TECHNISCHE UNIVERSITÄT WIEN
Vienna | Austria

ACIN
AUTOMATION & CONTROL INSTITUTE
INSTITUT FÜR AUTOMATISIERUNGS- & REGELUNGSTECHNIK

Stereo Vision for Obstacle Detection in Robotic Applications

DIPLOMARBEIT

Conducted in partial fulfillment of the requirements for the degree of a

Diplom-Ingenieur (Dipl.-Ing.)

supervised by

Ao.Univ.Prof. Dipl.-Ing Dr. techn. M. Vincze
Dipl.-Ing G. Halmetschlager-Funek

submitted at the

TU Wien
Faculty of Electrical Engineering and Information Technology
Automation and Control Institute

by
Bernhard Neuberger BSc.
Barichgasse 23/2/13
1030 Vienna
Austria

Vienna, June 2017

Vision for Robotics (V4R) Group
A-1040 Wien, Gusshausstr. 27, Internet: http://www.acin.tuwien.ac.at

Abstract

This thesis presents an obstacle detection algorithm and compares different stereo algorithms for robotic applications. Arising problems, such as reflections of sunlight on surfaces and problems with roll angles, are pointed out during the use of the obstacle detection algorithm in an robotic application.

The obstacle detection algorithm in this thesis uses disparity images as input and calculates the v-disparity image, which is used for floor detection. The step of floor detection makes also use of the Hough transform. Roll angle detection and correction is presented. The approach of multi v-disparity is created to compensate the problems caused by a roll angle. Noise reduction methods are discussed as well as an approach to deal with bright reflections.

The obstacle detection is tested on different experiments that are presented. The results show that the problems with the roll angle are eliminated due to the approach of roll angle detection and correction that is presented in this thesis. Wrong detections caused by reflections that appear due to bright illumination are also removed by the obstacle detection algorithm. The evaluation shows that obstacles with a height of 2cm are detectable up to a distance of 1m.

Kurzzusammenfassung

Diese Arbeit behandelt die Implementierung eines Algorithmus zur Objekterkennung im Bereich der Robotik. Dafür wird eine Stereokamera eingesetzt. Zudem werden unterschiedliche Stereo-Algorithmen verglichen.

Der präsentierte Algorithmus zur Objekterkennung verwendet dabei die Information der Stereokamera, welche räumlich versetzte Bilder aufnimmt. Der Unterschied zwischen den beiden Bildern gibt Auskunft über die Tiefe der abgebildeten Objekte. Dadurch wird die Objekterkennung ermöglicht.

Die präsentierte Methode verwendet die sogenannte v- Disparity um den Boden zu erkennen. Dafür wird zusätzlich die Hough Transformation eingesetzt. Bei der praktischen Anwendung des Algorithmus zur Objekterkennung kommt es zu falsch detektierten Objekten, welche durch Lichtreflektionen in stark belichteten Räumen entstehen. Zur Vermeidung dieses Problems wurde eine Strategie entwickelt, welche in dieser Arbeit vorgestellt wird.

Das Auftreten eines Rollwinkels der Stereokamera erschwert außerdem eine zuverlässige Objekterkennung. Hierfür wird eine Methode gezeigt, wie der Rollwinkel erfasst und in weiterer Folge korrigiert werden kann. Dieses Problem kann zudem durch die Verwendung mehrerer v- Disparity-Bilder vermindert werden. Es werden zusätzlich Möglichkeiten zur Reduzierung des Rauschens beschrieben.

Der Algorithmus zur Objekterkennung wurde an diversen Experimenten getestet. Die Resultate zeigen, dass die durch den Rollwinkel verursachten Probleme mithilfe der entwickelten Methoden beseitigt werden. Falsch erkannte Objekte, welche durch Lichtreflektionen in stark belichteten Räumen entstehen, können durch den Algorithmus entfernt werden. Die Auswertung der Daten zeigt, dass Objekte mit einer Höhe von 2cm auf einen Abstand von 1m erfasst werden.

Contents

List of Figures

List of Tables

1 Introduction

An autonomous robot requires the ability to navigate safely for an error-free interaction with its environment. An unwary action or movement of the robot may lead to danger for the robot itself, for the surrounding environment or, in the worst case, for any human interacting with the robot. In order to operate safely a robot needs to be aware of its environment and plan its actions depending on its surroundings.

A way for the robot to obtain awareness of the environment is through the use of a stereo vision system. This system delivers an image pair from cameras and then adds depth information to the recorded scene. This helps the robot to complete superior tasks, such as moving from one point to another or grasping objects.

Even if the robot knows its surrounding area well from past experiences and uses a map for navigation it is still necessary to obtain current information on the area, as it is possible that a new obstacle has appeared since then. When the robot fails to change its navigation properly to avoid an obstacle it may result in a crash. Such a scenario can cause damage in any form and must be avoided.

Another research topic that requires awareness of the environment is a driving assistance system. It warns the driver of possible danger and needs to be able to control an autonomous, evasive move in risky situations. In order to do this task successfully detection of obstacles is required, and can be done with a stereo vision system.

1.1 Motivation

A safe robot navigation arouses the need for reliable obstacle detection. It is necessary to plan a path in such a way that robots can navigate through it without danger. The use of a stereo vision system allows the sensing of the environment. However, for reliable use the limits of such a system need to be tested. The knowledge of the size of detectable obstacles enables the developer to select the appropriate stereo vision system for the task at hand.

A stereo vision system can be compared to human vision. Just like human vision, the stereo vision system in a robotic application allows an estimation of

the depth of objects in the nearby environment. In order to do this a scene has to be observed and needs to be classified in some way as well. One of these classifications is the detection of potential obstacles.

Taking safety and reliability into consideration, it can be pointed out that the recognition of possible danger from obstacles is a substantial requirement for a robot, most notably if it is intended to act autonomously. Humans have developed quite a few senses for spotting danger. One of the most relied upon senses for that purpose is human vision. It helps us to perceive the environment and plan actions accordingly. A robot requires a similar type of perception for achieving awareness of its surroundings. The robot should be able to work precisely, should not harm anybody around it and should be able to stay away from threats to itself. Stereo vision can be used to obtain this awareness of the robot's surrounding environment.

Cameras are used as a vision system to get this awareness. Machine vision is able to lend meaning to the generated images and interpret scenes for further tasks. Stereo vision improves this awareness by calculating the depth of points in images.

The requirements of a robotic vision system are dependent on the application. It is important to figure out if the stereo vision system should prioritize speed, precision, or reliability, and to what degree it is possible to maximize them all.

According to Kragic, Vincze, et al. [1], object detection is still an "open challenge," along with a number of other tasks that are important for a robot vision system. The tasks of a robot vision system are represented by the blocks in Figure 1.1.

1.2 Problems and Goals

One common challenge of stereo vision systems is the handling of noisy depth values. Due to different influencing factors, such as illumination and discretization, noise will be unavoidable. It restricts the detection of small objects because up to a certain size small obstacles are indistinguishable from noisy data. However, if the characteristics of the noise are well known the performance of a stereo system can be tuned accordingly. It is necessary to find a way to test the limits of a stereo system for obstacle detection. For an accurate characterization of the use of a stereo system it is important to know the dimensions of detectable objects and at which distance the detection works properly. This characterization helps developers to choose the right stereo system for an application.

Another important part is the use of the right stereo algorithm, which the purpose is the calculation of a disparity image. There are many different

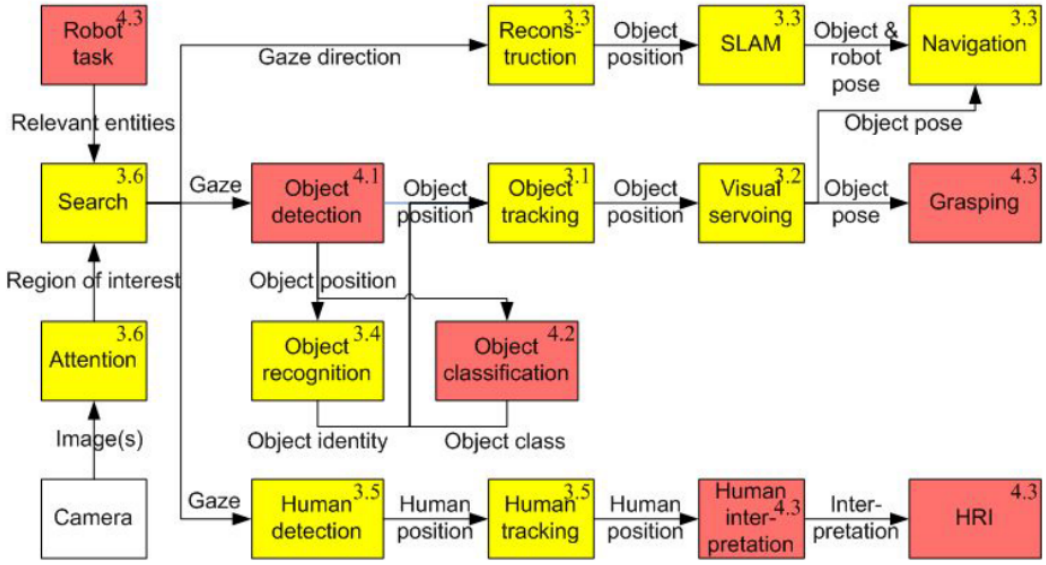


Figure 1.1: Main tasks of a robot vision system. Yellow blocks indicate working tasks and red blocks indicate "Open challanges" according to [1].

possibilities for designing an algorithm for disparity calculation. Every stereo algorithm has different requirements for a purposeful use. One of the most important requirements is the speed of the stereo algorithm's computation time because a slow pace would restrict that of the obstacle detection process. This thesis describes research on the advantages and disadvantages of different stereo algorithms in order to achieve a better understanding of the aforementioned influencing factors.

The problem of obstacle detection will not be the same for every environment. Different conditions of a robotic environment can lead to problems for object detection. Every scene presents its own unique set of challenges. However, being familiar with the common problem-causing factors can help one to anticipate which difficulties will most likely arise. The goal is to find and solve these typical problems with the use of a stereo vision system in a robotic application.

The goals of the thesis can be listed as follows:

- Implementation of an obstacle detection algorithm within a stereo vision system
- Testing the accuracy and reliability of the object detection
- Comparison of different stereo algorithms

- Finding solutions for typical problems in a stereo vision system

1.3 Approach Overview

For reliable detection of obstacles, an obstacle detection algorithm has been developed. In order to evaluate the reliability of the algorithm it is evaluated on a number of experiments for the purpose of robot navigation. The experiments are used to identify potential weaknesses of the obstacle detection algorithm. This allows one to improve it through adjustments and then evaluate those adjustments on a number of additional experiments.

One of the weaknesses arising from the approach suggested in this thesis is caused by the existing roll angle. Due to this fact, strategies to account for the roll angle are implemented. Automatic detection and correction help to solve this issue.

Another problem is caused by reflections of light on the floor, which are often responsible for wrongly identified obstacles. It is possible to detect those spots and remove them accordingly. In order to solve the problems assumptions, such as a flat floor in the environment, are made. The assumption of a flat floor allows detection through the use of the Hough transform. For the roll angle correction it is assumed that a certain area in the observed space of the stereo vision system is free from obstacles.

1.4 Structure of Work

In Chapter 2 related state-of-the-art research topics are presented and compared to this thesis.

The theory of stereo vision is then described in Chapter 3. The starting point of that chapter presents the functionality of cameras and a brief description of the physics behind cameras. The composition of a stereo vision system and how it makes depth calculation possible is discussed. Detailed information of the geometry of a stereo vision system is given. This helps one to understand the rectification process, which is always a component of stereo vision systems. Furthermore, an overview of stereo algorithms for disparity calculation is given.

In Chapter 4 the v-disparity [2] that represents a flat floor as a line is introduced. In order to detect this line the Hough transform is used and discussed. The floor detection in the v-disparity image enables one to remove the floor from the disparity image and make it possible to detect obstacles. Issues, such as problematic roll angles and complications caused by noise, are described. Strategies to minimize these problems are recorded. A comparison of

different stereo algorithms is also given. Finally, the obstacle detection process is described in Chapter 4.

The algorithm is evaluated on real-life data in Chapter 5. A noise-measuring experiment that helps to set the parameters of the obstacle detection algorithm properly is presented. Another experiment is used to show different problems for robotic indoor scenes using the obstacle detection algorithm. The limits of the stereo vision system and how different stereo algorithms compare to each other are tested. The strategies developed in Chapter 4 for the problems associated with the roll angle and bright reflections of light on the floor are tested and evaluated.

Chapter 6 summarizes the results of this thesis and gives an overview of possible future research.

2 State of the Art

In this chapter related research is presented. An overview of specific research topics is given with each of them summarized. Most of the presented topics in this chapter show similar tasks such as the v-disparity calculation and a roll angle correction, both of them are necessary for successful detection of surfaces and obstacles. The main application of the presented methods is the use for advanced driving assistance system (ADAS) and an autonomous navigation for robots. In addition the advantages and disadvantages of different stereo algorithms are discussed. Some of the presented research topics explain also the mapping of obstacles that is needed for the process of path planning.

The occurring problems, such as difficulties under changing lighting conditions and a correct floor detection are present in this research topic and overlap in the studies. These topics occur in this thesis as well. One of the problems is the dependancy of the v-disparity from the roll-angle of the stereo system. In Section 4.2.2 this is discussed in detail and solutions are presented. Another similarity between the presented scientific papers is the dependence on the application of an obstacle detection algorithm. The focus on rather small obstacles is not that important in ADAS but the presence of curvy and hilly roads results in slightly different approaches when compared to a robot in an indoor environment. In general it is important to know the environment of the possible tasks of a driving assistance system or an autonomous system. This can lead either to some sort of simplification or problems to be solved. A robotic indoor environment can be better classified if assumptions of the floor are taken. Because these assumptions make the floor detection easier to calculate and therefore a faster floor detection can be achieved. The lighting conditions need to be taken into account because the functionality of a camera is highly dependant from it. If the lighting is not handled properly a stereo system may have problems to calculate the disparity image and this results in a wrong perception of the environment in a robotic task.

Another important step is choosing the right stereo algorithm for disparity calculation and how to classify them properly. The interests of speed, precision and stability against changing environment, such as change in lighting or change of terrain, are important for any robotic application. However it is possible that an improvement in any of those interests results in a decline of the performance of another characteristic. For example if an improvement of a stereo system is

wanted, a sophisticated stereo algorithm leads to a more precise disparity image but at the same time the algorithm demands more processing power. This conflict of interest is best solved by looking at the demands of the application and therefore classifying the stereo algorithms according to those interests.

Multiple Lane Detection Algorithm Based on Novel Dense Vanishing Point Estimation

Ozgunalp et al. [3] presented 2017 a lane detection algorithm for assisted driving systems. It shows that challenges for assisted driving systems can be handled through the use of stereo vision.

The goal of lane detection is achieved by estimating a number of vanishing points which was carried out with stereo vision. It is used to estimate a disparity map that is then transformed into a v-disparity map. The v-disparity is an image that includes information about disparity values and the vertical coordinate (v-coordinate) of a disparity map. Basically it reduces the complexity of the disparity image by one dimension. Nevertheless it still includes enough information to operate in a reliable vanishing point calculation within a disparity map.

V-disparity is used for detecting the horizon and calculating the y-coordinate of the vanishing points V_p. This result helps to estimate the x-coordinate of V_p in a robust way. For this matter the lane markings of the road are considered as well. The v-disparity is also used to make a segmentation of images. The purpose of this is to distinguish between features on the road and between features that do not belong to the road.

The experimental set-up in this paper is a stereo camera rig that is mounted on a car and used for data collection. Problems with the roll angle are mentioned and a method for detecting and correcting it is presented. The detection of the roll angle is done through fitting a plane inside of a small part of the disparity image. The chosen area is in front and close to the vehicle and therefore assumed as part of the road.

The transformation from disparity image to v-disparity image in Section 4.1 is equal to the paper of Ozgunalps. The use of the v-disparity is slightly different because the authors in [3] use it for the vanishing point calculation and in Section 4.6 it helps to detect the floor of a robotic environment.

Also the approach from Section 4.2.2 is similar to the approach from Ozgunalp. But instead of a plane detection for the estimation of the roll angle a line detection is used.

Obstacle Detection in Stereo Sequences using Multiple Representations of the Disparity Map

Burlacu et al. [4] presented 2016 a way of object detection with the use of different disparity representations. The authors' approach is a disparity calculation using the ELAS (Efficient Large-Scale Stereo Matching) algorithm presented by Geiger, Roser and Urtasun in [5]. Furthermore they use three different transformations (v-disparity, u-disparity and θ-disparity) that represent the disparity and are used to detect the obstacles. The v-disparity is a row-wise and the u-disparity a column-wise histogram that reduces the disparity image information by one dimension. The θ-disparity is a polar representation of the disparity image. The v-disparity was used for the ground plane detection, the u-disparity and the θ-disparity were used for the object detection. For this purpose the ground plane detected with the v-disparity is removed from the u- and θ-disparity.

Furthermore in the paper problems with the roll angle are described that results in an impractical v-disparity map. For this purpose a stereo vision motion procedure is used to calculate the camera position and to correct it then by rotating the disparity image. For the evaluation virtual and real images are worked with. The presented results show the accuracy for the object detection with the use of either the u- or θ-disparity representations. The Results show also that combining both methods increases the robustness of the algorithm.

In Section 4.6 the v-disparity map was also used to detect the ground plane. But instead of the u- and θ disparity the disparity image was used to detect objects that pop out of the ground plane.

A Fast Dense Stereo Matching Algorithm with an Application to 3D Occupancy Mapping using Quadrocopters

Ait-Jellal and Zell [6] presented in 2015 a stereo matching algorithm and show an application for quadrocopters. The main focus is the development of an algorithm that is fast and efficient in calculation. This is done in a way that the quadrocopter calculates the disparity and the 3D reconstruction on board. The authors describe three steps for their stereo algorithm: 1.) an initial disparity calculation, 2.) a mismatch detection and correction, 3.) the final post-processing step through edge preserve filtering.

One part of the results in this work compare the presented stereo algorithm with other common stereo algorithms. The middlebury dataset[1] from Scharstein and Szeliski [7] is used to evaluate the algorithm. The score of the stereo

[1] http://vision.middlebury.edu/stereo/

algorithm was slightly better than the score from the global optimization based algorithms graph cut [8] and constant time belief propagation [9]. The running time of the algorithm was less than 20 milliseconds for any stereo pair of the middlebury dataset.

Another part of the paper, besides the stereo algorithm, is the 3D reconstruction of the disparity map. This is done by calculating an octomap as it is described in the paper from Wurm et al. [10]. The focus here is on a safe movement of Micro-Ariel Vehicles (MAV). For this purpose it is not necessary to detect the geometry of every possible object in detail. Instead of a highly detailed representation of the environment, the octomap only represents certain cubic areas that include a possible obstacle. This results in an octomap with a low resolution and therefore the octomap only delivers a rough representation of the environment for sufficient obstacle avoidance. The 3D reconstruction is tested with the stereo dataset from EuRoC Challange[2] and results in fast enough mapping for quadrocopters.

Obstacle detection using V-disparity: Integration to the CRAB rover

Wandfluh [11] in 2009 shows the implementation for obstacle detection in a robotic rover. The thesis gives an overview of a stereo system and an autonomous robot called CRAB. The object detection is described with the use of v-disparity and furthermore the handling of the generated map is described. Also a test simulation is described which is carried out in an virtual environment. The interplay of all components in the robotic system and addition of the object detection are presented.

In the thesis of Wandfluh the algorithm for object detection works with the use of the v-disparity for the floor detection and is presented in [12]. In further steps the obstacle detection is covered and the mapping of the obstacles is described. Wandfluh improved the algorithm with a roll angle detection and correction for better floor detection. The results from the object detection are then used to build a map which is needed to navigate the CRAB robot.

The testing of the map building process is also described in the thesis. It also covers the results of these tests in a simulated environment.

[2] http://www.euroc-project.eu/

Processing Dense Stereo Data Using Elevation Maps: Road Surface, Traffic Isle, and Obstacle Detection

Oniga and Nedevschi [13] presented in 2010 a way to distinct between road, traffic isles and obstacle points in stereo vision data. In their algorithm they fuse results from a road surface-based classification and a density based classification. This results in a classified map that structures stereo images of a road into the mentioned classes.

For the road surface-based classification the paper presents a method to fit a quadratic model of the road surface into the 3-D data. This is realized by minimizing an error function between modelled and a selected number of measured points. The points are chosen beforehand through a RANSAC (random sample consensus) approach . Additionally a distance dependent offset is added to get a spatial representation of the road. This data is used to calculate a digital evaluation map (DEM) that is divided into equal sized cells from which each of the cells contains the height information of the highest point in the cell. The DEM is then used as a partial result of the classification process.

The second approach in this paper is a density based classification. For this purpose the disparity map is used to calculate a map that contains information about the density of 3-D points inside each cell. With the road model the roads expected density map is estimated. The next step is to calculate the difference between the density map of the data and the estimated density map of the road. All remaining positive values represent possible objects which are needed for obstacle classification to distinguish between road and objects.

The data of both presented methods are fused into one optimized result. The fusion of the partial results allows a more reliable classification and gets rid of outliers.

Real-time Stereo Vision System at Nighttime with Noise Reduction using Simplified non-local Matching Cost

Xu et al. [14] in 2016 show a way to improve stereo vision results under low light conditions. The presented approach is intended to be used for advanced driving assistance system (ADAS) under night conditions. The main problem under such conditions is that the images from the stereo system are more noisy than under normal circumstances, caused by the lack of light during the exposure time. For the purpose of noise reduction the usage of a non-local means (NLM) filter [15] turned out to be most suitable in this case. Besides that other filters were tested but not implemented for real time simulation. They also implemented an image pyramid to reduce the size of the input date

for faster calculation time.

For an effective solution of the stereo correspondence problem an image enhancement algorithm is introduced. This is necessary since a night scene consists of mainly dark areas that need to be lightened up for better results. In the evaluation of this work the main focus is on detecting road surface with the help of the v-disparity algorithm. The data was tested on synthetic data and real night time scenes. The results show clearly an improvement of the road detection compared to common disparity algorithms. It is also stated that the approach of the work results in dense and accurate disparity data.

The HCI Stereo Metrics: Geometry-Aware Performance Analysis of Stereo Algorithms

Honauer, Maier-Hein and Kondermann [16] in 2015 classify different stereo algorithm performances under consideration of different geometric features in a scene. A metric for benchmarking stereo algorithms is presented. The goal is to combine two aspects of benchmarking. One of these sides is a mathematical way that evaluates the performances of the different algorithms and the other one puts the attention more on the application of the algorithm.

The authors focus on principals as depth discontinuities, planar surfaces and fine structures. Each of these recurring geometric characteristics might be important depending on the applications of the stereo system. In total nine different performance measures are presented that are used for the classifications. The proposed metrics structures the type and strength of errors occurring.

Furthermore the results of different stereo algorithms are presented and discussed. The focus is set on a good distinction between the strengths of different algorithms. The metric helps to look into specific geometric characteristics of a scene and ranks the stereo algorithms according to the score in each of the different challenges. Besides that the presented evaluation method supports parameter tuning in order to optimize an existing stereo algorithm.

Compared to the state-of-the-art research topics this thesis also uses the v-disparity approach for the floor detection. It implements a roll angle detection and correction and additional the approach of multi v-disparity that splits the disparity image into a number of sub disparity images. This thesis' focus is on the detection of rather small obstacles for the use in robotic indoor navigation. Furthermore it compares the strengths and weaknesses of different stereo algorithms that are used for the disparity calculation.

3 Methods

Sensors have a key role in terms of obstacle detection for robotic applications. They gather information of the environment and help to use this data for the purpose of obstacle detection. One type of sensors are cameras. They are able to catch information of the surrounding world and in particular digital cameras. The robot receives a digital representation of the gathered information.

A stereo system will deliver a pair of images as an input. This data is used for further processing, which results into information about the depth of a scene and enables to detect objects to a certain degree.

Before analyzing object detection it is essential to understand how a camera actually works and how a stereo system is built up. These basic principals are a framework that is used to detect objects in typical robotic scenes. The first part of this chapter deals with three different camera models, followed by stereo systems. A focus is on how the system is composed with a further look into the calculation of depth. For this purpose the knowledge of geometric relations is needed to work efficiently. Epipolar geometry is introduced and important parameters like the essential and fundamental matrix are discussed. Camera calibration is another essential step and is covered in this chapter as well. Looking deeper into the topic of camera calibration, models are presented that describe typical distortions caused by camera lenses. Moving on from there to the matter of rectification of image pairs for stereo vision.

Finally the last part of this chapter is focused on stereo algorithms. A short classification of stereo algorithms is discussed and an overview of the important steps of the stereo matching process is given.

The equations of this chapter are established according to Bradski and Kaehler [17].

3.1 Camera Models

In order to understand a stereo vision system a closer look is taken into basic camera models. The models describe how points from a scene are projected onto an image plane. It is necessary to understand how stereo systems try to reverse this process and extract the depth information. At first a short overview of the pinhole camera model is given and after that a model with a

frontal image plane is discussed. Finally the functionality of digital cameras is presented.

3.1.1 Pinhole Camera Model

The pinhole camera model describes how an object is projected onto an image plane. It consists of two planes which are perfectly parallel in the model. They are called the image plane and the pinhole plane. The distance between them is the focal length f. The pinhole plane is almost optical opaque except for a small hole which is big enough for a light ray to pass. Such a camera could be easily built with a cardboard, where a light sensitive paper is placed on one side inside the box and a pinhole is made on the opposite side. The optical axis is defined as the axis which is normal to both planes and passes through the center of the pinhole.

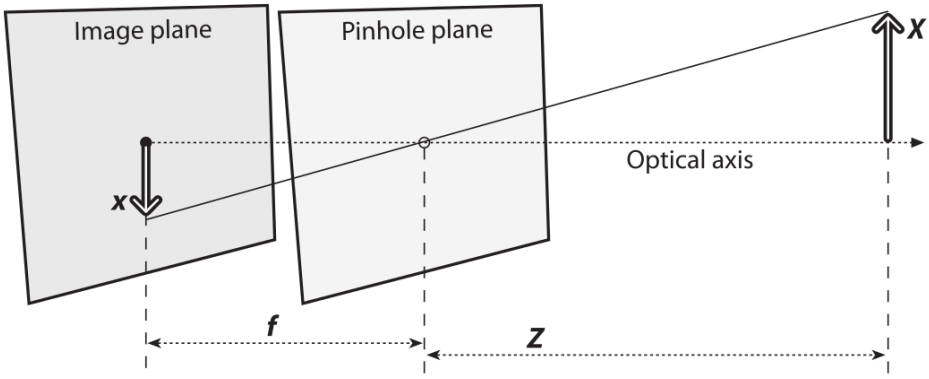


Figure 3.1: Pinhole Camera Model [17].

For further consideration a simple source point of light is placed in front of the pinhole plane. The distance normal to the plane is Z and the normal distance to the optical axis is X. The image plane is placed behind the pinhole plane, so only light from the point source passing through the pinhole is visible on the image plane. The normal distance of the point on the image plane to the optical axis is called x. Because of the relations in similar triangles it is calculated x from f,X and Z according to:

$$-x = f\frac{X}{Z} \tag{3.1}$$

Figure 3.1 shows the geometric relation between the distances. The position of the projection from the point source will be flipped compared to the original

position of the point source. Now if an entire object, that is made up of many single point sources is considered, it is clear that as a result the object is also upside down. The problem with a pinhole camera is that since of the small hole it does not produce bright enough images for proper use. In order to get rid of this problem it is necessary to move away from the simple pinhole camera and adjust the model.

3.1.2 Frontal Image Plane Model

Pinhole cameras do not produce bright enough images for proper use because the amount of light gathered through the hole is way too small. A bigger diameter of the pinhole would brighten up the image but comes with the disadvantage of a blurry image. The image becomes blurry because instead of one sharp light ray from the point source there will be many similar light rays which will be projected close to each other on the image plane. A solution for this is the use of a lens. It allows enough light to pass through it and focuses all light rays in one projection center. It is possible to move the image plane in front of the projection center with the advantage that points are no longer mirrored around the optical axis. This set-up is described by Fig. 3.2. Projected images now face the right direction. Adding the parameters c_x and c_y, they indicate the offset from the image center and allow to correct pixel coordinates if the image center does not fall onto the optical axis.
Points from the physical world $Q(X,Y,Z)$ now are projected on the plane with the coordinates $q(x,y,f)$. The coordinates on the screen x and y can be calculated through:

$$x = f_x \frac{X}{Z} + c_x \qquad y = f_y \frac{Y}{Z} + c_y \tag{3.2}$$

The reversal from image coordinates x and y to the world coordinates X, Y and Z do not deliver clear solutions but they deliver possible solutions along a line. With the information of a second pair of image coordinates, in another image plane, a clear solution for X, Y and Z can be calculated.

Digital cameras allow to process the data within a predictable time because the data consists of a fixed number of image points. The equations above are still valid in the model of a digital camera but have to be adjusted.

3.1.3 Digital Camera Model

So far models that describe how points are projected onto an image plane have been explained. This is still important for the digital camera model but for the processing of the data it is essential to know how a point is digitally represented.

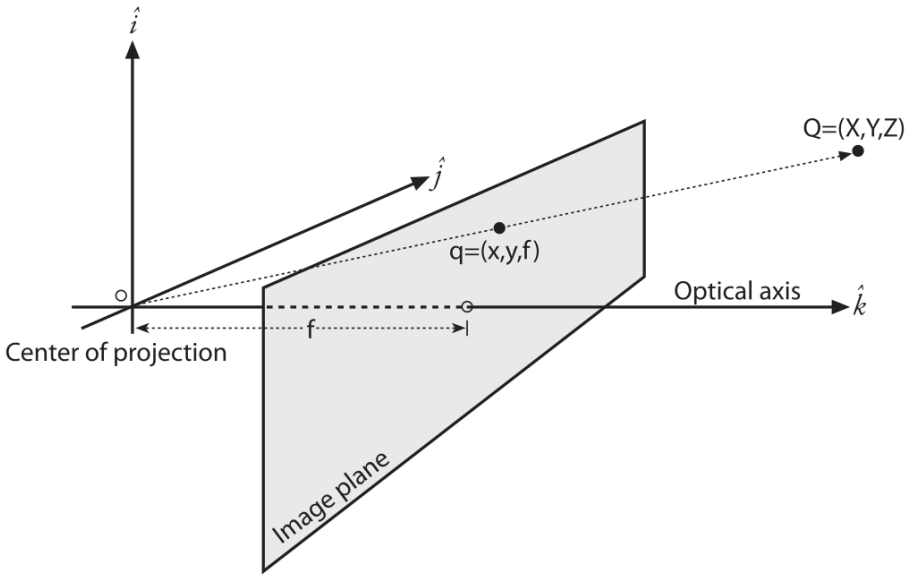


Figure 3.2: Frontal Image Plane Model [17].

The image plane of a digital camera consists of a computer chip, it is an array of photosensitive cells. The light from an object falls onto one of the cells and depending on the brightness of the object, a digital value is created for each pixel. If three digital values are used to represent one single point it is possible to gather information of color. These three values stand for (R) red, (G) green and (B) blue values which span a color space for each point in the image. In a digital image points can only fall onto discrete pixel coordinates instead of continuous image coordinates. The projection geometry is still the same as in Eq. (3.2), from here the pixel coordinates can be calculated by

$$u = \frac{x}{\rho_w} + u_0 \qquad v = \frac{y}{\rho_h} + v_0 \tag{3.3}$$

The values ρ_w and ρ_h represent the width and height per pixel, the origin of the pixel coordinate system is in the top left corner of the image, u_0 and v_0 represent the coordinates of the principal point.

So the data is a value of a grey level or three values for color information (RGB). The pixels are uniquely distinguishable through these pixel coordinates u and v.

3.2 Stereo Vision

With the help of stereo vision a digital reproduction of a real world scene is created. It is like a digital image with additional information. This information consists of a depth value for each pixel. The most natural comparison to a stereo vision system is the human vision. If one looks at objects within the grasping range, one is able to estimate their depth and grasp them without any problems. When one tries it with one eye closed, one will not be successful all the time. It gets harder once somebody else rearranges the objects. Probably one might still be able to estimate the depth to a certain degree since there are many clues in certain scenes to gather additional information of depth. Especially if the objects are familiar, with some effort one can assess and estimate the size of it and is able to judge the scene as a whole which leads to a feeling of depth.

However if both eyes are used it is an utterly easy task that is just natural to humans. Stereo systems are similar to the human vision, cameras try to imitate the eyes and a processor will do the work of the brain.

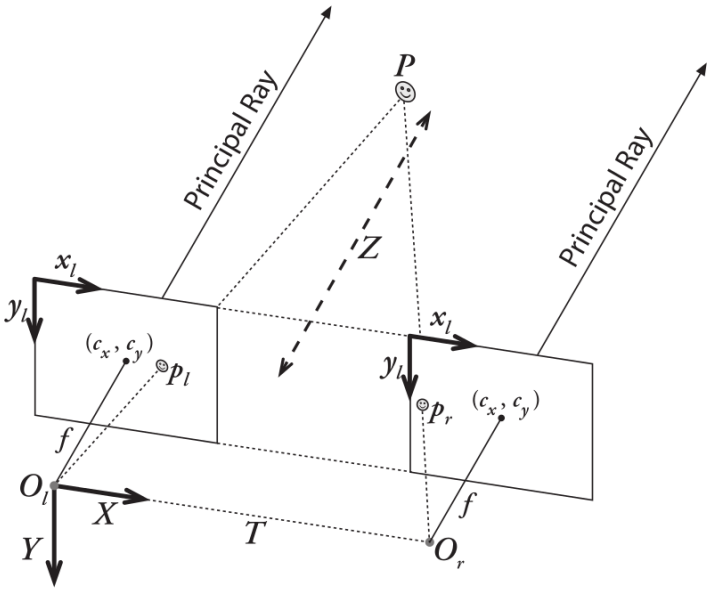


Figure 3.3: Stereo Vision System [17].

The equations from (3.2) show that stereo vision is needed to calculate a point's coordinates. It is necessary to use two images for an unambiguous result. When trying to recreate a three dimensional scene from two dimensional image the problem of gaining an infinite number of solutions for every point and so

for every object appears. A big object in the distance is not distinguishable from a small object close to the camera. By locating a single point in world coordinates from the image coordinates, the issue is that it results in an infinite number of solutions along a line. Now the additional information of a second image results in the advantage that it is possible to locate the point to a certain accuracy. This is possible because the second image results in a second line that represents the infinite number of all possible points according to the image coordinates of the second image. Ideally both lines have one point in common, this is the intersection of both lines and the location of the point. It is only manageable to calculate this position to a certain accuracy because of discrete image coordinates in a digital camera, as multiple similar positions deliver the same result.

This process needs to be repeated for every point of the scene but is not always possible. One reason is that not every point is visible in both images and not every pair of points is detectable since the points are not uniquely distinguishable from each other at times.

3.2.1 Composition of a Stereo System and Depth Calculation

The task of a stereo system is to record an image pair. This requires two cameras that are able to deliver the image pair time synchronised. Ideally both cameras have the same parameters and they are aligned in such a way that the image planes are coplanar. The y-axis of the coordinate system in the optical center of the cameras is parallel as well and the optical center of each camera falls on the x-axis of each other. This ideal composition is not necessarily fulfilled in practice, but this issue will be corrected through the camera calibration process.

Assuming a perfectly aligned calibrated stereo rig, the focus is on depth calculation. The optical center of the left camera is O_l, the optical center of the right camera is O_r and the displacement between them is the vector T. In this regard the principal points are c_x^{left} and c_x^{right}, which are the points where the principal ray intersects with the image plane. This point of intersection in each camera has the same pixel coordinates in both cameras. The focal length f of both cameras is the same as well.

Now a point P of the physical world is visible from both cameras and it is projected onto the image plane of each camera. The horizontal pixel coordinate of this point on the image plane of the left camera is x_l and the same coordinate on the right camera is x_r. These coordinates are the same if the point has an

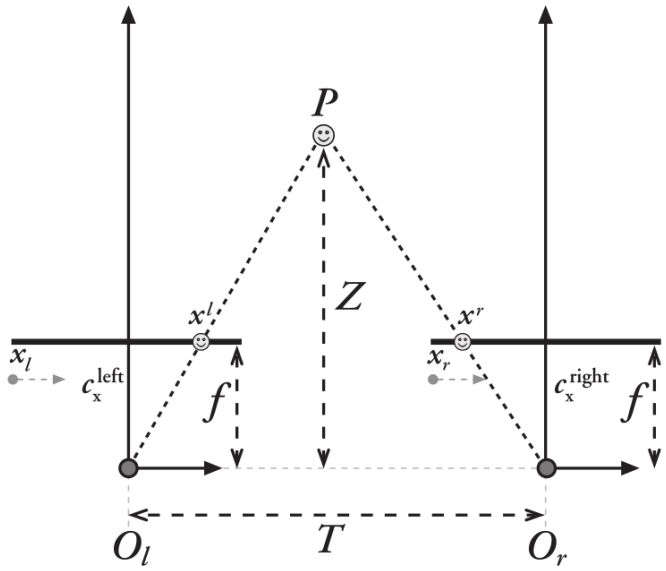


Figure 3.4: Depth Calculation [17].

infinite distance Z. With these coordinates the disparity d is defined as

$$d = x_l - x_r \tag{3.4}$$

The two triangles $O_l - O_r - P$ and $x_l - x_r - P$ are similar triangles which lead to a constant aspect ratio of

$$\frac{T - (x_l - x_r)}{Z - f} = \frac{T}{Z} \Rightarrow Z = \frac{fT}{x_l - x_r} \tag{3.5}$$

In Fig. 3.4 the geometric relations can be traced. The Equation (3.5) shows that depth and disparity are inversely related and nearby objects are easier to distinguish than objects further back. Considering that the depth value will be an integer value within a digital system, it is noticeable that the resolution decreases with decreasing disparity. This relation is demonstrated in Fig. 3.5.

3.2.2 Epipolar Geometry

The focus is on a fundamental concept that looks further in the geometry of a stereo system and helps to improve stereo algorithms. A pinhole model is used for each camera and the relation between the two cameras is described through the translation T and the rotation matrix R. In Section 3.2.1 it is considered that the image planes of each camera are coplanar, this is still a desirable arrangement but the mechanical construction of such a system will

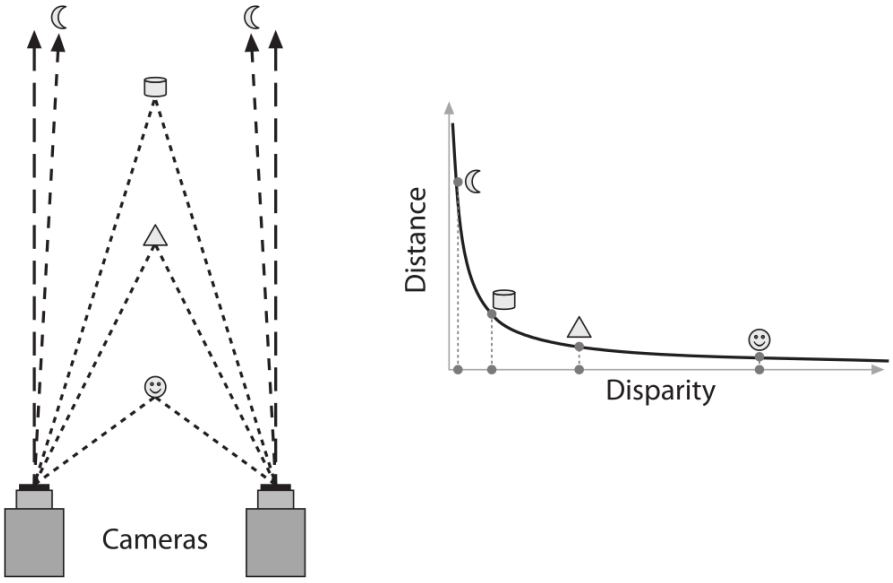


Figure 3.5: Relation between depth and disparity [17].

likely differ marginal. For further observations non perfectly aligned cameras are assumed.

A point P from the physical world can be projected on the image plane of each camera. If the line, from P to the optical center, is intersected with the image plane of the related camera it results in the point p_l in the left camera and in the point p_r in the right camera. As mentioned before the difference between the pixel coordinates in the two images deliver the disparity. In Eq. (3.4) it is considered that the disparity only depends on the x-coordinate of the pixel coordinates. This is only the case if the cameras are perfectly aligned.

Moving further to a more general approach in which the point pair p_l and p_r have an essential role. The approach of a stereo algorithm is the search for corresponding points in an image pair. If one point of the point pair is selected it will be necessary to find the other point and calculate the disparity between them. This is a step shared by each local stereo algorithm and it is also a part of the block matching algorithm described in Section 4.5.2. The task of searching point pairs in images can demand huge computation power with increasing image resolution, but with the help of epipolar geometry this task gets reduced to an one dimensional search along a line. At first the epipolar plane needs to be defined, along with the epipoles and finally the lines that reduce the search task, the so called epipolar lines.

The epipolar plane is spanned by the Point P and the two optical centers

O_l and O_r. The epipoles are the intersection of the line between the optical centers and the image planes of the cameras that lead to an epipole in the left image plane e_l and one on the right e_r. Connecting p_l with e_l will result in the epipolar line on the left side and connecting p_r with e_r will result in the epipolar line on the right side. These epipolar lines can also be gathered if the line $P - O_l$ is projected onto the right image plane and also through a projection of $P - O_r$ onto the left image. Figure 3.6 demonstrate the relation of epipolar lines in a stereo vision system.

The epipolar lines now help to find a corresponding point pair in an image pair. Starting with a point in one image, it is possible to find the corresponding point in the second image because the projection of the point P can only lie on the epipolar line as long as it is visible in both images. The difference in the points' position in one image compared to the other shows the disparity which helps to calculate the depth according to Eq. (3.5).

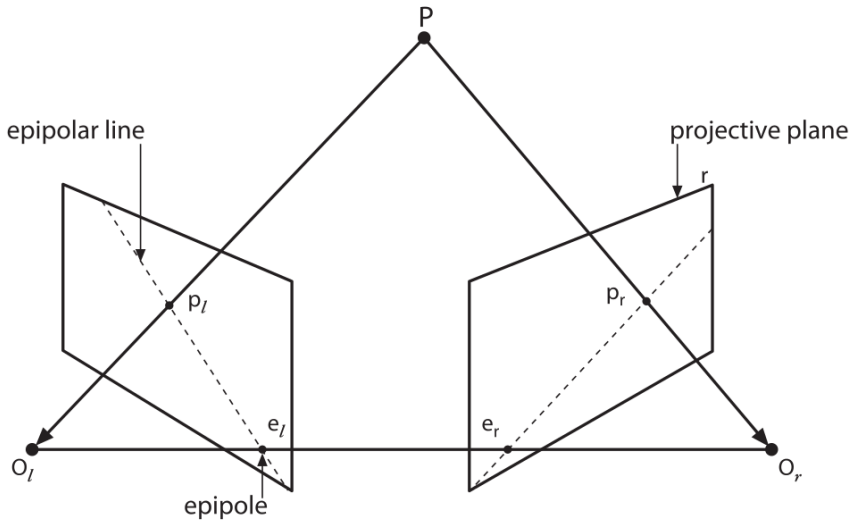


Figure 3.6: Epipolar lines in a stereo vision system [17].

The reduction to an one dimensional search task comes with the additional task of calculating the epipolar lines of each point. This task can be further optimized if the image is rectified. A pair of two rectified images has the advantage that a point in one image can be found in the same height along a horizontal line on the other image.

3.2.3 Essential Matrix

The essential Matrix E includes information about the geometric relation between the two cameras of a stereo system. With the help of E points on the image plane of the left camera can be related to points on the image plane of the right camera. For the calculation of the essential matrix the optical center of the left camera as the origin of our coordinate system is used (it would work equally well if choosing the right camera). The coordinates of a point P seen from the left camera is P_l and seen from the right camera is P_r. The vector T and the matrix R take the relation between the cameras into account and help to calculate P_r from P_l.

$$P_r = R(P_l - T) \tag{3.6}$$

The vectors P_l, P_r and T lie on the epipolar plane which can be specified through the normal vector representation. The following equation can be used

$$(P_l - T)^T (T \times P_l) = 0 \tag{3.7}$$

If the Equation (3.6) is rearranged it results into $(P_l - T) = R^{-1}P_r$. This can be substituted into Eq. (3.7), with the consideration of $R^{-1} = R^T$ because of the orthogonality of R the equation can be rewritten as:

$$(R^T P_r)^T (T \times P_l) = 0 \tag{3.8}$$

Next the matrix S is introduced and it helps to rewrite the cross product:

$$(T \times P_l) = SP_l \Rightarrow S = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \tag{3.9}$$

With the matrix S the Eq. (3.8) can be rewritten and additional the essential matrix is defined as $E = RS$

$$(P_r)^T RSP_l = 0 \;\; with \; RS = E \Rightarrow (P_r)^T EP_l = 0 \tag{3.10}$$

Furthermore the substitution of $P_l = \frac{p_l Z_l}{f_l}$ and $P_r = \frac{p_r Z_r}{f_r}$ is considered and the multiplication of Eq. (3.10) with $\frac{f_l f_r}{Z_l Z_r}$ results into:

$$p_r^T E p_l = 0 \tag{3.11}$$

The matrix E is a 3-by-3 matrix and has rank 2. Due to this the solutions of Eq. (3.11) will lead to an equation for a line. So the essential matrix does not exactly relate two points to each other but more precisely it relates to each point infinite points along a line. This helps to find corresponding point pairs. E only considers the relation between two cameras and does not take care of the intrinsic parameters of a camera. The relation between the points is in physical coordinates.

3.2.4 Fundamental Matrix

The fundamental Matrix F is very similar to the essential Matrix E. It contains all the information of the relation between the two cameras and additional information of the intrinsic parameters of both cameras. F relates two points in pixel coordinates to each other or more precisely it relates to each point an infinite number of points along a line. So it has the same purpose as the matrix E but instead of using the points p_l and p_r in physical coordinates it uses the points q_l and q_r in pixel coordinates. The relation between p and q is $q = Mp$ with M as the intrinsics matrix. Adding this relation to Eq. (3.11) it results in

$$q_r^T (M_r^{-1})^T E M_l^{-1} q_l = 0 \tag{3.12}$$

F is now defined as

$$F = (M_r^{-1})^T E M_l^{-1} \tag{3.13}$$

and the outcome is the relation between two points in pixel coordinates

$$q_r^T F q_l = 0 \tag{3.14}$$

3.2.5 Stereo Calibration

Stereo calibration is an offline process used to get the intrinsic and extrinsic parameters of a stereo system. Until now it was supposed that the relation between both cameras in the stereo system are well known, but this is only possible to a certain accuracy. The rotation and translation between two cameras are measurable and represented in the rotation matrix R and the translation vector T, but they can change over time if one camera position is slightly changed. Stereo calibration is one way to receive this information at a certain time. For this purpose usually a chessboard is used with well known measurements. An advantage of the chessboard is that its corner points are easy to recognize in the left and right image of the stereo system and the pattern is well known so the points in both images can be assigned to each other. The chessboard is then placed in front of the stereo system and pictures of it are taken in different positions. Moving on a point P is now projected on the left image with the coordinates P_l and the right image with the coordinates P_r. Considering the rotation matrices R_r and R_l from the cameras to the point in the scene as well as the translation vectors T_r and T_l, the projection is calculated through:

$$P_l = R_l P + T_l \qquad P_r = R_r P + T_r \tag{3.15}$$

Further relations are:

$$R = R_r (R_l)^T \qquad T = T_r - R T_l \tag{3.16}$$

Now with enough points P_r and P_l the relation between the two cameras R and T can be calculated through an optimization under the constraints of Eq. (3.6), Eq. (3.15) and Eq. (3.16). With enough points from the calibration process it is possible to solve for the optimal solution for every parameter. Different points do not always result in the same rotation matrix and translation vector because of different errors (mostly rounding errors and noise). The task of the optimization process is finding parameters so that the overall error will be minimized. According to Bradski and Kaehler [17] the Levenberg-Marquardt iterative algorithm, as implemented in [18] by J. J. Moré, delivers good and robust results. Ideally it is possible to find the parameters which describe the lens distortion with the help of stereo calibration. This allows to reverse the distortion and correct these types of errors.

3.2.6 Lens Distortion

In Section 3.1.2 the advantages of lenses are shown but the use of lenses also comes with the disadvantage of systematic errors and the possibility of distortion errors due to inaccuracies in the production process. Two lenses produced under similar circumstances may differ slightly from each other and create distortion errors. Of course more complex production processes would result in smaller errors to a certain degree. The errors can be described through mathematical models that make it possible to correct the most common ones. For this purpose camera calibration is used and delivers the necessary parameters.

Radial Distortion

Radial distortion appears due to the shape of the lens, light rays which pass the lens further away from the center are bent more compared to rays closer to the center. So a squared object is projected on a round lens at different radial coordinates and not every point is projected equally onto the image plane which results in a distorted object[1]. This error can be corrected through:

$$x_{corrected} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \tag{3.17}$$

$$y_{corrected} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \tag{3.18}$$

with x and y as original coordinates of a point on the image plane, r as distance from the radial distance from the center and k_1, k_2 and k_3 as the distortion parameters. Figure 3.7 shows how the effect of radial distortion influence the projection from a squared object.

[1] Also known as "barrel" or "fish-eye" effect

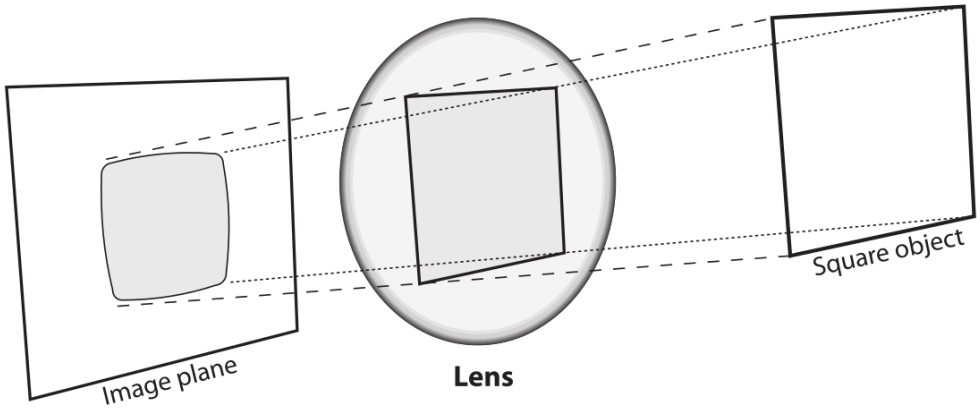


Figure 3.7: Radial Distortion [17].

Tangential Distortion

This kind of distortion appears if the lens and the image plane are not perfectly parallel to each other. The error is more severe if a point further away from the image center is considered, but it can be corrected through:

$$x_{corrected} = x + [2p_1y + p_2(r^2 + 2x^2)] \tag{3.19}$$

$$y_{corrected} = y + [p_1(r^2 + 2 * y^2) + 2p_2x] \tag{3.20}$$

3.2.7 Rectification

Before stereo algorithms are discussed the focus is on rectification, an important step to make the computation of the disparity much easier. As stated in Section 3.2.2 every point in one image of the stereo pair has the corresponding point in the other image along the epipolar line. If the image pair is rectified it ensures that point pairs always share their y-coordinate in the image. So it is much easier to find point pairs in two different images since the search process is always along the same horizontal line. Basically this would always be possible if the stereo system is set up in a way that the principal rays of both cameras are perfectly parallel and the image planes are arranged that both their y-coordinates are exactly the same[2]. Such a configuration is hard to adjust properly and so it is necessary to transform the images onto a new plane in order to achieve an optimal result.

For this purpose an undistorted image which is shown in Section 3.2.6 is needed as well as the rotation matrix R_{rect} which aligns the images along

[2]frontal parallel configuration: coplanar and row-aligned image planes

horizontal lines and two projection matrices for each camera P_r and P_l. In order to calculate R_{rect} the following vectors are substantial:

$$e_1 = \frac{T}{||T||} \qquad e_2 = \frac{\begin{bmatrix} -T_y & T_x & 0 \end{bmatrix}^T}{\sqrt{T_x^2 + T_y^2}} \qquad e_3 = e_1 \times e_2 \tag{3.21}$$

they lead directly to the rotation matrix:

$$R_{rect} = \begin{bmatrix} (e_1)^T \\ (e_2)^T \\ (e_3)^T \end{bmatrix} \tag{3.22}$$

The projection matrices are calculated under consideration of the focal lengths f_x, f_y, a skew factor α[3] and the parameters c_x and c_y:

$$P_l = \begin{bmatrix} f_{x\_l} & \alpha_l & c_{x\_l} \\ 0 & f_{y\_l} & c_{y\_l} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.23}$$

for the left side and for the right side:

$$P_r = \begin{bmatrix} f_{x\_r} & \alpha_r & c_{x\_r} \\ 0 & f_{y\_r} & c_{y\_r} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.24}$$

After aligning the images with R_{rect} the next step in the process of rectification is the projection onto the left and the right images through P_l and P_r. These rectified images are used to run stereo algorithms which do not need to calculate each epipolar line for every single point. They just search along horizontal lines in the two rectified images. To sum up these preparation steps the starting point is a raw image pair taken from the stereo system which is later undistorted, rectified and cropped. After these steps the result is a rectified image pair where both images have the same size. In consideration of these conditions stereo algorithms are following.

3.3 Stereo Algorithm

A stereo algorithm's purpose is to calculate a disparity image from a stereo image pair. This can be achieved by calculating the disparity between the left and the right image for every pair of points. Undoubtedly this is only possible

[3] this value is in most set ups almost 0

if it is clearly distinguishable that two points in different images represent the same point of the physical world. So a part of the algorithm's task is to find corresponding pairs of points in the left and right image. As stated previously this task gets much easier once the image pair is rectified. This allows to reduce the search task to a search along a line. Starting with the selection of a point in the left image and then searching along a horizontal line for the corresponding point in the right image. For every possible point the similarity is calculated and the point most similar is chosen and its disparity is determined.

In general this would work fine if every point of an image was clearly distinguishable from its surrounding points and the corresponding point of the pair of points in the other image would also be clearly distinguishable from its surrounding points. A number of different characteristics of a scene such as featureless regions, occlusions and multiple similar correspondences make the problem much harder. One way to compare points is through their digital values but these values are not unique and will result in different solutions for almost every point. The problem can be reduced if not only values of single points are compared to other points but rather is a comparison of regions that compound a number of pixels is realised. This allows to differ diverse image regions much better. If there is still a repeating pattern in an image it will be still a problem to distinguish between the regions. This problem is also present in homogeneous image regions that are present because of plane surfaces without any texture in a scene. There are different approaches to deal with these types of problems through different algorithm strategies that are discussed in Section 4.5. A rough categorisation between sparse and dense algorithms can be stated.

Sparse Output

The goal behind sparse stereo algorithms is recovering the coordinates of every feature visible in both images of the stereo vision system. So only regions with feature points can be recovered, which results in gaps in the disparity image. Algorithms of this kind mostly ignore similar regions which are hard to differ from each other.

Dense Output

Dense stereo algorithms calculate the coordinates of every pixel regardless whether a feature exists in a certain area or not. Global and local methods are commonly used to reach the goal of a dense disparity output.

3.3.1 Stereo Correspondence

For the disparity calculation it is necessary to find corresponding points in the image pair. Due to allocating an image area to a matching cost it is possible to compare image areas and find correspondences. Scharstein and Szeliski [7] show a taxonomy of different stereo algorithms. It includes a categorization for dense stereo algorithms and the important steps for most algorithms. Scharstein and Szeliski list the most common steps for a vast number of algorithms:

1. matching cost computation
2. cost (support) aggregation
3. disparity computation/optimization
4. disparity refinement

These steps are not always necessary and may vary depending on the specific algorithm. For the matching cost computation a region in the left image and a region from the right image is chosen. The regions contain either a fixed number of pixels or it could also be a variable number of pixels, as long as they have the same size in both images. The set of pixels in a certain area is represented in W. (x,y) are the coordinates of a pixel and d is the possible disparity. The matching cost can be a difference in intensities between two regions. The aggregation of the cost is done through different functions:

- sum of absolute differences (SAD):
  $SAD(x,y,d) = \sum_{(x,y)\in W} |I_R(x,y) - I_T(x+d,y)|$
- sum of squared differences (SSD):
  $SSD = \sum_{(x,y)\in W} (I_R(x,y) - I_T(x+d,y))^2$
- sum of truncated absolute differences (STAD)
  $STAD = \sum_{(x,y)\in W} min\{|I_R(x,y) - I_T(x+d,y)|,T\}$

The step of disparity computation can differ vastly between algorithm. In a sparse algorithm disparity values for features are calculated. In a dense algorithm disparity values for every pixel is computed. After that a refinement can be done to increase the resolution of the algorithm.

Local Methods

The important steps in local methods are the matching cost computation and the aggregation. The functions such as SAD, SSD and STAD can be considered for this. After this the disparity value with the minimal cost is chosen. This process is repeated for every considered image region and results in a disparity image. For some regions it is hard to calculate the right disparity value because of occlusions or homogeneous image regions. It is also possible that only certain regions of an image with detectable features are considered for the disparity calculation. This results in a sparse disparity output. The step of disparity refinement allows to correct wrongly calculated disparity values and can also be used to calculate a dense disparity image from a sparse disparity output.

Global Methods

Global methods consider the fact that neighbouring pixels have usually similar disparity values as long as the considered region does not represent the edge of an object. The disparity of every pixel is calculated through the minimization of a global energy function. It consists of a data term and a smoothness term. The matching cost for these kind of algorithms can be calculated in the same way as in local methods and is used for the calculation of the data term $E_{data}(d)$. The smoothness term $E_{smooth}(d)$ takes the neighbouring pixels into account and can be calculated through the sum of the difference between disparities. The energy function is the sum of both terms:

$$E(d) = E_{data}(d) + \lambda E_{smooth}(d) \tag{3.25}$$

The minimization of the energy function can be calculated through different algorithms. Common algorithms use methods like belief propagation [19] or graph cuts [8] to find an optimal solution.

4 Approach

The knowledge of camera models, stereo systems and stereo algorithms helps to understand physical connections and relations in the process of the disparity calculation. For a better understanding of the object detection, the focus is set on each single step of it. In this chapter the theory of those steps are discussed as well as their part in the object detection.

An important part is the disparity image, because it is used as the input for the process of object detection. One field of application of it is the v-disparity calculation, that is the base for the floor detection. Because an advantage of this image transformation from the disparity image is the transformation of a plane floor into a single line. This makes it possible to reduce the floor detection into a line detection, which will be less costly. In order to do this the floor needs to be the dominant plane in the observed area. The assumption that the floor is the dominant plane in a scene is valid for many indoor robotic scenes. It is considered that for the navigation purpose in robotic applications the stereo camera will be aligned in a way that the area in front of the robots moving direction is observed.

After the v-disparity calculation the goal is to detect a dominant line in the v-disparity image, which can be detected through different methods like RANSAC or Hough transform. In this thesis the Hough transformation is used because it showed fast and efficient results. This transformation will be discussed in this chapter.

The detection of the floor in the v-disparity image helps to end up with the objects in the disparity map. In order to do this the remapping process from the v-disparity back to the disparity is discussed. This process helps to remove the floor from the disparity image and only objects will be left in the scene.

One problem with the v-disparity image is that the floor is only transformed into a perfect line if the image horizon is parallel to horizontal lines. This is not the case if the roll angle is not zero. Because of that, two different approaches are introduced. One of them being the multi v-disparity which allows to correct small roll angle deviations, the other approach is an automatic roll angle detection and correction through image transformation.

A closer look is also taken into noise reduction because the stereo system is slightly influenced by noise and this leads to wrongly detected objects that are unwanted in the result. The theory of noise reduction is discussed.

Also wrongly detected pixels appear from bright light reflections on the floor caused by saturated intensity values in the cameras. This effect leads the stereo algorithm to partially wrong disparity images. A strategy to avoid such problems is also discussed.

Due to different tested algorithms in Chapter 5 a look into the functionality of the stereo algorithms is taken. Finally an overview over the object detection algorithm is given.

4.1 v-Disparity

An intuitive method for object detection is the segmentation from the objects in a 3D-space representation. So the disparity map is transformed into the (X,Y,Z)-space, the floor in the 3D-space is detected and the objects segmented from there. Nevertheless it is possible to calculate it more efficiently with the consideration of the v-disparity.

The v-disparity image is a reduced form of the disparity image. A pixel from the disparity image consist of the coordinates (u,v) and of a value that represents the disparity. In the v-disparity the information of the u-coordinate is lost but a dominant floor can be detected less costly than in the disparity image. In the field of robot navigation it is advantageous if the object can be detected in the disparity. For this purpose a remapping process can be used.

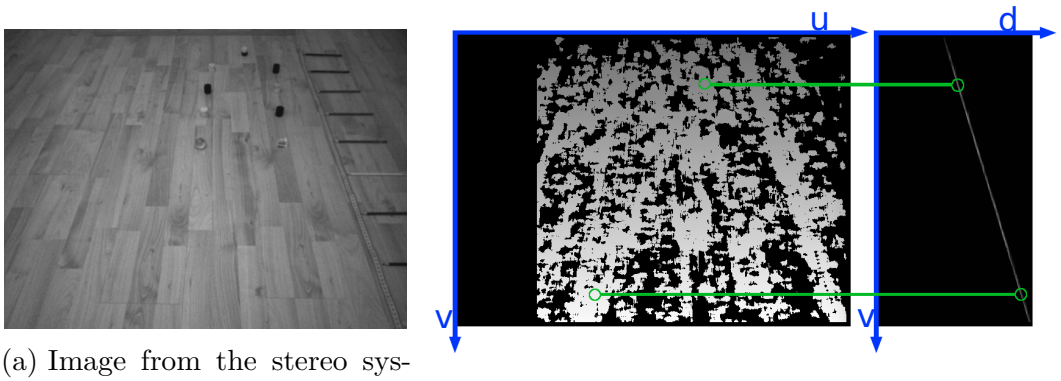
4.1.1 v-Disparity from Disparity

With the help of a stereo vision system the disparity map can be calculated which assigns a disparity value to each pixel. So every pixel represents a point in the physical world at a certain distance. The distance is indirectly represented in the value of the disparity as shown in Eq. (3.5). The coordinates of the pixel give information about their real world coordinates X and Y in the (X,Y,Z)-space. The horizontal coordinate in the disparity map is usually called the u-coordinate and the vertical coordinate is the v-coordinate and gives the name to the v-disparity. In order to get the v-disparity the disparity map (disparity as a function of u and v) d(u,v) is transformed to another space f(d,v) which is the v-disparity map. The v-disparity is similar to the disparity but instead of the u-coordinate of a pixel it is replaced with the disparity value as the new horizontal coordinate. The value f(d,v) represents the number of pixels with the depth value d with the same u-coordinate as in the initial disparity map.

Figure 4.1 shows the mapping from the disparity image to the v-disparity image. The color of the pixel can be directly associated with the disparity

value of the pixel. White pixels represent the maximum disparity value and the different intensities of grey mean values between the maximum and minimum disparity value. The brighter the grey is the bigger is the disparity value. A black value means the minimum disparity value. In the case of this scene the black spots are pixels without a calculated disparity value due to the sparse output of the block matching algorithm.

The advantage of the v-disparity is now that the floor is represented by a single line and objects in the disparity map are represented by pixels above this line.



(a) Image from the stereo systems left camera of a scene

(b) A sparse disparity image from the block matching algorithm is mapped into the v-disparity image

Figure 4.1: Example for the v-disparity calculation

4.1.2 Remapping: Disparity from v-Disparity

If the floor is detected and removed from the v-disparity image, only objects that pop out of the floor are left on it. However the problem is that the u-coordinate of the image is lost and it is not possible to calculate all the world coordinates of the objects. It is an issue concerning the task of path-planning of a robot.

In order to avoid this problem a transformation back to the disparity image needs to be done. This remapping process from the v-disparity image back to the disparity image allows to calculate the world coordinates of the detected obstacles. This is possible if the objects are still present in the disparity image after the remapping process and requires the coordinates u and v as well as the depth information.

One problem is that it is not achievable to create a transformation back to the disparity image if only the v-disparity image is considered. This is the

case because once the disparity image is transformed into the v-disparity image the information of the u-coordinate is lost. This lost information can not be restored just from the v-disparity image. However if additionally a look is taken at the original disparity image of the scene it is possible to remap the v-disparity image back to the disparity image. This remapping allows also to remove all floor pixels in the original disparity image and the result only includes objects that are potential obstacles. So in summary the objects are not transformed back to the disparity image, but instead the floor is just removed from there and all the objects are left in the image.

For this purpose the detected line from the Hough transform is considered in the v-disparity image. Now these lines include every floor pixel of the disparity image and its values of the v-coordinate and the disparity value. The u-coordinate is not accessible but it is restored from the original disparity image. A single pixel that represents a floor pixel in the v-disparity is considered and compared to the original disparity image. Every pixel in the disparity map with the same v-coordinate can be checked if its disparity value is equal to the disparity value of the floor pixel in the v-disparity. If this condition is fulfilled a floor pixel is detected in the disparity image. This is now done for every detected floor pixel of the v-disparity.

Additionally a certain threshold is added because the noise in disparity values result in the effect that the floor will not be transformed to a perfect line in the v-disparity image. This allows to assign noisy pixels to the floor in the scene to a certain degree. It mostly depends on the chosen threshold and the noise intensity of the stereo system. If a floor pixel is detected its disparity value is set to zero.

So in summary all the pixels in the disparity map are identified according to their disparity value in the v-disparity image. Then those pixels are identified in the input disparity image and set to zero if they are part of the floor.

This remapping process makes it possible to detect the floor as long as the line that represents the floor was detected properly. However under certain conditions the floor detection is difficult and needs to be treated differently. This is the case if the stereo systems baseline is not perfectly parallel to the floor.

4.2 Problems with the Roll Angle

The v-disparity works well for floor detection because it transforms the floor perfectly into a line and this line is detectable. The remapping allows the removal of the floor from the disparity image in a way that only objects are left in the image. These characteristics have a limited field of activity, because they

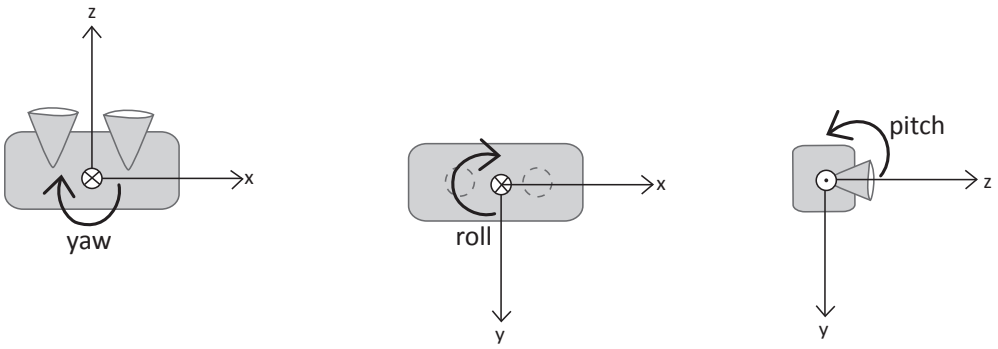
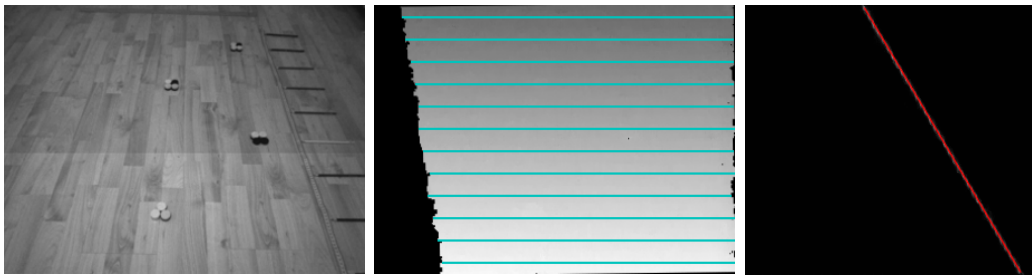


Figure 4.2: The roll-pitch-yaw angles

are highly dependable from the roll angle of the stereo camera. A disparity image from the floor with a perfectly aligned camera results in a straight line in the v-disparity image. The reason for this is that pixels with the same v-coordinate has the same disparity value, as long as noise plays a minor role, so they are transformed onto the same pixel in the v-disparity image. Nonetheless if the roll angle is varied, pixels along a horizontal line differ in disparity values and result in more than one point on the v-disparity image. The experiments from Chapter 5 show that this is not a problem for small roll angle changes which are smaller than 1°.

If Fig. 4.3 is compared to Fig. 4.4 it is visible how an applied roll angle to the stereo systems effects the v-disparity. The v-disparity with no roll angle allows a good detection of the line and the v-disparity image of Fig. 4.4 makes it hard to pick out a line. Instead a huge number of lines can be fitted into the v-disparity map since the floor results in a fanned out line.

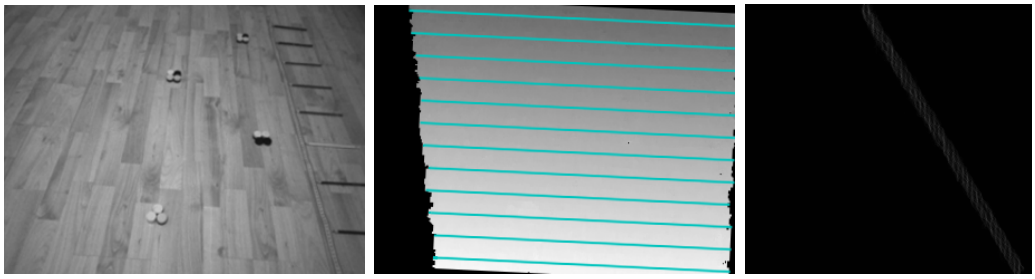


(a) Image from the stereo systems left camera of a scene

(b) Disparity map with the lines of equal disparity values

(c) v-Disparity with a clearly detectable line

Figure 4.3: A scene with a perfectly aligned stereo vision system



(a) Image from the stereo systems left camera of a scene

(b) Disparity map with the lines of equal disparity values

(c) v-Disparity with a difficult detectable broad line

Figure 4.4: A scene with a slight roll angle of the stereo vision system

One common method to avoid this problem is a roll angle correction by simply transforming the disparity image through rotation. In [3] the roll angle is measured in the calibration process in the first frame and then an affine transformation is used to correct it. In the further process it is assumed that the roll angle does not change significantly over time.

In this thesis a look is taken into two different ways to reduce the problems with roll angle changes. One of them is the multi v-disparity approach. It calculates more than one v-disparity and helps to reduce the effects of a slightly changing roll angle. The other way is a roll angle detection with the help of line fitting within a certain area of the disparity map. After the detection of the angle it is corrected by a rotation of the disparity image.

4.2.1 Multi v-Disparity

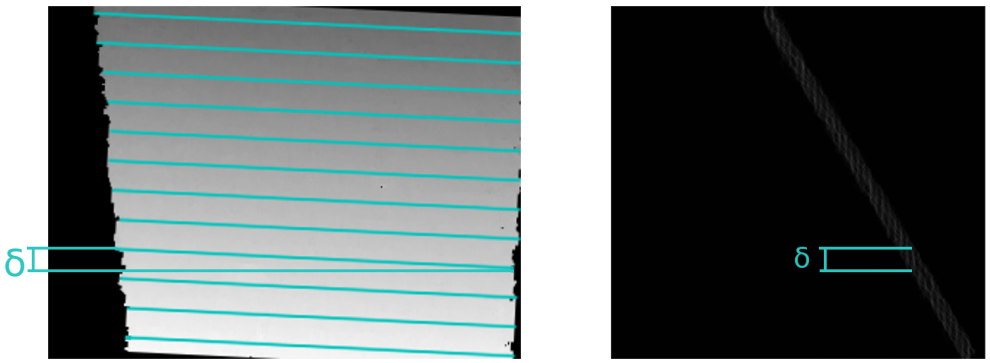
The goal with this approach is not the correction of the roll angle it is rather a reduction of the effect that is caused by the roll angle of the stereo system. As mentioned previously the roll angle results in a distorted v-disparity. The distortion affects the v-disparity map in a way that single lines that represent the floor are fanned out. These lines are similar to the lines detected if there is no roll angle but they are much thicker. These lines are still detectable but the problem now is that they lead to either too many wrongly detected floor pixels or too less detected floor pixels.

Now if the parameter for the thickness of the line is increased there is still the issue that possible points of an object represented in the v-disparity are covered by the line and are not considered in the remapping process. This results in a lower percentage of detected objects. If the parameter of the line thickness is too thin the remapping process does not consider all floor pixels

and areas with floor are detected as objects.

The main reason for the worse ground plane detection is the bigger difference in disparity values along a line of the disparity map. If the stereo system is perfectly aligned and the fluctuation of disparity values caused by noise is not considered, the values on the most left point and the most right point along a horizontal line are equal. Small changes and noise still occur in very similar values in disparity for these two points. If a roll angle is present a difference in the disparity value at the left edge and on the right edge along a horizontal line appears.

This can also be seen in the v-coordinate of the left pixel in a line of equal disparity values and the v-coordinate of the right pixel. The bigger the roll angle is the bigger the difference is in both v-coordinates of the edge pixel. This has also effects on the v-disparity image because both values have the same disparity value. The pixels fall one above the other in the v-disparity image and each of them having a different v-coordinate. This leads to the effect that the floor will be represented by a fanned out line. The difference is indicated as δ and can be seen in Fig. 4.5. The v-coordinate of the left most pixel can be considered as v_{left} and the one of the right as v_{right}. This make it possible to calculate δ as the difference between those coordinates $\delta = v_{right} - v_{left}$.



(a) δ indicated in the disparity image (b) δ indicated in the v-disparity image

Figure 4.5: Effect of the roll angle visible in the disparity and v-disparity image

So this fanned out line in the v-disparity is still detectable. It is possible to detect a broad line or a number of equal good fitting smaller lines. If all pixels are considered as floor pixels it is difficult to differ between small objects from the floor. Of course this is not an issue if the size of the object is much bigger than δ. The effect is also dependable of the position of the obstacle. Contingent upon a positive or negative roll angle either objects on the left or right boarder are still detectable.

The concept of the multi v-disparity is to split the disparity into a number of

sub disparity images. Now if the disparity is split into n equal sized sub disparity images the δ value in each of the sub disparity images are approximately $\frac{1}{n}$ of the δ value of the input disparity image. Each sub disparity image has the same height as the input disparity map but only has $\frac{1}{n}$ of the width. The v-disparity image of each sub disparity image still has the same size but the floor is represented by much less fanned out line. The effect of the roll angle is reduced with an increasing n.

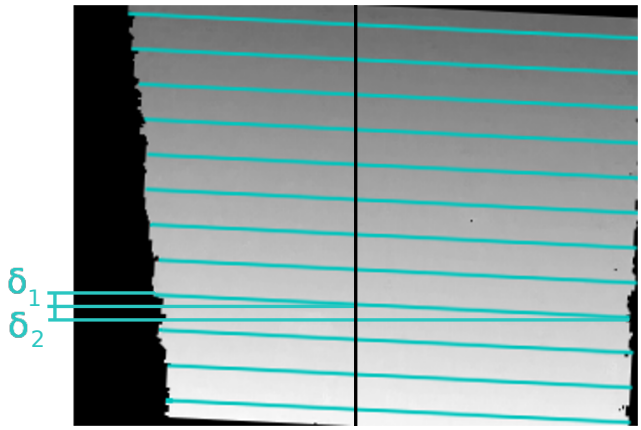


Figure 4.6: The multi v-disparity approach splits the disparity image in a number of sub disparity images. In this example n=2.

Figure 4.6 shows how the disparity image is split into two different sub disparity images. The left half of the image results in $\delta_1 \leq \delta$ and the right image in $\delta_2 \leq \delta$. The floor in the v-disparity of both sub disparity images has the values δ_1 and δ_2. This enables to detect a smaller line and differ better between floor and object pixels.

The disadvantage of the multi v-disparity is that n sub disparity images result in as many v-disparity images and therefore the floor detection needs to be calculated n times. This leads to the disadvantage of an increased calculation time.

4.2.2 Roll Angle Detection and Correction

Another way to make the floor detection more stable against a variation of the camera position is a roll angle correction. For this purpose it is necessary to detect an appearing difference in the roll angle. Theoretical it is wanted that the baseline of the camera is always parallel arranged to the ground. This setup results in a roll angle that is zero. However if the mechanical setup changes slightly it is likely that a roll angle appears.

The approach now is to calculate the rotatory deviation of to floor compared to the stereo system alignment and correct it through an image transformation. For a better understanding of the approach for the roll angle detection a look is taken into the disparity map of a perfectly aligned stereo rig and is compared with the disparity map of a rotated stereo system. Therefore the floor in a scene needs to be a dominant flat plane and can for now be considered free from any objects. Also noise might be possible but for now its influence can be considered without any effect. This is of course not always the case and therefore a look is taken into strategies to decrease the influence of noise. This principal is discussed after the explanation of the basics of the roll angle detection.

If the baseline of the stereo system camera is parallel to the floor the roll angle is zero. Every point of the floor with the same depth value is projected onto the same v-coordinate of each of the image planes. Considering the assumptions made previously and not taking into account a discrepancy due to discrete pixel values, the stereo system results in equal disparity values along a horizontal line.

A turned stereo system changed along the roll angle leads to a slightly different disparity map. Equal disparity values are horizontally orientated but equal values along diagonal lines can be seen. The slope of these lines is directly proportional to the roll angle. The goal is a roll angle calculation starting from the slope estimation of the lines with equal disparity values.

Once the angle is detected it can be corrected through an image transformation. In order to do this the disparity image is transformed in a way that the resulting new disparity map fulfils the requirements of equal disparity values along a horizontal line as well as possible. With non-ideal data it is not possible to meet the exact requirements. The task changes slightly to an error minimization of the sum of differences of disparity values along a horizontal line.

Angle Estimation from non ideal Environments

The non ideal data includes noise and possible objects in the scene. The noise arises in the possibility of different disparity values for points with the same depth value. An object in the scene will also influence the result of a roll-angle detection dependent of the geometry of the object. If the object is small enough it effects the detection only slightly and is marginal compared to the influence of the noise.

This error can not be ignored if the object exceeds a certain size. If the purpose of the object detection in a robotic application is considered this problem can be handled. One goal of the object detection is a safe path planning. This results in paths without obstacles and enables the assumption

that certain parts in the disparity map as free from obstacles. The consequence is that there is no object in the area close to the robot. In the disparity image this is the area close to the bottom of the image. The considered area is further reduced if a look is taken at disparity values in the center of the image. This is done because if the robot moves past an object on either side it might be visible to a certain degree in the boarders of the image. Now this area can be used for the roll angle detection.

For typical indoor robotic environments the assumption of an almost flat floor can be taken. If focused on the area of the floor as described above, this assumption can be expanded for a higher number of possible environments such as roads or more rough terrain. The reason for this is that the floor sections of this environments can be considered locally flat. The next step is described on a single horizontal line of the specific chosen area. Every pixels disparity value is taken and listed as y_i values, sorted by the u-coordinate of each pixel. The u-coordinate is recorded as the x_i values and can be renumbered from 1 to n without any change in result for the roll-angle detection. n is the number of considered disparity values. The v-coordinate is not important because for now the focus is on a single line with a constant v-coordinate. The goal is to fit a line in the data that minimizes the quadratic error. The following steps go into detail about the calculation of the linear least square model. The fitted line is expressed through the equation:

$$\hat{y}_i = \hat{\alpha} + \hat{\beta} x_i \tag{4.1}$$

and is calculated through linear least square fitting. The quadratic error is calculated with

$$S = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \hat{\alpha} - \hat{\beta} x_i)^2 \tag{4.2}$$

and the minimal error needs to fulfil the conditions:

$$\frac{\partial S}{\partial \hat{\alpha}} = 0 \qquad \frac{\partial S}{\partial \hat{\beta}} = 0 \tag{4.3}$$

The conditions from Eq. (4.3) result in the following equations after simplification:

$$\sum_{i=1}^{n} (y_i - \hat{\alpha} - \hat{\beta} x_i) = 0 \tag{4.4}$$

$$\sum_{i=1}^{n} x_i (y_i - \hat{\alpha} - \hat{\beta} x_i) = 0 \tag{4.5}$$

For further simplification the sum can be simplified with:

$$\sum_{i=1}^{n} x_i = n\bar{x} \qquad \sum_{i=1}^{n} y_i = n\bar{y} \tag{4.6}$$

in which $\bar{x}$ and $\bar{y}$ stand for the mean value of Eq. (4.4) and Eq. (4.5) have two unknown variables $\hat{\alpha}$ and $\hat{\beta}$ that are to be expressed. Rearranging Eq. (4.4) and using the relations from Eq. (4.6) results in:

$$n\bar{y} = \hat{\alpha}n + \hat{\beta}n\bar{x} \Rightarrow \bar{y} = \hat{\alpha} + \hat{\beta}\bar{x} \Rightarrow \hat{\alpha} = \bar{y} - \hat{\beta}\bar{x} \tag{4.7}$$

From the second condition for a minimum Eq. (4.5) rearranged to:

$$\sum_{i=1}^{n} x_i y_i = \hat{\alpha} \sum_{i=1}^{n} x_i + \hat{\beta} \sum_{i=1}^{n} x_i^2 \tag{4.8}$$

Now the result from Eq. (4.7) is used in Eq. (4.8) and rearranged with Eq. (4.6) to:

$$\sum_{i=1}^{n} x_i y_i = n\bar{x}\bar{y} - \hat{\beta}n\bar{x}^2 + \hat{\beta} \sum_{i=1}^{n} x_i^2 \tag{4.9}$$

Now the unknown variable $\hat{\beta}$ from the line equation (4.1) can be expressed and rewritten as follows:

$$\hat{\beta} = \frac{\sum\limits_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sum\limits_{i=1}^{n} x_i^2 - n\bar{x}^2} = \frac{n\sum\limits_{i=1}^{n} x_i y_i - \sum\limits_{i=1}^{n} x_i \sum\limits_{i=1}^{n} y_i}{n\sum\limits_{i=1}^{n} x_i^2 - \sum\limits_{i=1}^{n} x_i \sum\limits_{i=1}^{n} x_i} \tag{4.10}$$

With this result it is also achiveable to calculate the right value for $\hat{\alpha}$ from Eq. (4.7) and get a solution for Eq. (4.1) that minimizes the quadratic error. For the purpose of the roll angle detection and correction it is only necessary to consider the slope $\hat{\beta}$ of the line. From there the roll angle Φ can be calculated.

$$\Phi = \arctan \hat{\beta} \qquad with \qquad \Phi \in [-\frac{\pi}{2} ... \frac{\pi}{2}] \tag{4.11}$$

For a more stable detection of the angle it is necessary to not just look at a single line of data. If the number of considered rows is increased for the row angle detection it is realizable to compensate single outlier values that appear due to noise. The approach is to build the mean values of disparity values with the same u-coordinate. This can be done because the noise results in normally distributed values of the disparity. These arithmetical averaged values are the data in y_i and keep the outliers within a closer limit. This concludes in a more stable estimation of Φ under an increased noise level.

Affine Image Transformation for the Angle Correction

The estimated roll angle Φ can be corrected through an affine image transformation. The basic principal is that disparity values from the input image

with the coordinates (u_1,v_1) get transformed into new coordinates (u_2,v_2). One feature of an affine image transformation is that parallel lines are still parallel even after the transformation.

So if the focus is placed on points with equal disparity values in a disparity map, it is apparent that these points lie along a line. If the disparity map is recorded from a flat floor, there are a number of parallel lines created by equal disparity values. The goal is now that after the image transformation all these lines are still parallel to each other. Since this is not be the issue, it is desirable that all lines should also be horizontal compared to the image coordinate system. A problem with affine transformation is that angles are not preserved correctly. Nevertheless since the focus is on small angle correction this is not a matter.

An affine transformation consists of the following steps:

- Rotation
- Translation
- Scaling

To correct the angles the image simply needs to be rotated. In order not to exceeded the original image size it is necessary to crop the image. The image transformation is a matrix multiplication and uses the 2×3 matrix M:

$$M = \begin{bmatrix} a & b & (1-a)x_{center} - by_{center} \\ -b & a & bx_{center} + (1-a)y_{center} \end{bmatrix} \tag{4.12}$$

with

$$a = s\cos\Phi \qquad b = s\sin\Phi \tag{4.13}$$

The coordinate values x_{center} and y_{center} represent the image center of the input image. s is a scale factor and is set to the value $s = 1$. A positive value of Φ rotates the image counter-clockwise. Now the relation between the output image with coordinates (x_{output},y_{output}) and the input image with coordinates (x_{input},y_{input}) is related through:

$$\begin{bmatrix} x_{output} \\ y_{output} \end{bmatrix} = \begin{bmatrix} a & b & (1-a)x_{center} - by_{center} \\ -b & a & bx_{center} + (1-a)y_{center} \end{bmatrix} \begin{bmatrix} x_{input} \\ y_{input} \\ 1 \end{bmatrix} \tag{4.14}$$

After this affine image transformation the roll angle is corrected in a way that the ground plane in the v-disparity is fanned out to a minimum. This enables a unambiguously line detection in the v-disparity that represents the ground plane.

In 4.7 the process of the roll angle correction is summarized and can be listed as the following steps:

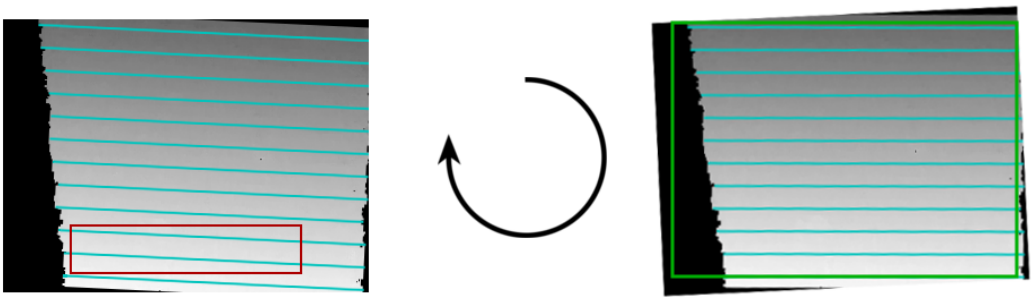


Figure 4.7: The steps of the angle detection and correction approach.

- Choose an area from the disparity image
- Fit a line in the data through error minimization
- Calculate the angle Φ of the lines slope
- Correct the image through transformation
- Use the newly generated disparity image for the further process

4.3 Line Detection with Hough Transform

The Hough transform is used to estimate the best fitting line in the v-disparity algorithm. This is done to detect the floor in the scene. After this the floor can be removed from the v-disparity map and only points that are part of an obstacle remain. After a remapping process of this point into the disparity map only objects are left and can be located in the image. The distance between the stereo system and an object is then calculated through Eq. (3.5) with the disparity values of the object.

For a recognition of complex pattern the Hough transform is first mentioned in the patent of Hough [20]. It transforms a point from an image space into an parameter space. The parameter space describes every possible line for a single point. In order to do this a line is expressed through the parameters ρ and θ:

$$x \cos\theta + y \sin\theta = \rho \tag{4.15}$$

The coordinates (x,y) can be replaced through the pixel coordinates (u,v). The point of the image is represented through a sinusoidal function in the parameter space (θ,ρ). When a second point is considered and transformed this leads to a second sinusoidal function in the parameter space. In the image space only one line can be considered that goes through two different points. In the parameter

space this line can be found at the intersection of the sinusoidal function of both image points. With the basic understanding of the Hough transformation the best fitting line for a higher number of points can be found.

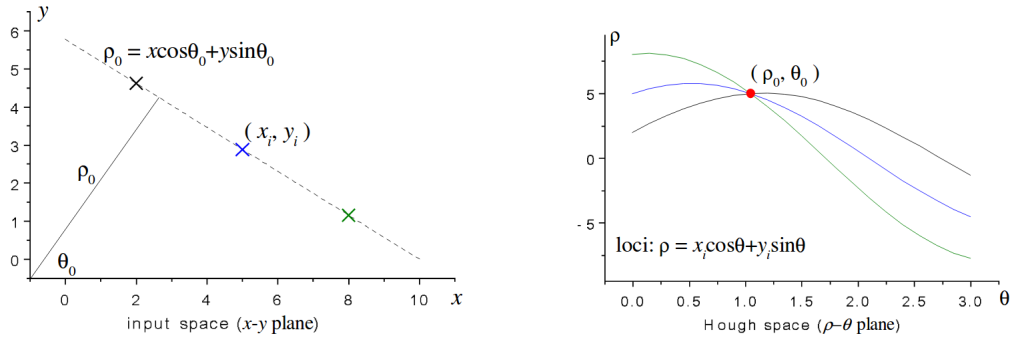


Figure 4.8: Hough transformation for three different points [21].

If more points are considered in an image space and transformed to the parameter space it leads to as many sinusoidal functions. Each of them intersect at one point with every other function. In the image space this means, it is assured that a line can be spawn between one point and every other point in the image. Figure 4.8 is a example for the transformation from the input space to the Hough space.

If there is one dominant line in the image, it means that a high number of functions intersect at the same point in the parameter space. The line is then identified through these parameters. Usually there is not just one dominant plane in the v-disparity map. In order to get a representing line in the v-disparity, all possible lines are considered that exceed a certain threshold. The threshold stands for the minimum number of intersections at a specific point.

Another important step is to dismiss results that are clearly not a representation of the floor in the v-disparity map. The floor is not parallel or close to parallel to the image plane of the stereo system. This assumption is valid since in a robotic system the stereo system is used to take images of the space in front of the robot and not just straight down to the floor. If this would be the case the floor would only consist of pixels with equal or similar disparity values and therefore be represented by vertical lines in the v-disparity image. However this is not the case for most robotic applications. So this means that the floor can not be represented through vertical lines in the v-disparity or lines similar to this. Now every possible line with a value for θ close to 90 ° can be dismissed.

The remaining number of possible lines have similar parameters and are close to each other. For the best representation of the detected floor these lines are averaged.

4.4 Noise Reduction

The disparity value of a single point at a constant distance varies over time. One of the reasons for this can be seen in the variation of lighting conditions over time. Two consecutive images have slightly changed values of intensity and this can emerge in slightly changed disparity values for a number of pixels. Experiments in Chapter 5 show that the disparity values are normally distributed with the standard deviation σ. The consequences can be seen in the v-disparity because the coordinate values of the floor pixels vary along a fitted line. This usually is detected as an object in the v-disparity. However for this purpose a threshold is added in the remapping process. The disadvantage of such a threshold is that it makes it harder to detect small objects particularly if it is an object with a small height. The choice of the value for the threshold is a conflict of interests. If it is too small the object detection algorithm detects objects wrongly and if it is too big little objects are not detected at all.

The approach in this thesis is that the threshold is chosen close to σ. Due to this there is still a number of wrongly remapped pixels but ensures that smaller objects can still be detected. In order to get rid of the wrongly detected objects a post processing step is carried out. For this purpose different filters have been tested and two of them have proven to be useful. One of them is the median blur filter and the other one is a morphological operation filter.

The goal of the filter is to get rid of wrongly detected objects. After the remapping process all detected floor pixels in the disparity map are set to zero. The probability that two pixels of a wrongly detected object are close to each other is low. The wrongly detected pixels have similarities to salt and pepper noise. So the goal of the filter is to get rid of this kind of noise but still obtain the correct size of the rightly detected objects.

4.4.1 Median Blur Filter

The median blur filter iterates every pixel of an image with a kernel. The center pixel inside the kernel is replaced with the median value. For this purpose the pixels inside the kernel are sorted by value. The center pixels value inside the kernel is then replaced by the middle value of the sorted pixels. The size of the kernel is $n \times n$ with n bigger than 1 and uneven.

Experiments show that in the disparity map after the floor removal the noisy

pixels are mostly singled out inside a close neighbourhood of the pixels. Floor pixels are all set to zero and if a wrongly detected pixels is surrounded by floor pixels the wrongly detected pixel will also be set to zero. This only fails if the number of wrongly detected pixels inside the neighbourhood is larger than 50 percent.

Another question is how this filter influences rightly detected objects. The biggest concerns are the edges of the object because it is possible that some of the edge pixels are possible surrounded by more floor pixels than object pixels. This results in a wrong dismissal of a rightly detected object pixel. The result of the filtering is that the object sizes are slightly reduced in the disparity map.

4.4.2 Morphological Operations

Several morphological operations can be used to transform images. In order to get rid of noise the morphological opening and closing operations are considered. Both of these operations are a combination of two other morphological operations. One of them is called dilation the other one is called erosion. The difference of opening and closing is just the order of dilation and erosion. The opening operation is an erosion followed by a dilation and the closing operation has the inverse order. Both of these operations have slightly different effects to pixels inside a kernel.

- dilation: expands a shape and fills holes inside the shape
- erosion: expands the background and fills holes inside the background

The kernel center iterates through every pixel of the whole input image and changes the pixels values inside a closed pixel neighbourhood. The size of the considered neighbourhood is equal to the kernel size. If the two operations are performed after another a shape will preserve its original size, because the shape gets increased and then decreased or the other way around. This also results in a removal of the noise because single pixels do not expand in the dilation step and in the erosion step they are removed. Another characteristic of this operation is that holes inside an object get filled. So if a floor pixel is surrounded by pixels with any disparity value its disparity value is changed to a interpolated disparity value of the surrounding pixels. The opening and closing operation can be used to reduce the noise. Both operations are useful in the object detection algorithm.

4.4.3 Problems with Reflections on the Floor

An issue that appears while detecting objects with a stereo vision system results from reflections of either sunlight or other bright light sources such as

light bulbs. Usually such reflections result in a number of wrongly detected pixels inside the area of reflection. The reason for this is mainly the saturated intensity values of the pixels. Due to the saturation the areas of reflection can not be distinguished for the calculation of the disparity image. The experiments from Chapter 5 show that the calculated disparity values inside the area of reflection is smaller than the expected disparity values compared to points of equal distance to the stereo system.

In order to avoid wrongly detected pixels in the area of reflection the approach is to dismiss every pixel that has smaller disparity values than the detected floor. This can be done by remapping the v-disparity image to the disparity image after the floor was detected. For this purpose pixels with smaller disparity values than floor pixels are set to zero while remapping.

4.5 A Comparison of Different Stereo Algorithms

One important step in a stereo vision system is the stereo algorithm that calculates the disparity map. The basic idea of a stereo algorithm is discussed in Section 3.3 but now three different stereo algorithms are introduced. All of these algorithms are tested for the purpose of object detection. The choice of the efficient large-scale stereo matching (ELAS) and the block matching algorithm is taken mainly because of the availability of these algorithms in the robot operation system (ROS) [22]. The semi-global matching was chosen because of the used stereo system. It uses an on-board processor to calculate the disparity with the semi-global matching algorithm. The other two algorithms run on an Intel® Core™ i7-2860QM CPU @ 2.50GHz × 8 processor that takes the input images from the same stereo camera as the semi-global matching calculation. Before the performance of the different algorithms is compared in Chapter 5, a brief overview of the functionality is given.

4.5.1 Efficient Large-Scale Stereo Matching ELAS

This algorithm was developed by Geiger, Roser and Urtasun [5]. The main purpose of it is to calculate the disparity for high-resolution images at a high frame rate close to real time. The functionality of this algorithm can be summarized with the following steps:

- Computation of the disparities for a sparse set of support points
- Generation of a two dimensional mesh via delaunay triangulation
- Calculation of a generative model

- Dense disparity calculation through solving local energy functions

At first the disparities of a set of points with unique texture are calculated. The points are chosen through the horizontal and vertical sobel filter response. The sobel filter is used to detect edge point, so mainly significant edge points are included in the set of support points. This set of points is further reduced to avoid ambiguities. It is done by a threshold for all points that exceed a certain ratio between the best and second best match. The set of points is further reduced by removing every point with a disparity that is not similar to its nearest neighbours.

After that the Delaunay triangulation [23] is used to calculate a rough estimation of the disparity values for pixels inside the set of support points. These results are used to generate a probabilistic model for every point inside of a set of support points. The generative model gives a probability for each possible corresponding pixel pair and the thereby resulting disparity.

The dense disparity is then calculated through the minimization of a local energy function. The energy function considers only disparity values that are close enough to the estimated value of the delaunay triangulation and takes also into account if points are along an epipolar line. In order to do this the algorithm requires rectified images.

The implementation of the algorithm is available as a package[1] in ROS.

4.5.2 Block Matching BM

The basic idea of the block matching algorithm is given in [17] and describes roughly the implementation of [24]. The block matching algorithm is a sparse algorithm that is able to detect good matching points. So points inside a region without any uniqueness are recognised with this algorithm. The main advantage of block matching is that it's functionality allows a fast calculation of disparity values. All the important steps of block matching are included in Section 3.3.1. Basically the steps can be summarized as following:

- Prefiltering: normalize image brightness and enhance the texture
- Correspondence search along horizontal epipolar lines using the SAD cost aggregation
- Selection of the disparity with the best fitting correspondence
- Postfiltering: eliminate bad correspondance matches

[1] `http://wiki.ros.org/elas_ros`

The implementation of the algorithm is available for ROS[2]. It can be tuned through a number of parameters such as correlation window size, prefilter size or an uniqueness ratio. The trade-off is mostly between accuracy and density of the disparity map. The parameters need to be tuned according to the application and are all experimental determined for satisfying results in Chapter 5.

4.5.3 Semi-Global Matching SGM

The semi-global matching algorithm from Hirschmüller[25] is a reliable dense stereo algorithm. The algorithm combines a pixelwise local approach and an approximation of a global energy function with a smoothness constraint. It consists of following steps:

- Pixelwise matching cost calculation
- Cost aggregation
- Disparity computation
- Disparity refinement

The semi-global matching is an iterative algorithm that calculates the disparity through a repetition of the steps listed above. For this purpose the input image pair is downscaled and an initial disparity map of the same size is needed. The initial disparity is used to calculate an improved disparity map which serves as initial disparity in the next step. The calculated disparity image is scaled up and is used as the input in the next iteration step.

For the pixelwise matching cost calculation the mutual information [26] between a set of image pairs can be used. As cost for matching intensities a probability distribution of corresponding intensities is used. The exact method is described in [27].

The cost aggregation includes a number of one-dimensional constrains that deals with non smooth neighbouring pixels. These constrains are represented by a global energy function that takes all values of the disparity image into account. Under these considerations a disparity dependent cost is calculated for each pixel. The minimum cost represents the best fitting disparity for one pixel.

The disparity is then calculated through finding the minimal cost for every pixel. After this step outliers in the disparity image are removed through

[2] `http://wiki.ros.org/stereo_image_proc`

the refinement step. It is done to remove peak values and also to manage untextured image regions.

The stereo system that recorded the data for the experiments in Chapter 5 uses the semi-global matching algorithm for a disparity image calculation. This data is used as input for the object detection algorithm. It was calculated on an on-board processor.

4.6 An Overview of the Obstacle Detection Algorithm

Through a combination of approaches from this chapter an object detection algorithm is presented. The algorithm's output delivers a disparity image where the possible obstacles are visible. This output can be used to navigate a robot around those obstacles. It can also be used for an emergency stop if an object appears unexpectedly within the robot's path.

The input for the object detection algorithm is the disparity image that is calculated within the stereo vision system. Therefore a rectified image pair is delivered by the cameras of the stereo system and processed into the disparity image.

The steps of the obstacle detection algorithm can be listed as follows:

- Roll angle detection and correction
- v-disparity calculation from disparity image
- Hough transform for line detection
- Floor pixel removal in the v-disparity
- Removal of pixel values that represent reflections on the floor
- Remapping from the v-disparity to the disparity image
- Noise reduction

The algorithm starts with the roll angle detection and correction. Therefore a certain image region in the central bottom of the input disparity image is selected. The disparity values in the selected area are considered to fit a line of equal disparity values inside it and the angle of slope is calculated. In order to correct this angle the disparity image is transformed with an affine image transformation.

The corrected disparity image is used for the v-disparity image calculation. After the transformation is done, another one takes place. This is the Hough

transform that is used to calculate the most dominant line in the v-disparity image. In order to detect a line that does not represent the floor, conditions of the θ value of the Hough transform are considered. It should not be close to zero because that would mean a close to vertical line in the v-disparity and most likely represents a wall in an indoor environment. Another condition is that the detected line needs to exceed a certain threshold value for intersections in the Hough transforms. This guarantees that only a dominant plane can be detected as floor.

Once the line detection is carried out successfully the algorithm continues with the removal of the floor pixels in the v-disparity image. If the line detection failed for some reason the algorithm stops and waits for further disparity inputs. After the removal of the floor pixels in the v-disparity, the remapping process starts. That means the removal of the floor in the input disparity image. Additional pixels, that result from reflections of light on the floor, are detected and removed. This removal of the pixels can be done because the reflections lead to disparity values that suggest that the pixels are further away than the detected floor.

After the remapping a filter is applied to reduce the noise in the disparity image. This can be done by either a median blur filter or a morphological open or closing operation.

Figure 4.9 is the block diagram of the implemented approach for the obstacle detection. It represents the approach with the roll angle detection and correction.

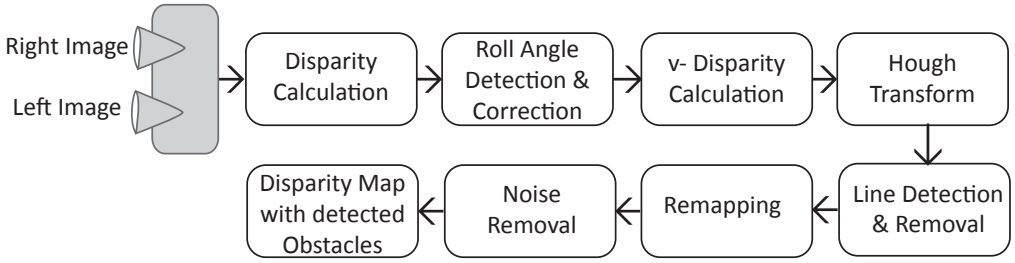


Figure 4.9: Obstacle detection approach with roll angle correction

In Fig. 4.10 the multi v-disparity approach is shown. The number of partial v-disparity calculations can vary and in the results of Section 5.4 two sub disparity images are used.

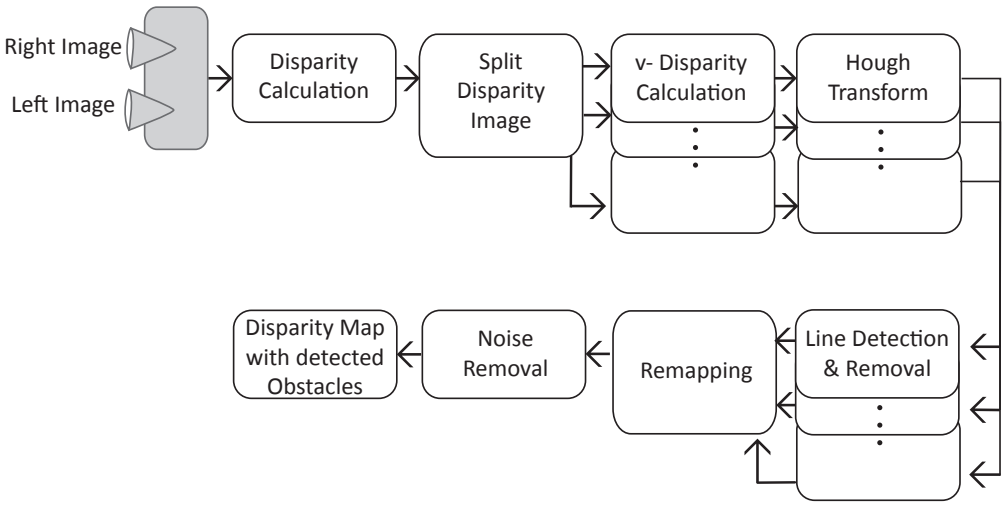


Figure 4.10: Obstacle detection approach with the multi v-disparity

5 Experiments

For an evaluation of the object detection algorithm a number of experiments are performed. The purpose of these experiments is to identify the algorithm's weaknesses. The results are then used to improve the performance of the obstacle detection algorithm. One of the experiments is applied in order to estimate the noise model of the stereo system. With these results the thresholds can be adjusted and they help to improve the performance for further results. The testing of the algorithm in an indoor environment helps to discover limits and give an insight into some of the occurring weaknesses. The knowledge gathered from the experiments helps to find strategies for avoidance and removal of problems such as an occurring roll angle to the stereo system or problems caused by reflections of light on the floor.

Other experiments are used to evaluate the performance of different stereo algorithms and allow to classify them for the purpose of obstacle detection. It advantageous to classify the reliability, speed and accuracy of the obstacle detection algorithm.

The problems of the roll angle are evaluated in another experiment that classifies the quality of the different strategies from Section 4.2.1 and Section 4.2.2. Another experiment tests strategies that handle light reflections on the floor.

All the experiments are performed with the stereo system described in Section 5.1.

5.1 Stereo Camera

The results of the experiments in this chapter are highly dependant of the used stereo system. Cameras with different specifications will result in different frame rates, noise models and success rates for the purpose of object detection. For all the experiments the same stereo system was used. For this reason a few of the important specifications of the stereo system are listed:

- Frames per second $\approx$ 12 fps
- Resolution: 640 px $\times$ 480 px

- Base width: 16 cm
- grey scale image

The stereo system is able to calculate the disparity image on board but for this purpose the images need to be downscaled. The on board calculation time was not as fast as the availability of the input image. The following specifications have to be considered.

- Number of disparity images per second $\approx$ 3 fps
- Resolution: 320 px $\times$ 240 px
- On board semi-global matching

The stereo system is also able to calculate its own odometry compared to a starting point. However this requires more processing power and reduces the disparity frame rate further. Due to this the internal odometry of the stereo system is not used in the experiments.

5.2 Measuring the Noise of the Stereo System

The measuring of the noise in the disparity calculation of a stereo systems helps to estimate the accuracy of the object detection algorithm. A well known noise model is applied for better noise reduction and gives a rough feeling for the limits in object size. A goal of these experiments is to know the noise behaviour, dependable on the distance between a considered point and the stereo system.

Nguyen, Izadi and Lovell [28] show a way for the noise model estimation that structures the noise into axial and lateral noise. In this thesis the main focus is on the axial noise. The axial noise is the variation of the disparity value along the z-axis which can be equated in a variation of depth.

For the noise model estimation only the semi-global matching stereo algorithm is taken into account. The block matching and ELAS algorithm deliver poor results in calculating the disparity of the ambiguous plane surface properly. Also the semi-global matching algorithm has some troubles with a smooth disparity calculation but still is reliable enough for a proper evaluation.

5.2.1 Experimental Setup

For this experiment a plane surface is placed in front of the stereo system. The surface is placed parallel to the stereo vision system and the data of the disparity is recorded for twelve different distances between surface and the

stereo system. The smallest distance Z for the plane is 35cm and the other distances are incremental increased in 15cm steps up to 230cm. Figure 5.1 shows the set-up.

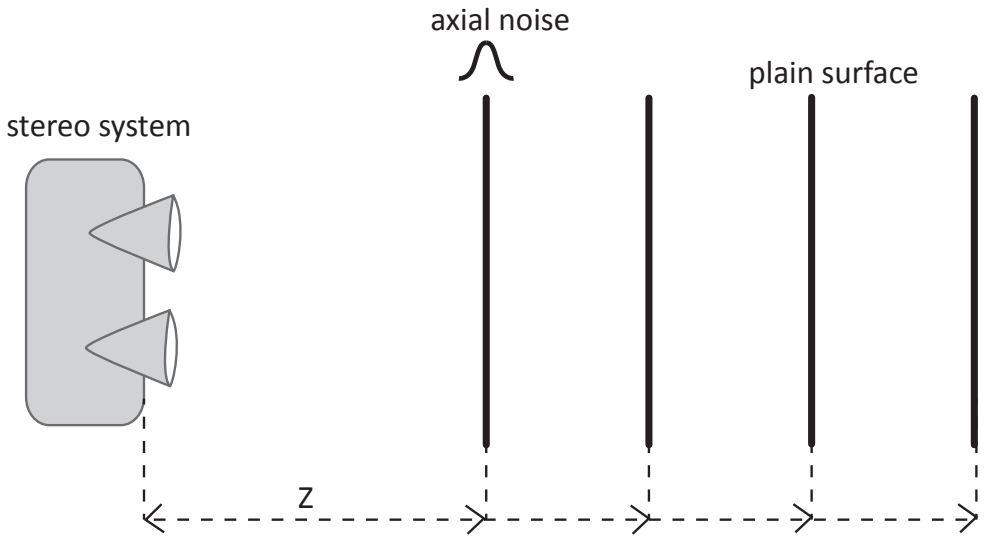


Figure 5.1: Set-up noise-measuring.

From the recorded data five frames of every distance are chosen to evaluate the axial noise of the disparity values. Furthermore regions without any disparity values due to ambiguities are discounted and the areas close to the edge of the plane are not taken into account as well. The image editing software GIMP (GNU Image Manipulating Program) is used to calculate the histogram of the valid pixels. The pixel value is a normalized disparity value in the range of 0 to 255. Figure 5.3 shows the valid area in light blue, the discounted pixels inside this area in magenta and the result of the histogram of a single frame. It is clear that in the histogram only pixels with a valid disparity value are selected for the evaluation. The orange area in the histogram shows the disparity values of the considered pixels.

5.2.2 Results

The results include the change of standard deviation of disparity values along the distance. In the disparity images from 5.2 it is recognizable that big areas of the disparity are not calculated. One reason for this are occlusions close to the boarder of the plane. The scene behind the plane is clearly visible in the left image but not visible in the right image. This makes it impossible

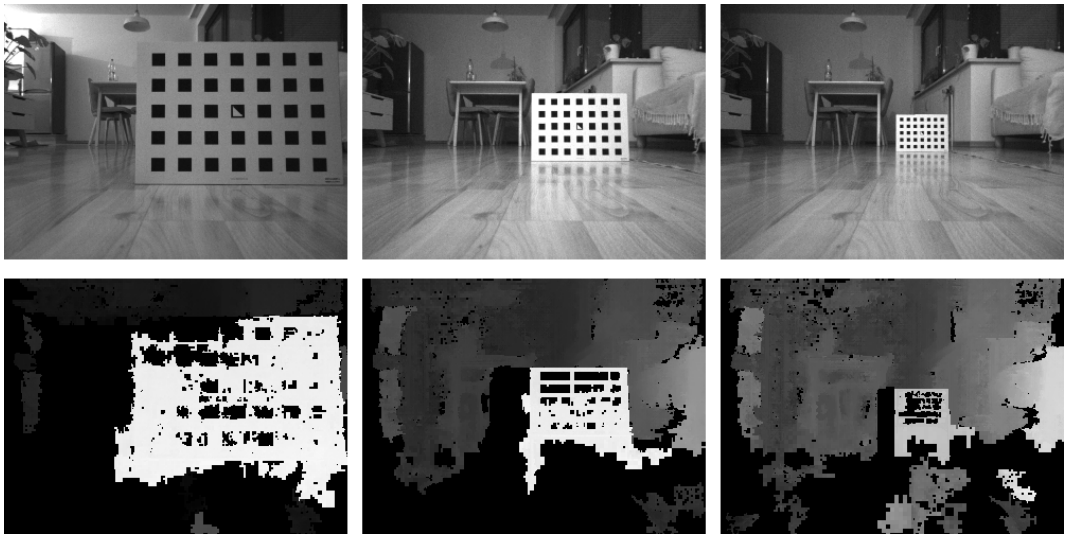


Figure 5.2: Input images from the left camera of the stereo system and disparity images from the stereo system.

Figure 5.3: Left: disparity image, center: Disparity Image with selected area, right: Histogram

to calculate the disparity values of these areas. It can be observed that these types of error decrease if the plane is further away. The reason for this is that the occluded area is smaller at a greater distance. It is detectable that the reflections on the floor make a calculation of the disparity value harder and the calculated value is most of the time wrongly calculated or not at all calculated. Figure 5.4 shows that the axial noise increases with the distance.

5.3 Obstacle Detection for a Robot Indoor Scenario

The requirements for an object detection algorithm are dependant on the application. In these experiments the goal is to find out how well the algorithm

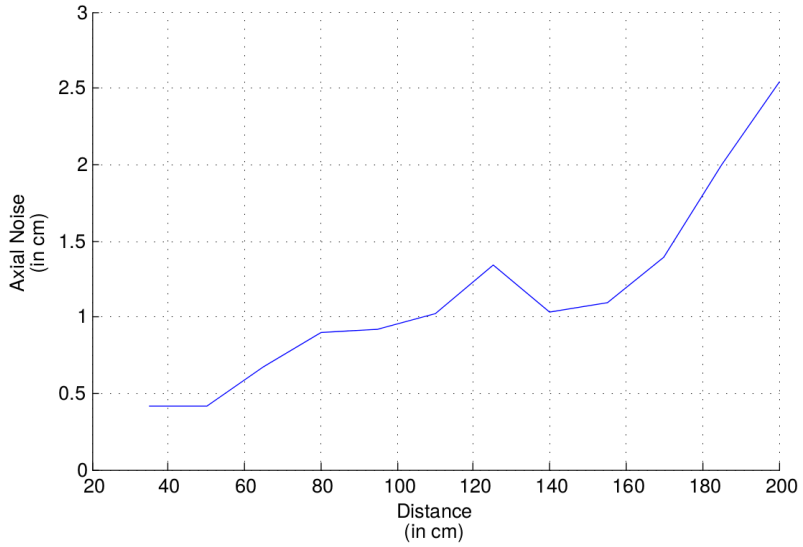


Figure 5.4: Result from noise-measuring

runs in an indoor environment. The aim is to discover what types of problems appear in this set-up and where does the algorithm fail to perform properly.

5.3.1 Experimental Setup

In this experiment the stereo system is mounted on a remote-controlled robot and navigatsd through a predefined path in an indoor scene. The navigation takes place in a hallway with several small objects spread out over a certain area. The size of the object differ slightly and have at least a height of 2cm, with the biggest object smaller than 8cm.

5.3.2 Results

The results of this experiment show disadvantages of the obstacle detection algorithm. As a satisfying result of the algorithm the floor is removed completely and only the objects on the plane are visible.

In Fig. 5.7 a decent result of the object detection algorithm is visible. It shows the input image from the left camera of the stereo system, the disparity map from the stereo system, the calculated v-disparity image and the calculated disparity map in which only obstacles are visible.

The identified problems of the obstacle detection algorithm in the robotic scenario can be listed as follows:

- Reflections of light on the floor

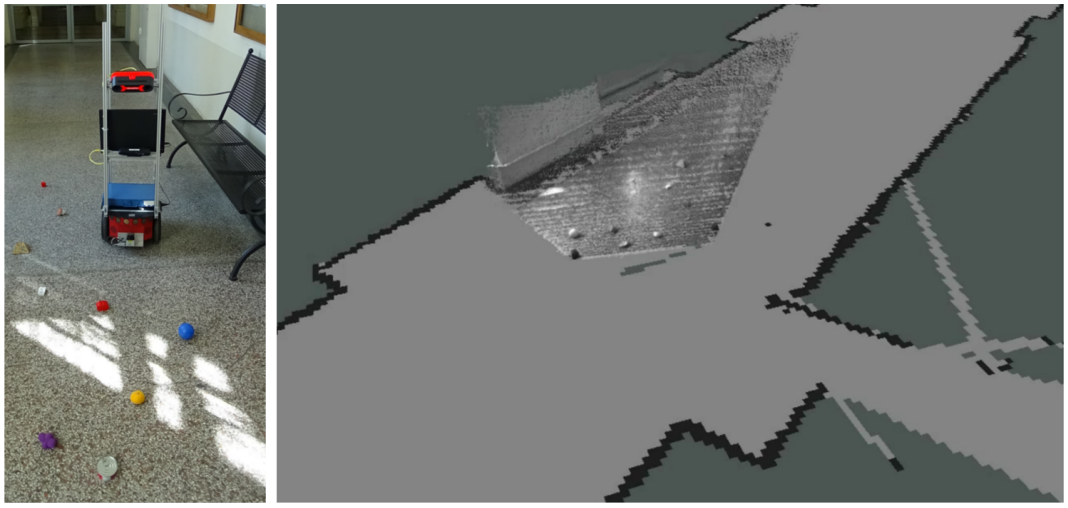


Figure 5.5: Robot and the point cloud of the stereo system overlaid to the map

- Problems with Hough transform when only a small area of the floor is visible
- Loss of reliability due to differences in roll angle

Fig. 5.8 shows that the reflections of sunlight on the floor are misinterpreted as obstacles. The reason for this is that the disparity calculation fails, due to saturation of the intensity values in this image areas. The stereo system is not able to adjust the exposure time properly because darker areas and bright areas are present in the image. Theis results in wrongly matched disparity values for the saturated areas.

Another problem occurs when there is no dominant floor visible in the input image pair. This is because of the fact that the robot is navigated close to the wall. Figure 5.9 is an example where the algorithm still manages to detect the floor properly, but it perfectly shows that the line in the v-disparity is not as dominant compared to the result from Figure 5.7. The problem arises from the Hough transform, because the points of the floor in the v-disparity image are too few. This causes that it is not possible to detect the line which represents the floor. With adjustment to the parameters in the Hough transform it is still possible to detect the floor in every single frame of this experiment. Figure 5.9 shows also how the disparity calculation fails in ambiguous image regions such as the radiator. This is not a problem for robot navigation because the surrounding image region is still calculated correctly.

The issue with the roll angle is pointed out as well. Due to the robot's movement the angular difference measured to the floor changed slightly over time. Also the stereo system was not mounted perfectly parallel to the floor. So

Figure 5.6: Different image pairs of the robot scenario

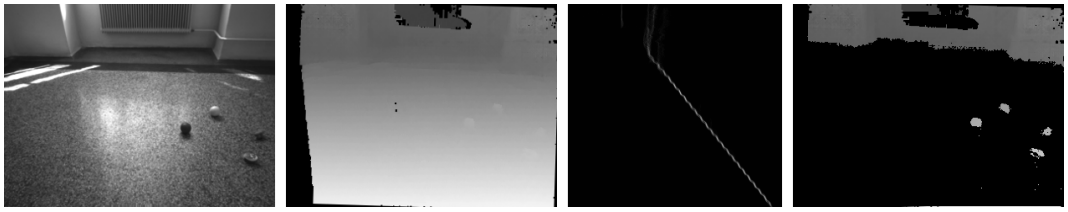


Figure 5.7: A good result of the object detection algorithm

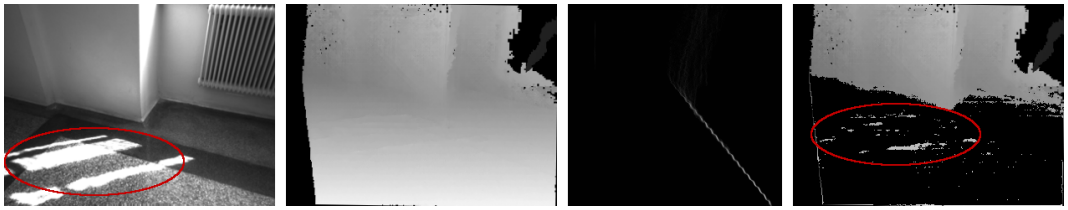


Figure 5.8: A scene with reflections of the sunlight on the floor

this results in scattered points in the v-disparity image and make the remapping much harder. Figure 5.10 shows that disparity map with the floor was not able to remove the whole floor properly. In the v-disparity image it is not possible to detect every floor pixel correctly. Some of the pixels are detected as objects and are not removed after the remapping process.

Figure 5.9: A scene with no dominant floor in the image

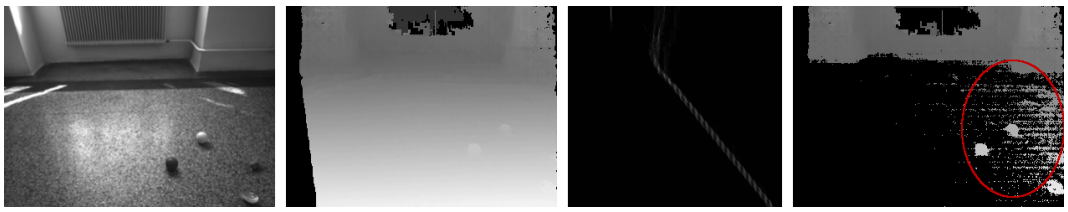


Figure 5.10: A scene with problems from the roll angle

5.4 Object Detection with Different Roll Angles

One of the results from Section 5.3 is that an occurring roll angle causes a number of wrongly detected pixels. In order to analyze this effect different roll angle changes are applied and data is recorded. For the testing of the different strategies to avoid the problem with the roll angle, the strategies are implemented in the object detection algorithm. This experiment is used to analyze the effects of the roll angle and how the multi v-disparity compares to the roll angle detection and correction.

5.4.1 Experimental Setup

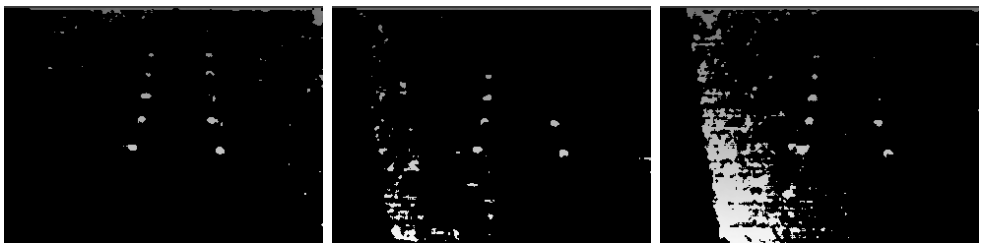
The experimental setup is similar to the one in Section 5.5 with the same objects as in Table 5.1 and under the same lighting conditions. The arrangement of the obstacle is the same as in Section 5.5. Additionally the stereo vision system is rotated by 1°and also 2°. The obstacles in the scene are stacked cylinders with a total height of 2.25cm and a diameter of 2.8cm. Figure 5.11 shows the arrangement for this experiment.

5.4.2 Results

The first Figure 5.12 shows how the roll angle affects the obstacle detection result without correcting the problem at all. Depending if the roll angle was applied clockwise or counter-clockwise, wrongly detected pixels appear on the boarder of the disparity image, in this example, on the left side. It is also

Figure 5.11: Arrangement of the scene for the evaluation of the experiment with roll angle

detectable in Fig. 5.12 that the objects on the right side are not detected because of the roll angle.

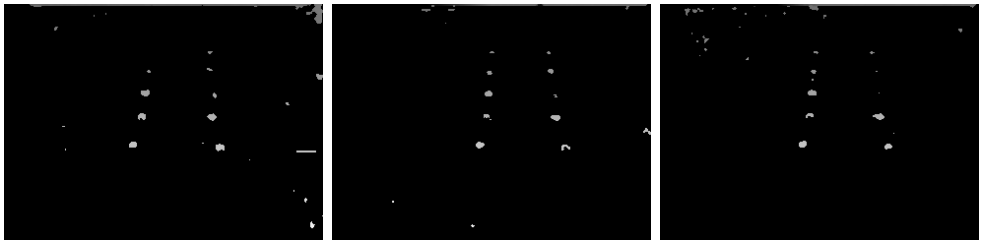


(a) Roll angle = 0° (b) Roll angle = 1° (c) Roll angle = 2°

Figure 5.12: Results without a roll angle correction

The results from Fig. 5.13 are calculated with the use of the obstacle detection with a roll angle correction. It is visible that the results are as well as they would be if the stereo vision system was perfectly aligned.

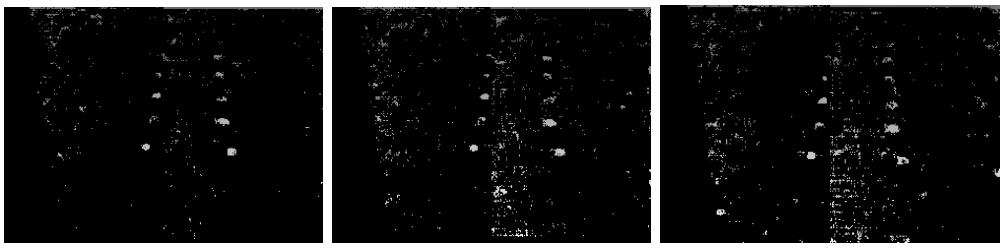
Figure 5.15 shows a reduction of the effects visible in Fig. 5.12 but there are still a number of wrongly detected obstacles. The reason for this can be better described with the results from Fig. 5.14. It displays the results from the obstacle detection before noise reduction is accomplished. It shows clearly that the effects from the roll angle are now split into two vertical halves of the disparity image. If the number of sub disparity images are increased for the multi v-disparity the effect is further reduced. Additionally this means an increased calculation time because for every single sub disparity, every step of the obstacle detection algorithm needs to be executed again. Nevertheless if



(a) Roll angle = 0° (b) Roll angle = 1° (c) Roll angle = 2°

Figure 5.13: Results with a roll angle correction

the terrain does not allow a proper roll angle correction this approach can be improved further.



(a) Roll angle = 0° (b) Roll angle = 1° (c) Roll angle = 2°

Figure 5.14: Results with multi v-disparity before noise reduction is done.

(a) Roll angle = 0° (b) Roll angle = 1° (c) Roll angle = 2°

Figure 5.15: Results with multi v-disparity

5.5 Object Detection for Small Objects

In order to evaluate the performance of the different stereo algorithms the obstacle detection algorithm was tested on a scene with small objects arranged within it. For this purpose the obstacle detection used the disparity images

of different stereo algorithms as input. The data is collected from an indoor environment and allows to test the performance of the block matching, ELAS and the semi-global matching algorithm. The intersection over union metric (IoU) is used to compare the success of the detection algorithm.

5.5.1 Experimental Setup

The stereo system was mounted static above the floor while observing an area of approximately $2.2m^2$. In the observed area objects were placed at different distances, measured to the stereo systems. Figure 5.16 displays the arrangement of the stereo vision system.

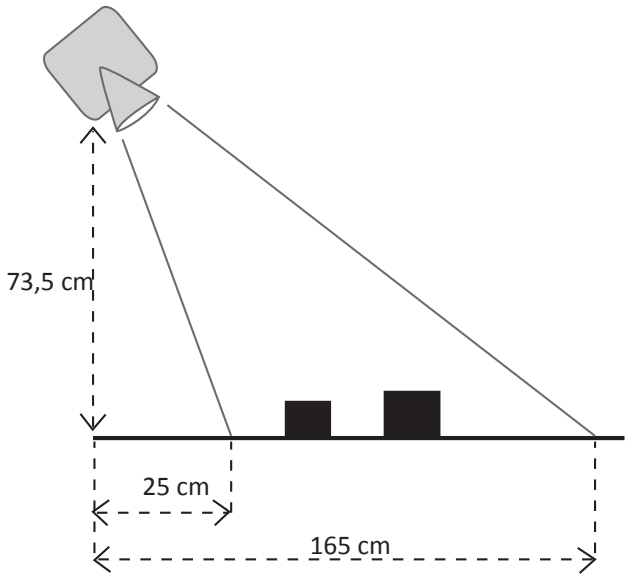


Figure 5.16: The set-up of the experiment for the detection of small obstacles.

The distances of the objects are indicated along the floor. Three different objects have been used:

object	geometry	size
Backgammon stone	cylinder	ø2.8cm × 0.75cm
Glass Object 1	cuboid	3cm× 3cm× 2.3cm
Glass Object 2	sphere	ø4cm

Table 5.1: List of objects

The backgammon stones were stacked in a set of either 2, 3 or 4 stones. The size of the stacks is delivered in the results of these experiments.

5.5.2 Intersection Over Union

The intersection over union (IoU) is used to evaluate the performance of the object detection algorithms. The IoU is also used in [29] for evaluation of pedestrian detection system on a moving vehicle. IoU allows to compare the success rate of a detected object. For the calculation of the IoU two inputs are needed, one of them is the bounding box of a marked object in the input image and the other one is the bounding box of the object in the output image. Now both bounding boxes are overlaying and they build the area of overlap I and the area of union U. The IoU can then be calculated from:

$$IoU = \frac{I}{U} \tag{5.1}$$

If the bounding box of the detected object perfectly overlaps with the bounding box of the input the IoU value is 1. If there is no overlap at all the score is 0. As a well matching results a score is over 0.5 and an insufficient matching result is below that. If small objects are considered it is more difficult to perform high IoU values for the detection because in general the area of union of the bounding box is rather small compared to the whole image size. Now if the detection has a slight offset or an error in size of one or two pixels, it leads to a low IoU value. If the bounding box of an object of small size is assumed with the size of 6 pixel $\times$ 6 pixel and the bounding box of the detected object perfectly overlaps it results in a perfect IoU score of 1. If an offset is applied to the position of the bounding box of the detected object of 1 pixel in either the vertical and horizontal direction the score drops down to approximately 0.53 and if the offset is increased to a total of 2 pixels in either direction the score is approximately 0.29. So with the consideration of possible errors calculated in Section 5.2 three different scores are considered:

- $IoU \leq 0.25$ are considered as insufficient matching results
- $0.25 < IoU < 0.5$ are considered as reasonable results
- $IoU \geq 0.5$ are considered as satisfying matching results

5.5.3 Results

The results of the performance of the obstacle detection algorithm under the use of different stereo algorithms are visible in Table 5.2. The obstacle in the

scene are stacked cylinders with a total hight of 2.25cm and a diameter of 2.8cm. Table 5.2 show that the best result came from the ELAS algorithm for obstacles at close distance but the matching results decreased over the distance. The SGM algorithm performed most consistent with almost constant result over the distance. The BM algorithm delivered only reasonable results.

object height	distance	∅ IoU SGM (σ)	∅ IoU BM (σ)	∅ IoU ELAS (σ)
2.25cm	40cm	0.600 (0.0912)	0.443 (0.167)	0.610 (0.113)
2.25cm	55cm	0.583 (0.070)	0.379 (0.165)	0.547 (0.099)
2.25cm	70cm	0.563 (0.072)	0.458 (0.140)	0.531 (0.098)
2.25cm	85cm	0.568 (0.094)	0.227 (0.208)	0.299 (0.165)

Table 5.2: comparison of IoU to different stereo algorithms.

Calculation Time for the Obstacle Detection

In Table 5.3 the calculation times of the different stereo algorithms are compared with each other. The roll angle correction of the obstacle detection algorithm is not used and allows slightly faster calculation times. The values are averaged over 20 frames and the standard deviation σ is also given. The hardware that runs the obstacle detection is mentioned in Section 4.5. The reason for the good result of the semi-global matching is that it uses half the resolution for the disparity image compared to the BM algorithm and the ELAS algorithm.

∅ time in ms SGM (σ)	∅ time in ms BM (σ)	∅ time in ms ELAS (σ)
4.58 ms (0.36)	11.48 ms (1.46)	10.65 ms (1.33)

Table 5.3: comparison of the calculation time for the obstacle detection using different stereo algorithms

The obstacle detection has not a big impact on calculation time compared to other parts of the stereo vision system. Compared to the stereo algorithm's calculation time for the disparity image the obstacle detection's calculation time is significantly faster. The semi-global matching delivers disparity images at a rate of approximately 3 Hz and the other two approximately 12 Hz. A faster rate of disparity image inputs allows a faster obstacle detection calculation. The advantage of the semi-global matching is that it is done completely off-board and therefore does not require additional computational resources.

5.6 Challenge of Reflections on the Floor

Section 5.3 shows that there are problems with reflections of the sunlight on the floor. This causes wrongly detected objects that can be avoided with the use of the strategy from Section 4.4.3 which removes pixels with bigger distance than the detected floor. For the evaluation of this improvement of the obstacle detection algorithm, different scenes with reflections of the floor are recorded. Objects are placed in the spots with the reflections on the floor.

5.6.1 Experimental Setup

The experimental setup is similar to the one in Section 5.5 but additional light sources are added as the environmental influences.

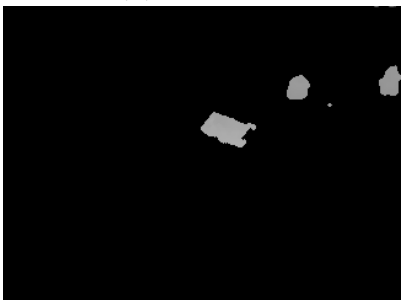
5.6.2 Results

The experiments show that the error caused by reflections of light can be compensated. It is also presented that the obstacle detection is not influenced by removing the error caused by the reflections. Figure 5.17 and Fig. 5.18 show a comparison of the results when the remapping process treats the reflections and when it does not.

(a) Image of the left camera

(b) Disparity

(c) Result without removal of reflections

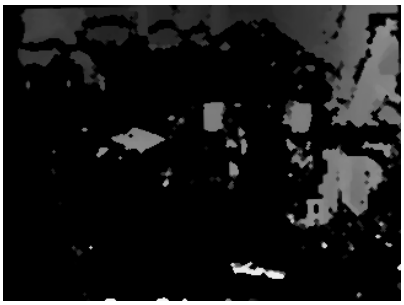


(d) Result with removal of reflections

Figure 5.17: Experiment with bright reflections on the floor scene 1

(a) Image of the left camera

(b) Disparity



(c) Result without removal of reflections

(d) Result with removal of reflections

Figure 5.18: Experiment with bright reflections on the floor scene 2

6 Conclusion

This thesis implements an obstacle detection algorithm based on stereo vision and evaluates the algorithm for real-life robotic indoor scenarios. This allows to classify the accuracy, speed and reliability of the interaction between software and hardware. The focus is on detecting rather small obstacles. The evaluation reveals strengths and weaknesses of the introduced algorithm. For better results the algorithm is improved and test scenarios are recorded to evaluate this progress. It is documented how the limits of a stereo vision system can be evaluated. A comparison of different stereo algorithms is given in order to understand their influence on the results of a stereo vision system.

The functionality of the obstacle detection algorithm is traced in Chapter 4 and defines the single steps. It is noticeable that the composition of sub tasks allows to fulfill the more sophisticated task of obstacle detection. The importance of experiments, that tune the parameters properly, can be understood with the experiment regarding noise-measuring. A compensation of external influences shows better results. Roll angle changes that may appear from the robots' movements lead to errors, that can be compensated though. Furthermore the lighting conditions can cause errors but a strategy is shown to account for that.

Some problems are still challenging and arouse interest for further research. One of these topics could investigate further the matter of floor detection. If the floor is not a flat dominant plane the presented algorithm can result in wrong obstacle detections. The idea behind the multi v-disparity of segmenting the disparity image serves as basis for an improvement in this case.

The experiments' results also show that objects are not equally good detectable in every single frame. If more frames are considered for the obstacle detection it can lead to an improved reliability, albeit this approach may result in a slower calculation time.

The obstacle detection algorithm can be implemented in a robotic system in order to improve safe robot navigation and might be useful for other robotic tasks, such as obstacle recognition.

Bibliography

[1] D. Kragic, M. Vincze, *et al.*, „Vision for robotics,“ *Foundations and Trends in Robotics*, vol. 1, no. 1, pp. 1–78, 2009.

[2] R. Labayrade, D. Aubert, and J.-P. Tarel, „Real time obstacle detection in stereovision on non flat road geometry through"v-disparity"representation,“ in *Intelligent Vehicle Symposium, 2002. IEEE*, IEEE, vol. 2, 2002, pp. 646–651.

[3] X. A. Umar Ozgunalp Rui Fan and N. Dahnoun, „Multiple Lane Detection Algorithm Based on Novel Dense Vanishing Point Estimation,“ IEEE, paper, 2017.

[4] A. Burlacu, S. Bostaca, I. Hector, P. Herghelegiu, G. Ivanica, A. Moldoveanul, and S. Caraiman, „Obstacle detection in stereo sequences using multiple representations of the disparity map,“ in *System Theory, Control and Computing (ICSTCC), 2016 20th International Conference on*, IEEE, 2016, pp. 854–859.

[5] A. Geiger, M. Roser, and R. Urtasun, „Efficient large-scale stereo matching,“ in *Asian conference on computer vision*, Springer, 2010, pp. 25–38.

[6] R. Ait-Jellal and A. Zell, „A fast dense stereo matching algorithm with an application to 3d occupancy mapping using quadrocopters,“ in *Advanced Robotics (ICAR), 2015 International Conference on*, IEEE, 2015, pp. 587–592.

[7] D. Scharstein and R. Szeliski, „A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,“ *International journal of computer vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[8] Y. Boykov, O. Veksler, and R. Zabih, „Fast approximate energy minimization via graph cuts,“ *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[9] Q. Yang, L. Wang, and N. Ahuja, „A constant-space belief propagation algorithm for stereo matching,“ in *Computer vision and pattern recognition (CVPR), 2010 IEEE Conference on*, IEEE, 2010, pp. 1458–1465.

[10] K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard, „Octomap: a probabilistic, flexible, and compact 3d map representation for robotic systems," in *Proc. of the ICRA 2010 workshop on best practice in 3D perception and modeling for mobile manipulation*, vol. 2, 2010.

[11] M. Wandfluh, „Obstacle detection using v-disparity: integration to the crab rover," Bachelor thesis, Swiss Federal Institute of Technology Zurich.

[12] C. Chautems, *Obstacle detection for the crab rover*, 2009.

[13] F. Oniga and S. Nedevschi, „Processing dense stereo data using elevation maps: road surface, traffic isle, and obstacle detection," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 3, pp. 1172–1182, 2010.

[14] Y. Xu, Q. Long, S. Mita, H. Tehrani, K. Ishimaru, and N. Shirai, „Real-time stereo vision system at nighttime with noise reduction using simplified non-local matching cost," in *Intelligent Vehicles Symposium (IV), 2016 IEEE*, IEEE, 2016, pp. 998–1003.

[15] A. Buades, B. Coll, and J.-M. Morel, „A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, IEEE, vol. 2, 2005, pp. 60–65.

[16] K. Honauer, L. Maier-Hein, and D. Kondermann, „The hci stereo metrics: geometry-aware performance analysis of stereo algorithms," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2120–2128.

[17] G. Bradski and A. Kaehler, *Learning OpenCV*. Sebastopol: O'Reilly Media Inc., 2008, vol. 1.

[18] J. J. Moré, „The levenberg-marquardt algorithm: implementation and theory," in *Numerical analysis*, Springer, 1978, pp. 105–116.

[19] T. Meltzer, C. Yanover, and Y. Weiss, „Globally optimal solutions for energy minimization in stereo vision using reweighted belief propagation," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, IEEE, vol. 1, 2005, pp. 428–435.

[20] H. C, *Method and means for recognizing complex patterns*, US Patent 3,069,654, 1962. [Online]. Available: `https://www.google.com/patents/US3069654`.

[21] X. Lin and K. Otobe, „Hough transform algorithm for real-time pattern recognition using an artificial retina camera," *Opt. Express*, vol. 8, no. 9, pp. 503–508, 2001. [Online]. Available: `http://www.opticsexpress.org/abstract.cfm?URI=oe-8-9-503`.

[22] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, „Ros: an open-source robot operating system," in *ICRA workshop on open source software*, Kobe, vol. 3, 2009, p. 5.

[23] L. P. Chew, „Constrained delaunay triangulations," in *Proceedings of the third annual symposium on Computational geometry*, ACM, 1987, pp. 215–222.

[24] K. Konolige, „Small vision systems: hardware and implementation," in *Robotics research*, Springer, 1998, pp. 203–212.

[25] H. Hirschmuller, „Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[26] P. Viola and W. M. Wells, „Alignment by maximization of mutual information," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, IEEE, 1995, pp. 16–23.

[27] J. Kim *et al.*, „Visual correspondence using energy minimization and mutual information," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, IEEE, 2003, pp. 1033–1040.

[28] C. V. Nguyen, S. Izadi, and D. Lovell, „Modeling kinect sensor noise for improved 3d reconstruction and tracking," in *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*, IEEE, 2012, pp. 524–530.

[29] M. Bajracharya, B. Moghaddam, A. Howard, S. Brennan, and L Matthies, „Results from a real-time stereo-based pedestrian detection system on a moving vehicle," in *Workshop on People Detection and Tracking, IEEE ICRA*, Citeseer, 2009.

Erklärung

Hiermit erkläre ich, dass die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt wurde. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Die Arbeit wurde bisher weder im In– noch im Ausland in gleicher oder in ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Vienna, June 2017

Bernhard Neuberger BSc.