98 | 1.99 | 2.00 | 2.01 | 1.99 | 2.03 | 2.02 | 2.00 | 2.02 | 2.02 | 2.00 | 2.02 | 2.02 |
| Cloth3 | 0.89 | 0.91 | 0.93 | 0.93 | 0.92 | 0.93 | 0.93 | 0.93 | 0.93 | 0.94 | 0.93 | 0.93 | 0.93 |
| Cloth4 | 1.06 | 0.98 | 0.99 | 0.99 | 1.00 | 1.01 | 1.01 | 0.99 | 1.01 | 1.01 | 1.00 | 1.01 | 1.01 |
| Dolls | 4.83 | 5.06 | 5.20 | 5.18 | 5.15 | 5.23 | 5.20 | 5.25 | 5.19 | 5.19 | 5.20 | 5.15 | 5.19 |
| Flowerpots | 19.55 | 20.01 | 19.81 | 20.68 | 20.01 | 20.69 | 20.68 | 20.73 | 20.68 | 20.69 | 20.73 | 20.68 | 20.69 |
| Lampshade1 | 8.71 | 8.16 | 8.23 | 8.33 | 8.19 | 8.61 | 8.61 | 8.30 | 8.64 | 9.86 | 8.29 | 8.63 | 10.02 |
| Lampshade2 | 8.20 | 8.70 | 8.75 | 9.46 | 8.70 | 9.46 | 9.78 | 8.74 | 9.78 | 9.97 | 8.74 | 9.78 | 9.75 |
| Laundry | 7.17 | 6.68 | 7.10 | 6.85 | 6.70 | 6.78 | 6.82 | 6.83 | 6.87 | 7.56 | 6.80 | 6.86 | 6.96 |
| Midd1 | 7.32 | 6.87 | 7.06 | 7.12 | 6.97 | 7.08 | 7.09 | 7.00 | 7.12 | 7.14 | 7.00 | 7.10 | 7.12 |
| Midd2 | 12.92 | 12.36 | 12.46 | 12.50 | 12.44 | 12.47 | 12.53 | 12.46 | 12.53 | 12.56 | 12.46 | 12.54 | 12.53 |
| Moebius | 9.40 | 9.53 | 9.51 | 9.35 | 9.48 | 9.33 | 9.43 | 9.52 | 9.42 | 9.41 | 9.51 | 9.43 | 9.42 |
| Monopoly | 14.54 | 14.73 | 14.87 | 14.92 | 14.86 | 14.86 | 14.86 | 14.91 | 14.87 | 14.90 | 14.92 | 14.86 | 14.87 |
| Plastic | 33.24 | 34.02 | 34.44 | 34.14 | 34.05 | 33.90 | 33.93 | 34.11 | 34.44 | 37.63 | 33.98 | 34.42 | 33.56 |
| Reindeer | 3.93 | 3.94 | 3.95 | 4.04 | 4.08 | 4.03 | 4.06 | 4.05 | 3.99 | 4.09 | 4.05 | 3.92 | 4.07 |
| Rocks1 | 3.45 | 3.34 | 3.35 | 3.36 | 3.36 | 3.36 | 3.36 | 3.35 | 3.37 | 3.36 | 3.35 | 3.36 | 3.36 |
| Rocks2 | 2.68 | 2.70 | 2.71 | 2.71 | 2.70 | 2.71 | 2.71 | 2.71 | 2.72 | 2.71 | 2.71 | 2.71 | 2.71 |
| Wood1 | 6.38 | 6.33 | 6.38 | 6.38 | 6.38 | 6.38 | 6.37 | 6.38 | 6.38 | 6.39 | 6.38 | 6.38 | 6.39 |
| Wood2 | 14.13 | 13.90 | 13.90 | 14.00 | 13.90 | 13.92 | 14.00 | 13.91 | 14.01 | 14.01 | 13.91 | 14.01 | 14.01 |
| **Average** | **8.91** | **8.89** | **8.95** | **9.01** | **8.92** | **9.02** | **9.05** | **8.99** | **9.07** | **9.28** | **8.98** | **9.07** | **9.10** |

Table D.2.14. – Percentage of bad pixels in estimated depth maps (textureless pixels only), experiment case 4. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 5.39 | 5.87 | 5.97 | 6.27 | 5.90 | 6.17 | 6.25 | 6.19 | 6.29 | 6.71 | 6.15 | 6.37 | 6.37 |
| Art | 8.59 | 6.87 | 7.20 | 7.21 | 7.11 | 7.23 | 7.21 | 7.22 | 7.24 | 7.28 | 7.23 | 7.24 | 7.30 |
| Baby1 | 6.68 | 6.87 | 6.91 | 6.90 | 6.90 | 6.93 | 6.92 | 6.91 | 6.91 | 6.99 | 6.90 | 6.93 | 6.97 |
| Baby2 | 4.71 | 4.72 | 4.72 | 4.92 | 4.81 | 4.92 | 4.84 | 4.81 | 4.84 | 4.85 | 4.82 | 4.90 | 4.83 |
| Baby3 | 7.71 | 7.72 | 7.73 | 7.74 | 7.74 | 7.72 | 7.80 | 7.71 | 7.81 | 7.78 | 7.71 | 7.81 | 7.81 |
| Books | 13.67 | 13.11 | 12.76 | 13.17 | 12.80 | 13.19 | 13.21 | 13.08 | 13.19 | 13.22 | 13.03 | 13.15 | 13.32 |
| Bowling1 | 18.90 | 19.28 | 19.40 | 19.51 | 19.38 | 19.46 | 19.64 | 19.46 | 19.50 | 19.47 | 19.46 | 19.50 | 19.50 |
| Bowling2 | 29.60 | 30.52 | 30.68 | 29.80 | 30.42 | 30.61 | 30.71 | 30.67 | 30.56 | 30.61 | 30.65 | 30.79 | 30.55 |
| Cloth1 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 | 0.16 |
| Cloth2 | 1.88 | 1.90 | 1.90 | 1.91 | 1.90 | 1.95 | 1.93 | 1.91 | 1.93 | 1.93 | 1.91 | 1.94 | 1.93 |
| Cloth3 | 0.79 | 0.79 | 0.82 | 0.83 | 0.81 | 0.82 | 0.82 | 0.82 | 0.82 | 0.82 | 0.82 | 0.83 | 0.82 |
| Cloth4 | 0.98 | 0.84 | 0.85 | 0.85 | 0.89 | 0.89 | 0.89 | 0.85 | 0.89 | 0.89 | 0.89 | 0.89 | 0.89 |
| Dolls | 4.98 | 5.35 | 5.56 | 5.52 | 5.50 | 5.60 | 5.55 | 5.64 | 5.54 | 5.54 | 5.56 | 5.46 | 5.54 |
| Flowerpots | 20.46 | 20.94 | 20.73 | 21.66 | 20.94 | 21.66 | 21.65 | 21.70 | 21.66 | 21.66 | 21.70 | 21.65 | 21.66 |
| Lampshade1 | 8.75 | 8.15 | 8.22 | 8.33 | 8.19 | 8.65 | 8.64 | 8.31 | 8.68 | 10.01 | 8.30 | 8.66 | 10.19 |
| Lampshade2 | 8.33 | 8.95 | 8.97 | 9.74 | 8.95 | 9.75 | 10.08 | 8.97 | 10.08 | 10.27 | 8.96 | 10.08 | 10.04 |
| Laundry | 6.98 | 6.24 | 6.85 | 6.48 | 6.27 | 6.36 | 6.44 | 6.45 | 6.45 | 6.43 | 6.40 | 6.49 | 6.43 |
| Midd1 | 9.03 | 8.48 | 8.74 | 8.78 | 8.62 | 8.77 | 8.78 | 8.65 | 8.80 | 8.80 | 8.65 | 8.77 | 8.80 |
| Midd2 | 17.05 | 16.37 | 16.52 | 16.53 | 16.50 | 16.53 | 16.54 | 16.52 | 16.53 | 16.56 | 16.52 | 16.54 | 16.53 |
| Moebius | 16.71 | 16.96 | 16.90 | 16.53 | 16.85 | 16.52 | 16.71 | 16.91 | 16.68 | 16.66 | 16.89 | 16.70 | 16.68 |
| Monopoly | 18.52 | 18.85 | 18.97 | 19.04 | 18.99 | 18.97 | 18.98 | 19.04 | 18.99 | 19.02 | 19.04 | 18.98 | 18.98 |
| Plastic | 34.50 | 35.34 | 35.77 | 35.46 | 35.37 | 35.21 | 35.23 | 35.43 | 35.77 | 39.13 | 35.29 | 35.75 | 34.85 |
| Reindeer | 3.98 | 4.04 | 4.03 | 4.15 | 4.24 | 4.17 | 4.17 | 4.17 | 4.10 | 4.22 | 4.18 | 3.98 | 4.17 |
| Rocks1 | 4.37 | 4.18 | 4.21 | 4.22 | 4.21 | 4.22 | 4.22 | 4.21 | 4.22 | 4.22 | 4.21 | 4.22 | 4.22 |
| Rocks2 | 3.51 | 3.53 | 3.54 | 3.54 | 3.53 | 3.53 | 3.53 | 3.53 | 3.55 | 3.53 | 3.54 | 3.53 | 3.53 |
| Wood1 | 8.40 | 8.33 | 8.40 | 8.40 | 8.41 | 8.40 | 8.39 | 8.40 | 8.40 | 8.41 | 8.40 | 8.39 | 8.41 |
| Wood2 | 15.22 | 14.98 | 14.98 | 15.09 | 14.98 | 15.00 | 15.10 | 15.00 | 15.10 | 15.10 | 14.99 | 15.10 | 15.10 |
| **Average** | **10.37** | **10.35** | **10.43** | **10.47** | **10.38** | **10.50** | **10.53** | **10.47** | **10.54** | **10.75** | **10.46** | **10.55** | **10.58** |

Table D.2.15. – Percentage of bad pixels in estimated depth maps (textured pixels only), experiment case 4. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | a=$\frac{1}{5}$ | a=$\frac{1}{12.5}$ | a=$\frac{1}{20}$ | a=$\frac{1}{25}$ | a=$\frac{1}{150}$ | a=$\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 2.94 | 2.98 | 3.06 | 3.17 | 2.99 | 3.15 | 3.17 | 3.15 | 3.18 | 3.21 | 3.12 | 3.17 | 3.19 |
| Art | 8.35 | 7.86 | 7.99 | 8.00 | 7.97 | 7.99 | 8.01 | 8.00 | 8.03 | 8.03 | 8.00 | 8.02 | 8.06 |
| Baby1 | 2.08 | 2.02 | 2.06 | 2.06 | 2.05 | 2.06 | 2.07 | 2.06 | 2.08 | 2.10 | 2.06 | 2.07 | 2.09 |
| Baby2 | 2.00 | 1.97 | 2.00 | 2.00 | 1.98 | 2.00 | 2.01 | 2.00 | 2.01 | 2.02 | 2.00 | 2.01 | 2.01 |
| Baby3 | 3.55 | 3.53 | 3.63 | 3.65 | 3.61 | 3.59 | 3.50 | 3.64 | 3.50 | 3.60 | 3.64 | 3.50 | 3.51 |
| Books | 3.82 | 3.70 | 3.49 | 3.83 | 3.59 | 3.84 | 3.84 | 3.59 | 3.83 | 3.84 | 3.63 | 3.84 | 3.86 |
| Bowling1 | 17.09 | 16.62 | 16.60 | 16.67 | 16.59 | 16.65 | 16.65 | 16.64 | 16.67 | 16.68 | 16.62 | 16.67 | 16.68 |
| Bowling2 | 3.08 | 2.99 | 2.99 | 2.99 | 2.98 | 2.99 | 2.99 | 2.99 | 3.00 | 3.01 | 2.99 | 2.99 | 3.00 |
| Cloth1 | 0.12 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 | 0.13 |
| Cloth2 | 2.11 | 2.11 | 2.12 | 2.14 | 2.11 | 2.13 | 2.13 | 2.12 | 2.14 | 2.14 | 2.12 | 2.14 | 2.14 |
| Cloth3 | 0.92 | 0.94 | 0.96 | 0.96 | 0.94 | 0.95 | 0.96 | 0.95 | 0.96 | 0.96 | 0.95 | 0.96 | 0.96 |
| Cloth4 | 1.11 | 1.06 | 1.07 | 1.07 | 1.07 | 1.08 | 1.08 | 1.07 | 1.08 | 1.07 | 1.07 | 1.08 | 1.08 |
| Dolls | 4.66 | 4.73 | 4.77 | 4.78 | 4.74 | 4.78 | 4.78 | 4.78 | 4.78 | 4.77 | 4.77 | 4.79 | 4.77 |
| Flowerpots | 4.87 | 4.94 | 4.94 | 4.98 | 4.94 | 5.01 | 5.01 | 4.98 | 5.01 | 5.01 | 4.98 | 5.01 | 5.02 |
| Lampshade1 | 8.35 | 8.20 | 8.33 | 8.31 | 8.19 | 8.30 | 8.33 | 8.28 | 8.35 | 8.52 | 8.24 | 8.34 | 8.48 |
| Lampshade2 | 7.11 | 6.43 | 6.78 | 6.91 | 6.45 | 6.78 | 6.99 | 6.69 | 7.06 | 7.22 | 6.65 | 7.01 | 7.09 |
| Laundry | 7.54 | 7.51 | 7.60 | 7.56 | 7.53 | 7.59 | 7.55 | 7.57 | 7.68 | 9.69 | 7.57 | 7.57 | 7.98 |
| Midd1 | 3.49 | 3.26 | 3.31 | 3.40 | 3.25 | 3.31 | 3.32 | 3.30 | 3.36 | 3.42 | 3.30 | 3.35 | 3.35 |
| Midd2 | 3.25 | 2.95 | 2.95 | 3.04 | 2.95 | 2.96 | 3.14 | 2.94 | 3.14 | 3.17 | 2.94 | 3.15 | 3.15 |
| Moebius | 3.12 | 3.14 | 3.16 | 3.17 | 3.14 | 3.15 | 3.17 | 3.16 | 3.18 | 3.18 | 3.15 | 3.18 | 3.18 |
| Monopoly | 2.99 | 2.76 | 2.98 | 2.96 | 2.88 | 2.92 | 2.92 | 2.96 | 2.93 | 2.96 | 2.96 | 2.91 | 2.94 |
| Plastic | 6.67 | 6.32 | 6.41 | 6.45 | 6.37 | 6.44 | 6.43 | 6.38 | 6.51 | 6.48 | 6.40 | 6.52 | 6.49 |
| Reindeer | 3.90 | 3.85 | 3.88 | 3.94 | 3.91 | 3.89 | 3.95 | 3.93 | 3.90 | 3.98 | 3.93 | 3.89 | 3.97 |
| Rocks1 | 1.93 | 1.93 | 1.93 | 1.94 | 1.94 | 1.94 | 1.94 | 1.93 | 1.94 | 1.94 | 1.93 | 1.94 | 1.94 |
| Rocks2 | 1.18 | 1.19 | 1.20 | 1.20 | 1.19 | 1.20 | 1.20 | 1.20 | 1.20 | 1.20 | 1.20 | 1.20 | 1.20 |
| Wood1 | 1.15 | 1.15 | 1.15 | 1.15 | 1.14 | 1.15 | 1.15 | 1.15 | 1.15 | 1.15 | 1.15 | 1.15 | 1.15 |
| Wood2 | 2.88 | 2.75 | 2.75 | 2.79 | 2.76 | 2.75 | 2.76 | 2.76 | 2.79 | 2.78 | 2.76 | 2.78 | 2.78 |
| **Average** | **4.08** | **3.96** | **4.01** | **4.05** | **3.98** | **4.03** | **4.04** | **4.01** | **4.06** | **4.16** | **4.01** | **4.05** | **4.08** |

Table D.2.16. – Percentage of bad pixels in estimated depth maps (only pixels near depth discontinuities), experiment case 4. Original cross-cost, weighting mode according to formula 5.4.9.

| | **Video analysis only** | **Proposed algorithm** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Step model | | | Linear model | | | Quadratic model | | | Gaussian model | | |
| | | b=±1 | b=±2 | b=±4 | $a=\frac{1}{5}$ | $a=\frac{1}{12.5}$ | $a=\frac{1}{20}$ | $a=\frac{1}{25}$ | $a=\frac{1}{150}$ | $a=\frac{1}{400}$ | σ=3 | σ=6 | σ=10 |
| Aloe | 12.25 | 12.68 | 12.94 | 13.35 | 12.70 | 13.27 | 13.38 | 13.22 | 13.46 | 13.65 | 13.12 | 13.46 | 13.50 |
| Art | 19.32 | 17.24 | 17.65 | 17.69 | 17.57 | 17.75 | 17.73 | 17.71 | 17.81 | 17.79 | 17.71 | 17.78 | 17.88 |
| Baby1 | 21.31 | 21.45 | 22.21 | 22.18 | 22.06 | 21.93 | 22.44 | 22.15 | 22.38 | 22.70 | 22.13 | 22.35 | 22.57 |
| Baby2 | 18.13 | 17.88 | 17.94 | 18.10 | 18.10 | 17.99 | 18.00 | 18.18 | 18.04 | 18.11 | 18.18 | 17.97 | 18.01 |
| Baby3 | 16.79 | 16.48 | 16.64 | 16.68 | 16.67 | 16.52 | 16.75 | 16.61 | 16.83 | 16.67 | 16.59 | 16.83 | 16.84 |
| Books | 22.72 | 22.15 | 22.04 | 22.65 | 22.25 | 22.73 | 22.72 | 22.64 | 22.73 | 22.87 | 22.46 | 22.67 | 22.90 |
| Bowling1 | 37.69 | 39.12 | 39.43 | 39.53 | 39.28 | 39.47 | 39.56 | 39.47 | 39.55 | 39.56 | 39.47 | 39.54 | 39.57 |
| Bowling2 | 28.20 | 28.55 | 28.70 | 28.32 | 28.57 | 28.64 | 28.21 | 28.67 | 28.60 | 28.69 | 28.65 | 28.32 | 28.42 |
| Cloth1 | 3.92 | 4.01 | 4.02 | 4.03 | 4.04 | 4.08 | 4.06 | 4.05 | 4.03 | 4.05 | 4.06 | 4.05 | 4.04 |
| Cloth2 | 9.25 | 9.29 | 9.34 | 9.40 | 9.30 | 9.48 | 9.45 | 9.34 | 9.48 | 9.45 | 9.34 | 9.48 | 9.45 |
| Cloth3 | 5.91 | 6.03 | 6.16 | 6.16 | 6.08 | 6.14 | 6.16 | 6.15 | 6.17 | 6.19 | 6.14 | 6.16 | 6.17 |
| Cloth4 | 7.76 | 7.22 | 7.27 | 7.30 | 7.34 | 7.38 | 7.39 | 7.28 | 7.38 | 7.37 | 7.36 | 7.39 | 7.38 |
| Dolls | 12.48 | 12.59 | 12.80 | 12.87 | 12.67 | 12.86 | 12.86 | 12.87 | 12.86 | 12.85 | 12.75 | 12.87 | 12.85 |
| Flowerpots | 33.35 | 34.95 | 34.78 | 34.96 | 34.95 | 34.81 | 34.76 | 35.01 | 34.79 | 34.81 | 35.01 | 34.80 | 34.83 |
| Lampshade1 | 23.47 | 21.68 | 21.71 | 21.80 | 21.54 | 22.23 | 22.36 | 21.65 | 22.58 | 23.78 | 21.65 | 22.31 | 23.83 |
| Lampshade2 | 17.17 | 16.07 | 16.46 | 16.51 | 16.08 | 16.43 | 16.53 | 16.35 | 16.60 | 17.14 | 16.32 | 16.55 | 16.60 |
| Laundry | 15.93 | 14.82 | 15.25 | 15.00 | 14.80 | 14.97 | 15.00 | 15.02 | 15.14 | 15.44 | 14.95 | 15.06 | 15.19 |
| Midd1 | 18.61 | 17.26 | 17.54 | 17.66 | 17.33 | 17.53 | 17.55 | 17.49 | 17.74 | 17.72 | 17.46 | 17.59 | 17.70 |
| Midd2 | 27.09 | 25.90 | 26.33 | 26.35 | 26.21 | 26.33 | 26.39 | 26.27 | 26.39 | 26.57 | 26.28 | 26.43 | 26.40 |
| Moebius | 16.74 | 16.90 | 16.91 | 16.72 | 16.82 | 16.64 | 16.88 | 16.92 | 16.84 | 16.80 | 16.89 | 16.87 | 16.85 |
| Monopoly | 19.72 | 20.11 | 20.20 | 20.35 | 20.44 | 20.05 | 20.06 | 20.40 | 20.12 | 20.33 | 20.42 | 20.04 | 20.10 |
| Plastic | 39.47 | 39.39 | 39.51 | 39.39 | 39.74 | 39.81 | 39.83 | 38.90 | 39.73 | 39.78 | 39.64 | 39.66 | 39.51 |
| Reindeer | 16.86 | 16.67 | 16.87 | 17.24 | 17.23 | 17.03 | 17.36 | 17.34 | 16.97 | 17.50 | 17.34 | 16.85 | 17.46 |
| Rocks1 | 12.75 | 12.38 | 12.48 | 12.51 | 12.49 | 12.54 | 12.50 | 12.48 | 12.55 | 12.52 | 12.48 | 12.52 | 12.52 |
| Rocks2 | 15.46 | 15.24 | 15.29 | 15.29 | 15.24 | 15.27 | 15.27 | 15.29 | 15.36 | 15.28 | 15.29 | 15.27 | 15.28 |
| Wood1 | 24.64 | 24.05 | 24.80 | 24.73 | 24.79 | 24.66 | 24.66 | 24.73 | 24.76 | 24.70 | 24.73 | 24.68 | 24.71 |
| Wood2 | 11.18 | 10.86 | 10.85 | 11.59 | 10.86 | 10.94 | 11.45 | 10.90 | 11.52 | 11.52 | 10.85 | 11.54 | 11.50 |
| **Average** | **18.82** | **18.55** | **18.74** | **18.83** | **18.71** | **18.80** | **18.86** | **18.78** | **18.90** | **19.03** | **18.79** | **18.85** | **18.96** |

# Bibliography

## Publications by the author and those co-authored by him

### *International journals*


[Domański_01] M. Domański, O. Stankiewicz, K. Wegner, M. Kurc, J. Konieczny, J. Siast, J. Stankowski, R. Ratajczak, T. Grajek, "High Efficiency 3D Video Coding Using New Tools Based on View Synthesis", IEEE Transactions on Image Processing, Special Issue on 3D Video Representation, Compression, & Rendering, Vol. 22, No. 9, September 2013, pp. 3517-3527.


### *International conferences*


[Domański_02] M. Domański, A. Dziembowski, A. Kuehn, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, "Experiments on acquisition and processing of video for free-viewpoint television", 3DTV Conference 2014, Budapest, Hungary, 2-4 July 2014.

[Domański_03] M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, Ł. Kowalski, M. Kurc, A. Łuczak, D. Mieloch, R. Ratajczak, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, "Methods of High Efficiency Compression for Transmission of Spatial Representation of Motion Scenes", IEEE International Conference on Multimedia and Expo ICME 2015, Torino, Italy, June 29-July 3 2015.

[Domanski_04] M. Domański, J. Konieczny, M. Kurc, A. Łuczak, J. Siast, O. Stankiewicz, K. Wegner, "Fast Depth Estimation on Mobile Platforms and FPGA Devices", 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 3DTV- Con 2015, Lisbon, Portugal, 8-10 July 2015.

[Domański_05] M. Domański, J. Konieczny, M. Kurc, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "3D video compression by coding of disoccluded regions", 19th IEEE International Conference on Image Processing (ICIP 2012), Orlando, Florida, U.S.A., 30 September - 3 October 2012, pp. 1317-1320.

[Domański_06] M. Domański, K. Klimaszewski, J. Konieczny, M. Kurc, A. Łuczak, O. Stankiewicz, K. Wegner, "An experimental Free-view Television System", 1st International Conference on Image Processing & Communications (IPC), Bydgoszcz, Polska, September 2009, pp. 175-184.

[Domański_07] M. Domański, T. Grajek, D. Karwowski, J. Konieczny, M. Kurc, A. Łuczak, R. Ratajczak, J. Siast, J. Stankowski, K. Wegner, "Coding of multiple video+depth using HEVC technology and reduced representations of side views and depth maps", 29th Picture Coding Symposium, PCS 2012, Kraków, Poland, 7-9 May 2012, pp. 5-8.

[Domański_08] M. Domański, T. Grajek, D. Karwowski, K. Klimaszewski, J. Konieczny, M. Kurc, A. Łuczak, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "New coding technology for 3D video with depth maps as proposed for standardisation within MPEG", 19th International Conference on Systems, Signals and Image Processing, IWSSIP 2012, Vienna, Austria, 11-13 April 2012, pp. 401-404.

[Domański_09] M. Domański, A. Dziembowski, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, Poznan University of Technology test multiview video sequences acquired with circular camera arrangement – "Poznan Team" and "Poznan Blocks" sequences, ISO/IEC JTC1/SC29/WG11 MPEG2015, Doc. M35846, Geneva, Switzerland, 14-20 February 2015.

[Domański_10] M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, [FTV AHG] Video and depth multiview test sequences acquired with circular camera arrangement – "Poznan Service" and "Poznan People", ISO/IEC JTC1/SC29/WG11 MPEG2015, Doc. M36569, Warsaw, Poland, 20-27 June 2015.

[Domański_11] M. Domański, K. Klimaszewski, M. Kurc, A. Łuczak, O. Stankiewicz, K. Wegner, "FTV: Poznan Laboratory - a test light-field sequence from Poznan University of Technology", ISO/IEC JTC 1/SC 29/WG 11, Doc. MPEG2014/m35071, Strasbourg, France, 20-24 October 2014.

[Domański_12] M. Domański, K. Klimaszewski, M. Kurc, R. Ratajczak, O. Stankiewicz, K. Wegner, "Super-multi-view light-field images from Poznan University of Technology", ISO/IEC JTC1/SC29/WG11 MPEG2015, Doc. M36566, Warsaw, Poland, 20-27 June 2015.

[Domański_13] M. Domański, T. Grajek, D. Karwowski, K. Klimaszewski, J. Konieczny, M. Kurc, A. Łuczak, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "Technical Desciption of Poznan University of Technology proposal for Call on 3D Video Coding Technology", ISO/IEC JTC1/SC29/WG11, MPEG 2011 / M22697, Geneva, Switzerland, 28 November-02 December 2011.

[Domański_14] M. Domański, T. Grajek, D. Karwowski, K. Klimaszewski, J. Konieczny, M. Kurc, A. Łuczak, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "Multiview HEVC – experimental results", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, MPEG 2011 / M22147, Geneva, Switzerland, 28 November - 02 December 2011.

[Domański_15] M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, K. Wegner, "Poznań Multiview Video Test Sequences and Camera Parameters", ISO/IEC JTC1/SC29/WG11, MPEG/M17050, Doc. m17050, Xian, China, 26-30 October 2009.

[Kurc_01] M. Kurc, O. Stankiewicz, M. Domański, "Depth map inter-view consistency refinement for multiview video", 29th Picture Coding Symposium, PCS 2012, Kraków, Poland, 7-9 May 2012, pp. 137-140.

[Kurc_02] M. Kurc, K. Wegner, M. Domański, “Transformation of Depth Maps Produced by ToF Cameras”, International Conference on Signals and Electronic Systems ICSES 2014, Poznań, Poland, 11-13 September 2014.

[Maćkowiak_01] S. Maćkowiak, J. Konieczny, M. Kurc, P. Maćkowiak, “A complex system for football player detection in broadcasted video”, 2010 International Conference on Signals and Electronic Systems (ICSES), 7-10 September 2010, pp. 119-122.

[Maćkowiak_02] S. Maćkowiak, J. Konieczny, M. Kurc, P. Maćkowiak, “Football Player Detection in Video Broadcast”, Proceedings of ICCVG, Lecture Notes in Computer Science, 20-22 September 2010, pp. 118-125.

[Ratajczak_01] R. Ratajczak, T. Grajek, K. Wegner, K. Klimaszewski, M. Kurc, M. Domański, “Vehicle Dimensions Estimation Scheme Using AAM on Stereoscopic Video”, 10th IEEE International Conference on Advanced Video and Signal-Based Surveillance, AVSS 2013, Workshop on Vehicle Retrieval in Surveillance (VRS) 2013, Kraków, Poland, 27-30 August 2013, pp. 478-482.

***National journals in Polish***

[Domański_16] M. Domański, T. Grajek, D. Karwowski, K. Klimaszewski, J. Konieczny, M. Kurc, A. Łuczak, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, “Poznański kodek obrazów trójwymiarowych”, Przegląd Telekomunikacyjny, No. 2-3, February/March 2013, pp. 81-83.

[Łuczak_01] A. Łuczak, M. Kurc, M. Stępniewska, J. Siast, “Interfejs komunikacyjny dla układów FPGA serii Virtex”, Pomiary Automatyka Kontrola, PAK, 2010, pp. 749-751.

[Łuczak_02] A. Łuczak, M. Kurc, M. Stępniewska, K. Wegner, “Platforma przetwarzania rozproszonego bazująca na sieci NoC”, Pomiary Automatyka Kontrola, PAK, 2009, pp. 690-692.

[Łuczak_03] A. Łuczak, M. Stępniewska, J. Siast, M. Domański, O. Stankiewicz, M. Kurc, J. Konieczny, “Network-on-Multi-Chip (NoMC) with monitoring and debugging support”, Journal of Telecommunications and Information Technology, No. 3/2011, 2011, pp. 81-81.

[Maćkowiak_03] S. Maćkowiak, J. Konieczny, M. Kurc, P. Mackowiak, “System detekcji i śledzenia piłkarzy w transmisjach widowisk sportowych w cyfrowym sygnale wizyjnym”, Elektronika, 2010, pp. 13-15.

***Patents and applications***


[Domański_17] M. Domański, J. Konieczny, M. Kurc, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "Method for predicting a shape of an encoded area using a depth map", Patent office: USPTO.Status: application, Application number: US 13/680740, Filling date: 19.11.2012, Publication number: US 2013/0128968A1, Publication date: 23.05.2013.

[Domański_18] M. Domański, K. Klimaszewski, J. Konieczny, M. Kurc, R. Ratajczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "Image coding method", Patent office: USPTO, Status: granted, Application number: US 13/680652, Filling date: 19.11.2012, Publication number: US 2013/0129235A1, Publication date: 23.05.2013, Patent number: US 8761527, Date of Patent: 24.06.2014,

[Domański_19] M. Domański, T. Grajek, J. Konieczny, M. Kurc, A. Łuczak, J. Siast, O. Stankiewicz, J. Stankowski, K. Wegner, "Method for coding of stereoscopic depth", Patent office: USPTO, Status: application, Application number: US 13/680822, Filling date: 19.11.2012, Publication number: US 2013/0129244A1, Publication date: 23.05.2013.

[Domański_20] M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, J. Stankowski, R. Ratajczak, K. Wegner, "A system and method for object dimension estimation", Patent office: USPTO, Status: granted, Application number: US 14/664983, Filling date: 23.03.2015, Patent number: US 9384417, Date of Patent: 5.07.2017.

[Domański_21] M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, J. Stankowski, R. Ratajczak, K. Wegner, "System do estymacji rozmiarów obiektu i sposób estymacji rozmiarów obiektu", Patent office: UPRP, Status: application, Application number: P.411656, Filling date: 21.03.2015.

[Domański_22] M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, J. Stankowski, R. Ratajczak, K. Wegner, "A system and method for object dimension estimation", Patent office: EPO, Status: granted, Application number: EP 15160202.06, Filling date: 23.03.2015, Publication number: EP 3073442, Publication date: 28.09.2016, Patent number: EP 3073442, Date of Patent: 01.03.2017.

## Other references


[3D-ATM] 3DV-ATM reference software v6.0, downloaded in 2012 from: *http://mpeg3dv.research.nokia.com/svn/mpeg3dv/tags/*

[3D-HTM] 3DV-HTM reference software v5.0, downloaded in 2012 from: *https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware*

[Arun_01] K.S. Arun, T.S. Huang, S.D. Blostein, “Least-Squares Fitting of Two 3-D Point Sets”, IEEE Transactions on Pattern Analysis, pp. 698–700, 1987.

[AVC_01] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", 04/2017.

[AVC_02] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", Annex H - "Multiview video coding", 04/2017.

[AVC_03] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", Annex I - "Multiview and depth video coding", 04/2017.

[AVC_04] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", Annex J - "Multiview and depth video with enhanced non-base view coding", 04/2017.

[Ballantine_01] J. P. Ballantine, A. R. Jerbert, "Distance from a line or plane to a point", American Mathematical Monthly, pp. 242–243, 1952.

[Bay_01] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359, 2008.

[Belhedi_01] A. Belhedi, S. Bourgeois, V. Gay-Bellile, P. Sayd, P. Bartoli, K. Hamrouni, "Non-parametric depth calibration of a TOF camera", 19th IEEE International Conference on Image Processing (ICIP), pp. 549-552, Sept. 30 2012-Oct. 3 2012.

[Besl_01] P.J. Besl, N. D. McKay, "A method for registration of 3-D shapes", IEEE Transactions on Pattern Analysis and Machine Intelligence vol.14, no.2, pp.239-256, February 1992.

[Bjontegaard_01] G. Bjontegaard, “Calculation of Average PSNR Differences between RD-curves,” ITU-T video Coding Experts Group document VCEG-M33, March 2001.

[Bolsee_01] Q. Bolsee and A. Munteanu, "Cnn-based Denoising of Time-Of-Flight Depth Images," 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 2018, pp. 510-514. doi: 10.1109/ICIP.2018.8451610

[Botezatu_01] N. Botezatu, S. Caraiman, D. Rzeszotarski and P. Strumillo, "Development of a versatile assistive system for the visually impaired based on sensor fusion," 2017 21st International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, 2017, pp. 540-547. doi: 10.1109/ICSTCC.2017.8107091

[Bovik_01] A.C. Bovik, “Handbook of Image and Video Processing”, Academic Press, July 2010, ISBN 9780080533612.

[Boyat_01] A. K. Boyat, B. K. Joshi, "A Review Paper: Noise Models in Digital Image Processing", Signal & Image Processing: An International Journal (SIPIJ) Vol.6, No.2, April 2015.

[Boykov_01] Y. Boykov, O. Veksler. R. Zabih, "Fast approximate energy minimisation via graph cuts", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.23, no.11, pp.1222,1239, November 2001.

[Brown_01] D. C. Brown. "Close-range camera calibration", Photogrammetric Engineering, 37(8):855–866, 1971.

[Buades_01] A. Buades, B. Coll, J.M. Morel, "A non-local algorithm for image denoising", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2005, vol.2, pp. 60-65, 20-25 June 2005.

[Calbaza_01] Calbaza, D.E.; Cordos, I.; Seth-Smith, N.; Savaria, Y., "An ADPLL circuit using a DDPS for Genlock applications," Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on , vol.4, no., pp.IV,569-72 Vol.4, 23-26 May 2004.

[CFP] "Call for Proposals on 3D Video Coding Technology", ISO/IEC JTC1/SC29/WG11, MPEG2011/W12036, Geneva, Switzerland, March 2011.

[Chiabrando_01] Chiabrando, F.; Chiabrando, R.; Piatti, D. “Sensors for 3D imaging: Metric evaluation and calibration of a CCD/CMOS time-of-flight camera”, Sensors 2009, 9, 10080–10096.

[Comaniciu_01] D. Comaniciu, P. Meer, “Mean shift: a robust approach toward feature space analysis”. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 24, no. 4, pp. 603–619, May 2002.

[Coxeter_01] H. S. M. Coxeter, "Barycentric Coordinates." §13.7 in Introduction to Geometry, 2nd ed. New York: Wiley, pp. 216-221, 1969.

[CTC] D. Rusanovskyy, K.Muller, A.Vetro, "Common Test Conditions of 3DV Core Experiments", ISO/IEC JTC1/SC29/WG11, MPEG2014/M27299, Shanghai, China, October 2012

[Cyganek_01] B. Cyganek, “Computer Processing of 3D images”, Academic Publisher House EXIT, ISBN 83-87674-34-6, Warsaw 2002.

[Dehkordi_01] A. Banitalebi-Dehkordi, M. T. Pourazad and P. Nasiopoulos, "3D video quality metric for 3D video compression," IVMSP 2013, Seoul, 2013, pp. 1-4. doi: 10.1109/IVMSPW.2013.6611930

[DERS] O. Stankiewicz, K. Wegner, M. Wildeboer, „A soft – segmentation matching in Depth Estimation Reference Software (DERS) 5.0″, ISO/IEC JTC1/SC29/WG11 MPEG2009/M17049, Xian, China, Oct. 2009.

[DIBR] C. Fehn, "Depth-image-based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV", in Proc. SPIE Conf. 5291, CA, U.S.A., pp. 93-104, Jan. 2004.

[Dwarakanath_01] D. Dwarakanath, A. Eichhorn, C. Griwodz, P. Halvorsen, "Faster and more accurate feature-based calibration for widely spaced camera pairs", Second International Conference on Digital Information and Communication Technology and its Applications (DICTAP) 2012, pp. 87-92, 16-18 May 2012.

[Eichhardt_01] I. Eichhardt, Z. Jankó and D. Chetverikov, "Novel methods for image-guided ToF depth upsampling," 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, 2016, pp. 002073-002078. doi: 10.1109/SMC.2016.7844545

[Evangelidis_01] G. D. Evangelidis, M. Hansard and R. Horaud, "Fusion of Range and Stereo Data for High-Resolution Scene-Modeling," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 11, pp. 2178-2192, 1 Nov. 2015. doi: 10.1109/TPAMI.2015.2400465

[Falie_01] D. Falie, V. Buzuloiu, "Distance errors correction for the Time of Flight (ToF) cameras", IEEE International Workshop on Imaging Systems and Techniques (IST), pp. 123-126, 10-12 Sept. 2008.

[Falie_02] D. Falie, V. Buzuloiu, "Distance errors correction for the time of flight (ToF) cameras", 4th European Conference on Circuits and Systems for Communications (ECCSC) 2008, pp. 193-196, 10-11 July 2008.

[Fischler_01] M.A. Fischler, R.C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". Comm. of the ACM 24 (6): 381–395, June 1981.

[Fuchs_01] S. Fuchs, G. Hirzinger, "Extrinsic and depth calibration of ToF-cameras", IEEE Conference on Computer Vision and Pattern Recognition, pp.1,6, 23-28 June 2008.

[Fukushima_01] N. Fukushima, "Icp With Depth Compensation For Calibration Of Multiple Tof Sensors," 2018 - 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Helsinki, 2018, pp. 1-4. doi: 10.1109/3DTV.2018.8478527

[Georgiev_01] M. Georgiev, A. Gotchev, M. Hannuksela, "Real-time denoising of ToF measurements by spatio-temporal non-local mean filtering", IEEE International Conference on Multimedia and Expo Workshops (ICMEW) 2013, pp. 1-6, 15-19 July 2013.

[Georgiev_02] M. Georgiev, A. Gotchev and M. Hannuksela, "Joint de-noising and fusion of 2D video and depth map sequences sensed by low-powered tof range sensor," 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), San Jose, CA, 2013, pp. 1-4. doi: 10.1109/ICMEW.2013.6618264

[Georgiev_03] M. Georgiev, R. Bregović and A. Gotchev, "Time-of-Flight Range Measurement in Low-Sensing Environment: Noise Analysis and Complex-Domain Non-Local Denoising," in IEEE Transactions on Image Processing, vol. 27, no. 6, pp. 2911-2926, June 2018. doi: 10.1109/TIP.2018.2807126

[Hansard_01] M. Hansard, S. Lee, O. Choi, R. Horaud, “Time of Flight Cameras: Principles, Methods, and Applications”, Springer Briefs in Computer Science, pp.95, Springer, 2012, ISBN 978-1-4471-4658-2

[Harris_01] C. Harris, M. Stephens, "A Combined Corner and Edge Detector", Alvey Vision Conference, 1988

[Hartley_01] R.Hartley, "Theory and Practice of Projective Rectification", International Journal of Computer Vision, vol. 35, no. 2, pp. 115-127, 1999.

[Hartley_02] R. Hartley, A.Zisserman, "Multiple View Geometry in Computer Vision", Cambridge University Press, March 2014, ISBN 0521540518, 9780521540513.

[Hecht_01] E. Hecht, “Optics (2nd edition)”, § 5.2.3, Addison Wesley, 1987

[Heikkila_01] J. Heikkila, "Geometric camera calibration using circular control points", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.22, no.10, pp. 1066,1077 ,October 2010.

[Herrera_01] D. Herrera, J. Kannala, J. Heikkilä, "Joint Depth and Color Camera Calibration with Distortion Correction,", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 10, pp. 2058-2064, October. 2012.

[HEVC_01] Recommendation ITU-T H.265, "High efficiency video coding", 02/2018.

[HEVC_02] Recommendation ITU-T H.265, "High efficiency video coding", Annex G - "Multiview high efficiency video coding", 02/2018.

[HEVC_03] G. Tech, K. Wegner, Y. Chen, S.Yea, "3D-HEVC Test Model 1", JCT-3V of ITU-T SG 16 WP 3 and ISO/IEC JTC1/SC29/WG11, Doc. JCT3V-A1005-d0, Stockholm, Sweden, 16-20 July, 2012.

[Hirschmuller_01] H. Hirschmuller, D. Scharstein, “Evaluation of cost functions for stereo matching”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, USA, June 2007.

[Huihuang_01] S.Huihuang, H. Bingwei, "A Simple Rectification Method of Stereo Image Pairs with Calibrated Cameras", 2nd International Conference onInformation Engineering and Computer Science (ICIECS) 2010, pp. 1-4, 25-26 December 2010.

[Huynh-Thu_01] Q. Huynh-Thu, M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment", Electronics Letters, vol. 44, no. 13, pp. 800-801, June 19 2008.

[IPC-7351] IPC-7351B, "Generic Requirements for Surface Mount Design and Land Pattern Standard", June 2010.

[IPC-SM-782] IPC-SM-782, "Surface Mount Design and Land Pattern Standard", 1993.

[Iqbal_01] J. L. M. Iqbal, S. S. Basha, "Real time 3D depth estimation and measurement of un-calibrated stereo and thermal images", 2017 International Conference on Nascent Technologies in Engineering (ICNTE), Navi, Mumbai, 2017, pp. 1-6.

[Jin-chao_01] L. Jin-chao, T. Hui-ming and L. Chao, "Noise Estimation in Video Surveillance Systems," 2009 WRI World Congress on Computer Science and Information Engineering, Los Angeles, CA, 2009, pp. 578-582. doi: 10.1109/CSIE.2009.541

[Jiyoung_01] J.Jiyoung, J.Yekeun, J.Park, H. Hyowon, K.J. Dokyoon, I.S. Kweon, "A novel 2.5D pattern for extrinsic calibration of ToF and camera fusion system", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011, pp. 3290, 3296, 25-30 September 2011.

[JTAG] "IEEE Standard for Reduced-Pin and Enhanced-Functionality Test Access Port and Boundary-Scan Architecture", IEEE Std. 1149.7-2009, pp.c1-985, February 10, 2010.

[Kahlmann_01] T. Kahlmann, F. Remondino, H. Ingensand, "Calibration for Increased Accuracy of the Range Imaging Camera Swissranger$^{TM}$", In proceedings of the ISPRS, Dresden, Germany, September 2006.

[Kang_01] Y.S. Kang, C. Lee; Y.S. Ho, "An Efficient Rectification Algorithm for Multi-View Images in Parallel Camera Array," 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video, pp. 61,64, 28-30 May 2008.

[Kang_02] Y.S. Kang, Y.S. Ho, "High-quality multi-view depth generation using multiple colour and depth cameras", 2010 IEEE International Conference on Multimedia and Expo (ICME), pp. 1405,1410, 19-23 July 2010.

[Kang_03] Y.S. Kang, Y.S. Ho, "Disparity map generation for colour image using TOF depth camera", 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp. 1,4, 16-18 May 2011.

[Kazmeir_01] L. J. Kazmeir, "Schaum's Outline of Business Statistics", McGraw Hill Professional, p. 359, 2003.

[Kim_01] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground-background segmentation using codebook model", Real-Time Imaging, vol. 11, no. 3, pp. 172–185, June 2005.

[Klim_01] K. Klimaszewski, K. Wegner, "Wpływ kompresji obrazów i map głębi na syntezę widoków w systemie wielowidokowym", Przegląd Telekomunikacyjny, pp. 380-383, No 6/2009.

[Klim_02] K. Klimaszewski, K. Wegner, M. Domanski, “Influence of views and depth compression onto quality of synthesized views”, ISO/IEC JTC1/SC29/WG11 MPEG Doc. M16758, London, UK, June 2009.

[Kovacs_01] J. Kovacs, “An Overview of Genlock”, MicroImage Video Systems, 2001.

[Lee_01] S. Lee, Y. Ho, "View-consistent multi-view depth estimation for three-dimensional video generation", 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp. 1,4, 7-9 June 2010.

[Lee_02] E.K. Lee; Y.S. Ho, "Generation of multi-view video using a fusion camera system for 3D displays". IEEE Transactions on Consumer Electronics, vol.56, no.4, pp. 2797-2805, November 2010.

[Liansheng_01] S. Liansheng, Z. Jiulong, C. Duwu, “Image Rectification Using Affine Epipolar Gemoetric Constraint”, International Symposium on Computer Science and Computational Technology (ISCSCT) 2008, vol. 2, pp. 582-588, 20-22 December 2008.

[Lin_01] G.Lin, X.Chen, W.Zhang, "A Robust Epipolar Rectification Method of Stereo Pairs", International Conference on Measuring Technology and Mechatronics Automation (ICMTMA) 2010, vol.1, pp. 322-326, 13-14 March 2010.

[Lindner_01] M. Lindner, A. Kolb, “Lateral and Depth Calibration of PMD-Distance Sensors”, Lecture Notes in Computer Science, vol. 4292, pp. 524-533, 2006.

[Liu_01] R. Liu, H. Zhang, M. Liu, X. Xia, T. Hu, "Stereo Cameras Self-Calibration Based on SIFT", International Conference on Measuring Technology and Mechatronics Automation (ICMTM) 2009, vol.1, pp. 352-355, 11-12 April 2009.

[Lowe_01] D.G. Lowe, "Object recognition from local scale-invariant features", Proceedings of the International Conference on Computer Vision 2, pp. 1150–1157, 1999.

[Ma_01] X. Ma, H. Tian, J. Tong, Z. Pan, "A 3D models acquiring method for complex surface objects", 2nd International Conference on Information Science and Engineering (ICISE) 2010, pp. 3392-3395, 4-6 Dec 2010.

[Matusiak_01] K. Matusiak, P. Skulimowski and P. Strumillo, "Improving matching performance of the keypoints in images of 3D scenes by using depth information," 2017 International Conference on Systems, Signals and Image Processing (IWSSIP), Poznan, 2017, pp. 1-5. doi: 10.1109/IWSSIP.2017.7965571

[Maxwell_01] J. Maxwell, “The Aesthetic Role of Depth of Field in Anamorphic Cinematography”, Film and Digital Times, Issue 61, June 2014.

[Mayer-Baese_01] U. Mayer-Baese, “Digital Signal Processing with Field Programmable Gate Array”, Springer-Verlag, Berlin, Heidelberg, 2004, ISBN: 978-3-662-06730-7.

[MesaImaging] Mesa Imaging company home page. Retrieved on 2011 from: *http://www.mesa-imaging.ch*

[Mori_01] Y. Mori, N. Fukushima, T. Yendo, T. Fujii and M. Tani- moto, “View Generation with 3D Warping Using Depth Information for FTV”, Signal Processing-Image Communication, Vol. 24, No. 1-2, 2009, pp. 65-72.

[MPEG2008/ N9468] “Call for Contributions on 3D Video Test Material”, ISO/IEC JTC1/SC29/WG11/N9468, October 2007.

[Mueller_01] M. Mueller, F. Zilly, C. Riechert, P. Kauff, "Spatio-temporal consistent depth maps from multi-view video", 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp. 1,4, 16-18 May 2011.

[Ningbo_01] G. Ningbo, C. Xiangning, J. Mingyong and L. Shengen, "A fusion method for TOF point cloud and digital photograph," 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, 2017, pp. 146-149. doi: 10.1109/ITOEC.2017.8122399

[Nozick_01] V. Nozick, "Multiple view image rectification", 1st International Symposium on Access Spaces (ISAS) 2011, pp. 277-282, 17-19 June 2011.

[Paris_01] S. Paris, P. Kornprobst, J. Tumblin, F. Durand, "A Gentle Introduction to Bilateral Filtering and its Applications", Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

[Perek_01] P. Perek, D. Makowski, A. Napieralski, "Efficient uncalibrated rectification method for stereo vision systems", 2016 MIXDES - 23rd International Conference Mixed Design of Integrated Circuits and Systems, Lodz, Poland, 2016, pp. 89-92.

[Pertile_01] M. Pertile, S. Chiodini, R. Giubilato and S. Debei, "Calibration of extrinsic parameters of a hybrid vision system for navigation comprising a very low resolution Time-of-Flight camera," 2017 IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace), Padua, 2017, pp. 391-396. doi: 10.1109/MetroAeroSpace.2017.7999604

[Plank_01] H. Plank, G. Holweg, T. Herndl and N. Druml, "High performance Time-of-Flight and color sensor fusion with image-guided depth super resolution," 2016 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, 2016, pp. 1213-1218.

[Porikli_01] F. Porikli, "Constant Time O(1) Bilateral Filtering", IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[Rakhshanfar_01] M. Rakhshanfar and M. A. Amer, "Estimation of Gaussian, Poissonian–Gaussian, and Processed Visual Noise and Its Level Function," in IEEE Transactions on Image Processing, vol. 25, no. 9, pp. 4172-4185, Sept. 2016. doi: 10.1109/TIP.2016.2588320

[RS232] EIA standard "RS-232-C: Interface between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Data Interchange", Washington Electronic Industries Association. Engineering Dept, 1969.

[Scharstein_01] D. Scharstein, C. Pal, “Learning conditional random fields for stereo”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, USA, June 2007.

[Scharstein_02] D. Scharstein, R. Szeliski, “High-accuracy stereo depth maps using structured light”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), vol. 1, pp. 195-202, Madison, WI, USA, June 2003.

[Scharstein_03] D. Scharstein, R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms”, International Journal of Computer Vision, 47(1/2/3):7-42, April-June 2002.

[Simanek_01] D. E. Simanek, “Mirror and Prism Methods for 3d Macro Photography”, Retrieved on 2011 from:

*http://www.lhup.edu/~dsimanek/3d/stereo/3dgallery16.htm*

[SMPTE240M] SMPTE standard 240M-1999, "1125-Line High-Definition Production Systems Signal Parameters".

[SMPTE274M] SMPTE standard 274M-2008, "1920 × 1080 Scanning and Analog and Parallel Digital Interfaces for Multiple Picture Rates".

[SR4000] Mesa Imaging SR4000 ToF camera datasheet. Retrieved on 2011 from: *http://www.mesa-imaging.ch/dlm.php?fname=pdf/SR4000_Data_Sheet.pdf*

[Stankiewicz_01] O. Stankiewicz, K. Wegner, M. Wildeboer, “A soft – segmentation matching in Depth Estimation Reference Software (DERS) 5.0”, ISO/IEC JTC1/SC29/WG11 MPEG 2009/M17049, Xian, China, October 2009.

[Stankowski_01] J. Stankowski, K. Klimaszewski, O. Stankiewicz, K. Wegner, M. Domański, "Preprocessing methods used for Poznan 3D/FTV test sequences", ISO/IEC JTC1/SC29/WG11 MPEG 2010 / M17174, Kyoto, Japan, January 2010.

[Stauffer_01] C. Stauffer, W. E. L. Grimson, “Adaptive background mixture models for real-time tracking”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 246–252, June 1999.

[Su_01] P. Su, R.L. Scot Drysdale, “A Comparison of Sequential Delaunay Triangulation Algorithms”, April 1996, Retrieved on 2011 from: *www.cs.berkeley.edu/~jrs/meshpapers/SuDrysdale.pdf*

[Tanimoto_01] M. Tanimoto, T.Fujii, M.P.Tehrani, N.Fukushima, K.Suzuki, M.Wildeboer, “Semi-automatic Depth Estimation for FTV”, ISO/IEC JCT1/SC29/WG11 MPEG 2009/M16391, Maui, USA, April 2009.

[Tektronix] Tektronix digital oscilloscope DPO4000 specification. Retrieved on 2011 from: *http://www.tek.com/oscilloscope/mso4000-dpo4000*

[Tikanmaki_01] A. Tikanmaki, A. Gotchev, A. Smolic and K. Miller, "Quality assessment of 3D video in rate allocation experiments," 2008 IEEE International Symposium on Consumer Electronics, Vilamoura, 2008, pp. 1-4. doi: 10.1109/ISCE.2008.4559441

[Tomasi_01] C. Tomasi, R. Manduchi, "Bilateral filtering for gray and color images", Proceedings of the International Conference on Computer Vision, pages 839–846. IEEE, 1998.

[VSRS] "View synthesis algorithm in view synthesis reference software 3.0 (VSRS3.0)", ISO/IEC JTC1/SC29/WG11 Doc. M16090, Feb. 2009.

[Wang_01] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity", IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April. 2004.

[Wang_01] Z. Wang, L. Lu, and A.C. Bovik, "Video quality assessment based on structural distortion measurement," Signal Processing: Image Communication, vol. 19, no. 2, pp. 121–132, 2004

[Wang_02] Y. Wang, "Survey of objective video quality measurements", Technical Report T1A1.5/96-110, Worcester Polytechnic Institute, 2006.

[Wegner_01] K. Wegner, O. Stankiewicz, M. Domański, "Stereoscopic depth estimation using fuzzy segment matching", 28th Picture Coding Symposium, PCS 2010, pp. 1-4, 8-10 December 2010, Nagoya, Japan.

[XH-G1] Canon XH-G1 camera datasheet, retrieved on 2011 from: *https://si.ua.es/es/lccm/documentos/av-camara-de-video-hdv-profesional-canon-xh-a1.pdf*

[Xu_01] P. Xu, Y. Ling, S. Li, "Research on camera calibration in track and field competition video", 4th IEEE International Conference on Software Engineering and Service Science (ICSESS) 2013, pp. 397-400, 23-25 May 2013.

[Yedida_01] J.S. Yedidia, W.T. Freeman, Y Weiss, "Understanding Belief Propagation and Its Generalisations" in Exploring Artificial Intelligence in the New Millennium, G. Lakemeyer, B. Nebel, ISBN: 1-55860-811-7, Chapter 8, pp. 239-236, Morgan Kaufmann Publishers, January 2003.

[Zhang_01] Z. Zhang. "A Flexible New Technique for Camera Calibration". IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11):1330-1334, 2000.

[3D-ATM] 3DV-ATM reference software v6.0, downloaded in 2012 from: *http://mpeg3dv.research.nokia.com/svn/mpeg3dv/tags/*

[3D-HTM] 3DV-HTM reference software v5.0, downloaded in 2012 from: *https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware*

[Arun_01] K.S. Arun, T.S. Huang, S.D. Blostein, “Least-Squares Fitting of Two 3-D Point Sets”, IEEE Transactions on Pattern Analysis, pp. 698–700, 1987.

[AVC_01] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", 04/2017.

[AVC_02] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", Annex H - "Multiview video coding", 04/2017.

[AVC_03] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", Annex I - "Multiview and depth video coding", 04/2017.

[AVC_04] Recommendation ITU-T H.264, "Advanced video coding for generic audiovisual services", Annex J - "Multiview and depth video with enhanced non-base view coding", 04/2017.

[Ballantine_01] J. P. Ballantine, A. R. Jerbert, "Distance from a line or plane to a point", American Mathematical Monthly, pp. 242–243, 1952.

[Bay_01] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359, 2008.

[Belhedi_01] A. Belhedi, S. Bourgeois, V. Gay-Bellile, P. Sayd, P. Bartoli, K. Hamrouni, "Non-parametric depth calibration of a TOF camera", 19th IEEE International Conference on Image Processing (ICIP), pp. 549-552, Sept. 30 2012-Oct. 3 2012.

[Besl_01] P.J. Besl, N. D. McKay, "A method for registration of 3-D shapes", IEEE Transactions on Pattern Analysis and Machine Intelligence vol.14, no.2, pp.239-256, February 1992.

[Bjontegaard_01] G. Bjontegaard, “Calculation of Average PSNR Differences between RD-curves,” ITU-T video Coding Experts Group document VCEG-M33, March 2001.

[Bolsee_01] Q. Bolsee and A. Munteanu, "Cnn-based Denoising of Time-Of-Flight Depth Images," 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 2018, pp. 510-514. doi: 10.1109/ICIP.2018.8451610

[Botezatu_01] N. Botezatu, S. Caraiman, D. Rzeszotarski and P. Strumillo, "Development of a versatile assistive system for the visually impaired based on sensor fusion," 2017 21st International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, 2017, pp. 540-547. doi: 10.1109/ICSTCC.2017.8107091

[Bovik_01] A.C. Bovik, “Handbook of Image and Video Processing”, Academic Press, July 2010, ISBN 9780080533612.

[Boyat_01] A. K. Boyat, B. K. Joshi, "A Review Paper: Noise Models in Digital Image Processing", Signal & Image Processing: An International Journal (SIPIJ) Vol.6, No.2, April 2015.

[Boykov_01] Y. Boykov, O. Veksler. R. Zabih, "Fast approximate energy minimisation via graph cuts", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.23, no.11, pp.1222,1239, November 2001.

[Brown_01] D. C. Brown. "Close-range camera calibration", Photogrammetric Engineering, 37(8):855–866, 1971.

[Buades_01] A. Buades, B. Coll, J.M. Morel, "A non-local algorithm for image denoising", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2005, vol.2, pp. 60-65, 20-25 June 2005.

[Calbaza_01] Calbaza, D.E.; Cordos, I.; Seth-Smith, N.; Savaria, Y., "An ADPLL circuit using a DDPS for Genlock applications," Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on , vol.4, no., pp.IV,569-72 Vol.4, 23-26 May 2004.

[CFP] "Call for Proposals on 3D Video Coding Technology", ISO/IEC JTC1/SC29/WG11, MPEG2011/W12036, Geneva, Switzerland, March 2011.

[Chiabrando_01] Chiabrando, F.; Chiabrando, R.; Piatti, D. “Sensors for 3D imaging: Metric evaluation and calibration of a CCD/CMOS time-of-flight camera”, Sensors 2009, 9, 10080–10096.

[Comaniciu_01] D. Comaniciu, P. Meer, “Mean shift: a robust approach toward feature space analysis”. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 24, no. 4, pp. 603–619, May 2002.

[Coxeter_01] H. S. M. Coxeter, "Barycentric Coordinates." §13.7 in Introduction to Geometry, 2nd ed. New York: Wiley, pp. 216-221, 1969.

[CTC] D. Rusanovskyy, K.Muller, A.Vetro, "Common Test Conditions of 3DV Core Experiments", ISO/IEC JTC1/SC29/WG11, MPEG2014/M27299, Shanghai, China, October 2012

[Cyganek_01] B. Cyganek, “Computer Processing of 3D images”, Academic Publisher House EXIT, ISBN 83-87674-34-6, Warsaw 2002.

[Dehkordi_01] A. Banitalebi-Dehkordi, M. T. Pourazad and P. Nasiopoulos, "3D video quality metric for 3D video compression," IVMSP 2013, Seoul, 2013, pp. 1-4. doi: 10.1109/IVMSPW.2013.6611930

[DERS] O. Stankiewicz, K. Wegner, M. Wildeboer, „A soft – segmentation matching in Depth Estimation Reference Software (DERS) 5.0″, ISO/IEC JTC1/SC29/WG11 MPEG2009/M17049, Xian, China, Oct. 2009.

[DIBR] C. Fehn, "Depth-image-based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV", in Proc. SPIE Conf. 5291, CA, U.S.A., pp. 93-104, Jan. 2004.

[Dwarakanath_01] D. Dwarakanath, A. Eichhorn, C. Griwodz, P. Halvorsen, "Faster and more accurate feature-based calibration for widely spaced camera pairs", Second International Conference on Digital Information and Communication Technology and its Applications (DICTAP) 2012, pp. 87-92, 16-18 May 2012.

[Eichhardt_01] I. Eichhardt, Z. Jankó and D. Chetverikov, "Novel methods for image-guided ToF depth upsampling," 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, 2016, pp. 002073-002078. doi: 10.1109/SMC.2016.7844545

[Evangelidis_01] G. D. Evangelidis, M. Hansard and R. Horaud, "Fusion of Range and Stereo Data for High-Resolution Scene-Modeling," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 11, pp. 2178-2192, 1 Nov. 2015. doi: 10.1109/TPAMI.2015.2400465

[Falie_01] D. Falie, V. Buzuloiu, "Distance errors correction for the Time of Flight (ToF) cameras", IEEE International Workshop on Imaging Systems and Techniques (IST), pp. 123-126, 10-12 Sept. 2008.

[Falie_02] D. Falie, V. Buzuloiu, "Distance errors correction for the time of flight (ToF) cameras", 4th European Conference on Circuits and Systems for Communications (ECCSC) 2008, pp. 193-196, 10-11 July 2008.

[Fischler_01] M.A. Fischler, R.C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". Comm. of the ACM 24 (6): 381–395, June 1981.

[Fuchs_01] S. Fuchs, G. Hirzinger, "Extrinsic and depth calibration of ToF-cameras", IEEE Conference on Computer Vision and Pattern Recognition, pp.1,6, 23-28 June 2008.

[Fukushima_01] N. Fukushima, "Icp With Depth Compensation For Calibration Of Multiple Tof Sensors," 2018 - 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Helsinki, 2018, pp. 1-4. doi: 10.1109/3DTV.2018.8478527

[Gargoum_01] S. Gargoum and K. El-Basyouny, "Automated extraction of road features using LiDAR data: A review of LiDAR applications in transportation," 2017 4th International Conference on Transportation Information and Safety (ICTIS), Banff, AB, 2017, pp. 563-574. doi: 10.1109/ICTIS.2017.8047822

[Georgiev_01] M. Georgiev, A. Gotchev, M. Hannuksela, "Real-time denoising of ToF measurements by spatio-temporal non-local mean filtering", IEEE International Conference on Multimedia and Expo Workshops (ICMEW) 2013, pp. 1-6, 15-19 July 2013.

[Georgiev_02] M. Georgiev, A. Gotchev and M. Hannuksela, "Joint de-noising and fusion of 2D video and depth map sequences sensed by low-powered tof range sensor," 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), San Jose, CA, 2013, pp. 1-4. doi: 10.1109/ICMEW.2013.6618264

[Georgiev_03] M. Georgiev, R. Bregović and A. Gotchev, "Time-of-Flight Range Measurement in Low-Sensing Environment: Noise Analysis and Complex-Domain Non-Local Denoising," in IEEE Transactions on Image Processing, vol. 27, no. 6, pp. 2911-2926, June 2018. doi: 10.1109/TIP.2018.2807126

[Glennie_01] C.L. Glennie, W. E. Carter, R. L. Shrestha, W. E. Dietrich, "Geodetic imaging with airborne LiDAR: the Earth's surface revealed", Reports on Progress in Physics 76(8):086801, July 2013, doi: 10.1088/0034-4885/76/8/086801

[Hansard_01] M. Hansard, S. Lee, O. Choi, R. Horaud, "Time of Flight Cameras: Principles, Methods, and Applications", Springer Briefs in Computer Science, pp.95, Springer, 2012, ISBN 978-1-4471-4658-2

[Harris_01] C. Harris, M. Stephens, "A Combined Corner and Edge Detector", Alvey Vision Conference, 1988

[Hartley_01] R.Hartley, "Theory and Practice of Projective Rectification", International Journal of Computer Vision, vol. 35, no. 2, pp. 115-127, 1999.

[Hartley_02] R. Hartley, A.Zisserman, "Multiple View Geometry in Computer Vision", Cambridge University Press, March 2014, ISBN 0521540518, 9780521540513.

[Hecht_01] E. Hecht, "Optics (2nd edition)", § 5.2.3, Addison Wesley, 1987

[Heikkila_01] J. Heikkila, "Geometric camera calibration using circular control points", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.22, no.10, pp. 1066,1077 ,October 2010.

[Herrera_01] D. Herrera, J. Kannala, J. Heikkilä, "Joint Depth and Color Camera Calibration with Distortion Correction,", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 10, pp. 2058-2064, October. 2012.

[HEVC_01] Recommendation ITU-T H.265, "High efficiency video coding", 02/2018.

[HEVC_02] Recommendation ITU-T H.265, "High efficiency video coding", Annex G - "Multiview high efficiency video coding", 02/2018.

[HEVC_03] G. Tech, K. Wegner, Y. Chen, S.Yea, "3D-HEVC Test Model 1", JCT-3V of ITU-T SG 16 WP 3 and ISO/IEC JTC1/SC29/WG11, Doc. JCT3V-A1005-d0, Stockholm, Sweden, 16-20 July, 2012.

[Hirschmuller_01] H. Hirschmuller, D. Scharstein, "Evaluation of cost functions for stereo matching", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, USA, June 2007.

[Horaud_01] R. Horaud, M. Hansard, G. Evangelidis, C. Ménier, "An Overview of Depth Cameras and Range Scanners Based on Time-of-Flight Technologies", Machine Vision and Applications Journal, 2016, 27 (7), pp.1005-1020.

[Huihuang_01] S.Huihuang, H. Bingwei, "A Simple Rectification Method of Stereo Image Pairs with Calibrated Cameras", 2nd International Conference onInformation Engineering and Computer Science (ICIECS) 2010, pp. 1-4, 25-26 December 2010.

[Huynh-Thu_01] Q. Huynh-Thu, M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment", Electronics Letters, vol. 44, no. 13, pp. 800-801, June 19 2008.

[IPC-7351] IPC-7351B, "Generic Requirements for Surface Mount Design and Land Pattern Standard", June 2010.

[IPC-SM-782] IPC-SM-782, "Surface Mount Design and Land Pattern Standard", 1993.

[Iqbal_01] J. L. M. Iqbal, S. S. Basha, "Real time 3D depth estimation and measurement of un-calibrated stereo and thermal images", 2017 International Conference on Nascent Technologies in Engineering (ICNTE), Navi, Mumbai, 2017, pp. 1-6.

[Jin-chao_01] L. Jin-chao, T. Hui-ming and L. Chao, "Noise Estimation in Video Surveillance Systems," 2009 WRI World Congress on Computer Science and Information Engineering, Los Angeles, CA, 2009, pp. 578-582. doi: 10.1109/CSIE.2009.541

[Jiyoung_01] J.Jiyoung, J.Yekeun, J.Park, H. Hyowon, K.J. Dokyoon, I.S. Kweon, “A novel 2.5D pattern for extrinsic calibration of ToF and camera fusion system", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011, pp. 3290, 3296, 25-30 September 2011.

[JTAG] "IEEE Standard for Reduced-Pin and Enhanced-Functionality Test Access Port and Boundary-Scan Architecture", IEEE Std. 1149.7-2009, pp.c1-985, February 10, 2010.

[Kahlmann_01] T. Kahlmann, F. Remondino, H. Ingensand, “Calibration for Increased Accuracy of the Range Imaging Camera Swissranger™”, In proceedings of the ISPRS, Dresden, Germany, September 2006.

[Kang_01] Y.S. Kang, C. Lee; Y.S. Ho, "An Efficient Rectification Algorithm for Multi-View Images in Parallel Camera Array," 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video, pp. 61,64, 28-30 May 2008.

[Kang_02] Y.S. Kang, Y.S. Ho, “High-quality multi-view depth generation using multiple colour and depth cameras”, 2010 IEEE International Conference on Multimedia and Expo (ICME), pp. 1405,1410, 19-23 July 2010.

[Kang_03] Y.S. Kang, Y.S. Ho, ”Disparity map generation for colour image using TOF depth camera”, 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp. 1,4, 16-18 May 2011.

[Kazmeir_01] L. J. Kazmeir, "Schaum's Outline of Business Statistics", McGraw Hill Professional, p. 359, 2003.

[Kim_01] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Real-time foreground-background segmentation using codebook model”, Real-Time Imaging, vol. 11, no. 3, pp. 172–185, June 2005.

[Klim_01] K. Klimaszewski, K. Wegner, “Wpływ kompresji obrazów i map głębi na syntezę widoków w systemie wielowidokowym”, Przegląd Telekomunikacyjny, pp. 380-383, No 6/2009.

[Klim_02] K. Klimaszewski, K. Wegner, M. Domanski, “Influence of views and depth compression onto quality of synthesized views”, ISO/IEC JTC1/SC29/WG11 MPEG Doc. M16758, London, UK, June 2009.

[Kovacs_01] J. Kovacs, “An Overview of Genlock”, MicroImage Video Systems, 2001.

[Lee_01] S. Lee, Y. Ho, "View-consistent multi-view depth estimation for three-dimensional video generation", 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp. 1,4, 7-9 June 2010.

[Lee_02] E.K. Lee; Y.S. Ho, "Generation of multi-view video using a fusion camera system for 3D displays". IEEE Transactions on Consumer Electronics, vol.56, no.4, pp. 2797-2805, November 2010.

[Liansheng_01] S. Liansheng, Z. Jiulong, C. Duwu, “Image Rectification Using Affine Epipolar Gemoetric Constraint”, International Symposium on Computer Science and Computational Technology (ISCSCT) 2008, vol. 2, pp. 582-588, 20-22 December 2008.

[Lin_01] G.Lin, X.Chen, W.Zhang, "A Robust Epipolar Rectification Method of Stereo Pairs", International Conference on Measuring Technology and Mechatronics Automation (ICMTMA) 2010, vol.1, pp. 322-326, 13-14 March 2010.

[Lindner_01] M. Lindner, A. Kolb, “Lateral and Depth Calibration of PMD-Distance Sensors”, Lecture Notes in Computer Science, vol. 4292, pp. 524-533, 2006.

[Liu_01] R. Liu, H. Zhang, M. Liu, X. Xia, T. Hu, "Stereo Cameras Self-Calibration Based on SIFT", International Conference on Measuring Technology and Mechatronics Automation (ICMTM) 2009, vol.1, pp. 352-355, 11-12 April 2009.

[Liu_01] J. Liu, Q. Sun, Z. Fan and Y. Jia, "TOF Lidar Development in Autonomous Vehicle," 2018 IEEE 3rd Optoelectronics Global Conference (OGC), Shenzhen, 2018, pp. 185-190. doi: 10.1109/OGC.2018.852999

[Lowe_01] D.G. Lowe, "Object recognition from local scale-invariant features", Proceedings of the International Conference on Computer Vision 2, pp. 1150–1157, 1999.

[Ma_01] X. Ma, H. Tian, J. Tong, Z. Pan, "A 3D models acquiring method for complex surface objects", 2nd International Conference on Information Science and Engineering (ICISE) 2010, pp. 3392-3395, 4-6 Dec 2010.

[Matusiak_01] K. Matusiak, P. Skulimowski and P. Strumillo, "Improving matching performance of the keypoints in images of 3D scenes by using depth information," 2017 International Conference on Systems, Signals and Image Processing (IWSSIP), Poznan, 2017, pp. 1-5. doi: 10.1109/IWSSIP.2017.7965571

[Maxwell_01] J. Maxwell, "The Aesthetic Role of Depth of Field in Anamorphic Cinematography", Film and Digital Times, Issue 61, June 2014.

[Mayer-Baese_01] U. Mayer-Baese, "Digital Signal Processing with Field Programmable Gate Array", Springer-Verlag, Berlin, Heidelberg, 2004, ISBN: 978-3-662-06730-7.

[MesaImaging] Mesa Imaging company home page. Retrieved on 2011 from: *http://www.mesa-imaging.ch*

[Mieloch_01] D. Mieloch, "Depth Estimation in Free-Viewpoint Television" (PhD dissertation), Poznan University of Technology, Faculty of Multimedia Telecommunication and Microelectronics, Poznań, 2018.

[Mori_01] Y. Mori, N. Fukushima, T. Yendo, T. Fujii and M. Tani- moto, "View Generation with 3D Warping Using Depth Information for FTV", Signal Processing-Image Communication, Vol. 24, No. 1-2, 2009, pp. 65-72.

[MPEG2008/ N9468] "Call for Contributions on 3D Video Test Material", ISO/IEC JTC1/SC29/WG11/N9468, October 2007.

[Mueller_01] M. Mueller, F. Zilly, C. Riechert, P. Kauff, "Spatio-temporal consistent depth maps from multi-view video", 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), pp. 1,4, 16-18 May 2011.

[Ningbo_01] G. Ningbo, C. Xiangning, J. Mingyong and L. Shengen, "A fusion method for TOF point cloud and digital photograph," 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, 2017, pp. 146-149. doi: 10.1109/ITOEC.2017.8122399

[Nozick_01] V. Nozick, "Multiple view image rectification", 1st International Symposium on Access Spaces (ISAS) 2011, pp. 277-282, 17-19 June 2011.

[Paris_01] S. Paris, P. Kornprobst, J. Tumblin, F. Durand, "A Gentle Introduction to Bilateral Filtering and its Applications", Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

[Perek_01] P. Perek, D. Makowski, A. Napieralski, "Efficient uncalibrated rectification method for stereo vision systems", 2016 MIXDES - 23rd International Conference Mixed Design of Integrated Circuits and Systems, Lodz, Poland, 2016, pp. 89-92.

[Pertile_01] M. Pertile, S. Chiodini, R. Giubilato and S. Debei, "Calibration of extrinsic parameters of a hybrid vision system for navigation comprising a very low resolution Time-of-Flight camera," 2017 IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace), Padua, 2017, pp. 391-396. doi: 10.1109/MetroAeroSpace.2017.7999604

[Plank_01] H. Plank, G. Holweg, T. Herndl and N. Druml, "High performance Time-of-Flight and color sensor fusion with image-guided depth super resolution," 2016 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, 2016, pp. 1213-1218.

[Porikli_01] F. Porikli, "Constant Time O(1) Bilateral Filtering", IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[Rakhshanfar_01] M. Rakhshanfar and M. A. Amer, "Estimation of Gaussian, Poissonian–Gaussian, and Processed Visual Noise and Its Level Function," in IEEE Transactions on Image Processing, vol. 25, no. 9, pp. 4172-4185, Sept. 2016. doi: 10.1109/TIP.2016.2588320

[RS232] EIA standard "RS-232-C: Interface between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Data Interchange", Washington Electronic Industries Association. Engineering Dept, 1969.

[Scharstein_01] D. Scharstein, C. Pal, “Learning conditional random fields for stereo”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, USA, June 2007.

[Scharstein_02] D. Scharstein, R. Szeliski, “High-accuracy stereo depth maps using structured light”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), vol. 1, pp. 195-202, Madison, WI, USA, June 2003.

[Scharstein_03] D. Scharstein, R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms”, International Journal of Computer Vision, 47(1/2/3):7-42, April-June 2002.

[Simanek_01] D. E. Simanek, “Mirror and Prism Methods for 3d Macro Photography”, Retrieved on 2011 from:

*http://www.lhup.edu/~dsimanek/3d/stereo/3dgallery16.htm*

[Sitnik_01] R. Sitnik, E. Bunsch, G. Maczkowski, W. Zaluski, K. Lech, J. Michonski, J. Krzeslowski, P. Forys, "Towards automated, high resolution 3D scanning of large surfaces for cultural heritage documentation", 3D Image Processing Measurement (3DIPM) and Applications, 2016, doi: 10.2352/ISSN.2470-1173.2016.21.3DIPM-051

[SMPTE240M] SMPTE standard 240M-1999, "1125-Line High-Definition Production Systems Signal Parameters".

[SMPTE274M] SMPTE standard 274M-2008, "1920 × 1080 Scanning and Analog and Parallel Digital Interfaces for Multiple Picture Rates".

[SR4000] Mesa Imaging SR4000 ToF camera datasheet. Retrieved on 2011 from: *http://www.mesa-imaging.ch/dlm.php?fname=pdf/SR4000_Data_Sheet.pdf*

[Stankiewicz_01] O. Stankiewicz, K. Wegner, M. Wildeboer, "A soft – segmentation matching in Depth Estimation Reference Software (DERS) 5.0", ISO/IEC JTC1/SC29/WG11 MPEG 2009/M17049, Xian, China, October 2009.

[Stankowski_01] J. Stankowski, K. Klimaszewski, O. Stankiewicz, K. Wegner, M. Domański, "Preprocessing methods used for Poznan 3D/FTV test sequences", ISO/IEC JTC1/SC29/WG11 MPEG 2010 / M17174, Kyoto, Japan, January 2010.

[Stauffer_01] C. Stauffer, W. E. L. Grimson, "Adaptive background mixture models for real-time tracking", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 246–252, June 1999.

[Sterp_01] E. Bebeselea-Sterp, R. Brad, R. Brad, "A Comparative Study of Stereovision Algorithms", (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 8, no. 11, 2017.

[Su_01] P. Su, R.L. Scot Drysdale, "A Comparison of Sequential Delaunay Triangulation Algorithms", April 1996, Retrieved on 2011 from: *www.cs.berkeley.edu/~jrs/meshpapers/SuDrysdale.pdf*

[Szeląg_01] K. Szeląg, G. Maczkowski, R. Gierwiało, A. Gebarska, R. Sitnik, "Robust geometric, phase and color structured light projection system calibration", Opto-Electronics Review 25 (4), 326-336

[Tanimoto_01] M. Tanimoto, T.Fujii, M.P.Tehrani, N.Fukushima, K.Suzuki, M.Wildeboer, "Semi-automatic Depth Estimation for FTV", ISO/IEC JCT1/SC29/WG11 MPEG 2009/M16391, Maui, USA, April 2009.

[Tektronix] Tektronix digital oscilloscope DPO4000 specification. Retrieved on 2011 from: *http://www.tek.com/oscilloscope/mso4000-dpo4000*

[Tikanmaki_01] A. Tikanmaki, A. Gotchev, A. Smolic and K. Miller, "Quality assessment of 3D video in rate allocation experiments," 2008 IEEE International Symposium on Consumer Electronics, Vilamoura, 2008, pp. 1-4. doi: 10.1109/ISCE.2008.4559441

[Tomasi_01] C. Tomasi, R. Manduchi, "Bilateral filtering for gray and color images", Proceedings of the International Conference on Computer Vision, pages 839–846. IEEE, 1998.

[VSRS] "View synthesis algorithm in view synthesis reference software 3.0 (VSRS3.0)", ISO/IEC JTC1/SC29/WG11 Doc. M16090, Feb. 2009.

[Wang_01] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity", IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April. 2004.

[Wang_01] Z. Wang, L. Lu, and A.C. Bovik, “Video quality assessment based on structural distortion measurement,” Signal Processing: Image Communication, vol. 19, no. 2, pp. 121–132, 2004

[Wang_02] Y. Wang, "Survey of objective video quality measurements", Technical Report T1A1.5/96-110, Worcester Polytechnic Institute, 2006.

[Wegner_01] K. Wegner, O. Stankiewicz, M. Domański, “Stereoscopic depth estimation using fuzzy segment matching”, 28th Picture Coding Symposium, PCS 2010, pp. 1-4, 8-10 December 2010, Nagoya, Japan.

[XH-G1] Canon XH-G1 camera datasheet, retrieved on 2011 from: *https://si.ua.es/es/lccm/documentos/av-camara-de-video-hdv-profesional-canon-xh-a1.pdf*

[Xu_01] P. Xu, Y. Ling, S. Li, "Research on camera calibration in track and field competition video", 4th IEEE International Conference on Software Engineering and Service Science (ICSESS) 2013, pp. 397-400, 23-25 May 2013.

[Yedida_01] J.S. Yedidia, W.T. Freeman, Y Weiss, "Understanding Belief Propagation and Its Generalisations" in Exploring Artificial Intelligence in the New Millennium, G. Lakemeyer, B. Nebel, ISBN: 1-55860-811-7, Chapter 8, pp. 239-236, Morgan Kaufmann Publishers, January 2003.

[Zhang_01] Z. Zhang. "A Flexible New Technique for Camera Calibration". IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11):1330-1334, 2000.