Faculty of Aerospace Engineering

Enhanced Sparse Depth Reconstruction Using Edge and Temporal Information

an Application to Micro Air Vehicles

T.A. Heil

June 30, 2017

TU Delft Delft University of Technology

Challenge the future

Enhanced Sparse Depth Reconstruction Using Edge and Temporal Information

an Application to Micro Air Vehicles

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

T.A. Heil

June 30, 2017

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Copyright © T.A. Heil
All rights reserved.

Delft University Of Technology
Department Of
Control and Simulation

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Enhanced Sparse Depth Reconstruction Using Edge and Temporal Information"** by **T.A. Heil** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: June 30, 2017

Readers:

Dr. ir. C.J.M. Verhoeven

Dr. G.C.H.E. de Croon

Ir. C. de Wagter

Preface

This thesis presents the results from all the conducted research of the past year. With this work I will conclude my time at the faculty of Aerospace Engineering at the Delft University of Technology, where I started my journey as a young aspiring student to become an engineer ready for the challenges of the future.

The basis for this thesis came during my internship at Temasek Laboratories at the National University of Singapore (NUS), where I did research in obstacle avoidance based on visual cues. The internship inspired me to focus on computer vision for autonomous navigation and seek a thesis topic in the field. With the help of dr. Guido de Croon I found the opportunity to conduct a part of the research work at Temasek Laboratories at NUS for several months. This led the graduation project to touch a wide range of topics in the field of autonomous navigation and computer vision, creating a large academic context for the main contribution of this thesis; the scientific article about Sparse Sensing based Depth Reconstruction.

I would like to thank all the people who helped me while working on this thesis and for making my internships at Temasek Laboratories possible. Firstly I want to thank my supervisor dr. Guido de Croon for his guidance and expertise which allowed me to gain new insights and focus my attention on the topics that mattered when needed. I would like to thank dr. Gao Zhi for his guidance and the countless discussions regarding numerous scientific and engineering challenges during my time in Singapore. Furthermore I would like to thank dr. Lin Feng, dr. Teo Swee Huat Rodney and dr Guido de Croon for their help in realizing my internship at Temasek Laboratories. Also I would like to thank Gerald van Dalen for proofreading this work. Lastly I would like to thank all my friends, colleagues and family for their support throughout the duration of this project.

With the finish line in sight I come to understand what this journey has brought me and I feel strengthened in my commitment to society and by helping to translate knowledge into technological innovations.

Tobias A. Heil
June 30, 2017
Delft, The Netherlands

Acronyms

BIT	Batch Informed Trees
C-space	Configuration Space
CNN	Convolutional Neural Network
FAST	Features from Accelerated Segment Test
FMT	Fast Marching Tree
LIDAR	Light Detection and Ranging
MAV	Micro Air Vehicle
MSAC	M-estimator Sample Consensus
NUS	National University of Singapore
OBVP	Optimal Boundary Value Problem
PD	Proportional Derivative
PRM	Probabilistic RoadMap
RABIT	Regionally Accelerated Batch Informed Trees
RANSAC	Random Sample Consensus
RRG	Rapidly exploring Random Graph
RRT	Rapidly exploring Random Tree
SAD	Sum of Absolute Differences
SEEDS	Superpixels Extracted via Energy-Driven Sampling
SLIC	Simple Linear Interactive Clustering
SST	Stable-Sparse-RRT
SVM	Support Vector Machine
UAV	Unmanned Aerial Vehicle

Contents

Chapter 1

Introduction

Recent advances in research have brought the capabilities of Micro Air Vehicles (MAVs) to a level at which versatile autonomous flight is within reach. These advances made the platform gain attention from governments and industry for various applications like exploration and reconnaissance, surveillance and delivery, and commercial applications like aerial photography.

With this increased interest also the demand on the MAVs capabilities are increased. Most current commercial MAVs are designed for outdoor flight where they use GPS way-point navigation and obstacle avoidance is of lesser importance as they mostly operate at a sparse obstacle altitude.

Future MAVs will be expected to show a great sense of situational awareness, such as the ability to identify no-fly-zones around airports and government facilities and allow for safe flight in highly cluttered and spatially constrained environments.

The latter has proven to be challenging and is still topic of research, with applications in exploration and search missions in earthquake affected villages. The goal of this thesis is to develop and implement a novel method to efficiently identify feasible vehicle-size dependent algorithm that is capable of identifying spatially permitting funnels in highly cluttered environments. The thesis objective is formulated as follows:

> *Design a novel vision-based algorithm capable of identifying free-space trajectories in highly cluttered environments.*

1-1 Research Context and Problem Statement

Current methods in the field of autonomous flight for MAVs require off-line computations due to computationally demanding algorithms, or are restricted in free flight in highly cluttered environments.

Vision-based approaches are currently most suited due to the weight, power and computational constrains of the platform, these restrict the perception sensors to be optical sensors.

The most recent computer vision approaches shows promising results as rapid advances are being made for various applications.

Methods with the ability to identify spatially constrained but feasible flight trajectories in real-time, with the potential for full on-board implementation have not been developed. They will allow for free exploration missions in environments which are currently inaccessible to MAVs.

The purpose of this Thesis is to contribute to the low-level vision-based obstacle avoidance methods developed at MAVLab[1] at the Delft University of Technology. Throughout the report it is assumed that solutions are intended to make use of a small in-house build stereo-camera board, which provides sparse disparity map using selective block matching. In order to increase the exploration capabilities, a novel and computationally efficient vision-based method has to be developed.

1-2 Research Questions

In order to proceed in a structured way, the following research questions are formulated. Based on the state-of-the-art methods presented in the next chapter, an approach will be proposed which shall form the basis for the development of a novel algorithm for feasible flight in highly cluttered environments.

- Which recent methods for real-time trajectory planning may be used without prior knowledge of the environment?
- Can superpixels segmentation be applied at real-time, for implementation on the MAVLab stereoboard?
- Can superpixel segmentation be used to expand a sparse configuration space?
- Which sensing method shows the most promising performance?
- What methods have been developed to represent the environments for trajectory planning?
- What type of representation method is most suitable for memory and computationally constrained platforms?
- Can the real-time trajectory planning be combined with efficient representation?

1-3 Thesis Layout

This thesis consists of four parts, the first part contains the scientific article and also forms the main part of this thesis. It introduces the key methods and their position within the relevant research, the results and the corresponding conclusions and recommendations.

[1] http://mavlab.tudelft.nl/

Part II consists of a broad literature study done at the start of this graduation project. It covers various topics relevant for vision-based autonomous navigation including; sensing, representation, obstacle-detection and path-planning. It's findings are the basis for the preliminary research in the subsequent part.

In part III the feasibility of several methods are tested in the context of implementing them on a computationally restricted Micro air Vehicle equipped with a stereo-camera system. The part concludes with a newly proposed research focus and research question. Further research into this direction is beyond the scope of this thesis.

Finally part IV provides the preliminary research that lead to the methods introduced in the scientific article and the in-depth background of the underlying research.

Part I

Scientific Article

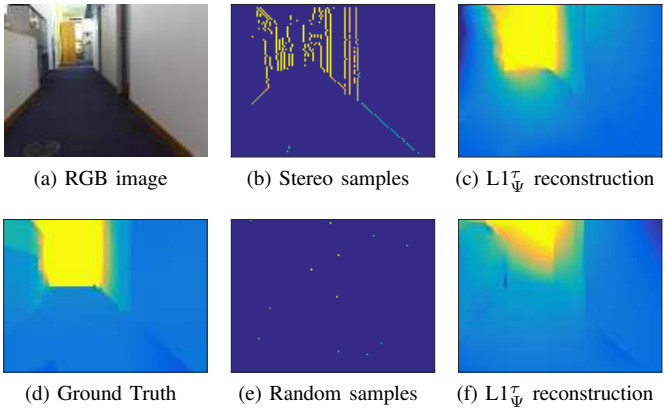
Enhanced Sparse Depth Reconstruction Using Edge and Temporal Information

Tobias A. Heil
Delft University of Technology
Email: tobiasaheil@gmail.com

Zhi Gao
Temasek Laboratories
National University of Singapore
Email: tslgz@nus.edu.sg

Guido C.H.E. de Croon
Delft University of Technology
Email: G.C.H.E.deCroon@tudelft.nl

***Abstract*—The reconstruction of dense depth maps is of great value to resource-constrained Mirco Air Vehicles (MAVs), in the pursuit of achieving autonomous flight with a high situational awareness. Most MAVs implement sensing methods which provide a sparse depth map, limiting their capabilities significantly. This article introduces two novel methods to enhance existing depth reconstruction algorithms in terms of geometric reconstruction, depth approximation and computational time. The first contribution is the introduction of a novel method that includes edge information from the image-domain into the depth-regularization problem. This to enhance the retrieval of the complete scene geometry. The second contribution is a novel scheme which includes temporal information in the reconstruction approach, allowing extremely sparse depth scenes to be reconstructed. By estimating the geometric transformation with optical flow, previous depth reconstructions can be used as initial solutions for the current depth-regularization problem. Empirical results show a consistent reduction reconstruction error, while at the same time reducing the computational time. Qualitative estimation shows significant improvement in the retrieval of scene geometry.**



(a) RGB image (b) Stereo samples (c) L1$^{\tau}_{\Psi}$ reconstruction

(d) Ground Truth (e) Random samples (f) L1$^{\tau}_{\Psi}$ reconstruction

Fig. 1. Demonstration of the enhanced sparse depth reconstruction. Using the stereo-edges from the image (1a) we obtain sparse depth samples (1b). Based on these samples we reconstruct the dense depth map (1c). Also with 0.1% randomly distributed sparse samples (1e) we are able to recover the scene to great detail (1f).

I. Introduction

Recent advances in research have brought the capabilities of Micro Air Vehicles (MAVs) to a level at which versatile autonomous flight is within reach [1]. These advances made the MAVs gain attention from governments and industry for various applications like exploration and reconnaissance, surveillance, weather observation, and commercial applications like aerial photography and delivery [2], [3]. With this increased interest also the requirements on the MAVs capabilities are raised to new levels. Most commercial MAVs are designed for outdoor flight where they use GPS way-point navigation in sparse obstacle environments [4], [5]. Future MAVs will be expected to show a great sense of situational awareness, such as the ability to identify restricted airspace and allow for safe flight in highly cluttered and GPS-denied environments. The latter has proven to be challenging and is topic of extensive research [6]. The most widely used sensors for depth estimation are active depth sensors like small LIDAR [7] or passive sensors like stereo [8] and monocular cameras [9], [10]. Because light weight MAVs are highly constrained in terms of weight, power and computational budgets, most of these vehicles are equipped with light sensors which only provide sparse information about the environment. This sparseness in sensing has large consequences in the ability of an MAV to autonomously navigate an unknown environment. Recently, in an effort to enable light weight MAVs to autonomously navigate using sparse sensing, Ma et al. [11] focused on reconstructing the dense depth map from the sparse measurements. They found that by assuming operation in highly structured environments, the geometric regularity and sparseness of edges can be leveraged for the reconstruction of the dense depth map. Building upon earlier work in the field of compressive sensing [12], they formulated theoretical conditions for which, using highly sparse input, a dense depth map can be reconstructed.

This article builds further on previous work of Ma et al. [11], [13] who presented a novel technique to reconstruct a dense depth map solely on depth measurements. To enhance the performance even more and extend the performance envelope in terms of sparseness, we explore the possibility of exploiting temporal and additional visual information (see Figure 1).

The goal of this article is to enhance and test a novel method to increase the situational awareness in terms of geometrical information recovery while using sparse depth sensing. State-of-the-art autonomous MAVs still require a significant number of data-points for obstacle avoidance or to recover geometric correct depth maps. We aim to reduce this considerably. Light weight range sensors [14] and miniaturised camera sensors

[15], [16], [17] that are used today would be, in combination with our technique, capable of operating in more environments opening the door to new applications and missions.

This article introduces three main contributions. The first contribution is the introduction of edge information from the image-domain to enhance the retrieval of the complete scene geometry with extremely sparse input data. The combination of visual and depth information will be investigated for various visual inputs. The second contribution is a novel scheme which includes temporal information in depth reconstruction for a sequence of image frames. The scheme aims at stimulating outlier resistance, better geometry recovery with extremely sparse input data and improving the computational time significantly. The last contribution is an extensive experimental evaluation of the performance of both approaches, using real world data.

The article is structured as follows, Section II provides an overview of related research in the field of resource-constrained MAV navigation, visual depth estimation, sparse sensing and compressive sensing. Section III describes the framework developed by Ma et al. [11], [13] which will be used throughout this article. In Section IV our approach is outlined, where in Section IV-A the visual- information and in Section IV-B the temporal-information enhanced approach are outlined. Section V and VI describe the experimental set-up and results, followed by the discussion and conclusions in Sections VII and VIII.

II. Related Research

This article, which combines visual information with sparse depth reconstruction, is related to several research topics. The closely related literature is briefly described in this section.

A. Resource-Constrained MAV Navigation

Due to the minimization of sensing electronics the possibilities for MAVs have been expanding rapidly, stimulating new research to be conducted in the field. One innovative approach uses bio-inspired robotics, where the design of the platform is based on nature's creations. Ma et al. [15] designed the *Robobee*: a 80-milligram flapping wing robot based on the morphology of flies. It is capable of stable hovering and basic controlled flight manoeuvres. Keennon et al. [16] developed the *Nano Hummingbird*: a 19-gram ornithopter with hovering and forward flight capabilities, together with a live video-downlink to a ground station. Similarly Dunkley et al. [18] demonstrated a 25-gram nano-quadrotor capable of stable hovering and providing a live video-downlink. The *DelFly Explorer* [17] was the first ornithopter able to perform onboard vision processing and achieving autonomous flight in unknown environments. McGuire et al. [19] further advanced the capabilities and achieved full autonomous flight on a 40-gram nano-quadrotor which performed depth and velocity estimation at 20 Hz.

B. Visual Information-based Depth Estimation

Learning approaches have been used for depth estimation [20], [21]. Bipin et al. [22] were able to generate a dense depth map for local navigation, using a supervised learning approach relating the direct video stream to depth. Likewise, Lamers et al. [23] used a supervised self-learning approach to relate object appearance with corresponding distances, and achieved successful implementation on a 19-gram MAV. More recently Facil et al. [24] recognised that monocular depth estimation approaches complement the weakness of stereo-vision based approaches, thus they proposed fusing monocular CNN-based depth estimation with stereo depth estimation. With it enhanced the sparse stereo depth map with visual information in the form of its CNN-based depth estimator.

C. Sparse to Dense Depth Map

Obtaining a dense depth map from sparse measurements greatly improves the situational awareness of autonomous MAVs and is therefore of great value. To gain such a dense map several approaches have been taken. Geiger et al. [25] introduced an effective method where an energy function that combines a likelihood estimator for feature correspondence with a linear disparity estimator, is minimized. The latter is modelled as a linear function which interpolates robust disparities using a triangulation computed on these robustly matched points. Alvarez et al. [1] use Parallel Tracking and Mapping for pose estimation [26], [27] and follow a multi-view stereo approach based on energy minimization. To gain the dense depth map they regularize the inverse depth map by minimizing the cost volume which contains a term that penalizes deviation from a spatially smooth inverse depth map. This approach has also been successfully applied to 3D laser point-clouds containing large gaps from removed objects [28], by using a Total Variation regularization with a new Kernel Conditional Density Estimation term which stimulates regularity when reconstructing surfaces. Piniés et al. [29] reconstruct a dense map from noisy range data through energy minimization with adaptive regularisers for which Bayesian optimisation is used to learn the needed parameters. A collective shortcoming of these approaches is the way they estimate depth in textureless regions, where most rely on interpolation. A more sophisticated and natural approach that includes additional visual information is expected to outperform conventional approaches in these textureless regions.

D. Compressive Sensing

The basis for our approach is found in compressive sensing literature [30], [31], [32]. Foucart et al. [12] proved with the *synthesis model* that a dataset can be completely recovered from a sparse subset, even for fewer samples than required by the Shannon-Nyquist sampling theorem, given the samples are sparse in the right domain. Recent work uses the *analysis model* [33], [34], which can extract a sparse subset, for a given dense dataset. Because required sparsity can be stimulated by adopting a l_1-minimization scheme and the reconstruction result can be improved with randomized samples, compressive sensing with the *analysis model* has proven successful in several applications. These include total variation minimization [35], 3D reconstruction [36] and regularization [37]. Using the

analysis model Ma et al. [11] set the theoretical conditions for which a dense depth map can be reconstructed based on a highly sparse depth map. By assuming operation in highly structured environments, the geometric regularity and sparseness of edges can be leveraged for the reconstruction of the dense depth map. Subsequently Ma et al. [13] implement the scheme in combination with a ad-hoc solver [38] reducing the computational time significantly. By only leveraging the sparseness of edges Ma et al. take the first step to include additional information in the depth reconstruction. In section IV we propose a reformulation and extension of the approach, to include extensive visual and temporal information.

III. Sparse Depth Map Reconstruction

This paper builds further on previous work by Ma et al. [11], [13], who took a novel approach to reconstruct a dense depth map solely on depth measurements and a regularity assumption. In this section the framework will be explained.

A. Definitions and Notations

Similar notations as in the paper of Ma et al. [11], [13] will be used. For matrices an upper case letter will be used, e.g. A, D. For scalars and vectors lower case letters e.g. z, y are used. Subsets are represented with calligraphic font, e.g. $\mathcal{M}$. The subset $\mathcal{M}$ of vector $z \in \mathbb{R}^n$ is denoted as $z_{\mathcal{M}}$. Indicating a subset $\mathcal{M}$ of a matrix D is done as $D_{\mathcal{M}}$, which represents the rows in subset $\mathcal{M}$ in matrix D. The following norms are widely used, (ℓ_∞-norm): $\|z\|_\infty = \max_{i=1,\dots,n}|z_i|$, ($\ell_0$-norm): $\|z\|_0 = |\mathrm{supp}\,(z)\,|$, and the ($\ell_1$-norm): $\|z\|_1 = \sum_{i=1,\dots,n}|z_i|$. It is important to recognise that the ℓ_0-norm corresponds to the number of none-zero elements in z. The depth reconstruction is based on the use of the *cosparsity model* Dz where the *analysis operator* D produces a sparse vector i,e. given $z \in \mathbb{R}^n$ and $D \in \mathbb{R}^{p\times n}$ we will have $\|Dz\|_0 \ll p$ [33], [34].

B. Problem Formulation

The theoretical basis of the reconstruction builds upon earlier work in the field of compressive sensing, where it was proven that a dataset z can be completely recovered from a sparse subset y given $y \in z$ [12]. The conventional model in the field is the *synthesis model*. It assumes that the dataset z is sparse given $z = D\alpha$ where the vector α is sparse in the domain of the matrix D. In more recent work a slight different representation in the form of the *cosparsity model* is proposed, where the vector z becomes sparse after multiplying it with a given matrix D, i.e. Dz, where z is the dataset and D is a given matrix [33], [34]. Ma et al. [11] found out that the ℓ_0-norm of the 2^{nd} order difference of the depth map can be used as an objective function to enforce the regularity assumption in the environment. By relaxing and reformulating the problem to a ℓ_1-norm problem, it becomes convex and fits the *cosparsity model*, allowing for a full reconstruction. In the remainder of this section the framework [11] is explained in detail, after which key-points are highlighted where we will depart from the original approach and introduce novel adaptations.

In order to reconstruct the dense depth map, it is assumed that sparse depth information is provided by sensing equipment. Lets define y as the measurement vector, η as measurement noise, $z^\diamond \in \mathbb{R}^n$ the depth map, and A the selection matrix. Then the measurements in y are found with $y = Az^\diamond + \eta$ with $A = \mathbf{I}_{\mathcal{M}}$, where $\mathbf{I}_{\mathcal{M}}$ is the identity matrix with only ones on the rows from subset $\mathcal{M}$. From this it can be clearly seen that $Az = z_{\mathcal{M}}$.

The assumption of operating in a structured environment means that the depth map shows a lot of regularity, i.e. the changes of the slope are mostly zero throughout the depth map. Such a change in slope for a measurement point is formulated as $\frac{\delta^2 z_i}{\delta {x_i}^2} = \frac{\delta z_i}{\delta x_i} - \frac{\delta z_{i-1}}{\delta x_{i-1}}$ which can be described as $\frac{z_{i+1}-z_i}{x_{i+1}-x_i} - \frac{z_i - z_{i-1}}{x_i - x_{i-1}}$. As this regularization is done in image-space we can assume $x_i - x_{i-1} = 1$, resulting in the second order derivative of z_i expressed as $z_{i+1} - 2z_i - z_{i-1}$.

It can be seen that the corner set $\mathcal{C}$ consists of indices for which $z_{i+1} - 2z_i - z_{i-1} \neq 0$. Keep in mind that with few corners in the environment $Dz^\diamond$ will be sparse. Defining matrix D as a *2^{nd}-order difference operator* (Equation 1) gives us the important equation $\|Dz^\diamond\|_0 = |\mathcal{C}|$.

$$D \doteq \begin{bmatrix} 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -2 & 1 \end{bmatrix} \in \mathbb{R}^{(n-2)\times n} \tag{1}$$

Now by leveraging the regularity in the environment, the full depth profile $z^\diamond$ can be reconstructed by solving the following optimization problem:

$$\min_z \|Dz\|_0 \quad \text{subject to} \quad Az = y \tag{2}$$

This noiseless optimization problem will force the depth profile to be consistent with the sparse measurements y, and minimises the number of corners, recall that $\|Dz^\diamond\|_0 = |\mathcal{C}|$. In order to allow for measurement noise and reformulate the problem into a linear programming problem the following relaxation is applied:

$$\min_z \|Dz\|_1 \quad \text{subject to} \quad \|Az - y\|_\infty \leq \varepsilon \tag{3}$$

Note that for this ℓ_1-minimization problem it is assumed that the noise is bounded $\|\eta\|_\infty \leq \varepsilon$. This is important for determining the tolerance for a specific application.

The optimization problems expressed in Equations 2 and 3 follow the *cosparsity models* [34], and allow us to reconstruct 2-dimensional depth maps, for the three dimensional case a we need to introduce a second *2^{nd}-order difference operator*.

For 3-dimensional depth reconstruction of $Z^\diamond \in \mathbb{R}^{r\times c}$ a operator, D_H, will be assigned to to the horizontal differences, and a difference operator, D_V, will be assigned to the vertical differences. The layout of the operators is identical as expressed in Equation 1, but with the appropriate sizes. Given depth map $Z^\diamond \in \mathbb{R}^{r\times c}$, we get $D_V \in \mathbb{R}^{(r-2)\times r}$ and $D_H \in \mathbb{R}^{(c-2)\times c}$. The corners are now encoded with

$D_V Z^\diamond \in \mathbb{R}^{(r-2)\times c}$ and $Z^\diamond D_H^{\mathrm{T}} \in \mathbb{R}^{r\times(c-2)}$, Thus the ℓ_1-minimization now becomes:

$$\min_{Z} \quad \|\mathrm{vec}(D_V Z)\|_1 + \|\mathrm{vec}(Z D_H^{\mathrm{T}})\|_1 \tag{4}$$

subject to $Z_{i,j} = y_{i,j}$ where $y_{i,j}$ is the sparse measurement map, Z the reconstructed depth map and $\mathrm{vec}(M)$ is the column wise vectorization of matrix M.

The next step is to reformulate the optimization in Equation 4 to correspond to the 2-dimensional case, the result is as follows:

$$\min_{z} \|\Delta z\|_1 \quad \text{subject to} \quad Az = y \tag{5}$$

where we define $n = r \times c$, and $z = \mathrm{vec}(Z) \in \mathbb{R}^n$ and the measurements are stored in $y \in \mathbb{R}^m$. The new matrix Δ is called the *Regularization matrix*, which is defined as follows:

$$\Delta = \begin{bmatrix} \mathbf{I}_c \otimes D_V \\ D_H \otimes \mathbf{I}_r \end{bmatrix} \tag{6}$$

where $\mathbf{I}_c$ is the identity matrix of size c, and $\otimes$ is the Kronecker product. The case where noise is present in the measurement, it is assumed the noise is bounded according to $\|\eta\|_\infty \leq \varepsilon$. This case is shown below:

$$\min_{z} \|\Delta z\|_1 \quad \text{subject to} \quad \|Az - y\|_\infty \leq \varepsilon \tag{7}$$

The problem as formulated in Equation 7 is referred to $\mathrm{L1}_\Delta^\varepsilon$ in Ma et al. [13]. In the remainder of this paper only noisy 3-D cases will be considered thus the epsilon (ϵ) and delta (Δ) subscripts will be dropped from the formulation. The new referencing is as shown in L1.

$$\min_{z} \|\Delta z\|_1 \quad \text{subject to} \quad \|Az - y\|_\infty \leq \varepsilon \tag{L1}$$

In order to solve the L1 optimization problem, Ma et al. [11] first used a off-the-shelf solver cvx/MOSEK [39] which proves to be impractical due to it's slow performance. In [13] they proposed the use NESTA, a variant of the more efficient first-order method developed by Becker et al. [38], which is based on earlier work by Nesterov for nonsmooth optimization [40], [41]. The method aims to solve

$$\min_{z} f_\mu(z) \quad \text{subject to} \quad z \in \mathcal{Q} \tag{8}$$

where $f_\mu(z)$ is the smooth approximation of the nonsmooth function $f(z)$ and $\mathcal{Q}$ the corresponding set for which $f(z)$ is convex. Ma et al. [13] uses the problem formulation as shown in Equation 7, thus resulting in $f(z) = \|\Delta z\|_1$ and $\mathcal{Q} = \{z : \|Az - y\|_\infty \leq \varepsilon\}$. To formulate it more concisely in terms of the ℓ_∞-ball

$$f(z) = \max_{u:\|u\|_\infty \leq 1} \langle u, Dz \rangle$$

Nesterov [40] proposes a method to approximate a non-smooth objective function by a function with Lipschitz-continuous gradient, applying this approach $f_\mu(z)$ will be defined as

$$f_\mu(z) = \max_{u:\|u\|_\infty \leq 1} \langle u, Dz \rangle - \mu \frac{\|u\|_2^2}{2} \tag{9}$$

where the gradient of $f_\mu(z)$ is Lipschitz-continues with constant L_μ. By minimizing $f_\mu(z)$ with an improved gradient method, Nesterov [40] achieved an optimal rate of convergence of $\mathcal{O}\left(\frac{1}{k^2}\right)$ The implementation of Nesterov's optimization technique as described above is formulated in Algorithm 2 from Ma et al. [13]. In the next section adaptations will be applied to the algorithm in order to include visual and temporal information.

IV. Enhanced Reconstruction Approach

This section explains the two main extensions proposed in this article. The first extension includes visual information, the original approach is solely based on sparse depth information leaving the information rich image-space out of scope. The second extension includes temporal information from previous frames, in an effort to stabilize the recovered geometry for extremely sparse input and significantly reducing the computational time, making the method suitable for real-time applications.

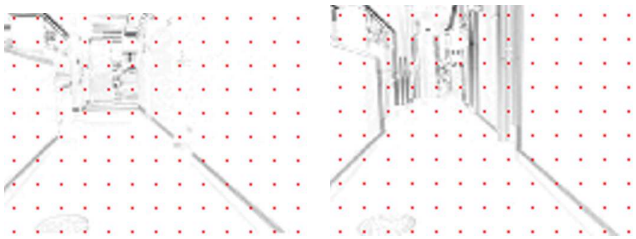
A. Edge-Information Enhanced Method

By including visual information in the depth reconstruction process will stimulate the recovery of the scene's geometry to a large extent. While Ma et al. [13] were only able to recover geometry based on information contained in the sparse depth values, we propose a method that extracts geometric information from image-space. The justification for this is that most biological systems use stereo or multi-vision to estimate distances from itself to the surroundings. These distances are estimated by neural networks that relate the visual input from multiple perspectives, it is therefore evident that (visual) distance estimation is only possible at distinguishable locations in image-space. The space of unobservable colour or intensity change is expected, and therefore assumed to be a interpolated region of it's observable boundary locations. As a consequence of this assumption the corner set $\mathcal{C}$ must a subset of observable set $\mathcal{O}$, i.e. $\mathcal{C} \in Z^\diamond_{\mathcal{O}}$, where $Z^\diamond_{\mathcal{O}}$ is the observable subset $\mathcal{O}$ of $Z^\diamond$.

To encourage the corner set to coincide with observable locations in image-space an adaptation is proposed to the problem formulation L1. The objective function $\|\Delta z\|_1$ will be adjusted by discounting the cost at observable locations with *discount function* ψ. The choice for *discount function* ψ is non-trivial due to largely unquantifiable relationship between observability and depth changes. For now ψ is chosen to be based on scaled image gradient. A successful and computationally attractive approach of identifying observable locations is applying a simplified Sobel filter onto the colour-intensity image [42] resulting in the gradient image. We propose the calculation of two directional gradient images $\delta_x \mathrm{I}$ and $\delta_y \mathrm{I}$ as shown below

$$\begin{aligned} \delta_x \mathrm{I} &= \mathrm{I} * \mathbf{G}_x, \quad \text{where} \quad \mathbf{G}_x = [1, 0, -1] \\ \delta_y \mathrm{I} &= \mathrm{I} * \mathbf{G}_y, \quad \text{where} \quad \mathbf{G}_y = [1, 0, -1]^T \end{aligned}$$

where I is the normalized colour-intensity image, $\delta_x \mathrm{I}$ and $\delta_y \mathrm{I}$ are the gradient images in horizontal and vertical direction.



(a) Vertical gradient map $\delta_y \mathrm{I}$ (b) Horizontal gradient map $\delta_x \mathrm{I}$

Fig. 2. Visualization of the Edge enhanced method using gridded sparse data. In Figure (2a) the vertical,- and in Figure (2b) the horizontal gradient map are shown. For both figures the sparse depth values are superpositioned in red. Its clear that a lot of information added about the scene's geometry.

$\mathbf{G}_x$ and $\mathbf{G}_y$ are the simplified Sobel filters in horizontal and vertical direction. Examples of the gradient maps are shown in Figure 2. To illustrate how they complement the sparse depth values gridded data points are added in red.

Combining the vectorized gradient images yields the image gradient vector $\mathrm{I}_{\mathcal{G}}$ which is defined as follows

$$\mathrm{I}_{\mathcal{G}} = \begin{bmatrix} \mathrm{vec}\left(\delta_x \mathrm{I}\right) \\ \mathrm{vec}\left(\delta_y \mathrm{I}\right) \end{bmatrix} \tag{10}$$

Subsequently the *discount function* ψ is defined as follows

$$\psi = \left(1 - \mathrm{I}_{\mathcal{G}}\right)^{\zeta} \tag{11}$$

where ζ is the scaling power which predominantly relates the likelihood of depth changes to observability in image-space, the value of ζ is empirically determined by minimizing the average euclidean error for a select dataset.

In order to incorporate the *discount function* ψ in the framework of Ma et al. [13] we will diagonalize the *discount function* according to

$$\Psi = \mathrm{diag}\left(\psi\right) \tag{12}$$

where Ψ is a diagonal matrix with the *discount function* ψ on its diagonal. Recall the problem formulation L1

$$\min_z \|\Delta z\|_1 \quad \text{subject to} \quad \|Az - y\|_\infty \leq \varepsilon \tag{L1}$$

We propose the following alteration to include the visual information in the depth reconstruction.

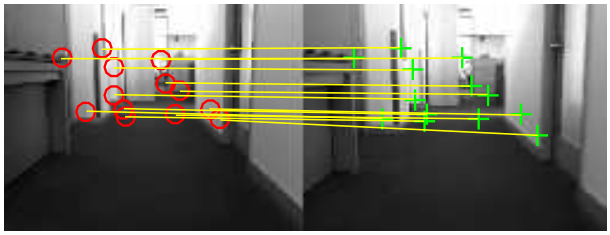
$$\min_z \|\Psi\left(\Delta z\right)\|_1 \quad \text{subject to} \quad \|Az - y\|_\infty \leq \varepsilon \tag{L1$_\Psi$}$$

This new formulation encourages the corner set $\mathcal{C}$ to coincide with gradients in image-space, which follows the intuitive thought that depth discontinuities in the depth map occur at edges in the image.

The corresponding Nesterov approximation function with Lipschitz-continuous gradient is as follows

$$f_\mu\left(z\right) = \max_{u:\|u\|_\infty \leq 1} \langle u, \Psi\left(Dz\right)\rangle - \mu \frac{\|u\|_2^2}{2} \tag{13}$$

(a) SURF features [43] frame 1 (b) SURF features [43] frame 2



(c) Feature Tracking from frame 1 to frame 2

(d) Geometric Transformation

Fig. 3. Visualization of the Temporal-Information Enhanced method. Figures (3a - 3b) show the feature detection step, followed by the feature tracking in Figure (3c). An illustration of the geometric transformation used to warp the previous depth map is show in Figure (3d)

B. Temporal-Information Enhanced Method

In this section the use of temporal information from previous frames is proposed to stabilize the recovered geometry based on extremely sparse input, and significantly reduce the computational time. This approach differs from the method proposed by Ma et al. [13] on three main points. Firstly, Ma et al. use multi-frame reconstruction where they project the measurements from a select number of previous frames onto the current sparse depth map. We propose to include the recovery of previous scene into the reconstruction of the current scene. Secondly, Ma et al. depend on sensor measurements to determine the relative position and pose for the projection of the sparse measurements onto the current sparse depth map. We propose the use of a geometric transformation of the previous dense depth reconstruction based on the current optical flow. Lastly where Ma et al. use a linearly interpolated depth map as a hot-start for the optimization problem L1, we use the expected dense map as initial solution for the optimization problem L1$_\Psi$ enabling large improvements in terms of computational time.

To determine the optical flow we propose Speeded-Up Robust Features (SURF) tracking over for instance more recent feature descriptors like FAST features [43], due to the relative robust and well performing SURF features in low

resolution images. When The SURF detector fails to match a sufficient number of points, we automatically switch to Harris feature tracking [44] which is less robust but is able to detect significantly more features. The approach is visualized in Figure 3 where Figures (3a - 3b) show the feature detection step and Figure (3c) the tracking step.

Tracking the features from the previous to current frame results into a sparse gradient field which is used to estimate the geometric transformation (see Figure 3d). Outliers are excluded using the M-estimator Sample Consensus (MSAC) algorithm, a variant of the Random Sample Consensum (RANSAC) algorithm [45], [46]. Lastly the initial solution to the $\mathrm{L1}_\Psi$ problem at time n, $z_n^{(0)}$ is determined by warping the previous dense reconstruction z_{n-1}^K using the estimated geometric transformation, where K is the last Nesterov iteration. To indicate the use of temporal information we shall use the upper case τ, e.g. for the problem formulation with edge-, and temporal-information we use $\mathrm{L1}_\Psi^\tau$.

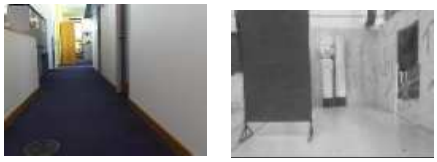
V. Experimental Set-up

In this section the experimental set-up is presented. The performance of the approach will be verified using real data, presenting us with empirical evidence. In order to compare the performance of the approach by Ma et al. [13] (in this paper referred to as L1), with the new problem formulation $\mathrm{L1}_\Psi$, both approaches will be tested on a 3.50GHz Intel i7-processor, using the ZED dataset from [13]. Furthermore to test the applicability of the method to visual navigation based MAVs, the performance is tested using a MAVLab[1] stereo-vision dataset. The accuracy of the dense depth map reconstruction is quantified using two different measures, firstly with the average euclidean distance error in centimetres, $\frac{1}{n}\|z^* - z^\diamond\|_1$, where z^* is the reconstruction and $z^\diamond$ the ground truth.

The second measure is based on the Scale-Invariant Error proposed by Eigen et al. [47] as defined

$$D\left(y, y^*\right) = \frac{1}{2n}\sum_{i=1}^{n}\left(\log y_i - \log y_i^* + \alpha\left(y, y^*\right)\right)^2$$

where $\alpha\left(y, y^*\right) = \frac{1}{n}\sum_{ij}\left(\log y_i^* - \log y_i\right)$. In order to cope with possible negative values, the magnitude is taken of the individual logarithms. The choice for the second error measure is based on the the desire to have a better quantifier for the recovery of the geometry of the scene.



(a) Example image (b) Example image

Fig. 4. Examples of ZED (4a) and MAVLab (4b) datasets

[1]TUDelft, Micro Air Vehicle Laboratory

The ZED dataset consists of 641 frames for which the depth ground truth and RGB images are downscaled to a resolution of 128×96, resulting in $12,288$ depth values per frame to be reconstructed (Figure (4a)).

The MAVLab dataset consists of 2875 stereo-pairs taken in 22 separate takes (Figure (4b)). The image-pairs are 128×96, the same as the downscaled ZED dataset. To estimate the sparse depth map, the block-matching scheme from [42] is used. Because there is no ground truth data for this set, only a qualitative evaluation is possible.

For the ZED dataset several sampling methods are proposed. The first method is to uniformly sample from the ground truth. This is done for varying sample sizes, ranging from 0.1% to 40%. Secondly the Canny edge detection method [48] is used on the gray-scale images where it finds edges by looking for local maxima of the image gradient. The Canny method is relatively robust against noise and is able to detect many edges. On these edge locations we will use the corresponding depth values as sparse input samples. Thirdly sampling in a regular-spaced grid is applied. This case is interesting as it corresponds to sampling from for instance laser-ranging sensors. The last sampling method is based on the block-matching approach described in [42], where at peaks in the image gradient block-matching between the image-pairs is applied. Because for the ZED dataset no stereo images are present, sampling is done at gradient locations that pass a certain threshold [42].

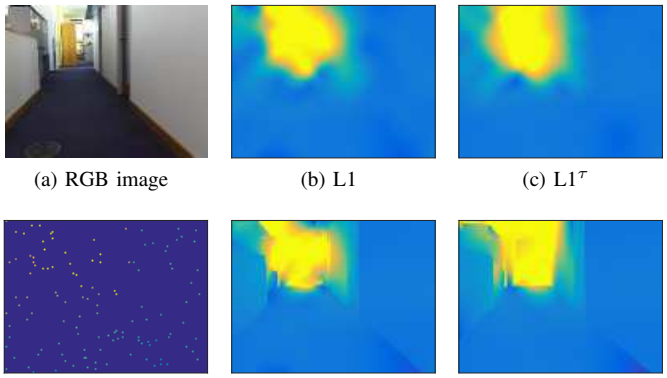
For the MAVLab dataset only the last sampling strategy is applied where the block-matching scheme is applied to estimate the corresponding depth values, in contrast to sampling them from the ground truth as done in the ZED dataset.

VI. Results

In this section the expected improvements are verified with experiments on real data from the ZED, and MAVLab dataset. The results from the experiments show significant improvements in terms of average error and scale-invariant error for all simulations, while improvements in terms of computational time are made for specific settings. Furthermore quantitative evidence is provided to prove the largely increased ability to recover scene geometry. The section is structured as follows, first in Section VI-A the performance using uniform random sampling is presented. Followed by regular-grid sampling in Section VI-B. Finally the performance using RGB-edge sampling and stereo sampling are presented in Section VI-C and VI-D respectively. Finally a qualitative performance evaluation is done in Section VI-E for the MAVLab dataset.

A. Uniform random sampling

In this section the performance of the methods are presented using the random sampling approach where samples are randomly drawn from a ground truth depth map. Firstly the influence of the addition of temporal-information is examined by comparing the L1 method (as mentioned in Section III-B, referred to by Ma et al. as $L1_\Delta^\epsilon$ [13]) to the temporal enhanced $L1^\tau$ method, and by comparing $\mathrm{L1}_\Psi$ to $L1_\Psi^\tau$. Secondly the influence of the addition of edge-information is examined by



(a) RGB image (b) L1 (c) $L1^{\tau}$

(d) random samples (e) $L1_{\Psi}$ (f) $L1^{\tau}_{\Psi}$

Fig. 5. Comparison of depth reconstructions using 1% random samples of different approaches. It is clear that the addition of edge information (5e - 5f) enhances the geometric recovery substantially.

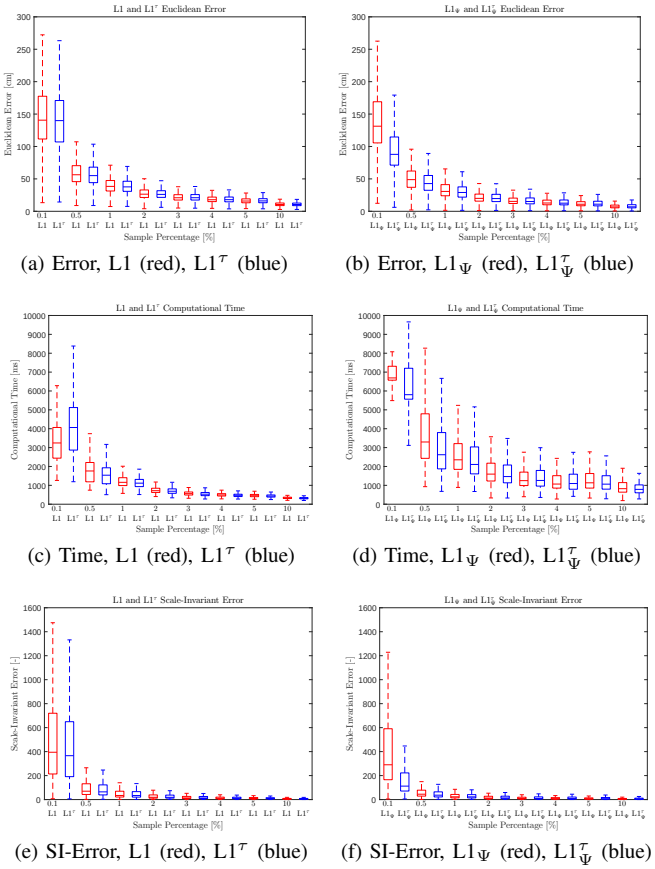
comparing L1 with $L1_{\Psi}$, and $L1^{\tau}$ with $L1^{\tau}_{\Psi}$. In order to represent the results, the bootstrap averages of the 641 images in the ZED dataset will be calculated with the Bias corrected and accelerated method [49]. The results are summarized in Figure 6. With the left column the performance of L1 (red) and $L1^{\tau}$ (blue) and in the right column $L1_{\Psi}$ (red) and $L1^{\tau}_{\Psi}$ (blue).

In Figure (6a) it is shown that introduction of temporal information does not significantly change the estimated average euclidean error for sampling percentages above 0.5%. L1 and $L1^{\tau}$ remain almost identical regardless of the sparseness of the samples. An exception is observed for the extreme sparse situation with only 0.1% samples. In this extreme sparse situation the combination of edge-information and temporal information, (Figure (6b)), shows that the introduction of temporal information reduces the estimated average euclidean error with 33% (see Table I).

The quantitative difference is clearly shown in Figure 5. The introduction of edge-information causes the reconstruction to recover the geometry of the corridor walls and floor to great detail. The influence of the temporal information is also visible in the improvement of Figure (5f) over Figure (5e).

When looking at the corresponding computational time in Figures (6c) and (6d) it is clear that for extremely sparse sample inputs the computational time increases with the introduction of temporal information, while for all other sparseness percentages the computational time improves by including temporal information. Comparing (6c) and (6d) shows that including the edge-information increases the computational time significantly, even up to 3 times, (see Table II). This is explained by the extra computational step in equation $L1_{\Psi}$ in the cost function, and the fact that the L1 and $L1^{\tau}$ approaches both do not recover any geometry while $L1_{\Psi}$ and $L1^{\tau}_{\Psi}$ do in great extent, (see Figure 7).

The effectiveness of the edge-information in terms of geometry recovery in extreme sparseness can also be seen in the Scale-invariant error in Figures (6e) and (6f). With a sampling percentage of 0.1% the scale-invariant error reduces with 14% from L1 to $L1_{\Psi}$ or even 63% from L1 to $L1^{\tau}_{\Psi}$, see Table III.



(a) Error, L1 (red), $L1^{\tau}$ (blue) (b) Error, $L1_{\Psi}$ (red), $L1^{\tau}_{\Psi}$ (blue)

(c) Time, L1 (red), $L1^{\tau}$ (blue) (d) Time, $L1_{\Psi}$ (red), $L1^{\tau}_{\Psi}$ (blue)

(e) SI-Error, L1 (red), $L1^{\tau}$ (blue) (f) SI-Error, $L1_{\Psi}$ (red), $L1^{\tau}_{\Psi}$ (blue)

Fig. 6. Uniform random sampling. A comparison between, L1, $L1_{\Psi}$ in red and $L1^{\tau}$, $L1^{\tau}_{\Psi}$ in blue, for mean Euclidean Error (6a-6b), mean Computational Time (6c-6d) and mean Scale-Invariant Error (6e-6f)

TABLE I
BOOTSTRAPPED MEAN EUCLIDEAN ERRORS USING UNIFORM RANDOM SAMPLING $[cm]$

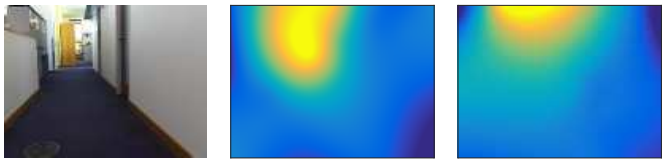
	Sampling Percentages [%]							
	0.1	**0.5**	**1**	**2**	**3**	**4**	**5**	**10**
Interp	145.74	57.73	40.05	27.62	22.04	18.77	16.56	10.87
$L1$	149.62	57.86	39.86	27.45	21.98	18.81	16.60	11.12
$L1_{\Psi}$	139.94	49.75	33.33	21.98	**17.22**	**14.61**	**12.68**	**8.20**
$L1^{\tau}$	137.66	56.92	39.21	27.23	21.95	18.86	16.54	11.03
$L1^{\tau}_{\Psi}$	**93.45**	**44.79**	**31.48**	**21.90**	17.29	14.92	13.31	8.58

TABLE II
BOOTSTRAPPED MEAN COMPUTATIONAL TIME USING UNIFORM RANDOM SAMPLING $[ms]$

	Sampling Percentages [%]							
	0.1	**0.5**	**1**	**2**	**3**	**4**	**5**	**10**
Interp	**11.56**	**13.04**	**14.43**	**17.07**	**19.39**	**21.20**	**24.09**	**24.11**
$L1$	**3488.43**	1839.35	1225.93	750.28	576.12	502.10	463.61	338.88
$L1_{\Psi}$	7735.67	3790.60	2741.47	1909.87	1523.79	1362.21	1402.17	952.37
$L1^{\tau}$	4269.24	**1609.61**	**1167.24**	**708.22**	**549.04**	**474.75**	**440.85**	**322.24**
$L1^{\tau}_{\Psi}$	6568.39	3033.14	2603.55	1801.64	1552.03	1330.00	1304.95	876.82

B. Regular grid sampling

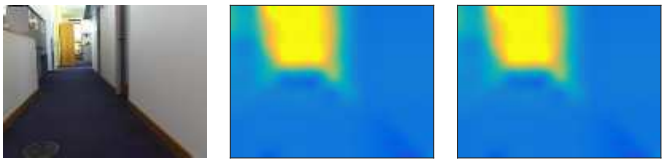
In this section the performance of the methods are presented using regular spaced samples. Similar to the previous section, firstly the influence of the addition of temporal-information is examined by comparing L1 to the temporal enhanced $L1^{\tau}$ method, and by comparing $L1_{\Psi}$ to $L1^{\tau}_{\Psi}$. Secondly the



(a) RGB image (b) L1 (c) $L1^{\tau}$

(d) random samples (e) $L1_{\Psi}$ (f) $L1^{\tau}_{\Psi}$

Fig. 7. Comparison of depth reconstructions using 0.1% random samples of different approaches. It is clear that the addition of temporal and edge information (7e - 7f), enhances the geometric recovery substantially.



(a) RGB image (b) L1 (c) $L1^{\tau}$

(d) regular-grid samples (e) $L1_{\Psi}$ (f) $L1^{\tau}_{\Psi}$

Fig. 8. Comparison of depth reconstructions using 1% regular-grid samples of different approaches. It is clear that the addition of edge information (8e - 8f) enhances the geometric recovery substantially.

TABLE III
BOOTSTRAPPED MEAN SCALE-INVARIANT ERROR USING UNIFORM RANDOM SAMPLING

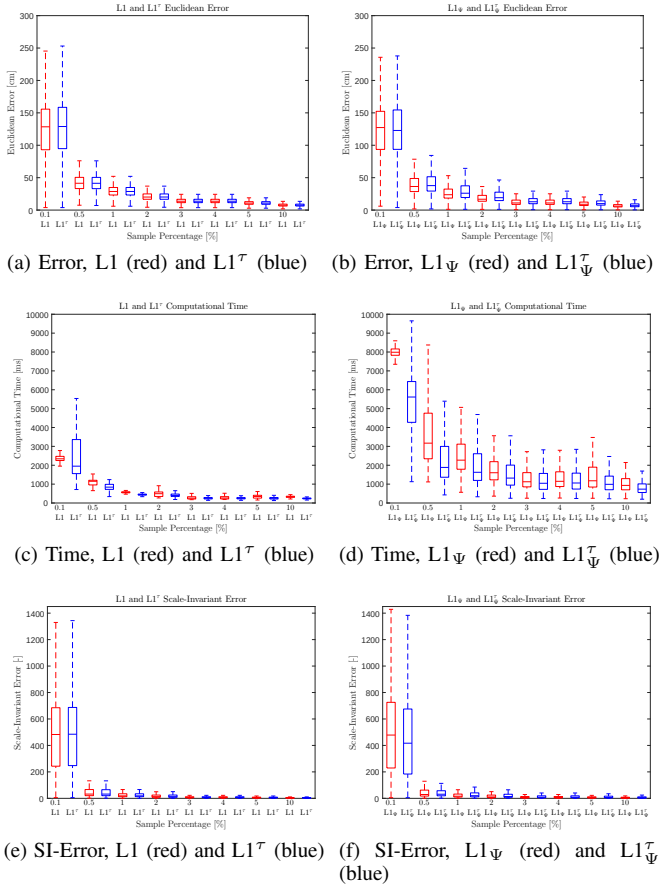
	Sampling Percentages [%]							
	0.1	**0.5**	**1**	**2**	**3**	**4**	**5**	**10**
$L1$	517.11	104.24	57.22	35.41	26.90	23.65	20.95	16.03
$L1_{\Psi}$	444.14	66.95	**39.59**	28.97	**23.24**	**20.95**	**19.22**	**15.79**
$L1^{\tau}$	455.61	100.68	57.18	35.23	27.85	23.52	20.96	15.85
$L1^{\tau}_{\Psi}$	**188.50**	**57.52**	40.06	**28.64**	24.07	22.99	21.02	16.82

influence of the addition of edge-information is examined by comparing L1 with $L1_{\Psi}$, and $L1^{\tau}$ with $L1^{\tau}_{\Psi}$. The results are summarized in Figure 9.

A quick look at Figure (9a) and Figure (9b) shows that the estimated average euclidean error is hardly affected by any of the approaches. This is confirmed when examining the bootstrapped averages in Table IV. The reconstruction error is similar to uniform sampling scenario in the previous section.

A quantitative assessment can be made by examining Figure 8. In contrast to the similar error values, the geometric recovery of the L1 and $L1^{\tau}$ is significantly higher for regularly spaced samples relative to uniform samples. Due to the evenly spread samples these methods do not disadvantage over the edge-information enhanced counterparts. A major difference that is present is the more crisp edges in the reconstruction for the edge-information enhanced methods $L1_{\Psi}$ and $L1^{\tau}_{\Psi}$.

When looking at the corresponding computational time in Figure (9c) and (9d) it becomes clear that in correspondence with the uniform sampling scenario, for all sparseness percentages the computational time improves by including temporal



(a) Error, L1 (red) and $L1^{\tau}$ (blue) (b) Error, $L1_{\Psi}$ (red) and $L1^{\tau}_{\Psi}$ (blue)

(c) Time, L1 (red) and $L1^{\tau}$ (blue) (d) Time, $L1_{\Psi}$ (red) and $L1^{\tau}_{\Psi}$ (blue)

(e) SI-Error, L1 (red) and $L1^{\tau}$ (blue) (f) SI-Error, $L1_{\Psi}$ (red) and $L1^{\tau}_{\Psi}$ (blue)

Fig. 9. Regular grid sampling. A comparison between, L1, $L1_{\Psi}$ in red and $L1^{\tau}$, $L1^{\tau}_{\Psi}$ in blue, for mean Euclidean Error (9a-9b), mean Computational Time (9c-9d) and mean Scale-Invariant Error (9e-9f)

TABLE IV
BOOTSTRAPPED MEAN EUCLIDEAN ERRORS USING REGULAR-GRID SAMPLING $[cm]$

	Sampling Percentages [%]							
	0.1	**0.5**	**1**	**2**	**3**	**4**	**5**	**10**
Interp	**113.99**	39.64	26.60	19.18	**12.80**	**12.80**	**10.08**	**7.10**
$L1$	122.27	43.03	29.69	21.58	14.43	14.43	11.33	7.90
$L1_{\Psi}$	119.44	**39.46**	**26.21**	**19.16**	12.84	12.84	10.41	7.21
$L1^{\tau}$	123.19	43.03	29.70	21.60	14.45	14.45	11.34	7.91
$L1^{\tau}_{\Psi}$	120.72	41.77	29.36	21.78	14.85	14.85	11.80	8.11

information. This correspondence also holds for the increasing effect on the computational time when including edge-information, due to the extra calculations made in equation $L1_{\Psi}$. This can be confirmed by comparing (9c) to (9d). The computational times are shown in Table V.

The effectiveness of the edge-information in terms of geometry recovery can not be seen in the scale-invariant error in

TABLE V
BOOTSTRAPPED MEAN COMPUTATIONAL TIME USING REGULAR-GRID SAMPLING $[ms]$

	Sampling Percentages [%]							
	0.1	**0.5**	**1**	**2**	**3**	**4**	**5**	**10**
Interp	**12.51**	**14.63**	**15.37**	**19.07**	**21.81**	**21.82**	**23.62**	**27.40**
$L1$	2794.63	1112.78	571.69	486.72	282.66	286.98	339.36	**322.55**
$L1_{\Psi}$	8869.99	3817.68	2723.77	1973.24	1476.62	1514.21	1644.04	1144.66
$L1^{\tau}$	**2347.40**	**840.69**	**443.85**	**405.95**	**254.90**	**255.17**	**255.10**	237.28
$L1^{\tau}_{\Psi}$	5759.93	2467.57	2178.69	1741.52	1335.98	1347.87	1179.49	858.02

TABLE VI
BOOTSTRAPPED MEAN SCALE-INVARIANT ERROR USING REGULAR-GRID SAMPLING

	Sampling Percentages [%]							
	0.1	**0.5**	**1**	**2**	**3**	**4**	**5**	**10**
$L1$	488.96	56.32	33.28	27.60	**17.63**	**17.63**	**15.74**	**13.41**
$L1_{\Psi}$	494.49	56.44	**33.02**	27.65	19.04	19.04	18.08	15.35
$L1^{\tau}$	496.59	56.28	33.40	**27.49**	17.65	17.65	15.76	13.42
$L1^{\tau}_{\Psi}$	**450.42**	**53.43**	36.95	30.00	22.64	22.64	20.19	16.65

Figures (9e) and (9f). The scale-invariant errors shown in Table VI are largely unaffected by the different approaches. This corresponds to the qualitative assessment based on Figure 8, where the outcome in Figures (8b, 8c, 8e, 8f) looks relatively similar in terms of geometry.

C. RGB-Edge sampling

In this section the performance of the methods are presented using rgb-edge samples. The samples are calculated using a search for local maxima of the image gradient, combining strong and weak edges.

Similar to the previous section, firstly the influence of the addition of temporal-information is examined by comparing L1 to the temporal enhanced $L1^{\tau}$ method, and by comparing $L1_{\Psi}$ to $L1^{\tau}_{\Psi}$. Secondly the influence of the addition of edge-information is examined by comparing L1 with $L1_{\Psi}$, and $L1^{\tau}$ with $L1^{\tau}_{\Psi}$. The results are summarized in Figure 11.

This sampling method differs significantly from the previous two in terms of the possibility of large areas in the depth map where no samples are taken. Figure 10 shows the results of the different approaches and in Figure (10d) the RGB-edge based samples are indicated.

From Figure 10 it is clear that the L1 (10b) and $L1^{\tau}$ (10c) create large errors in the reconstruction, as shown by the large dark blue areas. The edge-information enhanced method $L1_{\Psi}$ already shows a large improvement regarding these large dark blue regions, and the $L1^{\tau}_{\Psi}$ approach shows an even better result. To support this Figure 11a shows significantly higher average errors relative to the edge-enhanced approaches shown in Figure 11b.

In contrast to the two previous sampling methods, the RGB-edge method seems to have an increase in euclidean error when including the temporal information. This contradicts the qualitative assessment based on Figure 10. From Table VII we can deduct an euclidean error increase of 15% for including temporal information to L1, and a 34% increase for $L1^{\tau}_{\Psi}$ compared to $L1_{\Psi}$. The same phenomenon can be seen for the scale-invariant error in Figure 11e and Figure 11f, and the corresponding bootstrapped averages in Table VIII.

TABLE VII
BOOTSTRAPPED MEAN EUCLIDEAN ERRORS USING RGB-EDGE SAMPLING $[cm]$

	Euclidean Error [cm]	**Standard Error [cm]**
Interp	**61.73**	3.83
$L1$	126.25	5.28
$L1_{\Psi}$	67.64	3.88
$L1^{\tau}$	170.73	5.60
$L1^{\tau}_{\Psi}$	77.44	2.50

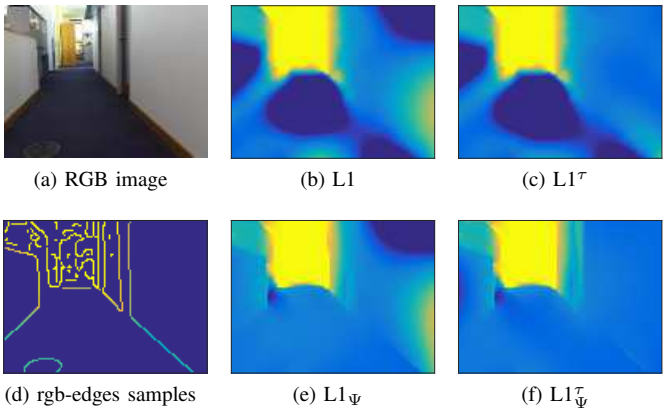
TABLE VIII
BOOTSTRAPPED MEAN SCALE-INVARIANT ERROR USING RGB-EDGE SAMPLING

	Scale-Invariant Error	**Standard Error**
$L1$	736.97	24.78
$L1_{\Psi}$	**346.29**	19.29
$L1^{\tau}$	1051.36	26.03
$L1^{\tau}_{\Psi}$	549.12	20.27

From Figure 11c it is shown that the computational time is significantly reduced when introducing the temporal information. From Table IX we can deduct a decrease of 38% for the L1 approach. For the $L1_{\Psi}$ approach we find a decrease of 10%. Remarkable is that the computational time is also decreased with 3% when introducing the edge-information to the L1 approach. It was expected that the computational time would increase with the introduction of extra calculations.

TABLE IX
BOOTSTRAPPED MEAN COMPUTATIONAL TIME USING RGB-EDGE SAMPLING $[ms]$

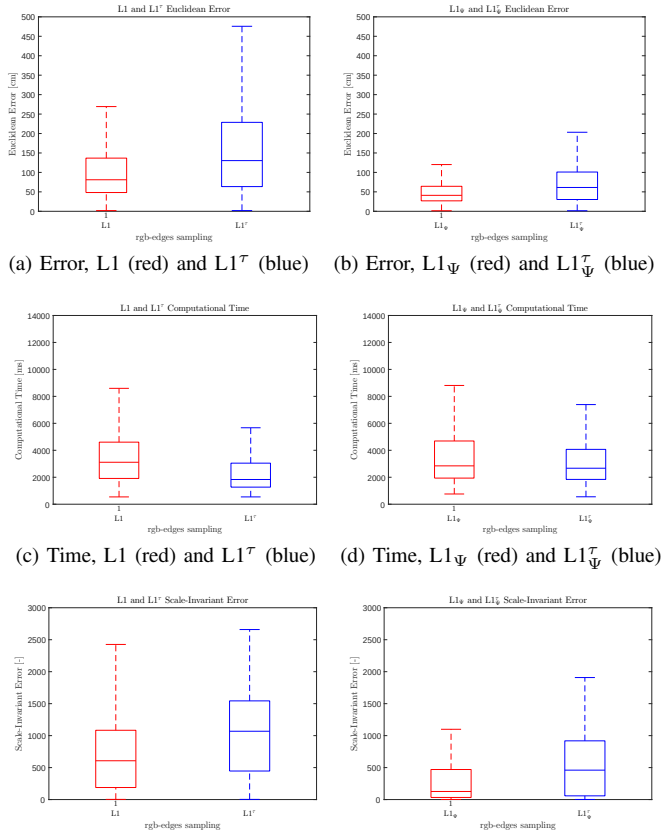
	Computational time [ms]	**Standard Error [ms]**
Interp	**26.99**	0.14
$L1$	4393.89	153.98
$L1_{\Psi}$	4240.29	145.19
$L1^{\tau}$	**2715.91**	100.57
$L1^{\tau}_{\Psi}$	3812.35	130.73



(a) RGB image (b) L1 (c) $L1^{\tau}$

(d) rgb-edges samples (e) $L1_{\Psi}$ (f) $L1^{\tau}_{\Psi}$

Fig. 10. Comparison of depth reconstructions using rgb-edges samples of different approaches. It is clear that the addition of edge information (10e - 10f) enhances the geometric recovery substantially.



(a) Error, L1 (red) and L1$^\tau$ (blue) (b) Error, L1$_\Psi$ (red) and L1$^\tau_\Psi$ (blue)

(c) Time, L1 (red) and L1$^\tau$ (blue) (d) Time, L1$_\Psi$ (red) and L1$^\tau_\Psi$ (blue)

(e) SI-Error, L1 (red) and L1$^\tau$ (blue) (f) SI-Error, L1$_\Psi$ (red) and L1$^\tau_\Psi$ (blue)

Fig. 11. RGB-edge sampling. A comparison between, L1, L1$_\Psi$ in red and L1$^\tau$, L1$^\tau_\Psi$ in blue, for mean Euclidean Error (11a-11b), mean Computational Time (11c-11d) and mean Scale-Invariant Error (11e-11f)

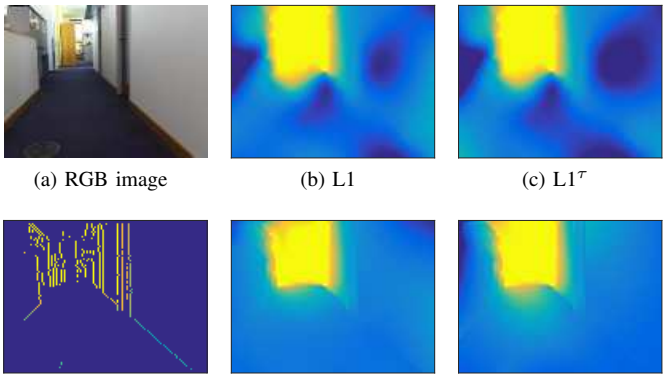
D. Stereo Sampling

In this section the performance of the methods are presented using stereo samples. Similar to the previous section, firstly the influence of the addition of temporal-information is examined by comparing L1 to the temporal enhanced $L1^\tau$ method, and by comparing L1$_\Psi$ to $L1^\tau_\Psi$. Secondly the influence of the addition of edge-information is examined by comparing L1 with L1$_\Psi$, and L1$^\tau$ with L1$^\tau_\Psi$. The results are summarized in Figure 13.

The stereo sampling method is chosen to test the validity of the edge,- and temporal-information enhanced approach for reconstructing a dense depth map for complex navigation tasks. The samples are taken at high horizontal gradient locations in the gray-scale image, in correspondence with [17].

An example of the samples is presented in Figure (12d). Comparing these samples with the rgb-edge samples from Figure (10d) it is clear that the stereo sampling approach results in fewer and less continuous lines of samples.

A qualitative assessment of the dense reconstructions is made based the reconstructions in Figure 12. Similar to the rgb-edge based reconstructions discussed in the previous section, the L1 and L1$^\tau$ approaches show a phenomenon of dark blue regions in areas where no samples are present,



(a) RGB image (b) L1 (c) L1$^\tau$

(d) stereo samples (e) L1$_\Psi$ (f) L1$^\tau_\Psi$

Fig. 12. Comparison of depth reconstructions using 1% stereo samples of different approaches. It is clear that the addition of edge information (12e - 12f) enhances the geometric recovery substantially.

making the approaches unsuitable for robot navigation. When edge-information is added the phenomenon disappears and a proper recovery of the scene geometry is achieved. Both L1$_\Psi$ and L1$^\tau_\Psi$ reconstruct the depth map successfully to a great extent. This improvement is confirmed by Figure 13, where in the top row the estimated average error in Figure (13a) is lower than Figure (13b) where edge-information is included. The estimated average errors are given in Table X. These values give a second validation, indicating a reduced error when including edge-information.

TABLE X
BOOTSTRAPPED MEAN EUCLIDEAN ERRORS USING STEREO SAMPLING $[cm]$

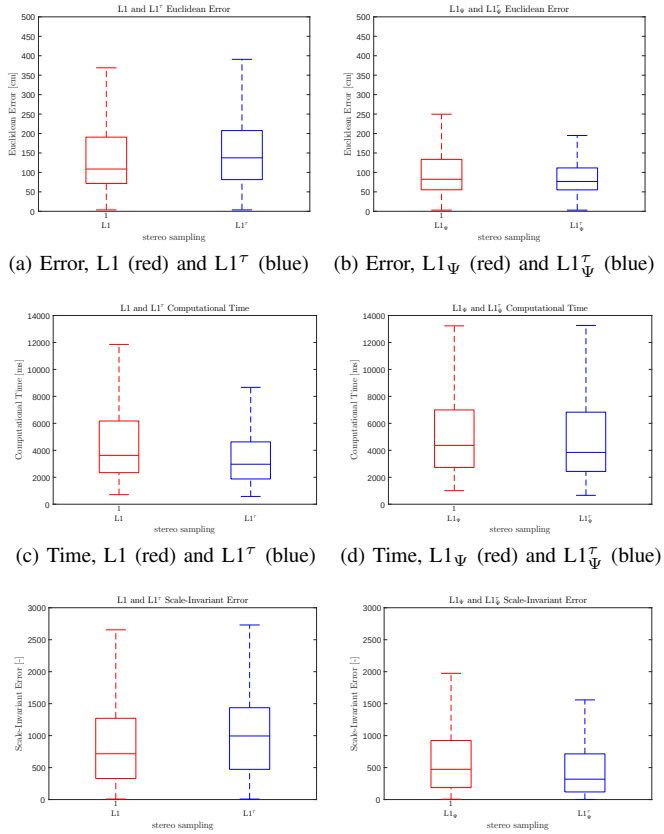
	Euclidean Error [cm]	Standard Error [cm]
Interp	129.03	7.18
$L1$	153.43	5.95
$L1_\Psi$	132.96	7.36
$L1^\tau$	161.80	5.27
$L1^\tau_\Psi$	**95.97**	4.64

Another observation from Table X is the significant reduction in average error when including temporal information to the edge-information enhanced approach L1$_\Psi$, while including temporal information to the standard L1 approach caused by a small increase in error.

The same is observed for the scale-invariant error; the introduction of edge-information reduces the error for both the L1 and L1$^\tau$, while the introduction of temporal information only causes a reduction of the error for the L1$_\Psi$ approach. The addition of temporal information to L1 causes an increase in error.

TABLE XI
BOOTSTRAPPED MEAN SCALE-INVARIANT ERROR USING STEREO SAMPLING

	Scale-Invariant Error	Standard Error
$L1$	852.05	25.10
$L1_\Psi$	650.79	23.79
$L1^\tau$	998.97	23.63
$L1^\tau_\Psi$	**461.63**	16.85



(a) Error, L1 (red) and L1$^\tau$ (blue) (b) Error, L1$_\Psi$ (red) and L1$^\tau_\Psi$ (blue)

(c) Time, L1 (red) and L1$^\tau$ (blue) (d) Time, L1$_\Psi$ (red) and L1$^\tau_\Psi$ (blue)

(e) SI-Error, L1 (red) and L1$^\tau$ (blue) (f) SI-Error, L1$_\Psi$ (red) and L1$^\tau_\Psi$ (blue)

Fig. 13. Stereo sampling. A comparison between, L1, L1$_\Psi$ in red and L1$^\tau$, L1$^\tau_\Psi$ in blue, for mean Euclidean Error (13a-13b), mean Computational Time (13c-13d) and mean Scale-Invariant Error (13e-13f)

Figure (13c) and Figure (13d) show the computational time. From (13c) it is clear that L1$^\tau$ outperforms the standard L1 approach, and that both L1$_\Psi$ and L1$^\tau_\Psi$ have a slightly higher computational time relative to L1. From Table XII we can deduct that L1$^\tau$ is 22% faster than L1, and that L1$^\tau_\Psi$ is 4% faster than L1$_\Psi$.

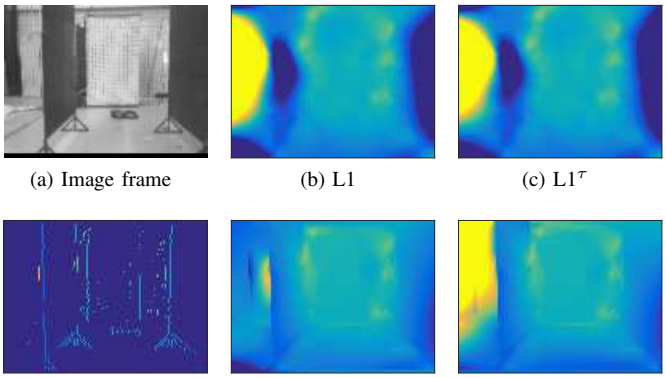
TABLE XII
BOOTSTRAPPED MEAN COMPUTATIONAL TIME USING STEREO SAMPLING $[ms]$

	Computational time [ms]	Standard Error [ms]
Interp	**28.42**	0.23
$L1$	5027.37	158.32
$L1_\Psi$	5938.69	181.78
$L1^\tau$	**3940.60**	124.02
$L1^\tau_\Psi$	5707.52	192.76

E. MAVLab dataset

In this section a qualitative evaluation is done of the performance of the approaches on the MAVLab dataset. A first remark is the image quality difference to the ZED dataset. Where the ZED dataset consists of downscaled high-resolution RGB images, the MAVLab dataset consists of stereo-pair images obtained with a extremely light and small stereo-camera system resulting. Because for the MAVLab dataset no ground truth depth maps are available, no quantitative assessment is made.

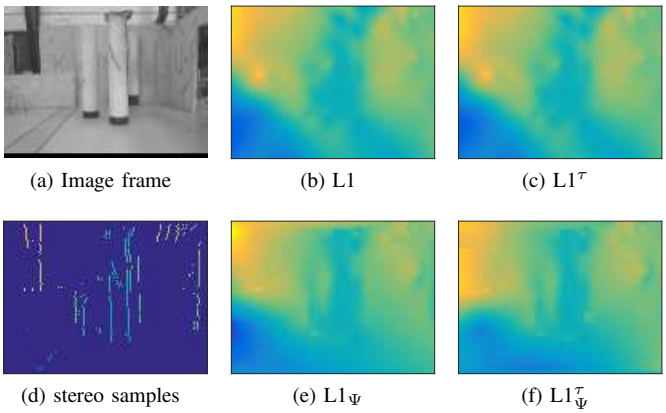
In Figure 14 the first result of the MAVLab dataset is shown. The frame has multiple prior fames to allow the temporal approach to take effect. It is clear that both L1 and L1$^\tau$ are unable to reconstruct a reliable depth map nor recover



(a) Image frame (b) L1 (c) L1$^\tau$

(d) stereo samples (e) L1$_\Psi$ (f) L1$^\tau_\Psi$

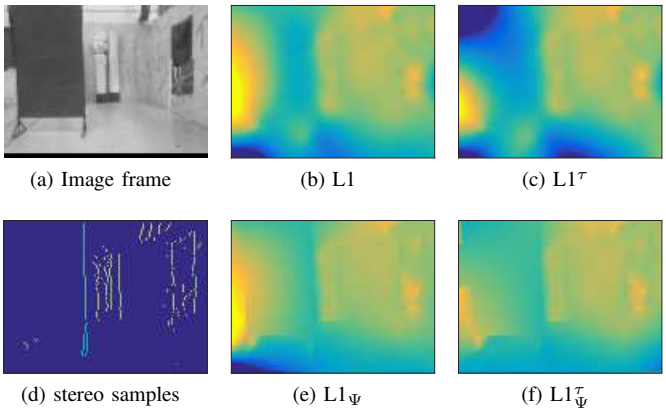
Fig. 14. Comparison of depth reconstructions using stereo samples of different approaches. It is clear that the addition of edge information (14e - 14f) enhances the geometric recovery substantially.



(a) Image frame (b) L1 (c) L1$^\tau$

(d) stereo samples (e) L1$_\Psi$ (f) L1$^\tau_\Psi$

Fig. 15. Comparison of depth reconstructions using stereo samples of different approaches. The addition of edge information (15e - 15f) enhances the geometric recovery marginally.



(a) Image frame (b) L1 (c) L1$^\tau$

(d) stereo samples (e) L1$_\Psi$ (f) L1$^\tau_\Psi$

Fig. 16. Comparison of depth reconstructions using stereo samples of different approaches. It is clear that the addition of edge information (16e - 16f) enhances the geometric recovery substantially.

the scene's geometry properly. The large dark blue regions indicate a strong sensitivity to outliers which falsely propagate into larger areas. Both edge-information enhanced methods $L1_\Psi$ and $L1_\Psi^\tau$ do not show this sensitivity and are able to reasonably reconstruct the scene's depth map, noting a large difference on the left side of the images. $L1_\Psi^\tau$ seems to be better equipped to reconstruct the distant region in the left side, relative to $L1_\Psi$ which falsely estimates this region to have a closer proximity.

In Figure 15 two objects are positioned at a relative large distance from the camera. In contrast to Figure 14, this frame seems to have no large outliers in the sparse sample set causing all four reconstructions to perform relatively similar in regard to depth approximation. In terms of geometric recovery the $L1_\Psi$ and $L1_\Psi^\tau$ produce the slightly more crisp maps, where in Figure (15f) the second pole is distinguishable, making $L1_\Psi^\tau$ the better performing approach.

In Figure 16 there is one large textureless screen positioned on the left side in the scene. All four approaches are able to reconstruct the screen reasonably but again both L1 and $L1^\tau$ seem to be largely influenced by outliers in the sample set. $L1_\Psi$ in this scene also suffers from a large outlier in the bottom left corner, indicating that including edge-information does provide sufficient robustness against outliers. In terms of recovery of the scene's geometry, both $L1_\Psi$ and $L1_\Psi^\tau$ outperform the other two methods, recovering a depth map which clearly distinguishes the screen, walls and floor from each other. The combination of edge,- and temporal-information proves to be the most successful, showing more robustness against outliers and recovering the scene's geometry to a larger extend, see Figure (16f).

VII. Discussion

In this paper two novel methods were introduced to enhance an existing depth reconstruction approach [13] in terms of geometric reconstruction, depth approximation and computational time. In the previous section the approaches were tested on the ZED dataset and subject to a extensive comparison between the approaches. This section will discuss the results and the implications they might have. First the uniform and regular-grid sampling approaches are discussed followed by the machine-vision methods; RGB-edge,- and stereo-sampling.

In Section VI-A we found that $L1_\Psi$ and $L1_\Psi^\tau$ have consistent lower estimated euclidean errors relative to L1 and $L1^\tau$. Given the randomness in the sampling method, some small regions of the scene might not contain a sample causing both L1 and $L1^\tau$ to treat those regions as regular surfaces between all surrounding samples. A consequence will be that discontinuities in the depth map (which are assumed to coincide with edges in the RGB image) will not be recovered. For $L1_\Psi$ and $L1_\Psi^\tau$, which both include edge-information in the reconstruction process, the depth discontinuities in these regions are to a certain extent recovered, causing a better resemblance to the ground truth. The scale-invariant error, which is more sensitive to geometric similarity, confirms that the addition of edge-information has great effect on the ability to recover the scene's geometry. When using regular-gridded samples (see Section VI-B) there are no regions without any depth-information. Therefore the addition of edge-information has very little effect on the average euclidean error. All reconstruction approaches shown in Figure 8 are able to reasonable recover the scene's geometry, with the only difference the more crisp result of $L1_\Psi$ and $L1_\Psi^\tau$ relative to L1 and $L1^\tau$. The addition of temporal-information only reduces the average euclidean,- and scale-invariant-error for extreme sparse input samples. This can be explained as the sparseness of some regions is virtually reduced by including depth-information from the previous frame in the form of a hot-start of the optimization problem with the previous warped depth-reconstruction.

In terms of computational time it is clear from the results in Section VI-A and Section VI-B that the addition of edge-information causes a large increase. This can partially be explained by the addition of the *discount function* ψ term in equation $L1_\Psi$, see Section IV. Another possible aspect that can contribute to the computational time is the more complex behaviour of the optimization problem given the *discount function* ψ is not regularly spread over the depth map. When temporal-information is included the computational time reduces significantly regardless of the sampling method. The additional optical flow calculations which are used to warp the previous depth reconstruction require additional computational time but are insignificant relative to the advantage of using a high-quality hot-start for the optimization problem.

Most MAV do not carry range sensors due to weight limitations, and therefore rely on optical depth sensing. The RGB-edge,- and stereo-sampling methods are conventional approaches for visual depth estimation for robots and aerial vehicles.

In Section VI-C the performance of the RGB-edge-sampling is presented. In Figure 10 the depth reconstructions of the four approaches show that L1 and $L1^\tau$ are incapable of providing usable results; the presence of large dark blue regions indicate an extreme close proximity to the camera, and will therefore cause navigation algorithms to take false evasion manoeuvres. The results in Table VII confirm the bad performance of L1 and $L1^\tau$ as the average euclidean error is almost double relative to $L1_{Psi}$ and $L1_{Psi}^\tau$. In terms of geometry recovery both edge-information approaches outperform their counterparts. This is shown in Figure 10 and confirmed by the scale-invariant errors in Table VIII. In contrast to the previous two sampling methods, the addition of temporal-information increases the average euclidean,- and scale-invariant-error. The estimated average euclidean errors in Table VII suggest a worsening effect of the introduction of temporal information. The reconstructions shown in Figure 10 show that for both with and without edge-information the reconstruction seems better in terms of dark blue regions and geometric recovery.

Similarly to the random,- and gridded-sampling methods,

the introduction of temporal-information reduces the computational time significantly. Even when the edge-information is included the computational time is reduced, suggesting that the computational benefit of the hot-start compensates the additional calculations and more complex optimisation domain due to the added edge-information.

To explore the possibility of implementing the approach to an actual platform like the TUDelft Delfly [17], the performance using stereo-samples is crucial. Figure 12 in Section VI-D shows high quality reconstructions for both $L1_{\Psi}$ and $L1^{\tau}_{\Psi}$ in contrast to L1 and $L1^{\tau}$, proving that the edge-information approach is highly effective in terms of depth reconstruction and geometry recovery. Both the estimated average euclidean error and scale-invariant error confirm this finding. Adding temporal-information only has a positive effect in combination with the edge-information approach, solely introducing temporal information causes a slight increase in average euclidean,- and scale-invariant-error. This suggests that using the naive linear interpolation approximation a better reconstruction in terms of euclidean error is possible, than when using the warped previous reconstruction.

The computational time however, reduces significantly when using temporal information relative to the standard approach which uses the naive interpolated approximation. Similar to the case with RGB-samples, the computational time is reduced even when the edge-information is included, suggesting that the computational benefit of the hot-start compensates the additional calculations and more complex optimisation domain due to the added edge-information.

The results of the different approaches using the MAVLab data is shown in Section VI-E. It is clear that the original approach of Ma et al. [13] is not able to produce reliable and acurate results that can be used for MAV navigation, see Figure (14b, 15b and 16b). The introduction of temporal-information improves the robustness slightly, but proves to be of most value in terms of reducing the computational time, as found in the results of the ZED dataset. Including edge-information proves to improve the depth map reconstruction considerably. The results of $L1^{\tau}_{\Psi}$ seem so qualitatively sound, showing geometric coincidence between the depth map and the image, with further development implementation on a real-world platform seems feasible.

VIII. Conclusion and Recommendations

In this paper we introduced two novel approaches to enhance an existing depth reconstruction algorithm in terms of geometric reconstruction, depth approximation and computational time. We have briefly summarized the basis approach and subsequently described a new and lean approach to include edge information from the image-domain into the depth-regularization problem in an effort to enhance the retrieval of the complete scene geometry. Results prove the effectiveness of the approach, broadening the operational envelope to extreme sparse inputs.

The introduction of a novel scheme which included temporal information was made possible by estimating the geometric transformation with optical flow, and subsequently warping the previous depth reconstructions to be used as initial solutions for the current depth-regularization problem. This contribution proved to reduce the computational time considerably in combination with adding robustness against disappearing samples due to vehicle movement.

Experimental results show that the introduced $L1^{\tau}_{\Psi}$ method, which combines both edge- and temporal-information is capable of reconstructing a dense depth map with a high degree of geometric recovery, based on highly sparse and noisy stereo-samples. Because the method works well with stereo-samples from the ZED and MAVLab datasets, the method in combination with the MAVLab stereo-board seems feasible. The stereo-board has already been used with edge detection algorithms but has not been used to construct a high quality dense depth map [42], [50]. With a faster C++ implementation of the method and when calculations are performed on a ground station, MAV navigation can be feasible based on the results.

Future work should focus on a few points. Several alterations can reduce the computational time significantly, namely a C++ implementation of the method will enable real-time testing on a UAV platform taking a step towards achieving autonomous flight with high situational awareness. Additionally improvements can be made with the development of a custom-made solver, reducing the number of cost-function evaluations. Tuning a gradient threshold to increase the sparseness of the *discount function*, the computational time can be decreased substantially. Furthermore research into different forms of the *discount function* ψ is expected to present improvements. The choice of ψ greatly influences the reconstruction capabilities for specific stereo-sampling algorithms. For these algorithms the stereo-edge locations are likely to coincidence with the by ψ discounted locations in the depth map, the influence of the only depth input is therefore reduced. A detailed study into the precise influence of the *discount function* could present a function that outperforms the one used in this article. Lastly the development of a hybrid approach which combines the warped previous depth-map with a naive linear interpolated depth-map of the current frame could increase the robustness of the temporal-approach and the combined approach $L1^{\tau}_{\Psi}$

References

[1] H. Alvarez, L. M. Paz, J. Sturm, and D. Cremers, "Collision avoidance for quadrotors with a monocular camera," in *Experimental Robotics*. Springer, 2016, pp. 195–209.

[2] R. Brockers, Y. Kuwata, S. Weiss, and L. Matthies, "Micro air vehicle autonomous obstacle avoidance from stereo-vision," in *SPIE Defense+ Security*. International Society for Optics and Photonics, 2014, pp. 90 840O–90 840O.

[3] A. Bachrach, R. He, and N. Roy, "Autonomous flight in unknown indoor environments," *International Journal of Micro Air Vehicles*, vol. 1, no. 4, pp. 217–228, 2009.

[4] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar, "Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft mav," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4974–4981.

[5] A. J. Barry and R. Tedrake, "Pushbroom stereo for high-speed navigation in cluttered environments," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 3046–3052.

[6] D. Dey, K. S. Shankar, S. Zeng, R. Mehta, M. T. Agcayazi, C. Eriksen, S. Daftry, M. Hebert, and J. A. Bagnell, "Vision and learning for deliberative monocular cluttered flight," in *Field and Service Robotics*. Springer, 2016, pp. 391–409.

[7] S. Shen, N. Michael, and V. Kumar, "3d indoor exploration with a computationally constrained mav," in *Robotics: Science and Systems*, 2011.

[8] F. Fraundorfer, L. Heng, D. Honegger, G. H. Lee, L. Meier, P. Tanskanen, and M. Pollefeys, "Vision-based autonomous mapping and exploration using a quadrotor mav," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4557–4564.

[9] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1449–1456.

[10] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 15–22.

[11] F. Ma, L. Carlone, U. Ayaz, and S. Karaman, "Sparse sensing for resource-constrained depth reconstruction," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 96–103.

[12] S. Foucart and H. Rauhut, *A mathematical introduction to compressive sensing*. Birkhäuser Basel, 2013, vol. 1, no. 3.

[13] F. Ma, L. Carlone, U. Ayaz, and S. Karaman, "Sparse depth sensing for resource-constrained robots," *arXiv preprint arXiv:1703.01398*, 2017.

[14] X. Chen, M. Zhao, L. Xiang, F. Sugai, H. Yaguchi, K. Okada, and M. Inaba, "Development of a low-cost ultra-tiny line laser range sensor," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 111–116.

[15] K. Y. Ma, P. Chirarattananon, S. B. Fuller, and R. J. Wood, "Controlled flight of a biologically inspired, insect-scale robot," *Science*, vol. 340, no. 6132, pp. 603–607, 2013.

[16] M. Keennon, K. Klingebiel, and H. Won, "Development of the nano hummingbird: A tailless flapping wing micro air vehicle," in *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2012, p. 588.

[17] C. De Wagter, S. Tijmons, B. D. Remes, and G. C. de Croon, "Autonomous flight of a 20-gram flapping wing mav with a 4-gram onboard stereo vision system," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4982–4987.

[18] O. Dunkley, J. Engel, J. Sturm, and D. Cremers, "Visual-inertial navigation for a camera-equipped 25g nano-quadrotor," in *IROS2014 aerial open source robotics workshop*, 2014, p. 2.

[19] K. McGuire, G. de Croon, C. De Wagter, K. Tuyls, and H. Kappen, "Efficient optical flow and stereo vision for velocity estimation and obstacle avoidance on an autonomous pocket drone," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1070–1076, 2017.

[20] K. van Hecke, G. de Croon, L. van der Maaten, D. Hennes, and D. Izzo, "Persistent self-supervised learning principle: from stereo to monocular vision for obstacle avoidance," *arXiv preprint arXiv:1603.08047*, 2016.

[21] S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, and M. Hebert, "Learning monocular reactive uav control in cluttered natural environments," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1765–1772.

[22] K. Bipin, V. Duggal, and K. M. Krishna, "Autonomous navigation of generic monocular quadcopter in natural environment," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1063–1070.

[23] K. Lamers, S. Tijmons, C. De Wagter, and G. de Croon, "Self-supervised monocular distance learning on a lightweight micro air vehicle," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 1779–1784.

[24] J. M. Fácil, A. Concha, L. Montesano, and J. Civera, "Deep single and direct multi-view depth fusion," *arXiv preprint arXiv:1611.07245*, 2016.

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

[26] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*. IEEE, 2007, pp. 225–234.

[27] J. Engel, J. Sturm, and D. Cremers, "Camera-based navigation of a low-cost quadrocopter," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 2815–2821.

[28] M. Tanner, P. Piniés, L. M. Paz, and P. Newman, "What lies behind: Recovering hidden shape in dense mapping," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 979–986.

[29] P. Piniés, L. M. Paz, and P. Newman, "Too much tv is bad: Dense reconstruction from sparse laser with non-convex regularisation," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 135–142.

[30] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE transactions on information theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[31] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[32] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[33] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," *Applied and Computational Harmonic Analysis*, vol. 34, no. 1, pp. 30–56, 2013.

[34] M. Kabanava and H. Rauhut, "Cosparsity in compressed sensing," in *Compressed Sensing and its Applications*. Springer, 2015, pp. 315–339.

[35] D. Needell and R. Ward, "Stable image reconstruction using total variation minimization," *SIAM Journal on Imaging Sciences*, vol. 6, no. 2, pp. 1035–1058, 2013.

[36] D. Reddy, A. C. Sankaranarayanan, V. Cevher, and R. Chellappa, "Compressed sensing for multi-view tracking and 3-d voxel reconstruction," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 221–224.

[37] S. Vaiter, G. Peyré, C. Dossal, and J. Fadili, "Robust sparse analysis regularization," *IEEE Transactions on information theory*, vol. 59, no. 4, pp. 2001–2016, 2013.

[38] S. Becker, J. Bobin, and E. J. Cands, "Nesta: A fast and accurate first-order method for sparse recovery," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011.

[39] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[40] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical programming*, vol. 103, no. 1, pp. 127–152, 2005.

[41] ——, "A method for unconstrained convex minimization problem with the rate of convergence o (1/k2)," in *Doklady an SSSR*, vol. 269, no. 3, 1983, pp. 543–547.

[42] S. Tijmons, G. de Croon, B. Remes, C. De Wagter, and M. Mulder, "Obstacle avoidance strategy using onboard stereo vision on a flapping wing mav," *arXiv preprint arXiv:1604.00833*, 2016.

[43] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," *Computer vision–ECCV 2006*, pp. 430–443, 2006.

[44] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.

[45] P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.

[46] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[47] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in neural information processing systems*, 2014, pp. 2366–2374.

[48] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.

[49] B. Efron, "Better bootstrap confidence intervals," *Journal of the American statistical Association*, vol. 82, no. 397, pp. 171–185, 1987.

[50] K. McGuire, G. de Croon, C. De Wagter, B. Remes, K. Tuyls, and H. Kappen, "Local histogram matching for efficient optical flow computation applied to velocity estimation on pocket drones," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3255–3260.

APPENDIX

TABLE XIII
BOOTSTRAPPED MEAN EUCLIDEAN ERRORS USING UNIFORM RANDOM SAMPLING $[cm]$

	Sampling Percentages [%]															
	0.1	**SE**	**0.5**	**SE**	**1**	**SE**	**2**	**SE**	**3**	**SE**	**4**	**SE**	**5**	**SE**	**10**	**SE**
Interp	145.66	3.06	57.75	0.93	40.05	0.64	27.63	0.46	22.04	0.38	18.77	0.33	16.56	0.29	10.87	0.20
$L1$	149.69	3.18	57.86	1.01	39.83	0.70	27.45	0.49	21.98	0.39	18.80	0.34	16.60	0.30	11.12	0.20
$L1_{\Psi}$	139.83	2.90	49.68	0.89	33.34	0.69	21.99	0.49	**17.21**	0.40	**14.61**	0.36	**12.67**	0.33	**8.20**	0.24
$L1^{\tau}$	137.55	2.33	56.92	1.00	39.23	0.69	27.22	0.48	21.96	0.39	18.86	0.34	16.54	0.29	11.04	0.20
$L1_{\Psi}^{\tau}$	**93.44**	1.76	**44.81**	0.85	**31.48**	0.68	**21.92**	0.51	17.31	0.42	14.95	0.38	13.29	0.40	8.58	0.26

TABLE XIV
BOOTSTRAPPED MEAN COMPUTATIONAL TIME USING UNIFORM RANDOM SAMPLING $[ms]$

	Sampling Percentages [%]															
	0.1	**SE**	**0.5**	**SE**	**1**	**SE**	**2**	**SE**	**3**	**SE**	**4**	**SE**	**5**	**SE**	**10**	**SE**
Interp	**11.56**	0.04	**13.04**	0.05	**14.43**	0.06	**17.07**	0.07	**19.39**	0.09	**21.20**	0.08	**24.08**	0.12	**24.11**	0.10
$L1$	**3493.07**	54.75	1839.97	29.95	1226.01	13.95	750.05	7.46	576.10	4.91	502.04	4.05	463.54	3.66	338.80	2.27
$L1_{\Psi}$	7745.08	108.45	3789.75	67.84	2739.04	54.28	1912.00	44.85	1522.89	36.74	1361.50	32.62	1401.38	33.95	951.73	18.24
$L1^{\tau}$	4266.91	70.08	**1611.17**	24.70	**1167.53**	13.52	**708.24**	7.47	**549.33**	5.29	**474.57**	3.92	**440.98**	3.52	**322.30**	2.08
$L1_{\Psi}^{\tau}$	6573.64	96.87	3033.99	60.67	2603.99	61.67	1801.56	44.27	1552.70	37.95	1328.84	31.38	1306.99	34.12	876.87	16.55

TABLE XV
BOOTSTRAPPED MEAN SCALE-INVARIANT ERROR USING UNIFORM RANDOM SAMPLING

	Sampling Percentages [%]															
	0.1	**SE**	**0.5**	**SE**	**1**	**SE**	**2**	**SE**	**3**	**SE**	**4**	**SE**	**5**	**SE**	**10**	**SE**
$L1$	517.75	16.40	104.19	3.98	57.13	2.40	35.35	1.82	26.87	1.58	23.61	1.56	20.95	1.52	15.96	1.46
$L1_{\Psi}$	443.12	16.19	66.92	2.71	**39.53**	1.86	28.97	1.84	**23.26**	1.66	**20.86**	1.64	**19.18**	1.60	**15.74**	1.53
$L1^{\tau}$	455.60	13.27	100.52	4.01	57.27	2.51	35.12	1.81	27.71	1.69	23.50	1.57	20.93	1.54	15.85	1.49
$L1_{\Psi}^{\tau}$	**188.92**	8.42	**57.48**	2.63	40.17	2.20	**28.69**	1.75	24.12	1.66	23.03	1.74	21.12	1.68	16.82	1.52

TABLE XVI
BOOTSTRAPPED MEAN EUCLIDEAN ERRORS USING REGULAR-GRID SAMPLING $[cm]$

	Sampling Percentages [%]															
	0.1	**SE**	**0.5**	**SE**	**1**	**SE**	**2**	**SE**	**3**	**SE**	**4**	**SE**	**5**	**SE**	**10**	**SE**
Interp	**113.96**	1.75	39.68	0.72	26.63	0.49	19.21	0.39	**12.81**	0.25	**12.81**	0.25	**10.09**	0.19	**7.11**	0.13
$L1$	122.22	1.97	43.01	0.80	29.67	0.53	21.57	0.42	14.42	0.27	14.42	0.27	11.33	0.21	7.90	0.15
$L1_{\Psi}$	119.46	1.87	**39.41**	0.79	**26.19**	0.54	**19.14**	0.43	12.82	0.32	12.82	0.32	10.41	0.27	7.20	0.19
$L1^{\tau}$	123.38	1.98	43.02	0.78	29.69	0.53	21.57	0.42	14.44	0.27	14.44	0.27	11.33	0.21	7.91	0.14
$L1_{\Psi}^{\tau}$	120.78	1.95	41.82	0.84	29.38	0.62	21.77	0.48	14.87	0.37	14.87	0.37	11.82	0.32	8.13	0.22

TABLE XVII
BOOTSTRAPPED MEAN COMPUTATIONAL TIME USING REGULAR-GRID SAMPLING $[ms]$

	Sampling Percentages [%]															
	0.1	**SE**	**0.5**	**SE**	**1**	**SE**	**2**	**SE**	**3**	**SE**	**4**	**SE**	**5**	**SE**	**10**	**SE**
Interp	**12.51**	0.04	**14.63**	0.04	**15.37**	0.05	**19.07**	0.06	**21.81**	0.07	**21.82**	0.07	**23.61**	0.09	**27.41**	0.09
$L1$	2794.27	39.46	1113.15	6.56	571.73	1.96	486.69	5.39	282.79	3.37	286.92	3.45	339.46	4.10	322.51	2.54
$L1_{\Psi}$	8865.62	117.57	3816.11	78.83	2726.02	59.35	1971.32	50.54	1476.91	46.12	1515.43	46.97	1641.72	55.14	1145.45	32.01
$L1^{\tau}$	**2346.65**	45.29	**840.85**	7.52	**443.88**	1.98	**406.16**	4.03	**254.90**	2.33	**255.20**	2.31	**255.11**	2.26	**237.24**	1.52
$L1_{\Psi}^{\tau}$	5759.70	94.67	2468.33	60.74	2179.35	58.04	1742.54	50.67	1335.34	37.77	1347.67	38.17	1181.51	31.39	858.10	20.72

TABLE XVIII
BOOTSTRAPPED MEAN SCALE-INVARIANT ERROR USING REGULAR-GRID SAMPLING

	Sampling Percentages [%]															
	0.1	**SE**	**0.5**	**SE**	**1**	**SE**	**2**	**SE**	**3**	**SE**	**4**	**SE**	**5**	**SE**	**10**	**SE**
$L1$	488.64	11.82	56.38	2.35	33.35	1.68	27.62	1.71	17.65	1.47	17.65	1.47	15.77	1.44	13.41	1.41
$L1_{\Psi}$	493.79	12.19	56.51	2.79	**33.06**	2.05	27.73	1.90	19.05	1.60	19.05	1.60	18.10	1.58	15.44	1.52
$L1^{\tau}$	497.04	12.00	56.13	2.38	33.26	1.68	**27.48**	1.73	**17.61**	1.50	**17.61**	1.50	**15.70**	1.45	**13.35**	1.43
$L1_{\Psi}^{\tau}$	**449.83**	11.99	**53.52**	2.58	36.96	1.96	30.05	1.74	22.61	1.60	22.61	1.60	20.18	1.54	16.60	1.47

Part II

Literature Study

Chapter 2

Literature Review

2-1 Introduction

This review will provide a concise introduction into the latest advances in autonomous obstacle avoidance and navigation methods. It briefly touches various recent findings with a special focus on optimal trajectory generation and obstacle representations. In an attempt to achieve autonomous flight in unknown cluttered environments several approaches have shown promising results. Various planners like Rapidly exploring Random Tree (RRT), Stable-Sparse-RRT (SST), Batch Informed Trees (BIT)*, RRT* and Fast Marching Tree (FMT) allow for rapid exploration of high dimensional configuration space in which system states and world obstacles are represented. Using advances in trajectory smoothing or analytical Optimal Boundary Value Problem (OBVP) points in high-dimensional configuration space can be connected. However to achieve full autonomous flight with on-line sensing, representation and control requires a computationally more efficient approach.

Efficient Image Space representations of expanded-inverse-disparity maps have been implemented on-line and can be combined with computationally efficient avoidance methods. Achieving further optimization of Image Space representation provides for an opportunity to tackle a major shortcoming of current obstacle avoidance approaches; autonomous flight in highly cluttered and spatially constrained environments. This review will expose the need for a novel method to efficiently identify feasible vehicle-size-based trajectory generation allowing for flight in spatially constrained environments.

In the following sections the focus will be applications for indoor where a, broadly used, GPS signal is not available and navigation has to be performed using alternative sensors and more advanced methods. As all sensing and processing has to be done on-board, the available payload capacity and power budget of the drone form major constraints on the options of various sensors and for the complexity of the algorithms that can be deployed.

Besides the limiting suitable types of sensors due to weight, size and power constraints, indoor flight impose heavy requirements on the performance of the sensors in terms of the information they provide about the environment in which the drone operates. As obstacles are expected to

be in close proximity to the drone and are highly cluttered at times, the situational awareness of the drone has to be high. In terms of sensing this imposes requirements on the horizontal and vertical field of view in order to perceive obstacles around the drone.

Vision-based sensors are highly suitable as they provide the necessary field of view in combination with dense information about the environmental. The research in optical sensors for consumer electronics have led to the development of high quality sensors of extremely low weight and power consumption. As optical sensors promise the best potential relative to other sensors, vision-based approaches to obstacle avoidance and navigation will be addressed in this review.

In order to provide a clear context for the vision-based obstacle avoidance approach, the scope of this review is slightly enlarged to include local trajectory planning and motion planning next to the in depth analysis of the latest achievements in visual obstacle detection and avoidance.

The problem of autonomous flight of MAVs can roughly be split in a global and a local local path planning. Where Sunberg et. al. focussed on global path planning in the form of collision avoidance policies in civic airspace (Sunberg, Kochenderfer, & Pavone, 2016), Nieuwenhuisen et. al. focussed on a layered navigation approach in a partially known environment combining global mission planning with local trajectory generation for obstacle avoidance (Nieuwenhuisen & Behnke, 2016). The problem of obstacle avoidance in cluttered unknown environments however does not allow for global path planning as most of the environment will be occluded, and no prior knowledge of the environment is assumed. The focus will therefore be on efficient obstacle detection and representation to be implemented on a light weight MAV.

In section 2-2 *sampling-based motion planning* algorithms are introduced. This type planners has successfully been implemented for robotics and MAVs and the latest are discussed in detail in Section 2-2-1. In the next Section (2-2-2) the latest work of Allen et. al. is presented where online kinodynamic motion planning is achieved by using a novel machine-learning-based framework in combination with a non-linear feedback controller, achieving successful real-time obstacle avoidance (R. Allen & Pavone, 2016).

In Section 2-3 recent work is presented in which a efficient configuration space expansion is introduced accelerating feasible trajectory checking and planning (Brockers, Kuwata, Weiss, & Matthies, 2014). Followed by Section 2-3-4 in which Brockers et. al. extend their previous work by introducing a novel egocylindrical image space representation allowing for direct obstacle free trajectory searches in image space (Brockers, Fragoso, & Matthies, 2016).

Lastly in chapter 3 a complement to the sparse disparity map is investigated in the form of two segmentation methods. A more direct method of obstacle avoidance is to use image space based detection and avoidance. Chapter 3 introduces segmentation as a method to extract information from an image and allowing for effective processing of image regions. Two influential approaches are described, first in Section 3-1 the Simple Linear Interactive Clustering (SLIC) approach is described (Achanta et al., 2010). Achanta et. al. achieve real-time high quality real-time segmentation using a novel new approach. Second in Section 3-2 the approach of Van den Bergh, Boix, Roig and van Gool is presented. They recently developed a more complex and flexible clustering approach called Superpixels Extracted via Energy-Driven Sampling (SEEDS), achieving faster segmentation than SLIC while retaining

the quality and even providing better edge detection (Van den Bergh, Boix, Roig, & Van Gool, 2015).

2-2 Sampling-based Motion Planning for MAV Applications

A of the challenges within the research field of the autonomous MAV navigation is real-time collision free motion planning. The modification of a collision free trajectory to a dynamically feasible one is not always easy, as most MAV and robotic systems are controlled by the time derivatives of their configuration (Choset, 2005). In order to take into account the constraints of the configuration derivatives in motion planning Donald et. al. introduced a method called kinodynamic motion planning, where he used the systems configuration directly to guarantee dynamically feasible trajectories(Donald, Xavier, Canny, & Reif, 1993).

2-2-1 Sampling-based Kinodynamic Planning

One class of methods to solve kinodynamic planning are *sampling-based methods* (Karaman & Frazzoli, 2011; Hsu, Kindel, Latombe, & Rock, 2002; Kavraki, Vestka, Latombe, & Overmars, 1996; LaValle & Kuffner, 2001). This class of methods is characterized by sampling a large amount of states and subsequently try to connect a select set of samples, to form a feasible trajectory. They have proven to be able to find feasible solutions in high-dimensional configuration space (LaValle & Kuffner, 2001; Hsu et al., 2002). The early developments used a probabilistic approach to sample the configuration for feasible trajectories, an early method was introduced by Kavraki et. al., the probabilistic roadmap method (Kavraki et al., 1996). Later developments were based on early work of LaValle (LaValle & Kuffner, 2001) who proposed RRTs to find ways to connect two different states in configuration space with each other. These methods are based on the probabilistic completeness of the methods to find a solution, i.e. the probability that the method finds a solution given the number of branches goes to infinity, converges to 1 (Kuffner & LaValle, 2000; Ladd & Kavraki, 2004).

A major leap in the development of sampling-based methods was done by Karaman and Frazzoli, they proved that incremental methods like RRT will surely not converge to the optimum-cost path (Karaman & Frazzoli, 2011). The same paper contributed with the introduction of Rapidly exploring Random Graph (RRG) which does converges to the optimum path. They introduce a tree version of RRG called RRT* which transfers the asymptotic optimal property of RRG to the tree structured methodology of RRT (Karaman & Frazzoli, 2011).

A contribution by Li is made with the development of the SST method which achieves asymptotic optimality for kinodynamic planning (Li, Littlefield, & Bekris, 2016), it based on an adaptation of sampling-based planner RRT (LaValle & Kuffner, 2001) called RRT-BestNear (Urmson & Simmons, 2003). Li accomplishes guaranteed better time performance with SST compared to RRT, where SST is able to converge to the optimal path over time faster.

Other adaptations of the RRT method have been developed by Gammell et. al., with their Informed RRT* and BIT (Gammell, Srinivasa, & Barfoot, 2014, 2015). These adaptations show a higher capability in finding feasible trajectories in narrow spaces in configuration space. The latest development showing a significant improvement in finding trajectories in narrow

spaces is developed by Choudhury et. al. called Regionally Accelerated Batch Informed Trees (RABIT) (Choudhury, Gammell, Barfoot, Srinivasa, & Scherer, 2016). One of the latest sampling methods which has already been implemented of MAV trajectory generation is the FMT (Janson, Schmerling, Clark, & Pavone, 2015). This method will be discussed in detail in Section 2-2-2.

2-2-2 Real-Time Kinodynamic Planning with Obstacle Avoidance

Kinodynamic motion planning is computationally expensive as all system dynamics are taken into account for the trajectory planning (LaValle, 2011). Because of this real-time implementations of kinodynamic motion planning have not been achieved until recently. In this section first a background of kinodynamic motion planning is given, followed by the latest approach of Allen and Pavone.

Development of Real-Time Kinodynamic Motion Planning

Based on the work of Mellinger (Mellinger & Kumar, 2011), major progress has been made in the field of obstacle avoidance. Mellinger proved that using a minimum snap trajectory a smooth dynamically feasible solution to the planning is found. Richter et. al. used this to generate feasible, polynomial, minimum-snap trajectories connecting waypoints directly in configuration space (Richter, Bry, & Roy, 2016, 2013). Richter used the dynamically flat dynamics of the quadrotor, meaning the systems inputs and states can en be expressed explicitly by it's output and it's derivatives (Mellinger & Kumar, 2011), to calculate the analytical inputs required for a feedforward controller for given generated trajectory (Richter et al., 2016).

Richter used the RRT* (Karaman & Frazzoli, 2011) algorithm for the waypoint planning. This was done off-line with prior knowledge of the obstacles in the environment. Also normal path-planning was implemented as Richter did not take into account the differential constraints of the quadrotor and thus did not do kinodynamic planning. Instead Richter developed a technique to automatically calculate the time per polynomial segment, hence limiting the velocities and actuator inputs (Richter et al., 2016).

Real-time motion planning has been accomplished in (Cowling, Yakimenko, Whidborne, & Cooke, 2007, 2010), and (Bouktir, Haddad, & Chettibi, 2008). Both accomplished this by predefining a limited amount of obstacles. This limits the number of types of obstacles for which the method will work properly, and thus safe flight in an unknown environment is not guaranteed.

Successful real-time kinodynamic motion planning was done by Frazzoli et. al. (Frazzoli, Dahleh, & Feron, 2002). Frazzoli used RRT (LaValle & Kuffner, 2001) and connecting the waypoints using a few select motion primitives from an available set of 25 primitives. Motion primitives are vehicle manoeuvres for which the input signals are predefined, for instance; turn 30 degrees. The method allowed for the path planner to find trajectories through an environment with sparse obstacles at real-time speed i.e. milliseconds. But because of the use of motion primitives the motion planner is unable to achieve completeness, i.e. the ability of reach the entire configuration space. The same holds for chess pieces, they each have

predefined motion primitives causing for instance a bishop not to be able to land on a different colour tile as the colour it started on. The inability of the planner to achieve completeness has large consequences, the motion planner may encounter difficulties in specific obstacle configurations making it impossible to guarantee the real-time computability of a feasible collision free trajectory (Frazzoli et al., 2002; R. Allen & Pavone, 2016).

Successes without prior knowledge of the environment have been made by Webb and van den Berg, with the introduction of the kinodynamic version of RRT (LaValle & Kuffner, 2001) called RRT* (Webb & Berg, 2013). Webb et. al. made kinodynamic motion planning possible without the requirement of prior information about the environment, but at a large computational cost. Because of this Webb et. al. were unable to run real-time simulations.

Allen et. al. recently demonstrated a real-time kinodynamic motion planning and trajectory control able to navigate without prior knowledge of the environment (R. Allen & Pavone, 2016). By using the real-time, machine-learning-based, kinodynamic framework from (R. Allen & Pavone, 2015), the minimum snap trajectory from (Mellinger & Kumar, 2011) and the nonlinear feedforward/feedback controller from (Lee, Leok, & Mcclamroch, 2013). The online computation times of this method were several times faster than other state of the art algorithms. The improvement was achieved by reducing the number of online OBVPs to be solved to a constant number. This reduction to constant number of OBVP is made possible by the use of machine-learning based estimates of reachability sets (R. Allen & Pavone, 2015). Using the Support Vector Machine (SVM) the target states are classified to be within or outside the reachability set, i.e. the estimated cost is within or over a certain threshold, (see Section 2-2-2).

In the following sections the real-time approach of Allen et. al. (R. Allen & Pavone, 2016) is described in detail.

Real-Time Kinodynamic Framework

Sampling-based motion planning algorithms are able to efficiently search even high-dimensional configuration spaces where states can be constrained by obstacles or differentially constrained by the abilities of the system (Lavalle, 2006). The sampling-based motion planning algorithms connect several short trajectories and does not solve a single computationally expensive optimization problem. Instead multiple OBVP are solved instead of one complex global optimization problem.

Although there are planners which do not require solving OBVPs like RRT (LaValle & Kuffner, 2001), but it has the downside that it does not guarantee optimality like algorithms like RRT*, Probabilistic RoadMap (PRM)* and FMT* (Karaman & Frazzoli, 2011; Janson et al., 2015) and is sensitive to drift. The recently proposed planner, described in Section 2-2-1 called SST, proposed by Li et. al., does provide optimality guarantees without the need to solve OBVPs. It uses forward propagated system dynamics, but also at large computational costs making a real-time implementation impossible (Li et al., 2016). Allen et. al. therefore choose to pursuit a strategy to limit the number of OBVPs to be solved.

After the planners sampled proper states, the OBVP solutions are checked for constraint violations, i.e. obstacle collisions, and form a tree when connected. A major challange for real-time execution is the number of OBVP to be solved, without additional information about

the systems reachability set the number of OBVPs is of $\mathcal{O}\left(N_s^2\right)$ where N_s^2 is the number of states (I. M. Ross & Fahroo, 2006).

Figure 2-1: The real-time framework for kinodynamic planning and control of the quadrotor, from Allen et. al. (R. Allen & Pavone, 2016)

The framework proposed in (R. Allen & Pavone, 2015) uses off-line and on-line computations to achieve successful flight. This framework is visualized in Figure 2-1, extended with the path smoothing step (R. Allen & Pavone, 2016).

In the off-line phase the following approach is used: First the function `Sample` builds a set of states V by taking a N_s number of randomly selected samples from configuration space without any obstacles. The second function `SampleData` draws N_{pair} samples randomly from V with replacement and where $N_{pair} \leq N_s\left(N_s - 1\right)$ and stores them in sets A and B. Matching these N_{pair} states in A and B gives $\frac{N_{pair}}{2}$ pairs for which the OBVPs are solved and saved in table `Cost` using the solve function `SolveOBVP`. The function `SolveOBVP` will be discussed in section 2-2-2. The look-up table `Cost` is used to train a SVM, `NearSVM`, to provide computationally efficient approximation of the reachable sets (i.e.*neighbourhoods*) for input states. The training and detailed workings of `NearSVM` are discussed in section 2-2-2. The reachable set is bounded by a cost threshold (i.e. *neighbourhood radius*) and is set manually.

In the on-line phase the following approach is used: At initialisation the algorithm is provided with the current state x_{init} and the goal region χ_{goal} which is updated on-line based upon the drone's environment. Next a set of N_{goal} samples are selected from the goal region χ_{goal} and put in the set X_{goal}. Using the SVM the neighbourhoods of the outgoing state x_{init} and the incoming state χ_{goal} are approximated rapidly, the results are stored in N_{init} and N_{goal} (See section 2-2-2). Now a limited amount of OBVPs can be solved from x_{init} and from χ_{goal} to

their closest neighbour states present in V.

Allen then uses the kinodynamic Fast Marching Tree (kino-FMT) sampling-based planner to compute the optimal path through pre-sampled states V (see Section 2-2-2). The kino-FMT algorithm introduced by Janson is asymptotically optimal and more efficient compared to other sampling-based methods (Janson et al., 2015; Schmerling, Janson, & Pavone, 2015).

The last on-line step is to smooth path, to obtain a minimum-snap, dynamically feasible trajectory. This is elaborated in section 2-2-2.

Analytical Solution to Optimal Boundary Value Problems

The quadrotor's non-linear dynamics are approximated by a double integrator system which allows for the calculation of the analytical solution to the minimum-time optimal control problem (Webb & Berg, 2013). The function `SolveOBVP` is used to calculate the analytical solution of the OBVPs between the way-point in the trajectory by using the method of Webb et. al. and Schmerling et. al. (Webb & Berg, 2013; Schmerling et al., 2015).

Machine Learning of Neighbourhoods

At initialisation of the system the current state and the goal state are connected with the pre-sampled states V which are in their neighbourhood. By doing so the number of OBVP to be solved is reduced. Allen uses the definition of the forward and backward reachable sets. The forward reachable set, i.e. neighbourhood, is the set of all states x_b for which the cost J to reach there from x_a, is less than the user defined threshold J_{th}. The backward reachable set is the set of all states x_a that are able to reach state x_b with a cost J that is less than the threshold cost J_{th}.

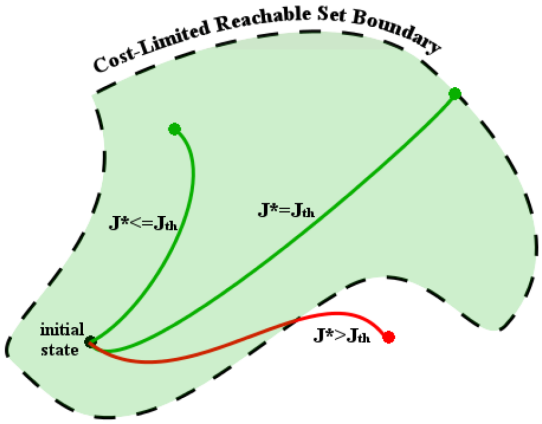


Figure 2-2: Illustrative image of a 2 dimensional cost-limited reachable set, from Allen et. al. (R. E. Allen et al., 2014)

In order to determine the computationally complex reachability sets in a real-time an approximation step is necessary (Stipanović, Hwang, & Tomlin, 2004). Allen et. al. use machine learning strategy from Allen, Clark, Starek and Pavone (R. E. Allen et al., 2014) to get an approximation of the reachability set. Allen et. al. train a support vector machine `NearSVM`

off-line with data stored in `Cost`, such that a quick approximation can be given at real-time. For the training Allen uses the method from Bishop to train the support vector machine (Bishop, 2006).

Kinodynamic Fast Marching Tree

At the heart of the motion planner Allen uses the kinodynamic version of the Fast Marching Tree, FMT* which is based on FMT (Janson et al., 2015), and adapted by Schmerling to efficiently calculate the optimal trajectory connecting the states in V from x_{init} to χ_{goal} (Schmerling et al., 2015). In the next paragraph the kino-FMT implementation as used by Allen is given in Algorithm 1, and visualised in Figure 2-4.

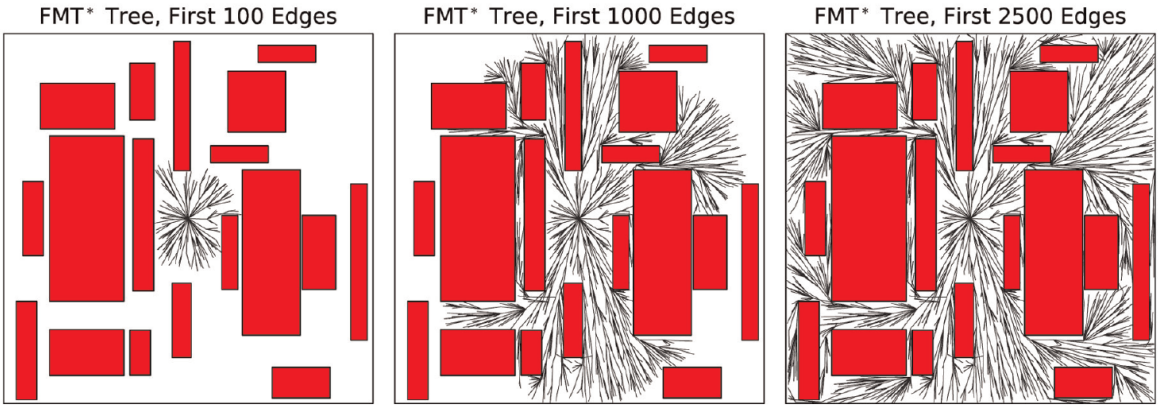


Figure 2-3: The tree development from the FMT* algorithm in a 2 dimensional cost-to-arrive space, from Janson et. al. (Janson et al., 2015)

The algorithm uses the set of pre-sampled states V, and assigns them to three subsets, $V_{unvisited}$, V_{open} and V_{closed}. The set $V_{unvisited}$ contains all pre-sampled states which are not part of the tree. The set V_{open} contains the frontier of the tree, the states which are part of the tree and are active to form potential new connections. The set V_{closed} contains all the pre-sampled states which are part of the tree.

(Line 1 - 4) Initially the current state of the MAV is stored in V_{open}, V_{closed} is empty and all other pre-sampled states are stored in $V_{unvisited}$. (Line 6) First the algorithm selects the state in V_{open} with the lowest cost-to-arrive, and stores this in the pivot variable z. (Line 7) Determine the forward-reachable set of z, called N_z^{fwd}, from the pre-sampled states, using a threshold J_th. This step uses the machine learning-based approximator `NearSVM` to speed-up the process, allowing for real-time operations. (Line 8) Calculate the discrete set X_{near} of the intersection of N_z^{fwd} and $V_{unvisited}$. (Line 9) Iterate over all states $x \in \mathrm{X}_{near}$. (Line 10) For each x the the backward reachable set N_x^{bwd} is approximated using `NearSVM`. (Line 11) Calculate the discrete set Y_{near} of the intersection of N_x^{bwd} and V_{open}. (Line 12) Calculate from the set Y_{near} the minimum cost-to-arrive node y_{min} which forms the optimal connection to the tree. (Line 13 - 16) Check for collisions between the optimal section from x to y_min. When this section is free of collisions, this section is added to the tree. And the state x is added to the tree frontier, and removed from the set of unconnected states. (Line 19) Finally

at the end of the loop over X_{near}, the pivot state z is removed from the frontier set V_{open} Finally the algorithm terminates when the pivot state z arrives in the goal region χ_{goal}.

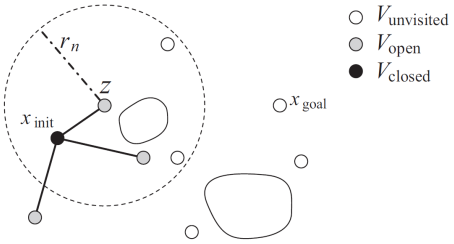
Algorithm 1: Kino-FMT implementation by (R. Allen & Pavone, 2016))

Store the pre-sampled configuration space in set V
Assign set V to $V_{unvisited}$
Assign current state to V_{open}
Initialize z with the current state
while $z \ni \chi_{goal}$ **do**
  $z \leftarrow \arg\min_{y \in V_{open}} \{\texttt{Cost}\,(y, T)\}$
  $N_z^{fwd} \leftarrow \texttt{NearSVM}\,(z, V, J_{th})$
  $\mathrm{X}_{near} \leftarrow \texttt{Intersect}\left(N_z^{fwd}, V_{unvisited}\right)$
  for *all* $x \in X_{near}$ **do**
    $N_x^{bwd} \leftarrow \texttt{NearSVM}\,(V_{unvisited}, x, J_{th})$
    $\mathrm{Y}_{near} \leftarrow \texttt{Intersect}\left(N_x^{bwd}, V_{open}\right)$
    $y_{min} \leftarrow arg\min_{y \in Y_{near}} \{\texttt{Cost}\,(y, T) + \texttt{Cost}\,(x\bar{y})\}$
    if *No collisions between* x *and* $y_m in$ **then**
      $V_{closed} \leftarrow V_{closed} \cup \{x, y_{min}\}$
      $V_{open} \leftarrow V_{open} \cup \{x\}$
      $V_{unvisited} \leftarrow V_{unvisited} \setminus \{x\}$
    end
  end
  $V_{open} \leftarrow V_{open} \setminus \{z\}$
end

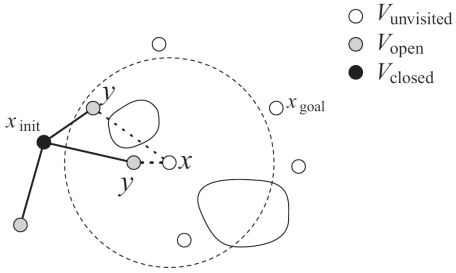
Trajectory Correction by Trajectory Smoothing

To correct for the simplifying the system dynamics as a double integrator and to further smooth the output trajectory. Allen et. al. use earlier work of Mellinger and work of Richter et. al. who formulated a high-order polynomial spline through the trajectory samples (Mellinger & Kumar, 2011; Richter et al., 2016). Richter reformulate the polynomial fitting problem where integral of the squared snap is minimised, (i.e. 4^{th} position derivative). The polynomials are constrained at the trajectory way-points (i.e. states), for position and time but the derivatives are kept as optimization parameters. In order to guarantee numerical stability the optimization is performed over these derivatives at the way-points (Richter et al., 2016, 2013). After obtaining the derivatives, the polynomial coefficients can be calculated easily for each spline (see Eq. 14 from (R. Allen & Pavone, 2016)).

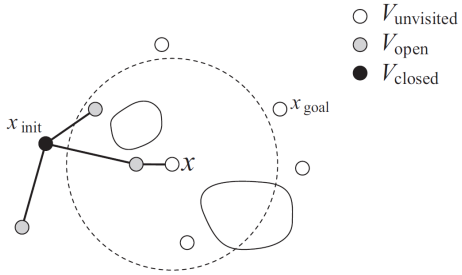
After the splines have been determined another collision check is performed, as the smoothing process will have altered the obstacle free straight line between y to x. If a collision is detected in a particular polynomial, the smoothing process of the trajectory is then redone with a new midway way-point in the segment where a collision occurred.



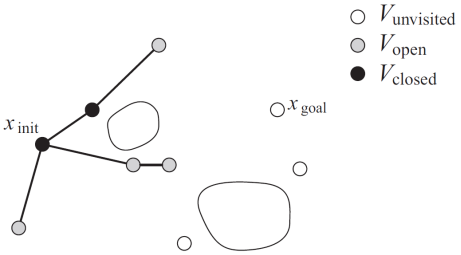
(a) Steps 1 to 3, pivot position z is selected from V_{open} and it's reachability set is determined using the machine learning-based SVM `NearSVM`. Image from Janson et. al. (Janson et al., 2015).



(b) Steps 4 to 5, for each x, the backwards reachable set N_x^{bwd} is approximated and the intersection (Y_{near}) of N_x^{bwd} and V_{open} is determined. Image from Janson et. al. (Janson et al., 2015).



(c) Steps 6 and 7, the minimum cost-to-arrive node y_{min} from Y_{near} is checked for collisions, and added to the tree V_{closed} when safe. Image from Janson et. al. (Janson et al., 2015).



(d) Step 8, after visiting all x in X_{near} all the successfully connected states are added to the tree and a new iteration starts. Image from Janson et. al. (Janson et al., 2015).

Figure 2-4: kino-FMT implementation as used by (R. Allen & Pavone, 2016), from (Janson et al., 2015)

Control Input Calculation

The splines that follow from Section 2-2-2 are continues up to the 4^{th} derivative. Mellinger et. al. provided the proof that the system representation from (Lee et al., 2013) is differentially flat and thus using the systems outputs the states and control inputs can be calculated (Mellinger & Kumar, 2011). The differential flatness proves that the smoothed trajectory is dynamically feasible and therefore the double-integrator approximation imposes no trouble. Using the control inputs and predicted states, Allen et. al. use the feedback/feedfoward controller from Lee et. al. (Lee et al., 2013), using the control input for feedforward control and a Proportional Derivative (PD) controller for the tracking of the system states.

2-3 Image Space-based Obstacle Avoidance

Autonomous flight for MAVs can be broken down into three main technical challenges, low weight and energy efficient on-board sensing, computationally efficient data representation and motion planning. This latter has been discussed in Section 2-2, so this section will focus on sensing and representation.

2-3-1 Optical Sensors for MAVs

Because MAVs are heavily constrained for size, weight and power consumption of the payload, only a few types of sensors are attractive for on-board sensing. Active optical sensors like Light Detection and Ranging (LIDAR) and structured light have been successfully used but performance is insufficient as LIDAR is only 1 dimensional and structured light is not reliable in outdoor environments (Shen, Michael, & Kumar, 2011; Bachrach et al., 2012). Normal optical cameras on the other hand provide dense reliable information in both indoor and outdoor environments. Camera modules are small, low weight, and have a very limited power consumption, this makes them as a on-board sensing device very attractive.

Optical flow has successfully been used in reactive controllers allowing for high speed obstacle avoidance but may fail in highly cluttered environments (Conroy, Gremillion, Ranganathan, & Humbert, 2009; Zingg, Scaramuzza, Weiss, & Siegwart, 2010; Chao, Gu, & Napolitano, 2014; Keshavan, Gremillion, Alvarez-Escobar, & Humbert, 2015). However they do not provide any information while the system is at rest and have trouble with obstacles near the focus of expansion (Conroy et al., 2009). To see depth while at rest or near the focus of expansion a stereo-camera set-up is often used (Fraundorfer et al., 2012).

Hrabar et. al. combined the effectiveness of optical flow with stereo-vision to improve perception at rest and around the focus of expansion (Hrabar, Sukhatme, Corke, Usher, & Roberts, 2005). This approach of combining multiple methods or sensors has been done successfully and allows for compensation of specific sensor failure cases(Shen, Mulgaonkar, Michael, & Kumar, 2013; Kendoul, 2012; Hausman, Weiss, Brockers, Matthies, & Sukhatme, 2016; Shen, Mulgaonkar, Michael, & Kumar, 2014; Droeschel et al., 2015; Nuske et al., 2015).

Monocular depth perception can use algorithms to get appearance cues from the scenery (S. Ross et al., 2013a; Hecke, De Croon, Maaten, Hennes, & Izzo, 2016; Tijmons, De Croon, Remes, De Wagter, & Mulder, 2016; Dey et al., 2016) or compute a depth map using motion

stereo (Engel, Sturm, & Cremers, 2013; Forster, Pizzoli, & Scaramuzza, 2014; Schops, Enge, & Cremers, 2014; Alvarez, Paz, Sturm, & Cremers, 2016). Although there is a lot of research in motion stereo, using a depth map calculated directly with stereo vision is less sensitive to errors and is able to provide a depth map without having to move (Goldberg & Matthies, 2011; Shen et al., 2013; Barry & Tedrake, 2014; Schmid, Lutz, Tomic, Mair, & Hirschmüller, 2014)

2-3-2 On-line Representation for MAVs

A full 3D representation of the environment is computationally heavy and requires large memory. Therefore many different approaches have been developed. For reactive control navigation, two dimensional image space is widely used (Beyeler, Zufferey, & Floreano, 2009; Conroy et al., 2009; S. Ross et al., 2013b). Also volumetric, three dimensional cartesian voxel structures as shown in Figure 2-5 have been used for world representation (Hornung, Wurm, Bennewitz, Stachniss, & Burgard, 2013; Shen et al., 2011; Bachrach et al., 2012; Fraundorfer et al., 2012). And two dimensional polar data representation has been used, which benefit from being body-frame based and corresponds to the sensory data input (Bakolas & Tsiotras, 2008; Yu & Beard, 2013).

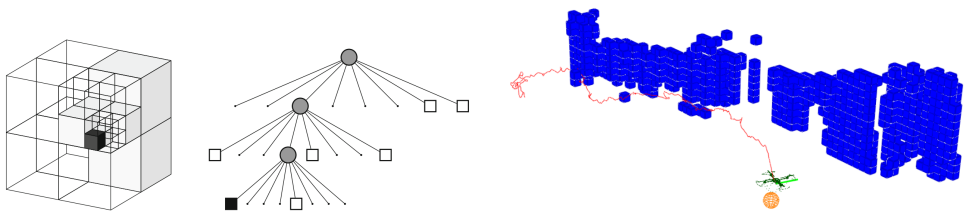


Figure 2-5: On the left, volumetric voxel representation. Occupied or partially occupied space is subsampled. And in the middle the efficient OctoMap data tree (Hornung et al., 2013). On the right, voxel representation of obstacles. (Fraundorfer et al., 2012)

An effective and memory efficient representation is image space and polar representation as they require little transformations from the raw sensory input from the camera (Brockers et al., 2014). This type of representation also allows for efficient fusion with inverse-range representation for systems using stereo-vision (Bajracharya, Howard, Matthies, Tang, & Turmon, 2009).

The representation approach of Brockers et. al. using polar representation in image-space, allows for collision checking in two dimensional image-space. Which had been done earlier by Otte et. al. who proved this reduced the computational load (Brockers et al., 2014; Otte, Richardson, Mulligan, & Grudic, 2009).

By using the computationally efficient image space for obstacle avoidance, Brockers et. al. realised on-board autonomous flight in cluttered environments. Using stereo-vision disparity maps for depth estimation and inverse range, polar-perspective representation for computationally efficient avoidance was achieved. In order to fly through the environment, they use a closed-loop RRT (LaValle & Kuffner, 2001) that simulates the systems dynamics to generate the control inputs. This closed-loop RRT planning is performed in 3D space, for maximum freedom in movement. While the collision checking is performed in a 2D Configuration Space (C-space), allowing for a real-time on-board implementation.

2-3-3 Image Space-based Obstacle Avoidance Framework

As described in the previous section, Because of the computational complexity of 3D space representation, Brockers et al. used an inverse-depth image-based representation based on the disparity map from a stereo-camera. By doing so an efficient method for trajectory planning and collision checking is possible.

The general outline of the approach is visualized in Figure 2-6 and is as follows. Using a frontal stereo-camera images are taken and used to calculate in a real-time manner a disparity map. Using a polar representation, for each disparity value the corresponding distance is calculated and then expanded with a predefined radius. This expansion is called a C-space expansion and creates a 2.5 dimensional C-space. Depth information is represented in 2D image-space. This C-space expansion allows them to treat the MAV as a point mass system as the systems size has been used for the expansion radius. The inverse-depth representation for obstacle detection works well for nearby objects whereas distant objects will be of little influence. The combination of polar representation and the 2.5 dimensional image space only a very limited amount of memory is required and efficient 2 dimensional searches can be performed for collision checking of the proposed trajectories. The collision checking approach is described in Section 2-3-3, and the motion planning and control is described in Section 2-3-3.

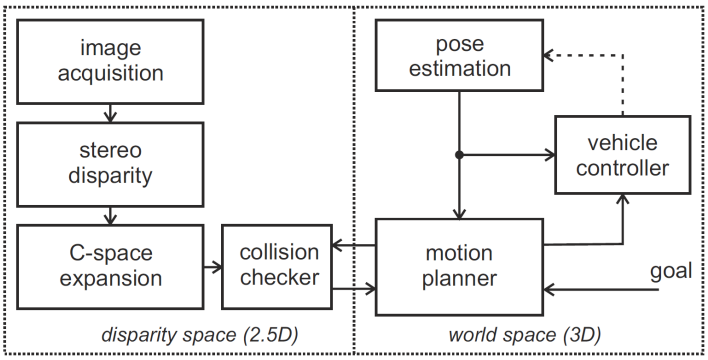


Figure 2-6: System architecture from Brockers et. al (Brockers et al., 2014)

Image Space Expansion

The expansion, visualized in Figure 2-7, is performed as follows. Each point $p(u, v, d)$ in the 2.5D disparity map is projected into 3D world space $P(x_w, y_w, z_w)$ using a polar representation. For the expansion a sphere radius based on the MAV's confining sphere is used. Instead of projecting the confining sphere onto image space, the smallest square covering the sphere is projected onto image space, see Figure 2-8. This square is assigned the lowest disparity value from the disparity map, covered by this square.

A real step in computational efficiency is made by pre-calculating the expansions for pixel-disparity combinations off-line. The resulting tables are efficiently searched on-line, a note has to be made that this does require relative large memory space for high resolution images and a high disparity range.

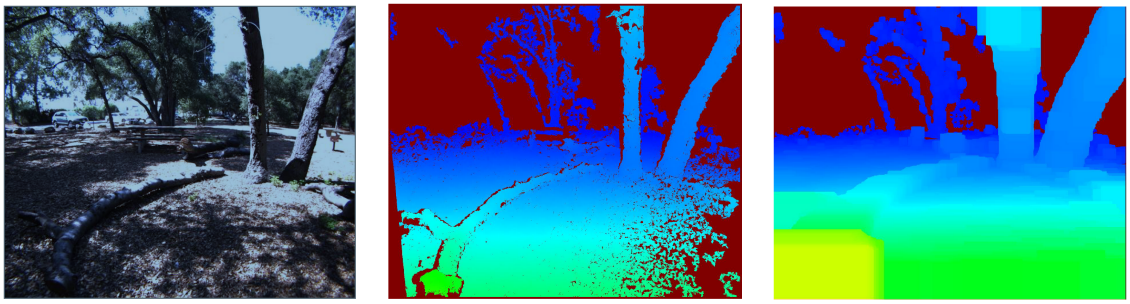


Figure 2-7: From left to right. The original left view image, stereo disparity map, C-space expanded disparity map. From Brockers et. al (Brockers et al., 2014)

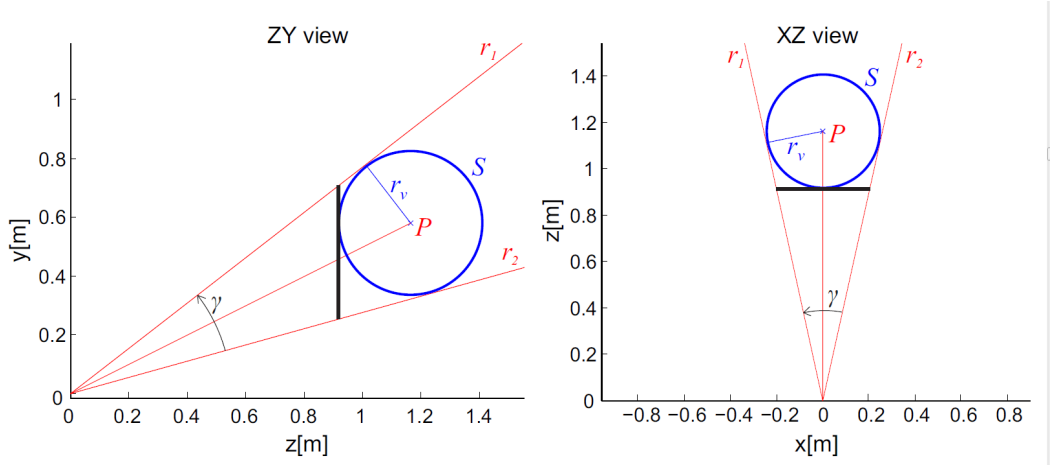


Figure 2-8: The configuration space explansion from Brockers et. al (Brockers et al., 2014)

Collision Checking in 2.5D Image Space

After obtaining the expanded image-space, collision cheing is performed. This procedure is reduced to projecting linear 3D trajectories onto 2.5D C-space, and comparing disparity values. By still considering trajectories that are a given k distance behind an obstacle, occluded trajectories that pass far behind obstacles become possible. When the disparity values of the trajectory are within this k distance from an observed obstacle in C-space, it is classified as a collision and the trajectory is infeasible.

Motion Planning and Closed-loop Control

The motion planner uses closed-loop RRT. The RRT algorithm is used to generate a tree of reference trajectories, next a closed-loop controller is used to simulate the system output and corresponding input commands to reduce the position error, see Figure 2-9. This approach allows for non-linear or unstable systems to be controlled by using a properly designed feedback controller. Also relative long dynamically feasible trajectories can be simulated.

The controller is designed such that safety invariance is guaranteed (Schouwenaars et al., 2001), this is accomplished by limiting the vehicles velocity such that at any time the MAV is

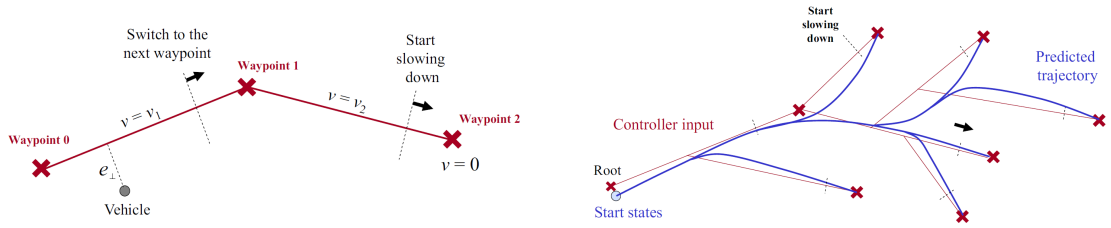


Figure 2-9: From left to right: . From Brockers et. al (Brockers et al., 2014)

able to reach a hovering state before it reaches the end of a validated trajectory segment it is following. This is important as trajectories behind observed obstacles, of an assumed size k, are assumed to be feasible and replanning might be necessary. The reference trajectory and simulation output are generally close (Luders, Karaman, Frazzoli, & How, 2010), Brockers et al. still re-simulate the trajectories and corresponding input signals using the current state and new way-points, this to reduce possible drift.

2-3-4 Egocylinderical Image Space Representation

Building further on the image space expansion approach from Brockers et. al. (Brockers et al., 2014), described in the previous sections, Brockers, Fragoso and Matthies introduce a novel image space representation for MAV navigation (Brockers et al., 2016). They propose to project the inverse range map onto a vehicle-centred cylinder (See Figure 2-10), called the egocylinder (Brockers et al., 2016). The C-space expansion from (Brockers et al., 2014) is subsequently applied directly on to this egocylindrical image space. The advantage of this egocylinderical representation is that reactive motion planning algorithm can be used directly on the egocylinder image. As the location of the obstacles in the image corresponds directly to their position in world space relative to the vehicle. This allows for high velocity flight with high situational awareness of the MAV and even efficient multi-sensor fusion as mentioned in Section 2-3-1.

Egocylindrical Projection Framework

Brockers et. al use stereo-vision in combination with the block matching algorithm from Hirschmuller et al. to obtain a disparity map (Hirschmüller, Innocent, & Garibaldi, 2002). Next they project this disparity map on the egocylinder which is a vehicle-centred, inverse-range cylindrical projection space. The straight forward method for projection is given in Brockers et. al. (Brockers et al., 2016). In this projection procedure from 3D coordinates to 2.5D, only the horizontal component of disparity value is considered, neglecting the vertical component. They choose a cylindrical representation over a spherical as in general MAVs move in the horizontal plane and altitude changes are uncommon. The framework is visualized in Figure 2-10.

The next step is to perform the C-space expansion, necessary to regard the MAV as a point in 3 dimensional space for practical motion planning and obstacle avoidance. The C-space expansion method from Brockers et al. (Brockers et al., 2014) is used, with the only difference in the calculation for the disparity value, as described above.

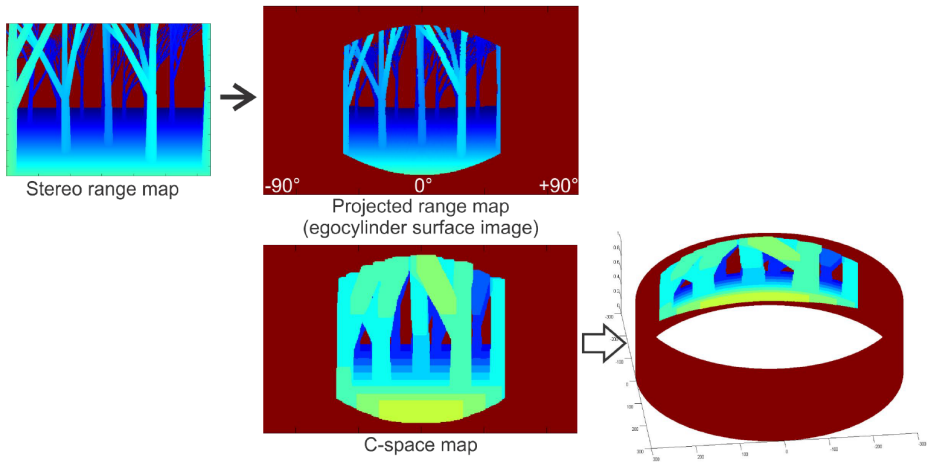


Figure 2-10: The projected disparity map on the egocylinder, in expanded image space. Brockers et. al. (Brockers et al., 2016)

Motion Planning in Egocylindrical Image Space

In contrast to the motion planning method used in Brockers et al. (Brockers et al., 2014), Brockers now implements a relative simple reactive method. A strong assumption on which the method is based is; the MAV is able to instantly change the direction of flight instantly. This allows the the velocity to be used as the planning horizon with the comparison between the stopping distance estimates and the observed disparity values for collision checking. The method assumes it has a general goal direction, and then using an image search on the egocylinder a flight direction can be found which is the closest direction towards the goal while avoiding collisions with obstacles. This is visualized in Figure 2-11.

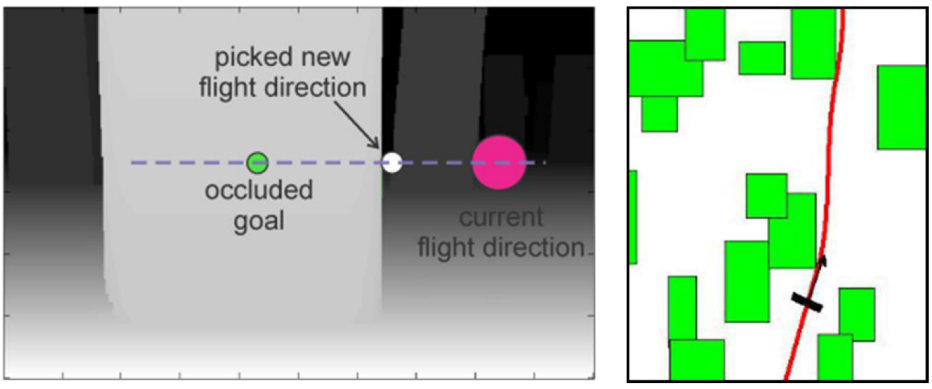


Figure 2-11: The motion planning algorithm. Left: Selected flight direction close to the goal, avoiding the object. Right: Top view of simulated flight. From Brockers et. al. (Brockers et al., 2016).

Chapter 3

Superpixel Segmentation for Object and Region Detection

In contrast to Section 2-2 where avoidance strategies were based on prior knowledge of the obstacles, this section will discuss approaches for detecting obstacles in real-time. This is important as the stereo-board will provide a sparse disparity map and thus a complete view of the environment is not available. Extending the information of the sparse disparity map with segmented objects might provide enough information for trajectory planning. The idea to use superpixels is to identify featureless regions as near obstacles or distant scenery.

A method to extract information from an image is to deploy image segmentation methods, which can identify image regions with similar appearance, indicating potential obstacles. The field has been researched extensively where various methods have been developed (Khan, 2013). A special interest goes to a segmentation approach called superpixels. Advances as made by Levinshtein et. al. have shown that progression in computationally efficient high-speed clustering is possible (Levinshtein et al., 2009).

One method of segmentation was introduced by Ren and Malik, called superpixels, where they over-segmented the image by clustering similar pixels with taking into account the compactness of each segment (Ren & Malik, 2003). Superpixel segmentation is an active topic of research (Moore, Prince, Warrell, Mohammed, & Jones, 2008; Moore, Prince, & Warrell, 2010; Veksler, Boykov, & Mehrani, 2010; Zhang, Hartley, Mashford, & Burn, 2011). Of which the latest work of Zhang et. al. is the fastest of this group with 2Hz. A large improvement in terms of quality has been achieved by Liu et. al. who introduced a balancing term to enforce compactness of the clusters, but still fail to achieve sufficient computational efficiency with a image processing time of 2.5 seconds per frame (Liu, Tuzel, Ramalingam, & Chellappa, 2011).

Continuing in the field Wang et. al. used superpixels to address the challenging problem of tracking in image space (Wang, Lu, Yang, & Yang, 2011). The optical sensors on the volatile MAV platform perceive large scale, motion and shape deformations due to the movements of the vehicle. The system also perceives many occlusions in the highly cluttered indoor environments it can operate. Wang et. al. propose a superpixels based method to store structural information, creating a mid-level appearance cue able to distinguish the target from the

background even when encountering large shape deformations and occlusions. Computational efficiency still remains an issue for real-time implementation.

A faster method is proposed by Achanta et. al. which goes by the name SLIC, with the use of SLIC real-time image segmentation is achieved (Achanta et al., 2010). A recent study comparing SLIC superpixels with state-of-the-art segmentation methods shows that SLIC outperforms most of the earlier developed approaches in terms of computational efficiency (Achanta et al., 2012). A in depth discussion of SLIC is given in Section 3-1. Another recent real-time performing superpixel method which is introduced by Van den Bergh called SEEDS (Van den Bergh et al., 2015), is discussed in Section 3-2.

3-1 SLIC SuperPixel segmentation

For superpixels to be applicable in obstacle avoidance algorithms for MAVs the have to be computationally efficient and produce reliable, in the sense of stable, high quality segmentations. Achanta et. al. introduce a novel clustering algorithm that uses five dimensions, L, a, b CIELAB colour space and 2D image space to calculate compact and uniform superpixels (Achanta et al., 2010). The segmentation is uniform in colour and each superpixel is compact, as can be seen in Figure 3-1. Achanta et. al. proved using the Berkeley benchmark dataset (Martin, Fowlkes, Tal, & Malik, 2001) that SLIC outperforms most earlier developed segmentation methods in terms of the quality of the segmentation and computational efficiency (Achanta et al., 2010), and distinguishes itself from other methods with its simplicity. In the next section a more recent and complex superpixel segmentation approach is presented, which outperforms the latest superpixel implementations.

Figure 3-1: Examples of image segmentation using SLIC, superpixel size are approximately 64, 256 and 1024 pixels. The compact superpixels show large colour uniformity. From Achanta et. al (Achanta et al., 2010)

3-1-1 SLIC distance measure

A novel distance measure ensures ensures a well balanced weighing of cluster compactness and colour uniformity. Centers of the superpixels will be approximately be spaced with an interval S of size $S = \sqrt{N/K}$. Where K is the number of superpixels and N is the total amount of pixels. At initialization the algorithm assigns the K superpixel cluster centers C_k with $C_k = [l_k, a_k, b_k, x_k, y_k]^T$, $k = [1, K]$ and spaced with regular interval S. To minimise the computational efforts it is assumed that for each pixel the associated cluster center it belongs to is within a $2S$ distance. Achanta et. al. propose a normalised distance measure D_s (Achanta et al., 2010), shown in Equation 3-1 and defined as follows:

$$\begin{aligned} d_{lab} &= \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \\ d_{xy} &= \sqrt{(x_k - x_i)^2 (y_k - y_i)^2} \\ D_s &= d_{lab} + \frac{m}{S} d_{xy} \end{aligned} \tag{3-1}$$

where D_s is the distance measure, the sum of the normalised euclidean image space distance and the euclidean distance in colour space (Achanta et al., 2010). The euclidean image space distance is scaled using parameter m to allow control on the compactness of the superpixels, a high value for m ensures highly compact superpixels, whereas a low value of m ensures the superpixels form high colour uniform clusters.

3-1-2 SLIC algorithm

The SLIC algorithm is interesting for MAV applications because of its computational efficiency which is enabled by its lean and simple algorithm. This Section summarises the algorithm. As described in Section 3-1-1, the cluster centers C_k are regularly spaced over image space, covering the entire image. To ensure the cluster centres are not positioned on image edges which would cause highly irregular superpixels. Each cluster center is relocated to the location with the lowest gradient in a 3x3 area around the center. Achanta et. al. define the gradients G are calculated as follows:

$$G(x, y) = \|\mathbf{I}(x+1, y) - \mathbf{I}(x-1, y)\|^2 + \|\mathbf{I}(x, y+1) - \mathbf{I}(x, y-1)\|^2 \tag{3-2}$$

where $\mathbf{I}$ is the *lab* vector at position (x, y) and $\|.\|$ is the $\mathtt{L}_2$ norm (Achanta et al., 2010).

After initialisation each pixel is assigned to the nearest cluster where the pixel is within the $2S$ search area of that cluster center. After all pixels are assigned to a cluster center, the cluster centers are relocated to the center of the cluster and its *lab* values are recalculated as the mean of the cluster. By iterating the center relocations and pixel assigning, convergence is achieved within four to ten iterations (Achanta et al., 2010). The SLIC pseudo-code is given in Algorithm 2.

Algorithm 2: Simple Linear Iterative Clustering. (From Achanta et. al. (Achanta et al., 2010))

Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling with a regular step-size S.
Move cluster centers in a n × n neighbourhood, to the lowest gradient position.
repeat
  for *each cluster center* C_k **do**
    Assign the best matching pixels from a $2S \times 2S$ search area around the cluster center according to the distance measure (Eq. 3-1) to the cluster.
  end
  for do
    Compute new cluster centers and residual error E { L_1 distance between previous centers and recomputed centers }
  end
until $E \leq$ *threshold*;
Enforce connectivity

3-2 SEEDS Superpixel segmentation

A new superpixel over-segmentation approach called SEEDS is based on hill-climbing optimization, i.e. gradually maximising a energy equation (Van den Bergh et al., 2015). Using an initial gird of superpixels (see Figure 3-2), the boundaries are continuously modified to refine each superpixel. A robust and efficient energy function is defined to enforce colour uniformity within the superpixel and the colour histogram. In the latter this method differs significantly from the simple clustering method proposed by Achanta et. al. (Achanta et al., 2010).

The boundary modifications performed in a hierarchical procedure using pyramid block-sizes, refining the updates with decreasing block-sizes up to pixel level. This pyramid scheme proves to be effective in reducing the computational effort of the segmentation.

By testing the algorithm van den Bergh et. al. show that SEEDS outperforms the state-of-the-art superpixel segmentation methods on the Berkeley benchmark dataset in terms of lowest computational effort (Van den Bergh et al., 2015).

SEEDS is developed from the interpretation of the definition of a high quality segmentation. The ability to group pixels together with similar colour and form cluster boundaries along object edges. To enforce this colour consistency an energy maximization problem is defined using the colour distribution within superpixels and the shape of its boundaries.

The SEEDS method as described in (Van den Bergh et al., 2015) is build up as follows. At initialisation all pixels are assigned to the superpixels $\mathcal{A}_k$ where all superpixels are restricted to be disjoint (i.e. pixels belong to a single superpixel), and each superpixel forms a continuous image region.

The set $\mathcal{S}$ is defined as all valid segmentations, $\bar{\mathcal{S}}$ is defined as the set of invalid segmentations, and $\mathcal{C}$ is combined set of $\mathcal{S}$ and $\bar{\mathcal{S}}$. Defining $s^\star$ as the segmentation what maximises the energy function:

$$s^\star = \arg\max_{s \in \mathcal{S}} E(s) \tag{3-3}$$

Van den Berg proposes an energy function $E(s)$ with two terms one based on the likelihood of the colour similarity within the superpixels $H(s)$ and one based on the shape of the superpixel $G(s)$. The colour similarity term $H(s)$ is taken to be the colour density distribution of each superpixel, with $\Psi(c_{\mathcal{A}_k})$ taken as the sum of squares, $H(s)$ can be expressed as:

$$H(s) = \sum_k \Psi(c_{\mathcal{A}_k}) = \sum_k \sum_{\mathcal{H}_j} (c_{\mathcal{A}_k}(j))^2 \tag{3-4}$$

where $c_{\mathcal{A}_k}$ is taken to be the colour histogram $\mathcal{H}$ of the set $\mathcal{A}_k$, and is calculated with:

$$c_{\mathcal{A}_k}(j) = \frac{1}{Z} \sum_{i \in \mathcal{A}_k} \delta(I(i) \in \mathcal{H}_j) \tag{3-5}$$

where Z is the total number of pixels part of set $\mathcal{A}_k$, δ is the indicator function giving 1 if pixel $I(i)$ is part of bin j and 0 otherwise.

The boundary term $G(s)$ is proven to be optional in terms of quality, due to the hierarchical refining approach which enforce edge smoothness (Van den Bergh et al., 2015). Still in order to provide control over the superpixel shape, van den Berg et. al. propose a similar term for $G(s)$:

$$G(s) = \sum_i \sum_k (b_{\mathcal{N}_i}(k))^2 \tag{3-6}$$

and

$$b_{\mathcal{N}_i}(k) = \frac{1}{Z} \sum_{j \in \mathcal{N}_i} \delta(j \in \mathcal{A}_k) \tag{3-7}$$

where $b_{\mathcal{N}_i}$ is the histogram of the number of superpixels present in a $N \times N$ patch around pixel i. Maximising $G(s)$ will favour as little as possible different superpixels present in the patches and therefore enforce smoothness at superpixel edges.

The computational efficiency of the SEEDS algorithm lays in the use of the histograms to asses the colour similarity and its hierarchical refinement. The hill-climbing optimization algorithm uses small local changes in order to optimize the global segmentation. When a proposed change increases the energy function the change is applied, as it assists in maximising the energy function. The algorithm starts with initializing large superpixels which can be divided in smaller blocks of 2×2, all the way to pixel level. An example is shown in Figure 3-2.

The algorithm starts by selecting a random block or pixel $\mathcal{A}_k^l$ from all boundary blocks or pixels and assign the pixel or block to a random neighbouring superpixel $\mathcal{A}_n$, when this creates a valid partitioning the energy function is evaluated to see if the change should be applied. According to Proposition 1 from van den Berg et. al. (Van den Bergh et al., 2015), the intersection between the two histograms is:

$$\textbf{int}(c_{\mathcal{A}_n}, c_{\mathcal{A}_n}) = \sum_j \min\{c_{\mathcal{A}_a}(j), c_{\mathcal{A}_b}(j)\}$$

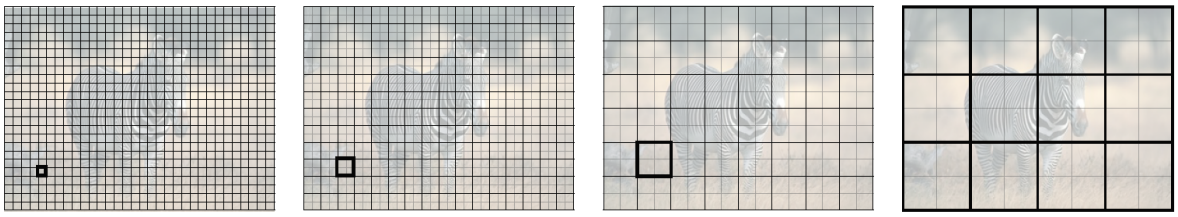


Figure 3-2: SEEDS initialization visualizing different block sizes. From (Van den Bergh et al., 2015)

and can be used to determine if the energy function will increase for a given segmentation:

$$\mathbf{int}\left(c_{\mathcal{A}_n}, c_{\mathcal{A}_n}\right) \geq \mathbf{int}\left(c_{\mathcal{A}\setminus\mathcal{A}_k^l}, c_{\mathcal{A}_k^l}\right) \Longleftrightarrow H\left(s\right) \geq H\left(s_t\right) \tag{3-8}$$

In order to validate Equation 3-8 two important assumptions are made. The first states; the size of the changing block or pixel is significantly smaller than the size of the superpixel. The second assumption states; the histogram of the change only consists of a single bin. Van den Berg showed that these assumptions hold in 93% of the cases (Van den Bergh et al., 2015).

When a new portioning is based on Equation 3-8, the new histogram are efficiently updated, at block level by subtracting $c_{\mathcal{A}_k^l}$ from $c_{\mathcal{A}_k}$ and adding to $c_{\mathcal{A}_n}$, and at pixel level by subtracting 1 from bin j and adding to bin j of histograms $c_{\mathcal{A}_k}$ and $c_{\mathcal{A}_n}$ respectively.

The refinement is terminated after a certain time t_{stop}, given a valid segmentation is obtained incredibly fast and the refinement process can be stopped directly after the last refinement iteration which takes almost no time due to its limited operations.

Figure 3-3: SEEDS over-segmentation results showing high quality superpixels. From (Van den Bergh et al., 2015)

In Figure 3-3 the over-segmented results of the SEEDS method is given. The ground truth is visualized by colours and the superpixel boundaries are shown in white.

Just like the distance measure of SLIC, the energy equation used by van den Berg can be adjusted to accommodate besides the colour uniformity and boundary smoothness, other inputs e.g. depth information (Van den Bergh et al., 2015). Enforcing depth uniformity within a superpixel by including depth information in the energy equation, it is expected that obstacles will become more distinguished from the background. This makes the SEEDS algorithm interesting for consideration in obstacle avoidance approaches.

Chapter 4

Literature based Conclusions and Recommendations

Major steps have been taken in an effort to achieve full autonomous flight of MAV in unknown cluttered environments. Various sampling-based planners have been developed in the last few years indicating an increasing interest in their applications. The combination of machine learning in the form of support vector machines with the Fast-Marching-Tree, has proven that sampling-based motion planners are able to search high-dimensional configuration space in a real-time manner. This makes this approach interesting for full autonomous flight for large Unmanned Aerial Vehicles (UAVs) if more computational optimization is possible. Because of the intention to develop a novel method which can be implemented on the MAVLab stereoboard, the discussed motion planners are expected to be too computationally demanding.

Alternatively, advances in Image Space representation in combination with on-line expansions promise higher potential on computationally heavily constrained systems. Current methods are able to identify obstacles to an extent that cluttered regions are avoided in total, limiting exploration missions considerably. This exposes the need for an vehicle-size dependent algorithm that is capable of identifying free-space trajectories in highly cluttered environments.

A proposed first step towards to such an algorithm will be the inversion of the configuration space expansion approach presented in this review. By adapting the way an expansion is used, a major improvement is made in terms of increasing the algorithms efficiency. In the space behind a C-Space expansion, only disparity values higher than a certain value have to be examined, as low disparity values will turn out to be fully occluded. Thus by reducing the disparity range in image regions where an expansion has already been performed, a large improvement in computational efficiency is expected.

Lastly the ability to identify obstacle volumes independent of disparity map generation will allow for allocating computational budget to relevant image space regions. In these concentrated image regions more advanced disparity calculations can be done, as the potential flight directions lie within these regions. In the next part the performance and feasibility are tested of the previously discussed methods.

Part III

Preliminary Problem Analysis

Chapter 5

Feasibility and Performance Analysis

In Chapter 2 the latest achievements regarding real-time trajectory planning, object detection and image space representation were presented. Based on this literature several simulations are performed and presented in this chapter, with the goal to answer the research questions presented in Section 1-2.

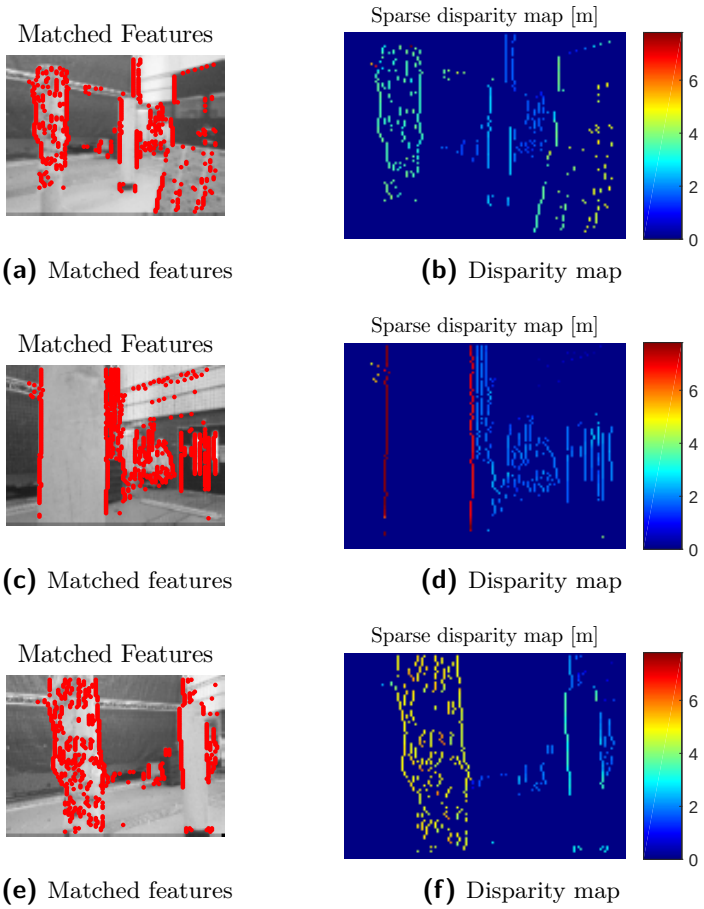
5-1 Efficient Disparity Calculation from Stereo-Image Pairs

As mentioned in section 1-1, the aim of this Thesis is to contribute to obstacle avoidance methods while using the stereo-board, developed by MAVLab. The stereo-camera board allows for on-board matching and disparity calculations. Tijmons et. al. found an efficient method to calculate a sparse disparity map by compromising between quality and efficiency (Tijmons, Croon, Remes, De Wagter, & Mulder, 2016). They calculate the horizontal differential convolution over single lines, and where the gradient exceeds a certain threshold a standard Sum of Absolute Differences (SAD) window matching is applied. In this preliminary problem analysis, three stereo-image pairs will be used which are selected from a test sequence performed in the flight arena of the TUDelft.

In Figure 5-1 edges features are shown in the left column, indicated in red. In the right column the corresponding disparity map is shown, where warmer colours represent a higher disparity value and thus a closer proximity to the camera.

In Figure 5-1a edges are found in nearly all image regions. The pole on the left side shows a lot of texture and is therefore easy to detect. The featureless pole in the center of the image is also easily detected as it has a large contrast to the background. The board on the right side of the image is also well detected with spread features all over it. It has to be noted that the dark background on the left side of the image, and the floor do not have many if any features.

The corresponding disparity map is shown in Figure 5-1b. Warmer colours indicate a higher disparity value and thus a closer proximity, pixels that do not have a disparity value are



(a) Matched features **(b)** Disparity map

(c) Matched features **(d)** Disparity map

(e) Matched features **(f)** Disparity map

Figure 5-1: Matched features are shown in the left column. The corresponding disparity maps are shown in the right column, warmer colours correspond to objects in closer proximity.

counted as having a disparity value of zero which corresponds to an infinite distance. In this figure the pole on the left is clearly identifiable, but the pole in the center of the image is only distinguishable by its edges, as the pole has a uniform texture and colour. On the board at the right side of the map, it is seen from the colour change on the board that its left edge is further away than its right side.

In Figure 5-1c edge features are found on the edges of the pole, which stands in the left side of the image. Spread over the background on the right side of the image many feature are found to due to the large texture differences. Because the pole is close to the camera it covers a large area of the image, this separates its edges a lot leaving a large featureless region.

The corresponding disparity map is shown in Figure 5-1d. it is clear that in the entire right side of the image no obstacles are found with a close proximity. All the blue coloured disparity values correspond to a distance of at least four meters, which at this point is assumed to be a safe distance away. The large featureless pole on the left side of the image is only identifiable by the two red edges, indicating a dangerously close obstacle, i.e. less than one meter in this case. A difficult situation presents itself in the featureless region between the pole edges, as its unclear if this represents a textureless obstacle or a window into a textureless room.

In Figure 5-1e edge features are found on the pole in the left side of the image, covering the entire texture-rich obstacle. On the right side of the image the textureless pole is identified by its edges. On the far left of the image no features are found creating uncertainty about the proximity in this region. To the right of the left pole various features are found from the texture-rich background.

The corresponding disparity map is shown in Figure 5-1f. It is clear that the pole on on the left side of the image is at a close proximity to the MAV. The pole on the right faces the same problem as the left pole in Figure 5-1d, a textureless region between the pole edges. The textureless region in the top-center part of the image is a safe region to fly but is not confirmed by disparity values due to the lack of found edges.

From the results shown in Figure 5-1 it can be concluded that research has to be done in object detection, in an effort to find a solution for the featureless regions. If textureless regions between pole edges can be identified as poles, all other textureless regions can be assumed to be distance scenery and thus represents a safe flight direction.

In the scenario where a textureless region represents a window into a textureless room, avoiding the region all together will limit the capabilities of the MAV to explore the environment. Additional research will have to be done in distinguishing textureless objects from windows.

5-2 Image Segmentation for Object Detection

In this section the SLIC approach is applied to the same three scenes as presented in the previous section. In an attempt to obtain more information from the image SLIC is applied. The idea is that the superpixels will assist in distinguishing objects from distant scenery, solving the problem described in Section 5-1. Three aspects will be investigated, firstly computational costs, secondly the influence on the quality by using gray-scale images, lastly the influence of increasing the number of superpixels.

5-2-1 SLIC Computational Costs

Firstly it is acknowledged that the literature presented in Chapter 2 suggest that the SEEDS superpixel approach outperforms the SLIC approach. But still the SLIC approach will be used to verify if superpixels in general could assist in distinguishing objects from distant scenery in a real-time manner.

In order to test the computational cost of the SLIC method, a C/C++ implementation is used as the intended platform, the MABLab stereo-board, also allows for C/C++ implementation. For the test. A test image from the Berkeley benchmark dataset (Martin et al., 2001) is used which is in correspondence with the work of Achanta et. al., see Figure 3-1. The image is downscaled to a resolution of 128×96 to correspond to the stereo-board resolution. The for comparison the computational time is expressed in computational frequency in Herz, the results are shown in Table 5-1.

The number of intended superpixels is increased from 24 to 240 in the following steps: $\{24, 48, 96, 192, 240\}$ with 500 simulations for each setting. The standard deviation σ and mean μ values for each set number of superpixels is presented in the table. With a only 24

Table 5-1: Computational effort of SLIC0 segmentation method of an 128x96 image expressed in [Hz], as function of the number of superpixels. Calculated with a 2.8 GHz Core 2 Duo processor.

	Number of superpixels [-]				
	24	48	96	192	240
σ [Hz]	8.9785	7.3284	8.1293	7.3338	4.8457
μ [Hz]	219.5517	193.3839	171.9013	150.4714	141.9897

superpixels the algorithm is able to run at a frequency of 220 Hz. With an increasing amount of superpixels the frequency decreases linearly to 142 Hz for 240 superpixels. The corresponding standard deviations provides a basis to conclude that the SLIC algorithm is able to run at real-time frequencies for all relevant number of segmentations and that the number of segments is of limited influence on the computational effort. For this simulation a laptop with a 2.8 GHz Core 2 Duo processor is used, this is a lot faster than the embedded processor on the stereoboard, which runs on 168 MHz. Sub-sampling will be considered to decrease the computational effort, but heavy concerns remain about the feasibility to implement any superpixel segmentation method on the stereoboard.

5-2-2 SLIC Quality Decrease due to Gray-Scale Image

In this section the impact of using gray-scale images in combination with the SLIC algorithm is investigated. The SLIC algorithm is developed to be used with the l,a,b values from the CIELAB colour space, which contains a lot more information than just intensity values. Due to bandwidth restrictions on the stereo-board, instead of colour images we are provided with gray-scale images and thus the impact on the segmentation quality has to be investigated. For this investigation the same test image as in the previous section is used. It originates from the Berkeley benchmark dataset (Martin et al., 2001), see Figure 3-1 and Figure 5-3

The image is segmented using three different amounts of superpixels: $\{48, 96, 293\}$, first using colour space and subsequently in gray-scale. The resulting segmentations are shown in Figure 5-2. To compare the results with the segmented image, Figure 5-3 shows the image on the background.

The segmentations are visualized as follows, in green the colour-based SLIC is shown, in magenta the gray-scale-based SLIC is shown. The superpixels edges that stay the same for colour space and gray-scale are indicated in white.

In Figure 5-2a large difference are seen in the center of the image. This region contains little texture causing the the colour based segmentation to be relatively steered more than the intensity-based segmentation.

In Figure 5-2b large difference are seen only in the top left and a small region in the center of the image. The same holds for this number of superpixels; the small amount of texture in the center is causing the the colour based segmentation to be relatively steered more than the intensity-based segmentation. In the top left of the image a recurrent texture caused the two methods to deviate around a image spot.

Finally in Figure 5-2c differences can be seen in in the entire right region, outer top region and on a small region in the left of the image. Due to the relative small size of the superpixels

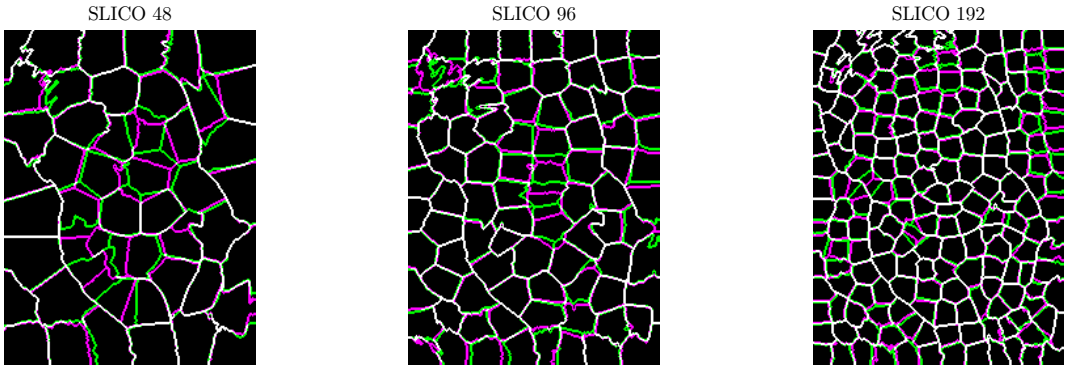


Figure 5-2: Segmentation results, in green: colour based SLIC, in magenta: gray-scale based SLIC superpixels.

the deviations are also small, with an exception in the relative small region in the left part of the image. In this specific region it is suspected that at initialization the cluster center is positioned on top of a edge. At initialization cluster centers are moved in a local region to the lowest gradient location, this gradient has a colour component which is suspected to push the superpixel cluster to a different position than when using intensity values.

Figure 5-3: Reference image together with the segmentation results, in green: colour based SLIC, in magenta: gray-scale based SLIC superpixels.

After examination it can be concluded that the quality of the segmentation decreases with the use of gray-scale images instead of colour-space. The reduction of information in the images causes the clusters to be less capable of detecting colour based edges. Furthermore it can be concluded that with increasing number of superpixels the quality decrease from colour-space to gray-scale-space is limited. Thus better quality segmentation is obtained with a larger amount of superpixels.

5-2-3 SLIC Number of Superpixels

In the previous two sections the computational effort of SLIC and the impact of gray-scale based segmentation was investigated. It was found that the computational effort increases marginally with the amount of superpixels. Furthermore it was found that the impact of using gray-scale images on the segmentation quality decreases with an increasing amount of superpixels. This raises the question; to which extent does increasing the number of superpixels improve the quality of the segmentation in terms of object detection? This section will investigate the influence of varying the amount of superpixels.

Figure 5-4, Figure 5-5 and Figure 5-6 show the SLIC segmentations with an increasing number of superpixels. From left to right the number of intended superpixels is increased from 24 to 240 in the following steps: $\{24, 48, 96, 192, 240\}$.

In Figure 5-4 it can be seen that when using 24 superpixels none of the obstacles is segmented properly. The edges of the pole to the left are only detected in the center of the pole, and the pole in the center of the image falls in a larger segment which covers a large part of the background to. The edges of the board are also not found using this setting. In the second image from the left shows the segmentation using 48 superpixels. The pole on the left is detected properly, excluding its top quarter. The pole in the center is only in the center properly segmented. The board on the right side is almost fully segmented along its edges. In the center image, 96 superpixels were used for the segmentation. With this amount the same observation is done regarding the pole on the left and the pole in the center. The board to the right is segmented along most of its edges, similar as when using 48 superpixels. The second image from the right shows the segmentation using 192 superpixels. Using this number of superpixels all three obstacles are are segmented along their edges, indicating that when combining certain superpixels a full obstacle can be identified. In the right image, 240 superpixels are used for the segmentation. again all three obstacles are segmented along their edges, and even smaller regions on the background are segmented better. Major difference between this segmentation and the previous discussed one is minimal in respect to the three main obstacles present. For this scene it is concluded that the best detection is done using 192 superpixels.

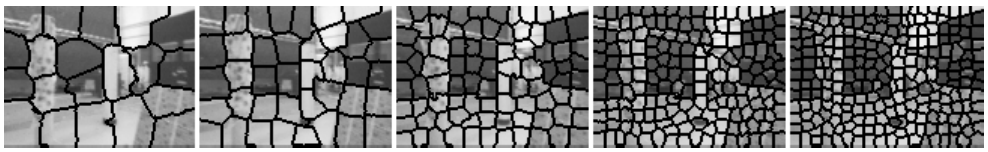


Figure 5-4: SLIC0 segmentation with increasing number of superpixels, fltr 24, 48, 96, 192, 240

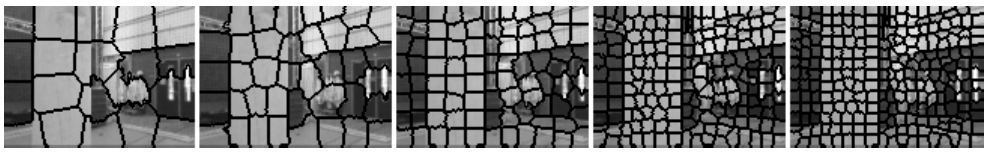


Figure 5-5: SLIC0 segmentation with increasing number of superpixels, fltr 24, 48, 96, 192, 240

In Figure 5-5 it can be seen in the left image that using 24 segments leads to under-segmentation and the one obstacle is not detected properly. In the second image from the left,

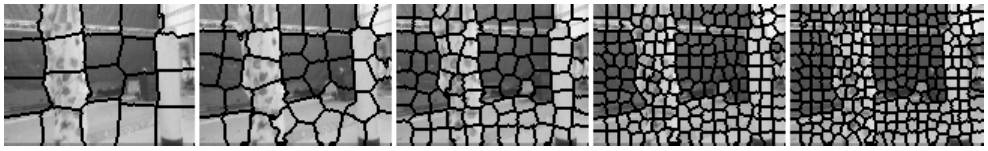


Figure 5-6: SLIC0 segmentation with increasing number of superpixels, fltr 24, 48, 96, 192, 240

48 superpixels are used. Again it can be seen quickly that the superpixels do not border the right edge of the pole and thus the more superpixels is desired. In the center image 96 superpixels are used for the segmentation. Using this segmentation the entire obstacle is segmented along its edges. Both right images showing the 192 and 240 superpixel segmentations, show over-segmentation in terms of edge following of the obstacles. For this scene it is concluded that the best detection is done using 96 superpixels.

In Figure 5-6 it can be seen in the left image that using 24 segments allows for the successful identification of the left pole but not the right pole of smaller size. In the second image from the left, 48 superpixels are used for the segmentation. In this case the edges of both poles are not followed by superpixel boundaries in a satisfying way, indicating that the segmentation using only 24 superpixels was only successful due to properly positioned cluster centers. In the middle image both of the obstacles are identified as the edges are followed by superpixel boundaries. Using 192 and 240 superpixels in this scene only leads to over-segmentation without higher quality boundary following superpixels. For this scene it is concluded that the best detection is done using 96 superpixels.

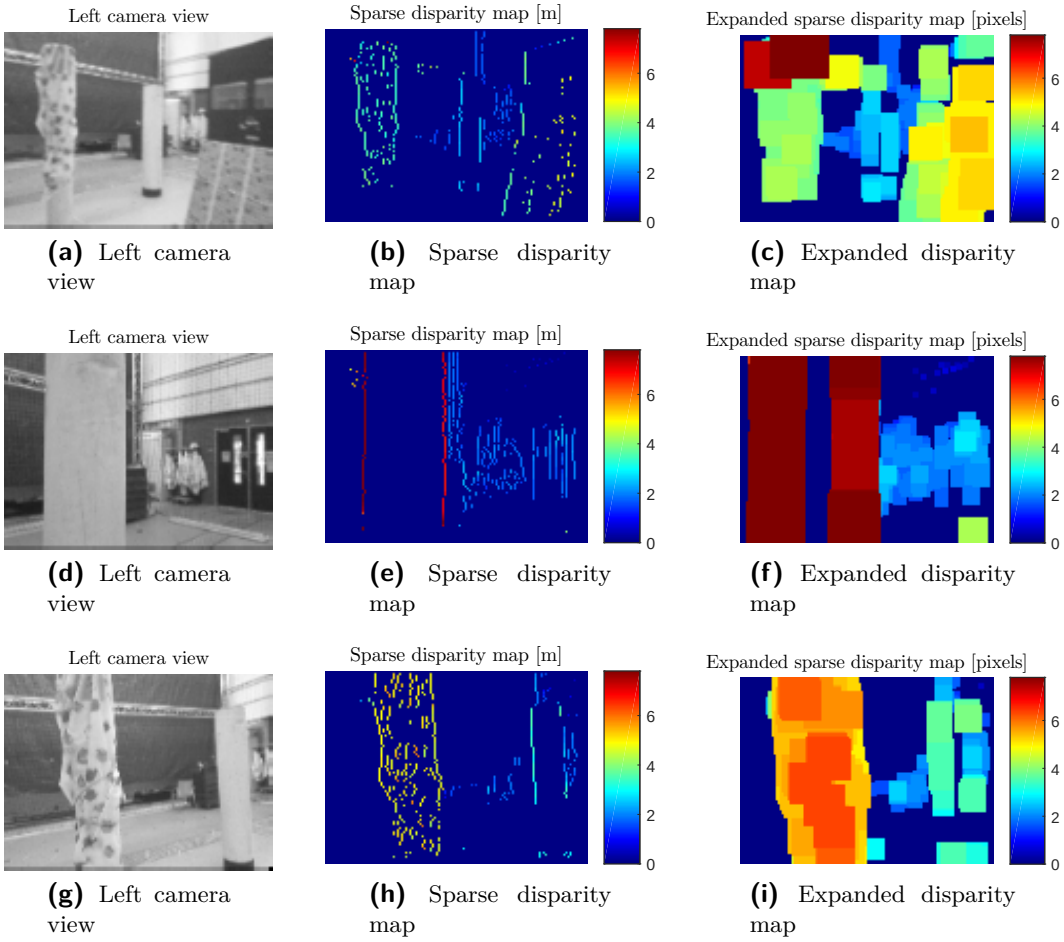
In order to determine the number of preferred superpixels an extra interest should go to the spatial size a superpixel projects into the 3D world. It could be interesting to investigate if superpixels are able to identify regions which indicate a safe flight trajectory. More about this idea will be discussed in Section 5-4.

5-3 Configuration Space Expansion

In this section the configuration space expansion approach as presented in Section 2-3-3, is applied to the space disparity maps obtained from the stereo-board. In the original paper Brockers et. al. applied the C-Space expansion to a dense disparity map, allowing them to consider the MAV as a point mass system and with the use of the egocylinderical representation allow for image search based obstacle avoiding trajectory planning.

In Figure 5-7 the scenes are shown in the left column, and the corresponding sparse disparity map in the middle column. In the right column the expanded disparity map is shown on a scale that corresponds to a proximity scale of infinity to 1 meter, where hot represents a close proximity and cold represents a large distance.

While examining Figure 5-7c it immediately becomes clear that the disparity map contains outlier values, representing wrong disparity values. This can be seen by the dark red expansions in the top of the image. All other disparity values of the pole's location are in the light green region, representing a smaller disparity value than the outlier. The same can be found in Figure 5-7f. In this figure a outlier is expanded in the bottom right corner, representing a disparity value of around 4. In Figure 5-7i no outliers are found allowing for the conclusion



(a) Left camera view **(b)** Sparse disparity map **(c)** Expanded disparity map

(d) Left camera view **(e)** Sparse disparity map **(f)** Expanded disparity map

(g) Left camera view **(h)** Sparse disparity map **(i)** Expanded disparity map

Figure 5-7: In the left column the left camera view is shown. The corresponding disparity maps are shown in the middle column, and the C-Space expansion is shown in the right column. Warmer colours correspond to objects in closer proximity.

to be made that the presence of outliers produced by the sparse disparity map algorithm is depended on the scene. In Section 5-3-1 the consequences of outliers are examined in more detail.

Besides outliers, it can be seen after inspecting the expanded disparity maps in Figure 5-7 that the configuration space expansion is an effective approach to indicate obstacles. In Figure 5-7c the left most pole is, when ignoring the outliers, labelled properly with disparity values. Also the board in the right side of the image is filled with disparity values. The center pole is identifiable by two expanded edges, this suggests that the configuration space expansion is not solely capable of solving the uncertainty that come with the sparseness of the disparity map. Additional depth information is required from this featureless zone, or avoiding it in total would be an alternative.

In Figure 5-7f this suggestion is confirmed. The large and mostly textureless pole in on the left side of the image is only able to provide disparity values at the edges. Even after expansion there is a large region left which does not contain any depth information which could indicate

if this area is an obstacle of potential funnel to fly through.

In Figure 5-7i the left pole is successfully expanded without outliers or regions without disparity information. The pole on the right which is also mostly textureless suggest that a region without depth information will be created between the edges, once the MAV approaches the pole.

From the three scenes it becomes clear that the configuration space expansion is a efficient and very effective tool provide extra depth information. Objects which are rich in texture will be completely filled with depth information, but objects with little to no texture will still give problems when they are approached. In Section 5-4 the the results will be discussed in more detail.

5-3-1 Outlier detection

In the previous section it was found that the sparse disparity algorithm (Tijmons, Croon, Remes, De Wagter, & Mulder, 2016) can produce outliers in the sparse disparity map. This cause faulty expansions, which becomes problematic for high disparity values which are expanded over a large image region.

In order to obtain a better idea of the number of outliers and their values compared to the other values, for each of the three scenes a histogram is constructed for the number of found disparity values. The three histograms are presented in Figure 5-8.

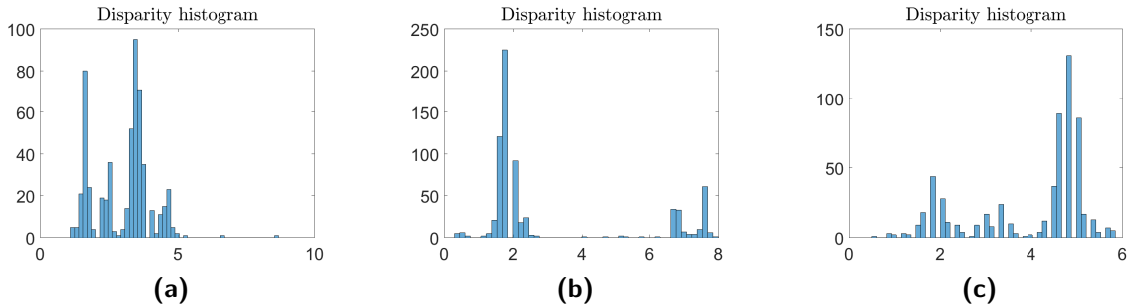


Figure 5-8: Histograms of the disparities found in the three different scenes.

In Figure 5-8a the histogram of the disparity map of scene one is shown (see Figure 5-7a). Form the histogram is becomes clear that the closest obstacle is represented by disparity values between 4.3 and 5. Three outliers are detected at disparity values of 5.2, 6.7 and 8.8.

In Figure 5-8b the histogram of the disparity map of scene one is shown (see Figure 5-7d). From the histogram is does not become very clear how many outliers are present as one obstacle is found at disparity values between 6.5 and 8. The background is found at disparity values of 1.4 and 2.2. The disparity values in between contain a outlier, but solely based on this histogram these values could represent floor or ceiling features for instance.

In Figure 5-8c the histogram of the disparity map of scene one is shown (see Figure 5-7g). From the histogram no outliers can be detected as multiple disparity values are found for each bin. This observation corresponds to the Expanded sparse disparity map, shown in Figure 5-7i.

The histograms provide a indication in the amount of false disparity values, caused by miss-matches of the block-matching algorithm. In the next section these results are discussed in more detail.

5-4 Preliminary Results and Discussion

In Section 5-1 it was found that the sparse disparity map contains large regions without depth information. These regions can represent a distant scenery but also a textureless obstacle, and without additional processing the distinction cannot be made. If objects can be identified with a large degree of certainty, textureless regions can be assumed to be distant scenery allowing for image-space based obstacle avoidance. Thus an effort has to be made in object detection using, for instance, extracting texture information with the use of computational efficient superpixels.

In Section 5-2-1 it was found that the SLIC method is able to run at real-time frequencies for different numbers of superpixels. Because of the relative simplicity of the SLIC algorithm it was chosen for implementation to verify if superpixel calculation methods could run real-time. During simulations very high frequencies were achieved which provides a basis to conclude that superpixel calculations could be implemented in an effort to extract texture information from the images.

In Section 5-2-2 it was found that when a sufficient amount of superpixels were used, the quality decrease due to the use of gray-scale instead of CIELAB colour space insignificant. In Section 5-2-3 three different scenes, from the stereo-board, were segmented using the SLIC algorithm to asses the ability to create superpixels that consist of uniform texture. By varying the number of superpixels, it was found that obstacles can distinguished from distant scenery by using at least 96 superpixels. In order to distinguish obstacles further away from the MAV more superpixels are needed, using 192 superpixels provide satisfying results. As mentioned in the Section 5-2-3 it could be of special interest to investigate is the size of the superpixels can be used to identify funnels which indicate a safe flight trajectory. When the she size of the MAVs bounding box is projected into image-space, its dimensions would depend on the depth in image-space on which the bounding box is projected. Disparity values near a superpixel which could be such a funnel can be taken as a reference for which depth the bounding box should be projected. When such a bounding box is smaller than such superpixel, and the superpixel is classified as distant scenery, this superpixel can be considered as a safe funnel for flight.

In Section 5-3 the configuration space expansion of the sparse disparity map is examined. The C-Space expansion clearly shows good performance in terms identifying highly textured obstacles. But a sensitivity to textureless regions is also identified, indicating that C-Space expansion alone does not solve the problem with identifying textureless obstacles. Also the impact of outliers in the sparse disparity map is made visible, high disparity values are heavily expanded covering large portions of the image with false information. Therefore in Section 5-3-1 the outliers in the sparse disparity map are examined using histograms. It can be seen that not in all sceneries outliers are found in the same way. In the first two scenes outliers could be identified as they represented lone bins in the histogram, largely separated from the other filled bins. In the third scene however a wide spread of disparity values was

found, making the detection of outliers in the histogram impossible. In order to cope with the outliers, pre-filtering is could help but this would also compromise the total number of disparity values that can be found. Using temporal information could help to identify outliers, for instance estimating the probability of a disparity value based on disparity histograms of previous frames. Detecting outliers with a very high disparity can be done by examining only the highest disparity values. Setting a threshold on the gradient of the sorted top 5 to 10 percent of the values, will allow for the detection of large disparity values, for which only a very limited amount if found. This approach relies on the assumption that an obstacle with a close proximity will have many features, whereas large outliers will come in very limited numbers.

5-5 Proposed Research Focus

In Chapter 2 a broad literature review was presented of the latest achievements regarding real-time trajectory planning, object detection and image space representation. Subsequently in Chapter 5 the performance of these methods was investigated. This section will focus the research into a proposed approach based on these results.

As previously described the main purpose of this Thesis is to contribute to the low-level vision-based obstacle avoidance methods developed at MAVLab. Throughout the Thesis work the intention is to find solutions suitable to be implemented fully-on-board a MAV and preferably on the in-house-build stereo-board of the MAVLab. This restricts us to the use of a efficiently computed sparse disparity map.

In section 5-3 it was found that the sparse disparity map may contain outliers which are of major impact on the depth perception when expanded in image-space. In order to develop a novel algorithm capable of identifying free-space trajectories in highly cluttered environments, a solution has to be found to prevent, cope or filter out these outliers.

In an effort to distinguish textureless objects from background scenery superpixels are considered as an additional source of information. In section 5-2 it was shown that SLIC is able to create a superpixel segmentation that respects obstacle boundaries, at real-time. When superpixels prove to provide essential information in order to distinguish textureless obstacles from the background, the computationally more efficient but more complex SEEDS algorithm will be implemented.

From Chapter 2 it was found that inverse-distance image-space is a computationally efficient representation for obstacle avoidance methods. By using image-space expansion, the MAV can be considered a point-mass-system and obstacle avoidance becomes a 2D search problem. In section 5-3 look-up tables where used to perform this computationally efficient expansion and the results were promising. The only drawback is its sensitivity to disparity outliers, and textureless regions.

For future research the following is proposed; conduct further research in implementing image-space expansion in combination with obstacle segmentation. And investigate the adaptation of egocylindrical representation for efficient obstacle avoidance by means of 2D image searches. In the next subsection a new research question is formulated together with six sub-questions, to guide possible future research.

5-5-1 Research Questions

In the previous section a proposed research focus is described. Based on this a new research question is formulated together with several sub-questions, in an effort to reach the research objective using the chosen approach. The proposed research question is:

> *How can image-space expansion be used to computationally efficient identify free-space trajectories, regardless of the sparseness of the disparity map?*

To structure the research with the aim of answering this question, several sub-questions are formulated:

- Which method can be used to prevent, cope or filter out outliers found in the disparity map?
- How to process low-texture regions in image-space, in order to classify it as free-space or obstructed.
- Can superpixels be used to distinguish textureless obstacles from distant scenery?
- Can the SEEDS algorithm be used to provide superpixels of equal or of higher quality compared to SLIC, at a fraction of the computational cost?
- How can the proximity of a detected obstacle be determined when various differing disparity values are assigned to the same obstacle?
- Can egocylindrical image-space representation assist in developing a computationally more efficient obstacle avoidance approach?

Part IV

Sparse Sensing based Depth Reconstruction

Chapter 6

Preliminary Depth Reconstruction Problem Analysis

Abstract

The reconstruction of dense depth maps is of great value to resource-constrained MAVs, in the pursuit of achieving autonomous flight with a high situational awareness. Most MAVs implement sensing methods which provide a sparse depth map, limiting their capabilities significantly. This preliminary analysis will assess the feasibility and performance of using a novel depth reconstruction algorithm in combination with a computationally constrained stereo-camera system. The stereo-camera is designed for flapping wing platforms which operate indoors due to their sensitivity to turbulent air. The algorithm complements this indoor environment in terms that it leverage the regularity of indoor environment to reconstruct dense depth maps. Performance of the method on synthetic data using randomly distributed,- and edge-samples, shows that the method performs best with edge-samples. Before the method is applied with the stereo-camera, a neighbourhood search based outlier removal approach is developed. The large improvement in robustness which this approach brought is overshadowed by the even better performing weighted constraints and weighted recursive approaches which are introduced in this part. The recursive weighted approach shows exceptional potential in providing robust geometry reconstruction using temporal information.

6-1 Introduction

In this part the feasibility is studied of implementing a sparse sensing depth reconstruction method by (Ma, Carlone, Ayaz, & Karaman, 2016) on a computationally constrained stereo-camera system.

Lightweight MAVs are highly constrained in terms of weight, power and computational budgets. Most of these vehicles are therefore equipped with sensors which only provide sparse

information about the environment. In a recent publication by Ma, Carlone, Ayaz and Karaman (Ma et al., 2016), geometric information of the environment is leveraged for the reconstruction of the full environment from a highly sparse depth map. By assuming operation in man-made environments the assumption is made that the environment shows a high level of regularity in terms of flat surfaces with few edges. Ma et. al. showed high quality robust depth reconstruction results on synthetic datasets and real datasets in which the assumption of a structured environment is violated. These results provide a basis for an investigation to the feasibility of implementing this depth reconstruction method on a highly resource-constrained stereo-camera system.

Because the sparse depth maps from the stereo-camera can contain outlier values, a computationally efficient neighbourhood search approach is developed to identify outliers. By pre-filtering the sparse depth map, the robustness of the entire reconstruction approach is improved.

In contrast to the approach of Ma et. al. where no distinction is made between samples with a high confidence and uncertain samples, in this report the use of weighted constraints is proposed. By weighting the constraints based on a confidence measure the overall reconstruction method becomes robust against outliers, without the need of a pre-filter, e.g. the neighbourhood search method.

In an attempt to improve the robustness of the method and the overall quality of the reconstructed geometries, the use of recursive samples is proposed. By sub-sampling a previous reconstruction and merging these samples with a current sparse map it is expected that the quality improves and becomes more tolerant to faulty samples.

In section 6-2 the reconstruction method of Ma et. al. is explained and how the method can be implemented on a computer using MATLAB[1] together with the CVX/MOSEK package for specifying and solving convex programs (Grant & Boyd, 2014, 2008). In section 6-3 the performance on synthetic data is tested first for two dimensional data and subsequently for three dimensional data. In section 6-4 the results of the feasibility study are discussed and phenomena are explained. Lastly in section 6-5 conclusions are made about the performance of the method and the potential for implementation on a computationally constrained MAV platform.

In Section 6-6 a computationally efficient neighbourhood search method is proposed, followed by a image space mean filter in an attempt to remove outliers. In Section 6-7 the use of weighted constraints is proposed to provide an alternative to using a pre-filter and still provide robustness against outliers. In Section 6-9 the weighted constraints approach is extended with a method of adding samples of previous reconstruction to the current sparse map, before reconstructing it. The necessary steps include estimating the pixel shift using optical flow, sub-sampling the previous reconstruction and allocating the appropriate weights to the different samples. The results of this recursive depth reconstruction are given in Section 6-10. The report is finalized with a discussion in Section 6-11 and the conclusion in Section 6-12.

[1] https://nl.mathworks.com/

6-2 Sparse Sensing Depth Reconstruction

In effort to enable lightweight MAVs to autonomously navigate using sparse sensing, Ma et. al. focused on reconstructing the dense depth map from the sparse input data. In a recent publication, Ma et. al. they set the objective to formulate theoretical conditions for which a dense depth map can be reconstructed based on a highly sparse depth map, and develop the algorithms to verify their theories (Ma et al., 2016). By assuming operation in highly structured environments, the geometric regularity and sparseness of edges can be leveraged for the reconstruction of the dense depth map.

The theoretical basis of the reconstruction builds upon earlier work in the field of compressive sensing, where it was proven that a dataset z can be completely recovered from a sparse subset y given $y \in z$ (Foucart & Rauhut, 2013). A convencial model in the field is the *synthesis model*. It assumes that the dataset z is sparse given $z = D\alpha$ where the vector α is sparse in the domain of the matrix D. In more recent work a slight different representation in the form of the *cosparsity model* is proposed, where the vector z becomes sparse after multiplying it with a given matrix D, i.e. Dz, where z is the dataset and D is a given matrix (Nam, Davies, Elad, & Gribonval, 2013; Kabanava & Rauhut, 2015). Ma et. al. found out that the ℓ_0-norm of the 2^{nd} order difference of the depth map can be used as a objective function to enforce the regularity assumption in the environment. By relaxing and reformulating the problem to a ℓ_1-norm problem, it becomes convex and fits the *cosparsity model*, allowing for a full reconstruction. In the following sections the used notation and the algorithm is explained in detail.

6-2-1 Notations

In this report the same notations are used as in the paper of Ma et. al. (Ma et al., 2016). For matrices the upper case will be used, e.g. A, D, and for scalars and vectors lower case letters e.g. z, y are used. Subsets are represented with calligraphic font, e.g. $\mathcal{M}$. The subset $\mathcal{M}$ of vector $z \in \mathbb{R}^n$ is denoted as $z_{\mathcal{M}}$. Indicating a subset $\mathcal{M}$ of a matrix D is done as $D_{\mathcal{M}}$, which represents the rows in subset $\mathcal{M}$ in matrix D. The following norms are widely used, (ℓ_∞-norm): $\|z\|_\infty = \mathtt{max}_{i=1,\dots,n}|z_i|$, ($\ell_0$-norm): $\|z\|_0 = |\mathtt{supp}\,(z)\,|$, and the ($\ell_1$-norm): $\|z\|_1 = \sum_{i=1,\dots,n}|z_i|$. It is important to recognise that the ℓ_0-norm corresponds to the number of none-zero elements in z. The depth reconstruction is based on the use of the *cosparsity model* Dz where the *analysis operator* D produces a sparse vector i,e. given $z \in \mathbb{R}^n$ and $D \in \mathbb{R}^{p\times n}$ we will have $\|Dz\|_0 \ll p$.

In the next section the the algorithm is explained.

6-2-2 Depth Reconstruction

In order to reconstruct the dense depth map, it is assumed that sparse depth information is measured by sensing equipment. Lets define y as the measurement vector, $z^\diamond \in \mathbb{R}^n$ the depth map, and A the selection matrix. Then the measurements in y are found with $y = Az^\diamond + \eta$ with $A = \mathbf{I}_{\mathcal{M}}$, where $\mathbf{I}_{\mathcal{M}}$ is the identity matrix with ones on the rows from subset $\mathcal{M}$. Therefore it can clearly be seen that $Az = z_{\mathcal{M}}$.

The assumption of operating in a structured environment means that the depth map shows a lot of regularity, i.e. the changes of the slope are mostly zero throughout the depth map. Such a change in slope for a measurement point is formulated as $\frac{\delta z_i}{\delta x_i} - \frac{\delta z_{i-1}}{\delta x_{i-1}}$ which can be described as $\frac{z_{i+1}-z_i}{x_{i+1}-x_i} - \frac{z_i-z_{i-1}}{x_i-x_{i-1}}$, where we can assume $x_i - x_{i-1} = 1$, resulting in the second order derivative of z_i expressed as $z_{i+1} - 2z_i - z_{i-1}$.

It can be seen that the corner set $\mathcal{C}$ consists of indices for which $z_{i+1} - 2z_i - z_{i-1} \neq 0$. Keep in mind that with few corners in the environment $Dz^\diamond$ will be sparse. Defining matrix D as a *$\mathcal{Z}^{nd}$-order difference operator* (see equation 6-1) gives us the important equation $\|Dz^\diamond\|_0 = |\mathcal{C}|$.

$$D \doteq \begin{bmatrix} 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -2 & 1 \end{bmatrix} \in \mathbb{R}^{(n-2)\times n} \tag{6-1}$$

Now by levering the regularity in the environment, the full depth profile $z^\diamond$ can be reconstructed by solving the following optimization problem:

$$\min_z \|Dz\|_0 \qquad \text{subject to} \qquad Az = y \tag{6-2}$$

This noiseless optimization problem will force the depth profile to be consistent with the sparse measurements y, and minimises the number of corners, recall that $\|Dz^\diamond\|_0 = |\mathcal{C}|$. In order to allow for measurement noise and reformulate the problem into a linear programming problem the following relaxation is applied:

$$\min_z \|Dz\|_1 \qquad \text{subject to} \qquad \|Az - y\|_\infty \leq \epsilon \tag{6-3}$$

Note that for this ℓ_1-minimization problem it is assumed that the noise is bounded $\|\eta\|_\infty \leq \epsilon$. This is important for determining the tolerance for a specific application.

The optimization problems expressed in Equations 6-2 and 6-3 allow us to reconstruct 2-dimensional depth maps, for the three dimensional case a we need to introduce a second *$\mathcal{Z}^{nd}$-order difference operator*.

For 3-dimensional depth reconstruction of $Z^\diamond \in \mathbb{R}^{r\times c}$ a operator, D_H, will be assigned to to the horizontal differences, and a difference operator, D_V, will be assigned to the vertical differences. The layout of the operators is identical as expressed in Equation 6-1, but with the appropriate sizes. Given depth map $Z^\diamond \in \mathbb{R}^{r\times c}$, we get $D_V \in \mathbb{R}^{(r-2)\times r}$ and $D_H \in \mathbb{R}^{(c-2)\times c}$. The corners are now encoded with $D_V Z^\diamond \in \mathbb{R}^{(r-2)\times c}$ and $Z^\diamond D_H^{\mathrm{T}} \in \mathbb{R}^{r\times(c-2)}$, Thus the ℓ_1-minimization now becomes:

$$\min_Z \qquad \|\mathrm{vec}(D_V Z)\|_1 + \|\mathrm{vec}(Z D_H^{\mathrm{T}})\|_1 \qquad \text{subject to} \qquad Z_{i,j} = y_{i,j} \tag{6-4}$$

where $y_{i,j}$ is the sparse measurement map, Z the reconstructed depth map and $\mathrm{vec}(M)$ is the column wise vectorization of matrix M.

The next step is to reformulate the optimization in Equation 6-4 to correspond to the 2-dimensional case, the result is as follows:

$$\min_z \|\Delta z\|_1 \qquad \text{subject to} \qquad Az = y \tag{6-5}$$

where we define $n = r \times c$, and $z = \text{vec}(Z) \in \mathbb{R}^n$ and the measurements are stored in $y \in \mathbb{R}^m$. The new matrix Δ is called the *Regularization matrix*, which is defined as follows:

$$\Delta = \begin{bmatrix} \mathbf{I}_c \otimes D_V \\ D_H \otimes \mathbf{I}_r \end{bmatrix} \tag{6-6}$$

where $\mathbf{I}_c$ is the identity matrix of size c, and $\otimes$ is the Kronecker product. The case where noise is present in the measurement, it is assumed the noise is bounded according to $\|\eta\|_\infty \leq \epsilon$. This case is shown below:

$$\min_z \|\Delta z\|_1 \qquad \text{subject to} \qquad \|Az - y\|_\infty \leq \epsilon \tag{6-7}$$

In the next section it will be shown how the problems as formulated in Equations 6-2, 6-3, 6-5 and 6-7 can be implemented in MATLAB.

6-2-3 Matlab Implementation

In this section the practical implementation of the optimization problems as described in the previous section, is given. As mentioned before to solve the convex optimization problems a commercial MATLAB package called CVX/Mosek is used (Grant & Boyd, 2014, 2008).

The first step is to download the CVX/Mosek package[2], and obtain a academic or commercial licence to use the package. To install the package it is reccomanded to follow the instructions which provided on the same website[3]. In short the following steps are performed, first change the working directory to the folder where the CVX package is stored. Second the command `cvx_setup` will install the CVX package and integrate the solver in the current MATLAB installation. Finally the licence file has to be installed using the command `cvx_setup C://cvx_license.dat`. The three main steps are shown below.

```
1 cd C:\personal\cvx
  cvx_setup
  cvx_setup C:\cvx_license.dat
```

After the package is installed successfully, it is time to look at the usage in terms of it's programming environment. An extensive user's guide can be found online[4] and is highly recommended for any new user.

The 2^{nd}-order difference operators D_V and D_H can be constructed in the following way:

[2] http://cvxr.com/cvx/doc/mosek.html
[3] http://cvxr.com/cvx/doc/install.html
[4] http://cvxr.com/cvx/doc/CVX.pdf

```
% Define the dimension parameters
h = 96;      % height;
w = 128;     % width;

% Construct 2D 2nd-order difference operator, Dv
Dv = zeros(h-2,h);
for i = 2:h-2+1;
    Dv(i-1,i-1:i+1) = [1 -2 1];
end

% Construct 2D 2nd-order difference operator, Dh
Dh = zeros(w-2,w);
for i = 2:w-2+1;
    Dh(i-1,i-1:i+1) = [1 -2 1];
end
```

The next step would be to generate the sparse measurement samples. In this case we generate in line 2 the edge values present in the synthetic dataset y. The selection matrix A is calculated in line 5, and the corresponding sparse measurement samples are selected from y in line 8.

```
% Set the samples to be the edge values
samples_2D = [1; diff(y,2,1)~=0; 1];

% Calculate the selection matrix A
A_samples = diag(samples_2D);

% Calculate the sampled values
y_sampled = A_samples*y;
```

Next the CVX environment can be used to solve the linear optimization problem (a relaxed version of Equation 6-2), which is shown below:

$$\min_{z} \|Dz\|_1 \qquad \text{subject to} \qquad Az = y$$

In the code block below the CVX environment is started in line 2, the addition of quiet suppresses the output of the internal optimization process. Line 3 initiates the variable X of size n which is the 2-dimensional depth map. The cost function is shown in line 4, the one-norm of Dz. Line 5 indicates that the constrains are provided to the CVX environment. Line 6 constrains the measurement samples to remain present in the depth map while regularity in X is enforced in line 4.

```
% Calculate the prediction
cvx_begin quiet
variable X(n);
minimize( norm(D*X,1) );
subject to
A_samples*X == y_sampled;
cvx_end
```

In the next code block the depth map is reconstructed with noisy sparse measurement samples. The optimization problem (Equation 6-3) is repeated below:

$$\min_{z} \|Dz\|_1 \qquad \text{subject to} \qquad \|Az - y\|_\infty \leq \epsilon$$

The code used for this problem is identical as for the noiseless case shown above except for line 6, where the infinity-norm is taken to be less or equal to the tolerance.

```
% Calculate the prediction
cvx_begin quiet
variable X2(n);
minimize( norm(D*X2,1) );
subject to
norm(A_samples*X2 - y_sampled_noisy,inf) <= tol;
cvx_end
```

In the remainder of this section the implementation of the 3-dimensional reconstruction is explained. First the *Regularization Matrix* has to be computed, this is done in the next code block. The *Regularization matrix* is calculated in the following lines. First the identity matrices are predefined in lines 2 and 3, subsequently the Kronecker products are appended in line 4 in order to get the *Regularization matrix*.

```
% Calculate Regularization Matrix
Iy = eye(h);
Ix = eye(w);
Regularization_Matrix = [sparse(kron(Ix,Dv));sparse(kron(Dh,Iy))];
```

The noiseless 3-dimensional optimization problem (Equation 6-5) is repeated below.

$$\min_z \|\Delta z\|_1 \qquad \text{subject to} \qquad Az = y$$

The MATLAB code shown in the following code block is similar to the previously discussed code. Note that the depth map is now initiated by the variable $X2D$ of size $n2D = w \times h$ which is product of the width and height.

```
% Calculate the prediction
cvx_begin quiet
variable X3D(n3D);
minimize( norm(Regularization_Matrix*X3D,1) );
subject to
A_samples*X3D == z_sampled;
cvx_end
```

The noisy 3-dimensional optimization problem (Equation 6-7) is repeated below.

$$\min_z \|\Delta z\|_1 \qquad \text{subject to} \qquad \|Az - y\|_\infty \leq \epsilon$$

The MATLAB code shown in the following code block is similar to the noiseless case shown above. The only difference can be found in line 6 where the regularity of the depth map is enforced by the infinity-norm being less or equal to a given tolerance value.

```
% Calculate the prediction
cvx_begin quiet
variable X3D2(n3D);
minimize( norm(Regularization_Matrix*X3D2,1) );
subject to
norm(A_samples*X3D2 - z_sampled_noisy,Inf) <= tol_3D;
cvx_end
```

In the next section the simulation results are given.

6-3 Reconstruction Results

In order to asses the performance of the reconstruction method four simulations are performed using a synthetic dataset. The first simulation will be a 2-dimensional reconstruction without any added noise. This will be solved using the optimization problem shown in Equation 6-2. The second simulation will be a 2-dimensional reconstruction with added white noise, the optimization problem will be formulated as shown in Equation 6-3. The third simulation will be a 3-dimensional reconstruction without noise, formulated as Equation 6-5. And lastly a 3-dimensional reconstruction with added white noise will be done using the problem formulation as shown in Equation 6-7.

6-3-1 Two Dimensional Depth Reconstruction

In this section the 2-dimensional depth reconstruction method is tested using a synthetic dataset. The dataset is shown in red in Figure 6-1. It is clear that the data consists of six constant slope segments, with indicated in black, seven edge values. Note that the beginning and end of the dataset have been added to the set of edges. No noise is added to the sparse measurement samples, the resulting solution of the optimization problem shown in Equation 6-2 is indicated with the blue line.

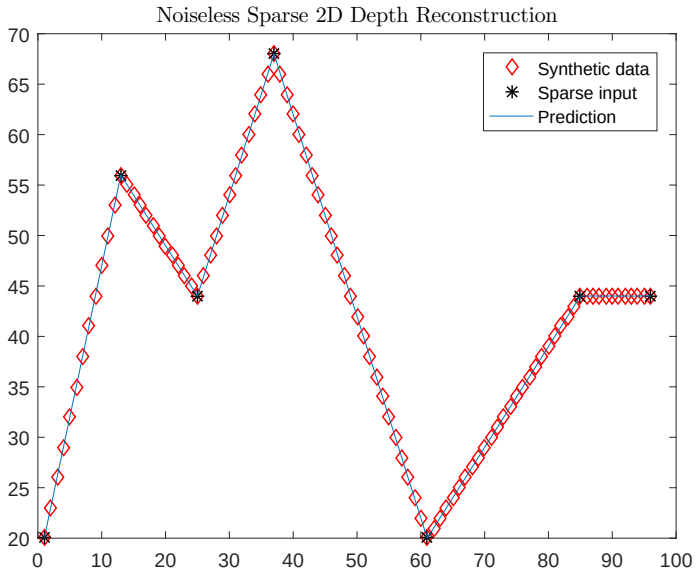


Figure 6-1: 2-Dimensional depth map reconstruction. In red the synthetic ground truth, in black the sparse measurement samples and finally in blue the perfect reconstruction.

From Figure 6-1 is can be seen that without any noise in the sampled measurements the solution reconstructs the synthetic dataset perfectly.

With added white noise ($\sigma = 1$) to the measurement samples the reconstruction differs significantly, which can be shown in Figure 6-2. In red the noiseless synthetic data is shown, whereas in black the sparse samples are indicated which are subject to the added white noise. The tolerance ϵ is taken to be 2.1226 which is chosen as the minimal valid tolerance through the relation $\|\eta\|_\infty = \epsilon$. As can be seen in the figure, the reconstruction, indicated in blue, shows smaller absolute gradient values. This is because minimization of the cost function tries to minimise the change in slope and thus naturally, when given a higher tolerance, will produce a solution with smaller absolute gradient values.

A note has to be made about the samples at the start and end of the dataset. These were added to ensure the first and second segment would not be a linear continuation of the second and second last segment. In the case where there are no samples near the dataset edges, the last segment will be extrapolated towards the dataset boundaries. In the next section this can be observed as random sampling will be shown besides just samples at corner points.

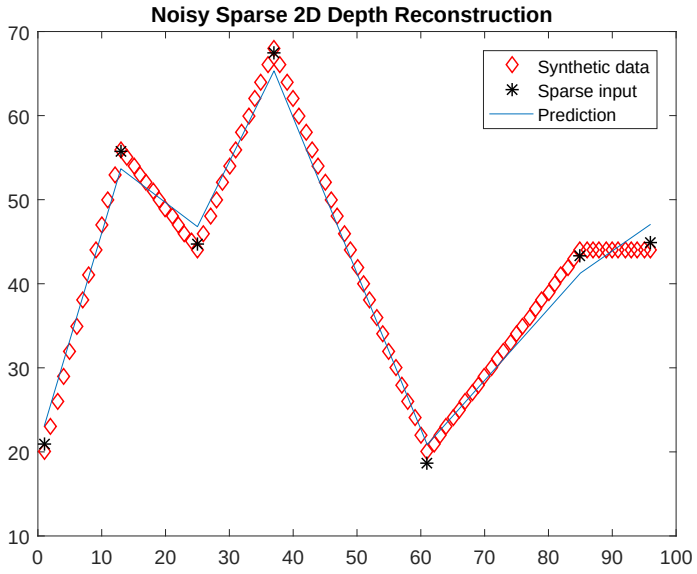


Figure 6-2: 2-Dimensional depth map reconstruction with added white noise ($\sigma = 1$). In red the synthetic ground truth, in black the sparse measurement samples and finally in blue the reconstruction. It is clear that use of a tolerance value allows for a reconstruction with less sharp edges.

6-3-2 Three Dimensional Depth Reconstruction

In this section the 3-dimensional depth reconstruction method is evaluated using the synthetic dataset shown in Figure 6-3. The synthetic depth map consists of various constant gradients along it's x and y axes, forming a regular geometry.

In the next two sections the reconstruction method will be tested using this synthetic depth map. First randomly distributed samples are taken and used for the reconstruction. Followed by a reconstruction using edge samples, in both x and y direction. For both cases first a noiseless case will be tested, after which the depth value of the samples is disturbed by white noise with a variance of 1 ($\sigma = 1$).

Randomly distributed samples

In this section the reconstruction results using 200 randomly distributed samples are given. In Figure 6-4 the samples are indicated with red points, scattered evenly over the reconstruction surface. It can be seen clearly that the reconstruction in Figure 6-4 has less sharp edges relative to the ground truth shown in Figure 6-3. This is because the relaxed objective function is written in such a way that the second order derivative is minimised, this causes a rejection of sudden geometric shape changes. It therefore will act as a smoothing function when it can, this is clearly seen in the lower x-axis region where the depth map shows an attempt to smooth-out an entire edge in x-direction.

In order to asses the accuracy of the reconstruction a histogram is made of the difference with the ground truth. The histogram is shown in Figure 6-5. As can be seen from the

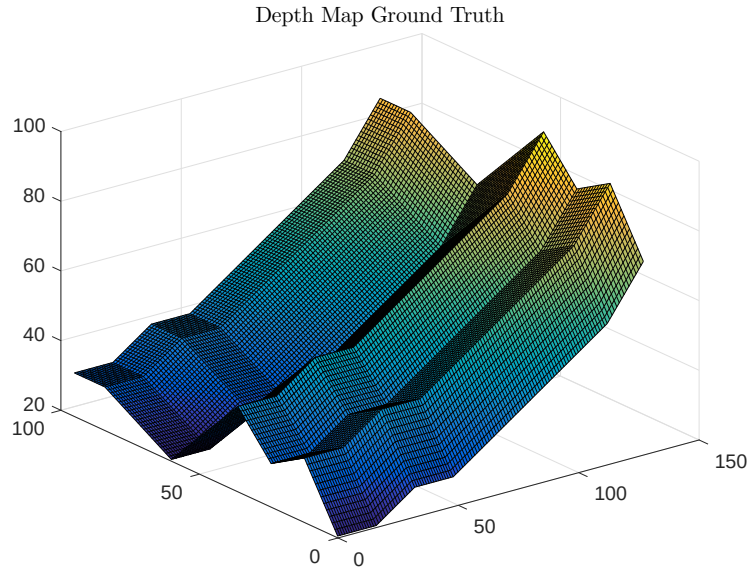


Figure 6-3: Synthetic depth map showing the ground truth for the reconstruction simulations. The map is 128 by 96 pixels and has an arbitrary chosen depth range.

histogram, almost 50% of the reconstructed data-points lay within 0.1 of the ground truth. The histogram shows an extreme steep slope indicating a great reconstruction of the dataset based on the sparse sample-set. The reconstruction error in this case can be explained by the random samples not covering the entire dataset, and thus not all information about the edges and dataset borders are encoded in the samples. Some regions the reconstruction will therefore minimise the second order derivative without being constrained by a data-point.

The next step is to simulate the performance in the scenario where the samples are subject to disturbances. The disturbances of the samples are simulated by adding white noise ($\sigma = 1$) to the depth values. In Figure 6-6 the disturbed samples are indicated with red dots, and the surface represents the reconstructed depth map.

From the figure it is clear that the results seem to be smoothed. Given the nature of the objective function; trying to minimise the sharpness of the edges, this is to be expected. The white noise disturbance causes the samples to represent the ground truth less clearly and certain edges are disregarded completely. To asses the performance of the reconstruction disturbed by white noise, a histogram is made of the differences with the ground truth. The histogram is given in Figure 6-7.

The error histogram shows a clear normal distribution which can be explained by the introduced normal distributed white noise. The white noise does not cause a significant increase in large deviations but does introduce an overall larger error, e.g. the number of data-points with a error of less than 0.1 has gone down with almost 70%. In the next section the reconstruction performance is assessed using instead of randomly distributed samples, edge samples.

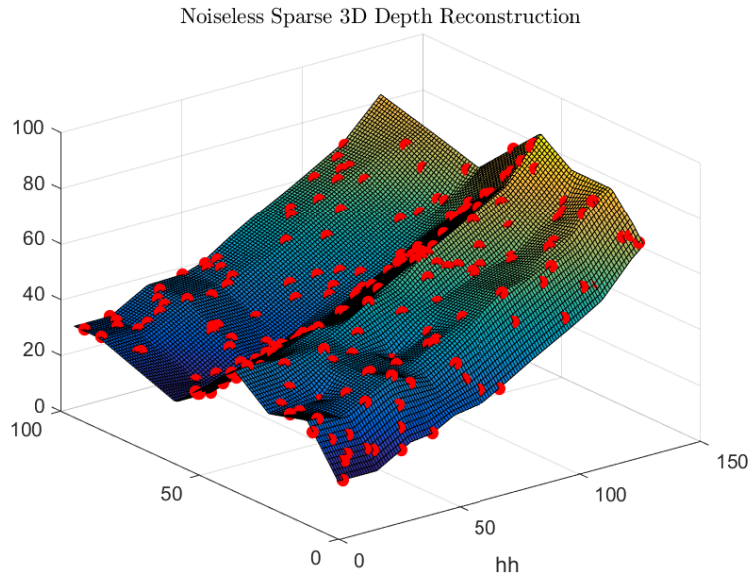


Figure 6-4: 3D reconstruction using 200 randomly distributed samples from the ground truth depth map. The samples are indicated with red dots.

Corner point samples

In this section the results of the reconstruction using corner samples are given. Because most stereo-matching algorithms are able to determine the distance to edge-points easily and most accurate, the assessment of this method using edge-samples is very interesting. First the edges are detected by calculating the 2^{nd} order differences of the dataset. As the synthetic dataset has constant slopes on it's planes, all none-zero 2^{nd} order differences indicate an edge. These samples are selected for the reconstruction.

In Figure 6-8 edge samples are indicated using red dots, the surface is the corresponding depth reconstruction. When comparing the reconstruction with the ground truth in Figure 6-3 it seems that using edge values greatly improves the ability to reconstruct the depth map relative to random sampling. The smoothing characteristic as observed in Figure 6-4 and Figure 6-6 is not present. The only clear deviation is seen at the edges of the dataset where the slope of the previous plane is extended to the dataset edge.

To examine the performance of the reconstruction, again a histogram is made the error of all datapoints. The histgogram is shown in Figure 6-9. In the histogram it becomes clear that the fit is of high quality as almost 50% of the data-points have an error of less than 0.1. Also, as observed in Figure 6-8, the deviations near the dataset edges can be seen in the histogram. A constant error distribution is found between -5 and -25, this indicates that the reconstruction extrapolated the planes near the edge to the edge.

In Figure 6-10 the samples, shown in red, are exposed to white noise ($\sigma = 1$). The edges are still clearly reconstructed but errors are introduced to some extend. Again to get a better view of the errors a histogram of the errors is made and shown in Figure 6-11.

The error histogram clearly shows the influence of the introduced white noise, the data-points

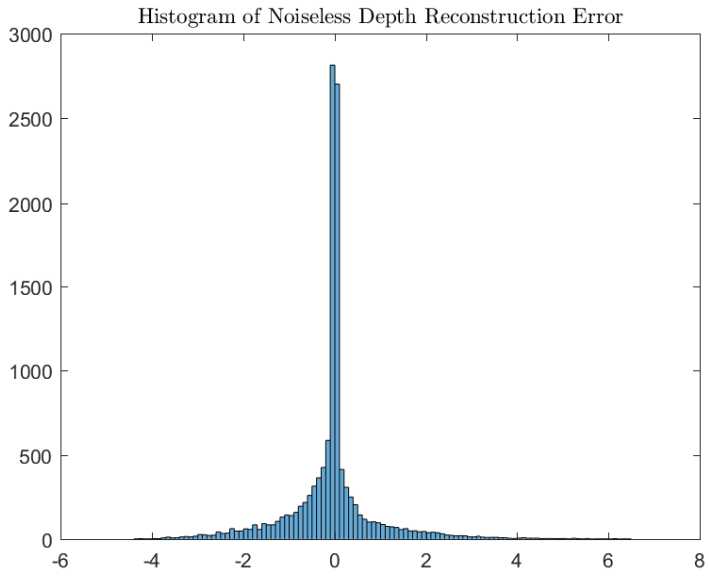


Figure 6-5: Histogram of the differences between the noiseless reconstruction and the ground truth. It is clear that most samples show a minimal error of less than 0.1.

which first had a error of less than 0.1, now are normally distributed around zero. This effect is shown more clearly in Figure 6-12, where the errors values from the dataset boundaries where removed. Although the total error has increased due to the noise, the method is still able to reconstruct the original dataset to a large extend.

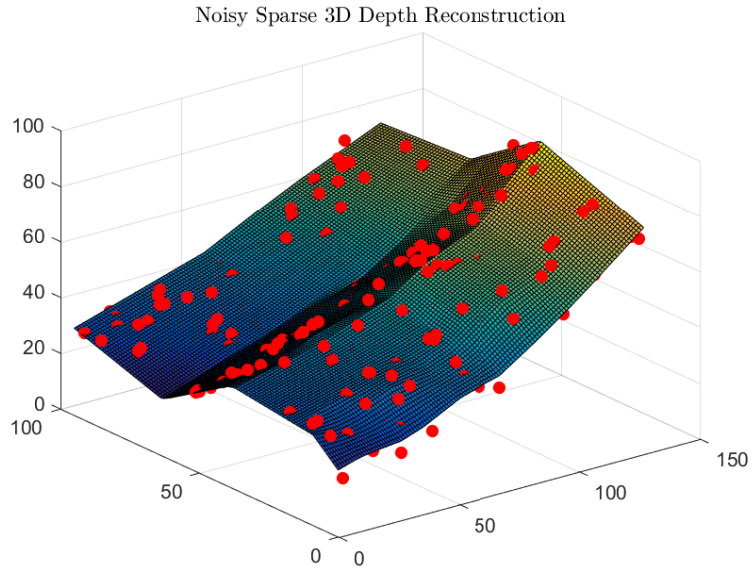


Figure 6-6: 3D reconstruction using 200 randomly distributed samples from the ground truth depth map and disturbed by white noise ($\sigma = 1$). The samples are indicated with red dots.

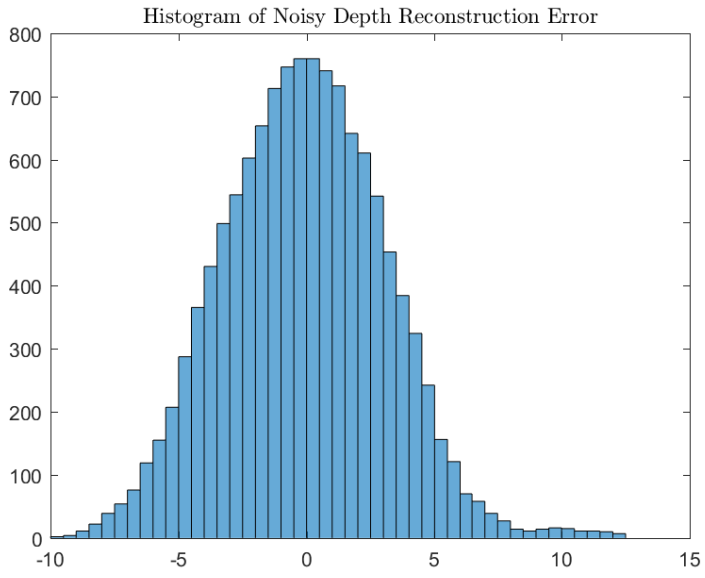


Figure 6-7: Histogram of the differences between the reconstruction and the ground truth. It is clear that the samples show the error is influenced by the introduced white noise ($\sigma = 1$).

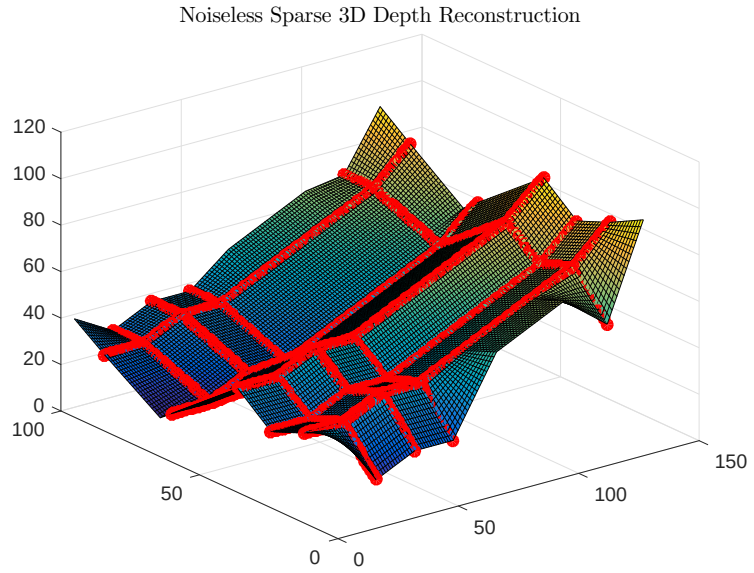


Figure 6-8: 3D reconstruction using edge samples from the ground truth depth map. The samples are indicated with red dots.

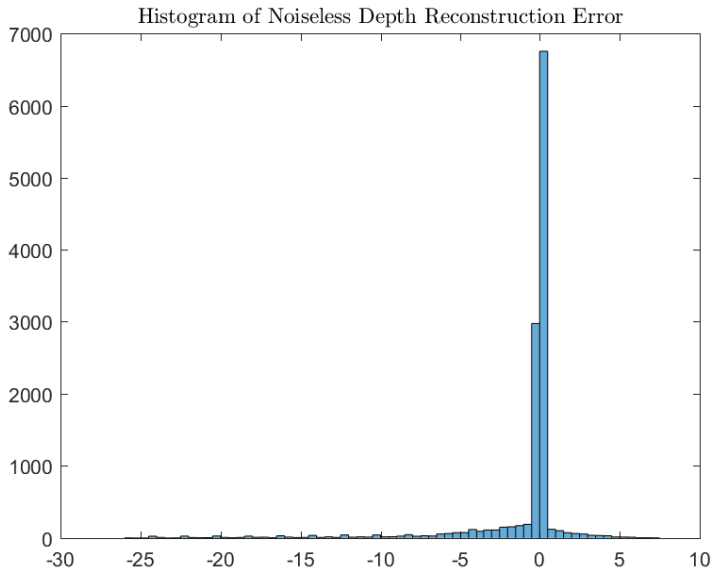


Figure 6-9: Histogram of the differences between the noiseless reconstruction and the ground truth. It is clear that most samples show a minimal error.

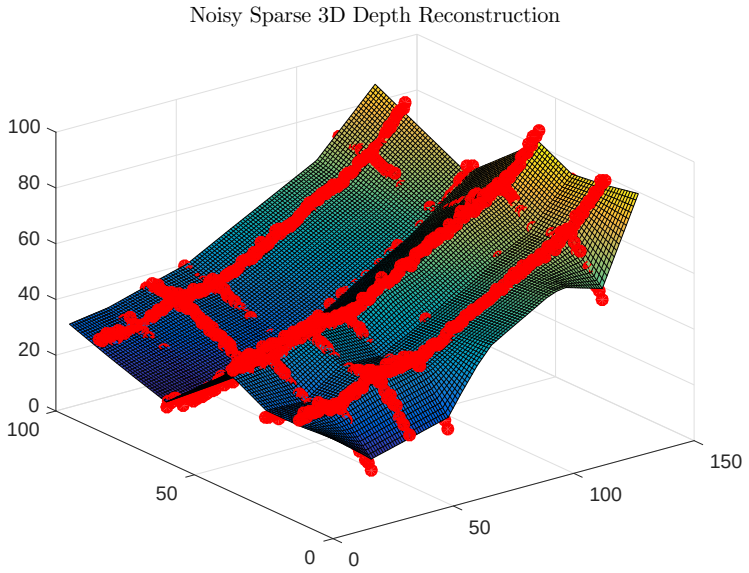


Figure 6-10: 3D reconstruction using edge samples from the ground truth depth map and disturbed by white noise ($\sigma = 1$). The samples are indicated with red dots.

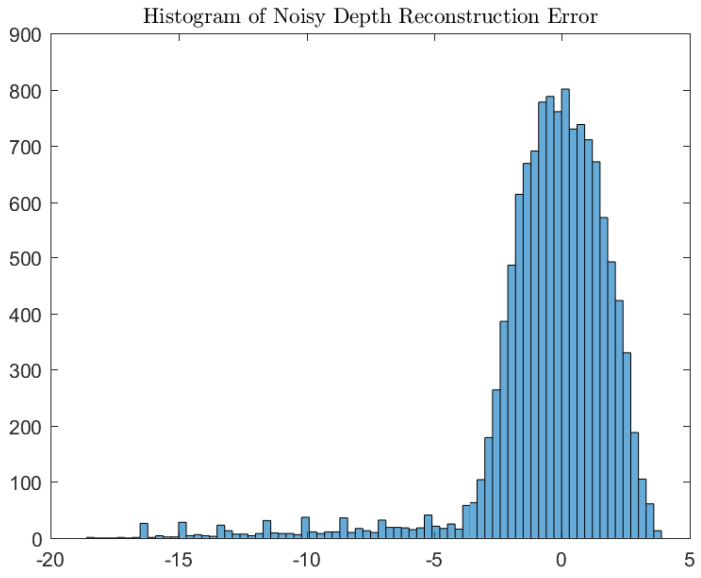


Figure 6-11: Histogram of the differences between the reconstruction and the ground truth. It is clear that the samples show the error is influenced by the introduced white noise ($\sigma = 1$).

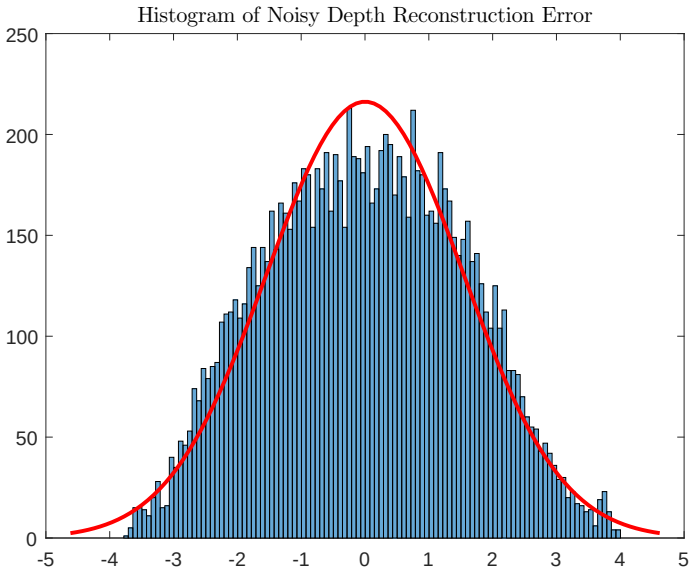


Figure 6-12: Histogram of the differences between the reconstruction and the ground truth. The errors due to deviations near the dataset boundaries have been removed to indicate the influence of the introduced white noise ($\sigma = 1$).

6-4 Intermediate Performance Discussion

In the previous section the results were presented where the sparse depth reconstruction method was tested on different sample sets.

Firstly the set was tested on randomly distributed samples without any noise disturbances (Figure 6-4). This resulted in a reconstruction where a smoothing effect was found on the output data. This can be explained by the way the objective function is formulated. The objective function minimises the sharpness of the depth map, so when samples do not fall directly on an edge that edge will be smoothed out significantly.

The second simulation used the same randomly distributed samples, but now the depth values where disturbed by white noise ($\sigma = 1$) (Figure 6-6). The white noise caused a even stronger smoothing effect and thus sharp edges which are close to each other would be blend together, to minimise the sharpness while the matching error with the samples was kept at a minimum. In regions where several samples where taken on an edge, the reconstruction does show that the edges can be reconstructed properly. This encourages for the use of edge samples instead of randomly distributed ones.

In the third simulation all edge values from the ground truth dataset were used an impressive reconstructed map was produced (Figure 6-8). It is clear that in order to reconstruct a depth map, having samples at the edges is of utmost importance. A new phenomenon was observed this simulation, the lack of samples near the dataset borders causes the planes near the border to extrapolate their gradient all the way to the dataset edge. This is to be expected as the sharpness is minimised, but this causes larger disturbances near the dataset edges. In the corresponding histogram (Figure 6-9) this can be seen by the evenly distributed errors between bin -5 and bin -25.

In the fourth and last simulation the depth values of the edge samples where disturbed by white noise ($\sigma = 1$), and the reconstructed depth map showed signs of disturbances (Figure 6-10). Regardless the reconstruction looks to approximate the ground-truth rather well as the edges are still identifiable. The corresponding histogram of the reconstruction error is shown in Figure 6-11. As could be seen in the simulation with randomly distributed samples, the presents of white noise causes the error distribution to widen around zero. When disregarding the error near the dataset boundary the error shows a symmetric distribution with a zero-mean, this is visualized in Figure 6-12. Note that the bin-size has been reduced in order to show a smoother distribution.

6-5 Intermediate Conclusion

The results presented in the previous section verify the reconstruction capabilities of the method, introduced by Ma et. al. By examining the performances with disturbed samples the robustness of the method could be assessed. While random sampling provides a better approximation at the dataset boundaries, it was the simulation with edge samples that is more capable of reconstructing a regular geometry. This was explained with the sharpness rejecting formulation of the objective function, thus when samples don't sufficiently mark an edge, this edge will be smoothed out.

Because the method works best using edge values, the use of the method in combination with the MAVLab stereo-board seems a natural next step. The stereo-board has already been used with edge detection algorithms but has not been used to construct a high quality dense depth map (Tijmons, Croon, Remes, De Wagter, & Mulder, 2016; McGuire et al., 2016). Modifying and implementing the novel reconstruction method of Ma et. al. would greatly increase the environmental awareness of a MAV equipted with the MAVLab stereo-board camera system.

6-6 Outlier Removal using Neighbourhood Search

Because the sparse depth map, used as input, often contains outliers which will deteriorate the dense reconstruction an attempt will be made to remove them. In this section a neighbourhood search based method will be discussed.

Due to errors in the stereo-matching procedure, the sparse depth map often contains wrong disparity values. These values often lay between $d = 0$ and $d = 1$, which correspond, for the MAVLab Stereo-camera, to distances ranging from ≈ 7.8 m to ∞. Because of this, these values tend to have no to very few neighbouring points in 3D space. This observation will be used to identify these points, these outliers and removed from the dataset.

The first step is to calculate the points in vehicle-centred 3D space. This is done using elementary equations;

$$z = \frac{fb}{d} \qquad x = \frac{zu}{f} \qquad y = \frac{zv}{f}$$

where z is the distance from the camera, x the distance along the horizontal axis, y the distance along the vertical axis, u and v the horizontal and vertical pixel location with the image centre defined as the origin. The result are 3 vectors containing the coordinates of all measured depth values.

The second step is calculating the euclidean distance between the first measurement point and all other points.

$$\bar{d}_1 = \sqrt{(x_1 - \bar{x})^2 + (y_1 - \bar{y})^2 + (z_1 - \bar{z})^2}$$

Next the neighbourhood of the point is defined as a vector of boolean:

$$\bar{N}_1 = \bar{d}_1 < \tau_{neighbourhood}$$

To classify as an outlier, it is set that the number of neighbours $|\bar{N}_1| < 1\%$ of the total number of measurement points. In order to decrease the computational load, all elements in this neighbourhood are also classified as outliers. When the number of neighbours lies between 1% and 5% of the total number of measurements, only the single value under examination is classified as an outliers, but all it's neighbours are not collectively classified as outliers. When the number of neighbours is larger than 5%, the examined value and all it's neighbours are classified as non-outliers and their individual neighbourhoods will not be examined any more. The algorithm is shown in Algorithm 1.

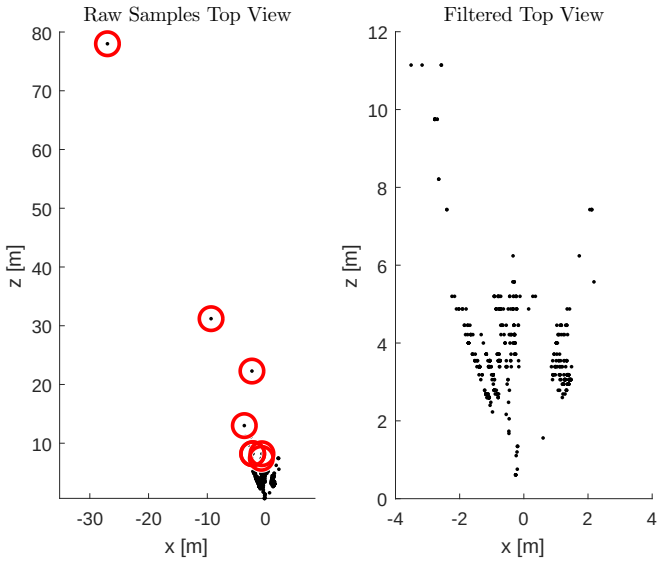


Figure 6-13: Top view of 3D coordinates shown in the xz-plane. On the left the identified outliers are marked with red circles with $\tau_{neighbourhood} = 2$. On the right the remaining samples are shown.

Due to these extra policies it was found empirically that the number of neighbourhoods examined is reduced to maximally 10% of all measurements throughout a test sequence of 1000 images. In Figure 6-13 the effect of the neighbourhood search based outlier removal approach is visualized.

In the left figure the raw samples are visualized in the xz-plane. Visualized with red circles, the outlier neighbourhoods with $\tau_{neighbourhood} = 2$m are shown. The all identified outliers lay far away from the camera, upto 79m from the origin, approximately $d = 0.1$ pixels. Given the used stereo-matching method, small disparity values tend to be falsely matched image block pairs.

After removing the identified outliers, the remaining samples are visualized in the right figure. It can be observed that the maximum distance has reduced from 79m to around 11m. The result is promising in removing extreme values, but several points remain for $z \geq 6$. From extensive testing it is in practice assumed that distance values greater than 6m are not reliable for obstacle avoidance methods. Therefore There still exists a need for a more reliable and effective approach to dealing with the outliers.

The main reconstruction results are given in Section 6-8 and more extensively in the Appendix 7 at the end of this part.

6-6-1 Sparse to Dense Filtering using a Mean-filter

An alternative approach to deal with outliers was developed in the form of a mean-filter. The filter's advantage lays in it's simplicity, the general approach is as follows.

The filter sweeps a window over the map, calculating the mean value of all non-zero samples within the window. The second step is to calculate a lower and upper boundary from this

Algorithm 3: Neighbourhood-Search Outlier Identification and Removal

Initialize a boolean list for all samples idx_{list}
Loop over all samples
for $i = 1$ *to* $n_{samples}$ **do**
    Check if the sample i is in the check list idx_{list}
    if $i \in idx_{list}$ **then**
        Select the sample coordinates and all sample coordinates
        $C_i = \{x_i, y_i, z_i\}$
        $C_{all} = \{x, y, z\}$
        Calculate the distance from sample i to all other samples
        $\bar{d}_{list}\sqrt{\sum (C_i - C_{all})^2}$
        $idx_{matches} = \texttt{find}\left(\bar{d}_{list} <= \tau_{min}\right)$
    Test if index i is in an outlier neighbourhood(1% neighbourhood)
    else if $|idx_{matches} < \textit{max}(2 + 1, c/100)\,|$ **then**
        Add neighbourhood to empty list as it is a outlier
        $idx_{emptyList} = [idx_{emptyList}, idx_{matches}]$
        Remove neighbourhood from check list
        $idx_{list}(idx_{matches}) = false$
    Test if index i has a neighbourhood of $\geq 5\%$ of the dataset
    else if $|idx_{matches} > \textit{max}(2 + 1, c/20)\,|$ **then**
        Neighbourhood is safe so remove neighbours from check list
        $idx_{list}(idx_{matches}) = false$
    else
        Else the sample i is NOT assumed to be an outlier, and
        it's neighbourhood is NOT assumed to be safe.
        $idx_{list}(i) = false$
    end
end

mean value, and assigning the mean value of all samples within the window that fall within the lower and upper boundary to the center position of the window. Because in sparse maps the chance that no samples within the window fall between the boundaries is small, the boundaries get widened and the window is searched for samples again. If no samples are found or when a mean value is assigned the algorithm proceeds to shift the window, sweeping over all points on the map. The method requires multiple iterations, where the starting boundary values are enlarged with each iteration to stimulate sharp edges to be retained and at the same time fill the entire map with values.

A simulation of the approach using 4 iterations is visualized in Figure 6-14. The biggest step is observed in the first iteration, where the map gains a large amount of additional samples. In the consecutive 3 iterations the gaps between highly varying depth regions are closed until in the fourth iteration a fully dense map is obtained.

The approach is proven to have a smoothing effect removing measurement errors, but also makes alterations on the observed geometry as empty image regions are filled based on neigh-

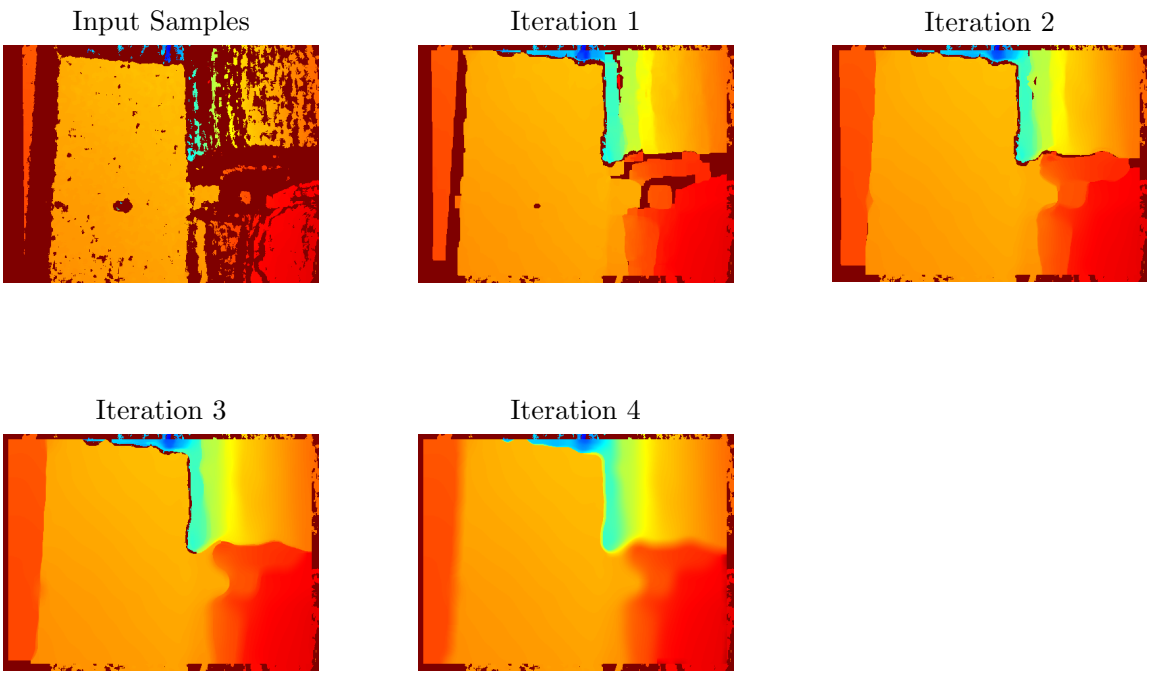


Figure 6-14: Results of the Mean-filter approach, using 4 iterations of varying boundary values. The boundaries are set at 10, 10, 20 and 80 cm for depth values ranging from 0.4 to 5 m.

bouring samples. The current implementation in MATLAB does not show real-time performance and a more direct implementation in C/C++ and using closest neighbour instead of a mean operation are expected to significantly improve the computational speed. For now this filter approach will not be considered for outlier removal or to reconstruct dense depth maps.

6-7 Sparse Sensing Depth Reconstruction using Weighted Samples

This section introduces the use of distance dependent weights for the constraints of the reconstruction optimization problem. This approach will form as an alternative for the neighbourhood search approach, as discussed in the previous section.

The optimization problem and corresponding constraint is given in Equation 6-7, and repeated below.

$$\min_{z} \|\Delta z\|_1 \qquad \text{subject to} \qquad \|Az - y\|_\infty \leq \epsilon$$

The main idea is to, instead of filtering out the outlier values, to minimise their influence on the reconstruction problem. By allocating weights to individual depth measurements, based on their distance, it becomes possible reduce the effect wrong values have on the reconstruction. As mentioned in the previous section, values larger than 6m are in practice regarded as untrustworthy and therefore it is proposed to allocate distance dependent weights to all measurements

The vector w will contain all weights corresponding to measurements y. For values of 2m or less, a weight of 1 is assigned while values of more than 6m are assigned a weight of 0. For measurements between 2m and 6m the weight is linearly interpolated between 1 and 0.

In order to use the weights, the constraint will be adjusted to the following:

$$\min_{z} \|\Delta z\|_1 \qquad \text{subject to} \qquad \|W\left(Az - y\right)\|_1 \leq \epsilon \tag{6-8}$$

where W is $diag\left(w\right)$. Note that the infinity norm is relaxed to the 1-norm reducing the impact of difficult to reconstruct image regions. The main reconstruction results are given in Section 6-8 and more extensively in the Appendix 7 at the end of this part.

6-7-1 Stereo-Matching Confidence Based Weights

In this section it is proposed to use a quality measure of the stereo-matching procedure to base the weights on, which are used for reconstruction.

In the previous section the use of weights to reduce the impact of outliers is proposed as function of the measured distances. Although proven to be effective it makes more sense to use a quality measure of the stereo-matching procedure as the base for the weights. It is expected that outlier values will have barely passed the matching confidence test and should therefore carry a lesser weight than values which are matched with high confidence.

6-8 Pre-filtered and Weighted Reconstruction Results

In this section the reconstruction results are given for the outlier removal approach using neighbourhood search, the weighted constraints approach. In order to evaluate their relative performance the results are presented next to each other enabling a quick comparison.

The results of 4 approaches are given, firstly the standard reconstruction method introduced by Ma et. al. (Ma et al., 2016), by with the constraint as formulated in Equation 6-7, being loosened from a infinity-norm to a 1-norm. The second approach uses the standard reconstruction method but the input samples have been pre-filtered with the outlier removal approach using neighbourhood search. The third approach uses the unfiltered samples in combination with the weighted samples as shown in Equation 6-8. And lastly the fourth approach uses the combination of pre-filtered input samples where the outliers are removed using the neighbourhood search method, and the weighted samples method from Equation 6-8. The first test scene is visualized in Figure 6-15. The corresponding reconstruction results of three consecutive frames are shown in Figure 6-16.

Looking at Figure 6-16, in the first column the sparse depth maps are shown. These are calculated onboard the MAVLab stereo-camera and send to a workstation. Throughout the results dark red will indicate a close proximity while the colder towards dark blue indicates a larger proximity. The results will be discussed per column and from map 1 to map 3.

In the second column the reconstruction results of the standard method of Ma et. al. with loosened constraints are given (Ma et al., 2016). The reconstruction of map 1 is clearly able

Figure 6-15: Photo taken of scene 22 with a High-Definition camera sensor.

to obtain the general geometry of the scene. The corridor is visible and depth values are coherent with the sparse map. Notably in the right hand side a large outlier causes the reconstruction to have a extreme dark blue patch, visualizing a tunnel. This shows that the standard approach is vulnerable to outliers. For map 2, the standard reconstruction is not able to reconstruct the scene at all. Large differences in depth of cause the method to be incapable of reconstructing it with a given tolerance of $\epsilon = 0.1$. For map 3 the standard reconstruction method is able to reconstruct the geometry of the scene properly.

In the third column the pre-filtered reconstruction results are given. The reconstruction of map 1 clearly shows the general geometry of the scene, with a small blue patch on the right side. The reconstruction of map 2 and 3 also clearly recover the general geometry of the scene, with again an indication of vulnerability to outliers on the right hand side.

In the fourth column the weighted approach without outlier removal shows highly similar results as the pre-filtered results in column 3. This indicates that the weighted and the pre-filtered approaches have highly similar performance for this particular scene.

In the fifth and last column, the combined results of the pre-filtered and the weighted constraints approaches are given. Because the pre-filtered and the weighted approach already perform similarly, the combination of both approaches do not show any significant improvement.

The second test scene is visualized in Figure 6-17. The corresponding reconstruction results of three consecutive frames are shown in Figure 6-18.

In the first column of Figure 6-18 three consecutive sparse depth maps from a particular scene are shown. Columns two to five show the standard, pre-filtered, weighted and the combined pre-filtered weighted approach respectively.

From examining the standard reconstruction results in the second column it becomes clear that the standard approach is sensitive to outliers and it is not able to reconstruct all three maps.

By pre-filtering, shown in the third column, the reconstruction becomes possible for all three scenes and to a large extent robustness against outliers is introduced.

The fourth column shows the results of the weighted approach which shows an even larger robustness against outliers relative to the pre-filtered approach.

Just like the pre-filtered and the weighted approach, the combined approach is able to reconstruct the scene's geometry to a large extent. But it is noted that the combined approach does not show any significant improvement over the weighted approach.

More test results can be found in the Appendix 7 at the end of this part.

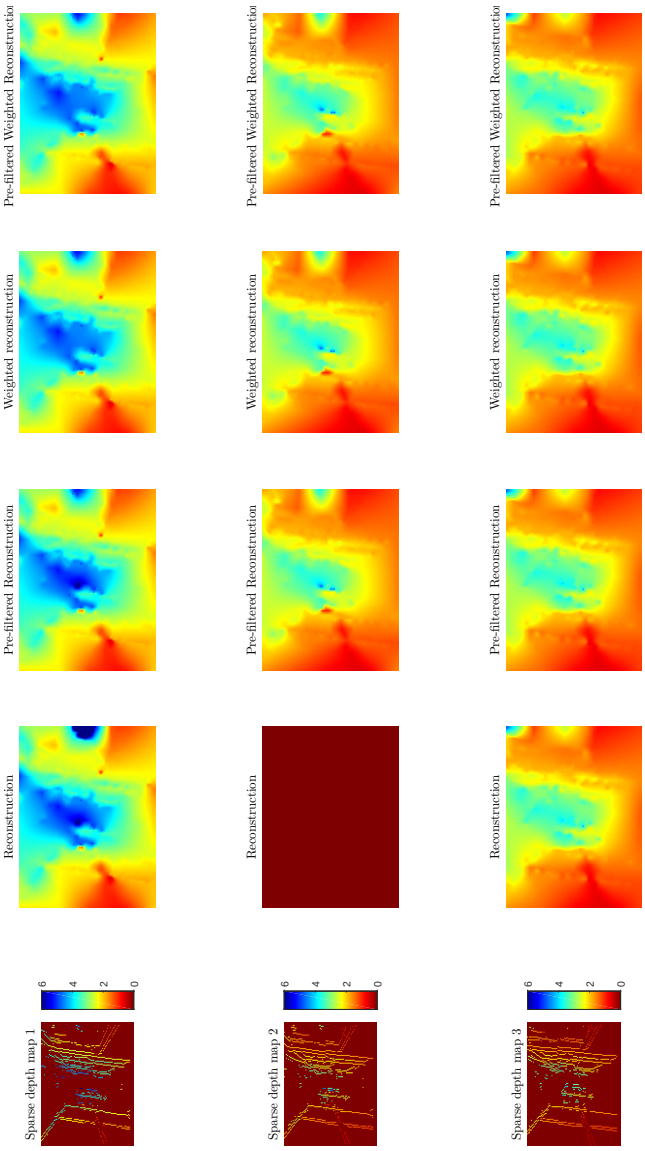


Figure 6-16: Reconstruction results of frame 22. First column, sparse depth map is shown. Second to fifth column show, standard reconstruction, pre-filtered, weighted and combined approach respectively.

Figure 6-17: Photo taken of scene 69 with a High-Definition camera sensor.

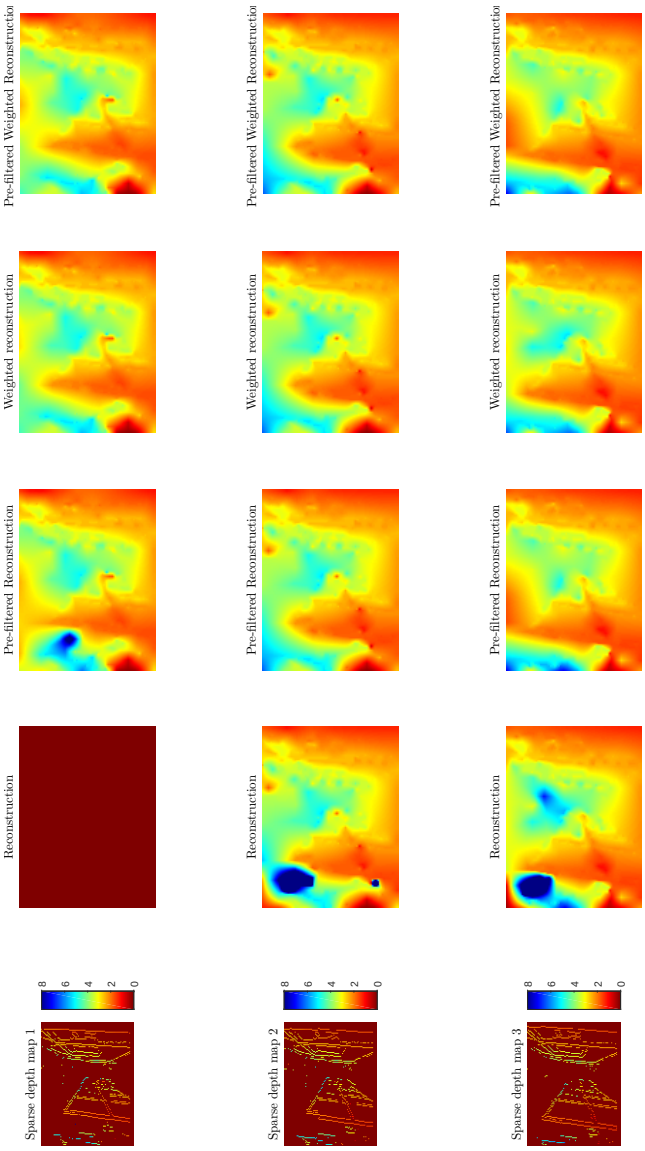


Figure 6-18: Reconstruction results of frame 69. First column, sparse depth map is shown. Second to fifth column show, standard reconstruction, pre-filtered, weighted and combined approach respectively.

6-9 Recursive Depth Reconstruction using Temporal Information

In order to increase the robustness of the reconstruction method against outliers even further, the use of temporal information is discussed in this section. Using information from previous reconstructions can assist in reconstructing a robust map in present time.

Before depth information from previous frames can be used for the reconstruction of the current depth map, the camera movement has to be estimated. This is the first of 4 main steps before the reconstruction is possible. In the following sections the main steps are explained in detail. First the pixel shift is estimated using optical flow algorithms. Second the sub-sampling approach of the previous reconstruction is discussed. Third the approach of allocating weights corresponding to the sparse depth values is discussed. And finally the overlay of the newly measured sparse depth map (T_{i+1}) onto the sparsely sampled reconstruction of T_i is discussed.

6-9-1 Pixel Shift Estimation using Optical Flow

In order to merge the depth reconstruction at Ti and the new sparse depth map at $Ti+1$, the maps have to be aligned such that the geometry they represent overlaps.

The quickest way is to apply feature matching in 2D image space, using the MATLAB[5] function `matchFeatures()` which implements the approach described by Lowe (Lowe, 2004). As input the sparse depth map Z_{sparse_i} at T_i and the sparse depth map $Z_{sparse_{i+1}}$ at T_{i+1}. The result is a $m \times 2$ vector of the matched features.

The next step is to calculate the pixel shift from T_i to T_{i+1}. A quick way is to select the median horizontal shift du and vertical shift dv and use it for the complete sample grid. But because of the fundamental workings of optical flow, using a single vector $[du, dv]$ to describe the flow is risky as it will be zero when moving perfectly forward or when subject to a roll movement.

Alternatively a fast and more precise method as introduced by McGuire et. al. (McGuire et al., 2016) could be implemented. This would allow for column and row wise determination of pixel shifts du and dv. But at this stage to test the concept of recursive sparse depth map reconstruction, the median shift will be used as a single value to describe the shift in the entire image.

The next step is to determine the approach of sub-sampling the depth map reconstruction of T_i.

6-9-2 Reconstruction Sub-Sampling

Super-positioning the sparse map of T_{i+1} onto the entire previous reconstruction would lead to large difficulties for the solver to pass the constraints. It is therefore considered to sub-sample the reconstruction, this will still introduce depth values to extremely sparse regions in the sparse depth map, guaranteeing a minimum information density.

[5] https://nl.mathworks.com/

It is this information density that is expected to introduce large improvements to the reconstruction approach. If for instance a single outlier value is positioned in an extremely sparse region, it's impact would be greatly minimised by the introduced samples from the previous frame.

The choice of a evenly space grid of samples taken from the previous reconstruction is trivial, as it guarantees a minimum information density while easy to implement. Other more sophisticated methods have not been considered at this point, but could consist of random sampling or method which guarantee a constant information density throughout the sparse depth map.

In two top figures in Figure 6-19 represent the sparse depth map at T_{i+1} on the left side, and the recursive map on the right side, e.g. the sparse depth map at T_{i+1} super-positioned onto a grid of samples from the reconstruction at T_i.

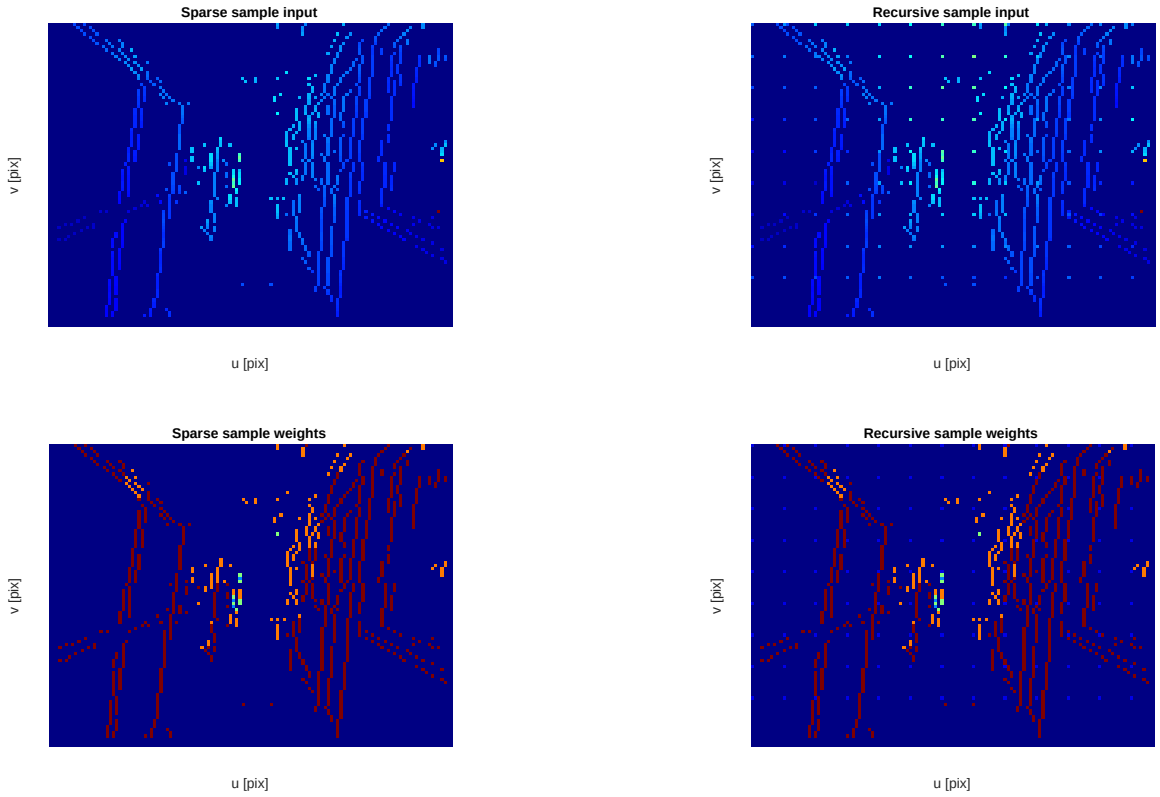


Figure 6-19: Top left; Sparse depth map T_{i+1}. Top right; Gridded samples from the reconstruction of T_i with the sparse depth map T_{i+1} super-positioned on top e.g. the recursive depth map. Bottom left; Weights corresponding to the sparse depth map. Bottom right; Weights corresponding to the recursive depth map.

In the figure, the depth values are encoded in the used colour. The smaller the proximity the darker blue and the larger the proximity, the lighter the colour blue. While comparing the sparse map in the top left and the recursive map on the top right it is clear how large the influence of the gridded samples are on the information density in the top and bottom of the map.

Table 6-1: Weight allocation as function of distance.

Weight allocation		
Segment	Distance [m]	Weight [-]
1	0 - 2	1
2	2 - 6	1 - 0
3	6 - ∞	0

The bottom two figures visualize the corresponding weights. The weight allocation will be discussed in the next section.

6-9-3 Weight Allocation

In order to gain a robust performance, the assignment of the right weights is key. The two bottom figures in Figure 6-19 visualise the weights. The dark red corresponds to a high confidence and thus a weight of 1, while colder colours all the way to blue correspond to less confident samples and lower weights. The gridded samples, shown in blue on the weight maps are assigned a constant value of 0.1. The weighting scheme is shown in Table 6-1.

In the table three segments are defined for which the weights are a function of the distance of the respective samples. In the first segment, where samples lay within 0 to 2 m, a weight of 1 is allocated. In the second segment which spans from 2 to 6 m, the confidence decreases linearly from 1 to 0 m. The third and last segment consists of all samples from 6 m and beyond, which all are assigned a weight of 0.

The result is that samples beyond 6 meters will not have any influence on the reconstruction as their constraint has a weight of zero. The choice for linear interpolation in segment 2, (2 to 6 m), is due to it's straight forward computation and the constant relation to the distance. The reconstruction results are shown in Figure 6-20 in Section 6-10.

Alternatively a natural choice for weights would be based on the confidence test which is used for the stereo-matching (Hu & Mordohai, 2012). In the stereo-matching algorithm a block-matching scheme us used to construct a cost function, from which the minimum is chosen as the disparity value. To determine the reliability of each disparity value and filter out low quality matches, a naive version of the peak ratio test is used (Tijmons, Croon, Remes, De Wagter, & Mulder, 2016). A threshold value on this ratio test is used for the computation of the sparse depth map on the MAVLab stereo-camera (Tijmons, Croon, Remes, De Wagter, & Mulder, 2016). Because there already exists this quantitative measure for the confidence of each measured sample, in the future it will be considered to base the weights on this test.

Further improvements in weight allocations can be sought in the direction of exploiting the temporal information. Stable samples which are measured in sequential frames are more likely to be correct rather than samples which are measured in only a single frame. Further work will address this issue.

6-9-4 Sparse Depth Map Overlay

In order to implement the recursive reconstruction approach, the gridded samples from T_i have to be merged with the new sparse depth map from T_{i+1}. The recursive samples are all given a constant weight of 0.1 and are considered less representative for the current relative to the new sparse samples. Therefore when a gridded sample and a new sample coincide on the same map coordinates (u, v), the sparse sample overwrites the gridded sample. This can be interpreted as; the new sparse depth samples overwrite the sparsely gridded samples.

A more sophisticated approach for the map overlay procedure can be considered when two improvements have been implemented. Firstly the naive peak ratio test, as described above, should be implemented and has to be proven to be a robust alternative to a constant confidence. And secondly a method is to be implemented to estimate the confidence of the recursive samples, e.g., incorporating higher confidence in stable samples. A natural approach would then be to, when two samples coincide on one coordinate (u, v), is to select the sample with the highest confidence.

In the next section the results of recursive depth reconstruction are presented where the gridded recursive samples are overwritten by the new sparse samples, regardless of their weights.

6-10 Recursive Depth Reconstruction Results

In this section recursive reconstruction results are discussed. The results are obtained using linear weight allocation for the new sparse samples, and a constant weight for the gridded recursive samples. To assess the effectiveness Figure 6-20 shows both the result without recursive samples on the top row and with recursive samples on the bottom row.

The top left figure shows the sparse depth map, and the top right figure shows the depth reconstruction using the weighted reconstruction as described in the previous sections. The bottom left figure shows the sparse depth map with recursive samples, and the bottom right shows it's respective depth reconstruction.

When looking at both sparse maps it immediately becomes clear that the added recursive samples add a lot of information in some regions of the image where little sparse samples where matched. In the reconstructions on the right side the consequences become clear immediately. The largest improvement is seen in the geometry of the light blue corridor in the center of the image. The non-recursive reconstruction uses the (faulty) blue samples in the top-center to reconstruct the top-side of the image, while the recursive reconstruction is able to retain most of the large-proximity values in the center and top-center of the image.

It can be said that the current sparse map has faulty values at a range smaller than 6m. The weights of these values are clearly shown in orange in the bottom left figure of Figure 6-19. The weights which are based on the distance are significant and therefore the non-recursive reconstruction does not show any clear geometry of the corridor anymore. The recursive reconstruction on the bottom left uses these gridded samples to retain the large proximity, do note that traces of the (faulty) sparse samples are still present in the reconstruction.

When examining the recursive reconstruction in detail, traces of the gridded samples can be found on the right side as lighter dots in the blue plane. These are the result because the

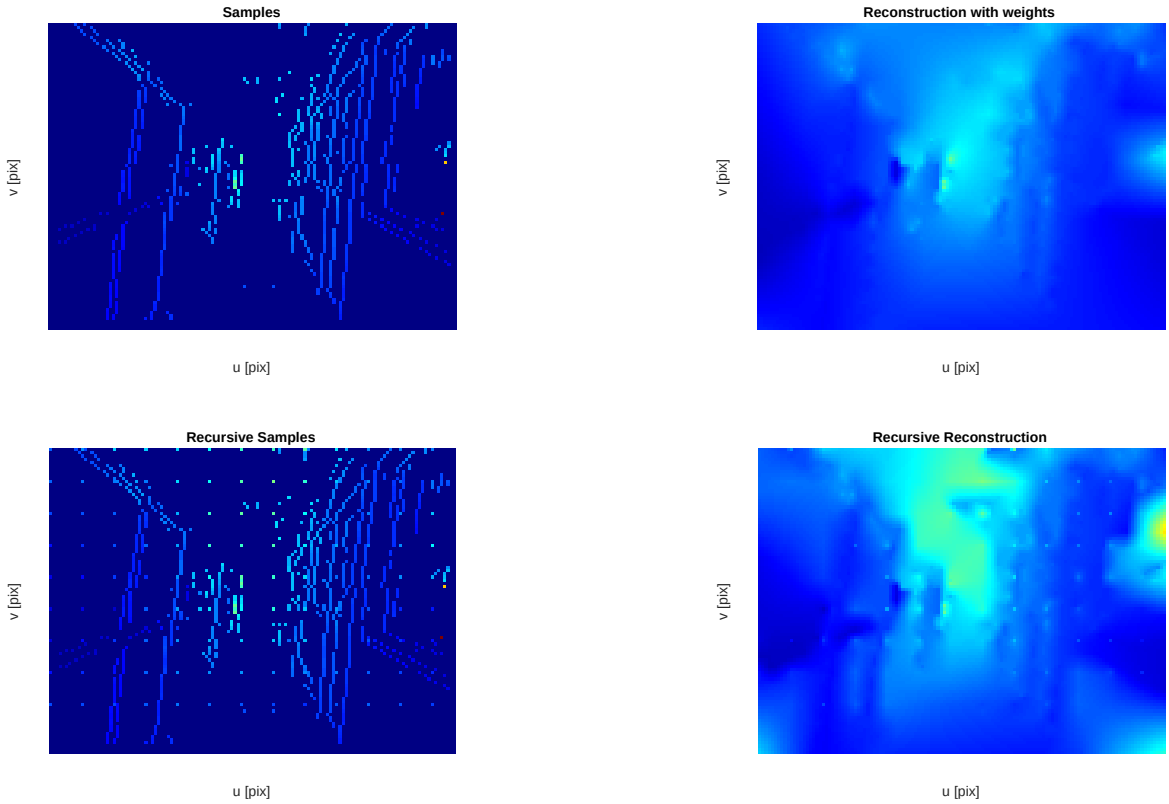


Figure 6-20: Top left; sparse depth map T_{i+1}. Top right; weighted reconstruction. Bottom left; gridded samples from the reconstruction of T_i with the sparse depth map T_{i+1} super-positioned on top e.g. the recursive sparse depth map. Bottom right; reconstruction using the recursive samples.

the camera has moved forward around 20 cm in between the frames, causing the scene in its entirety to get closer. Improvements are to be expected when the recursive samples are, except from shifted (see Section 6-9-1), also adjusted for the movement of the camera.

6-11 Discussion

In this report several methods were introduced and their results presented. Firstly it is described how the reconstruction method as introduced by Ma et. al. is implemented using MATLAB on a desktop system. Secondly a performance evaluation of using the reconstruction method is done on synthetic data and a description of how to implement it on a desktop system is given. This was followed by using the sparse depth map as calculated by the MAVLab stereo-board for depth map reconstruction. Before this could be done a method to handle mismatched disparity values had to be developed. The third contribution is a outlier removal approach using a neighbourhood search which has been successfully applied and proved effective. The fourth contribution is an alteration of the method by Ma et. al. by introducing weighted constraints to incorporate tolerance against outlier values and increase the robustness overall. The fifth and last contribution is the introduction of recursive samples

to improve the robustness against mismatches and improve the geometrical reconstruction capabilities significantly.

This section briefly discusses the results, significance and potential of the previously mentioned contributions.

6-11-1 Sparse Sensing Depth Reconstruction Method and Implementation

The method by Ma et. al. is a lean and powerful approach in reconstructing dense depth maps from sparse samples. By levering the assumed highly regular geometry in terms of flat surfaces and straight edges, the method proves to be effective in reconstructing the original dense map to a great extent.

For the implementation the same approach as Ma et. al. is chosen; the combination of the MATLAB development environment and the CVX/MOSEK solver. This solver is available on a academic licence and provides an easy to use convex optimization environment for the problem.

It has to be noted that for anything other than conceptual research, such as an implementation for UAV or MAV flight will require a dedicated solver written in C/C++ or similar programming language as real time performance is impossible in the MATLAB implementation.

After successful implementation it is shown that the method is able to fully reconstruct synthetic depth maps, given no open boundaries are given. When randomly sampling 1.6% of the original map, the method is able to reconstruct the dense map to a great extent. When providing all edge values of the synthetic map, the reconstruction matches perfectly with the only errors present at the unconstrained dataset boundaries.

6-11-2 Outlier Removal Using Neighbourhood Search

In order for the reconstruction approach of Ma et. al. to work on the sparse datasets, the data cannot contain large outliers. Because the MAVLab stereo-board camera provides sparse maps which are not guaranteed to be without outliers, a method is proposed to remove these potential outliers.

A neighbourhood search approach is a widely used method to identify outliers and a computationally efficient implementation has proven to be effective in identifying and removing outliers. The result is that where with the original sparse maps the reconstruction failed several times to provide a solution to the optimization problem, the pre-filtered sparse maps could be reconstructed in all tested scenes.

6-11-3 Sparse Sensing Depth Reconstruction using Weighted Samples

A new solution to deal with outliers came with the introduction of weighted constraints on the sparse samples. Where the method of Ma et. al. does not take into account any confidence measure with the samples, this report introduced depth depended weights to increase the robustness significantly.

As the sparse depth map from the MAVLab stereo-board is considered reliable for measurements closer than 2m and unreliable for measurements further than 6m, depth depended weights are added to the sample constraints of the optimization problem. It is shown that for all tested scenes this approach is successful in reconstructing the geometry and robust to any outliers. This can be explained that most outliers correspond to large distances which are given a weight of zero, and are thus do not constrain the reconstruction.

In several scenes it was shown that the weighed approach outperforms the the neighbourhood search method in terms of reconstructing the geometry of the environment.

Besides relating the weights on the distance of the values, it is considered to base the weights on the confidence measure of the stereo-matching method. The block-matching algorithm on the MAVLab stereo-board uses a naive peak ratio test to assess the confidence of finding a local minimum in the cost function. Using this ratio as a base for the weights will automatically put weights on good matches, and mismatches or outliers will be assigned a low weight.

It is expected that instead of interpolating the weights between a distance of 2m and 6m, using weights based on the naive peak ratio will result in qualitatively better results. Assigning a weight of zero to all values past 6m is still advised.

6-11-4 Recursive Depth Reconstruction using Temporal Information

The next step would be to incorporate temporal information in determining the weights. To use information from previous frames in the current map reconstruction, the pixel shift was estimated using the median optical flow. Although results showed to be effective, it would be better to incorporate a more pixel shift method. The histogram method proposed by McGuire et. al. is expected to facilitate pixel shifts at a more local level, future implementation is therefore recommended.

After determining the pixel shift from the previous frame to the current, the previous frame can be sub-sampled and merged with the current sparse depth map. The sub-sampling is done in an equally spaced grid because this ensures a minimum information density in the merged sparse map. Additional research should be done in finding an optimal step size of the grid. From the results it was noticed that most of the gridded samples were further away than the new sparse samples. This is explained by the forward movement of the camera with about 20cm, this had not been taken into account and will have to be done in futute work.

The merger of gridded samples from the previous reconstruction and new sparse samples has proven to be more successful than all other methods presented in this report. The main focus from this point onwards should be to assign a higher and appropriate weight to samples which are proven to be stable over time. Because these samples would have been consistently been measured multiple times, the chances of the sample to be wrong can be considered smaller than a sample which has been found in only one frame.

6-12 Conclusion

In this report the use of the depth map reconstruction method by Ma et. al. was proven feasibly in combination with the sparse depth map from the MAVLab stereo-camera. The

original method by Ma et. al. does not provide the required robustness against outliers to reconstruct each test scene, therefore a outlier removal algorithm was developed. Using a neighbourhood search based approach, outliers were successfully identified and could be filtered out before the reconstruction began.

A more significant contribution is made with the introduction of weighted constraints as an alternative to pre-filtering the samples. Samples are given a weight that corresponds to it's confidence level. At first this confidence level is directly coupled to the distance, this resulted in a large improvement in the quality of the reconstruction and the required robustness against outliers. For future work it is proposed to couple the weights to the naive peak ratio, used to assess the confidence of the stereo-matching.

Finally recursive depth reconstruction is proposed in the sense that previous reconstructions are sub-sampled and merged with the current sparse depth map. The result is that the reconstructions are now dependent on current sparse measurements and on sparse values from previous frames. The use of optical flow to shift the samples from previous frames has proven to be sufficient, but a more sophisticated local approach should be considered in future work. The addition of recursive samples brings the biggest improvement to the reconstruction of the geometries in this report. Where a purely weighted reconstruction was still prone to outliers, the recursive weighted approach shows to a large extent robustness against these outliers.

In future work the use of temporal information should be extended even more. Current weights were coupled to the distance, coupling them to the naive peak ratio test from the stereo-matching presents potential improvement. But also assigning higher weights to samples which are measured in consecutive frames and lower weights to samples which are only observed once, is expected to improve the robustness of the quality of the reconstruction significantly.

Chapter 7

Appendix

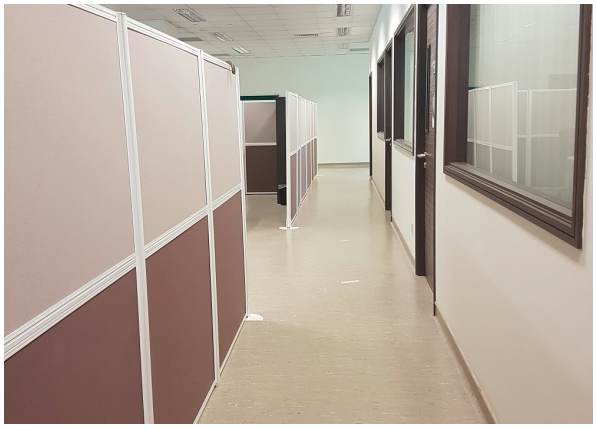


Figure 7-1: Photo taken of scene 15 with a High-Definition camera sensor.

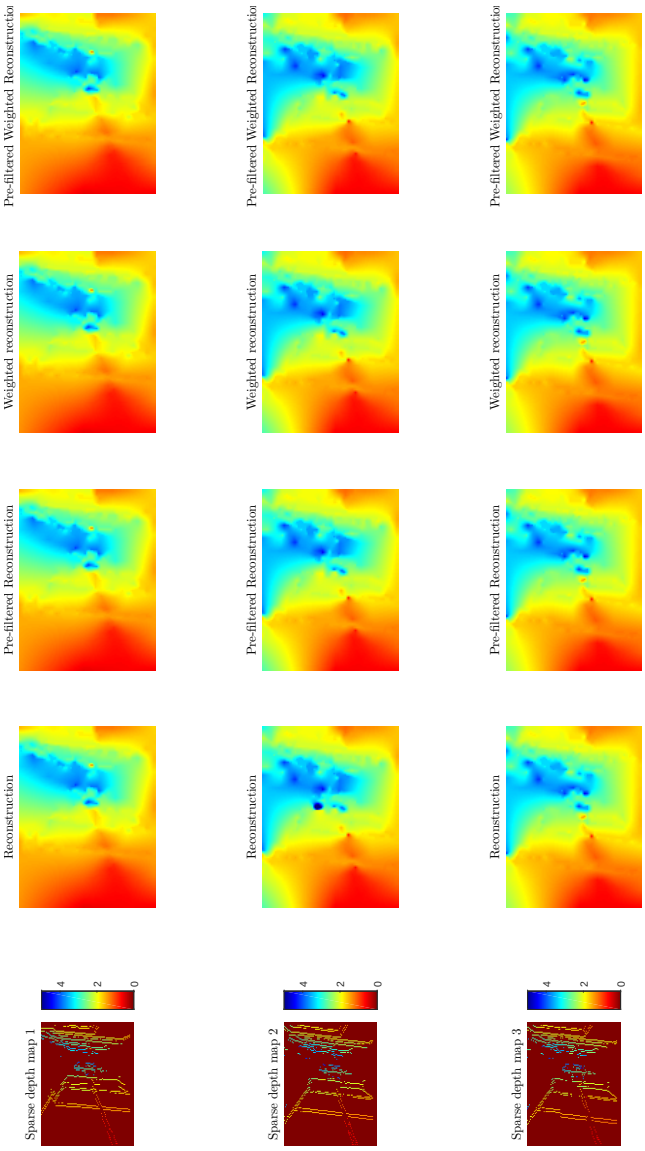


Figure 7-2: Reconstruction results of frame 15. First column, sparse depth map is shown. Second to fifth column show, standard reconstruction, pre-filtered, weighted and combined approach respectively.

Figure 7-3: Photo taken of scene 495 with a High-Definition camera sensor.

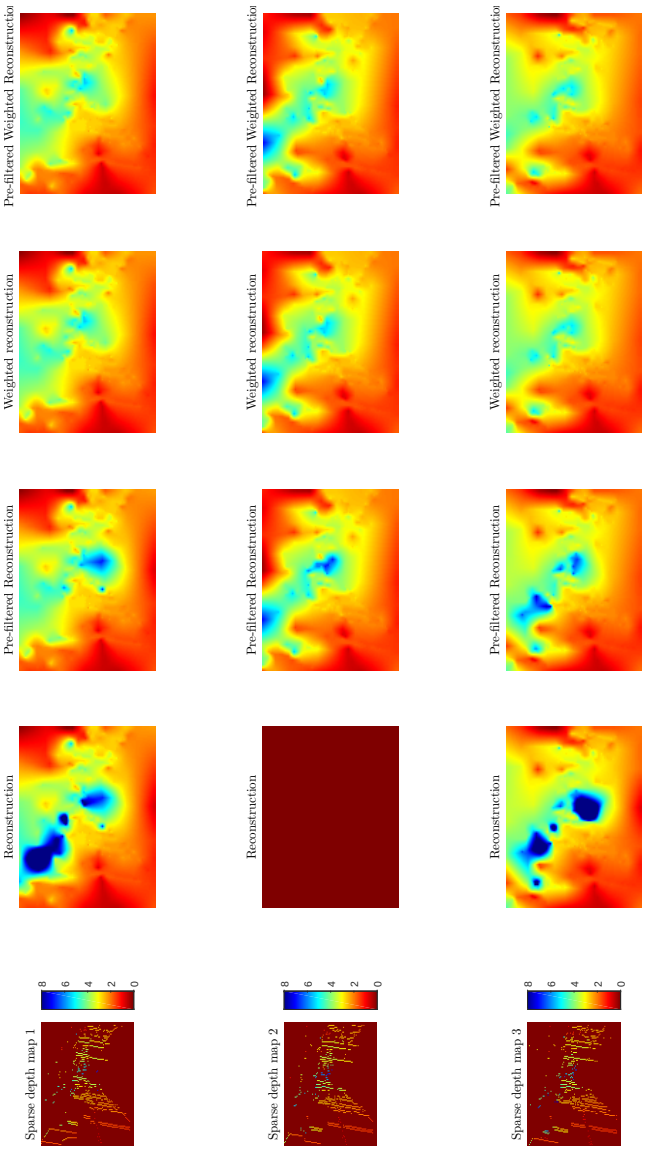


Figure 7-4: Reconstruction results of frame 495. First column, sparse depth map is shown. Second to fifth column show, standard reconstruction, pre-filtered, weighted and combined approach respectively.

Figure 7-5: Photo taken of scene 985 with a High-Definition camera sensor.

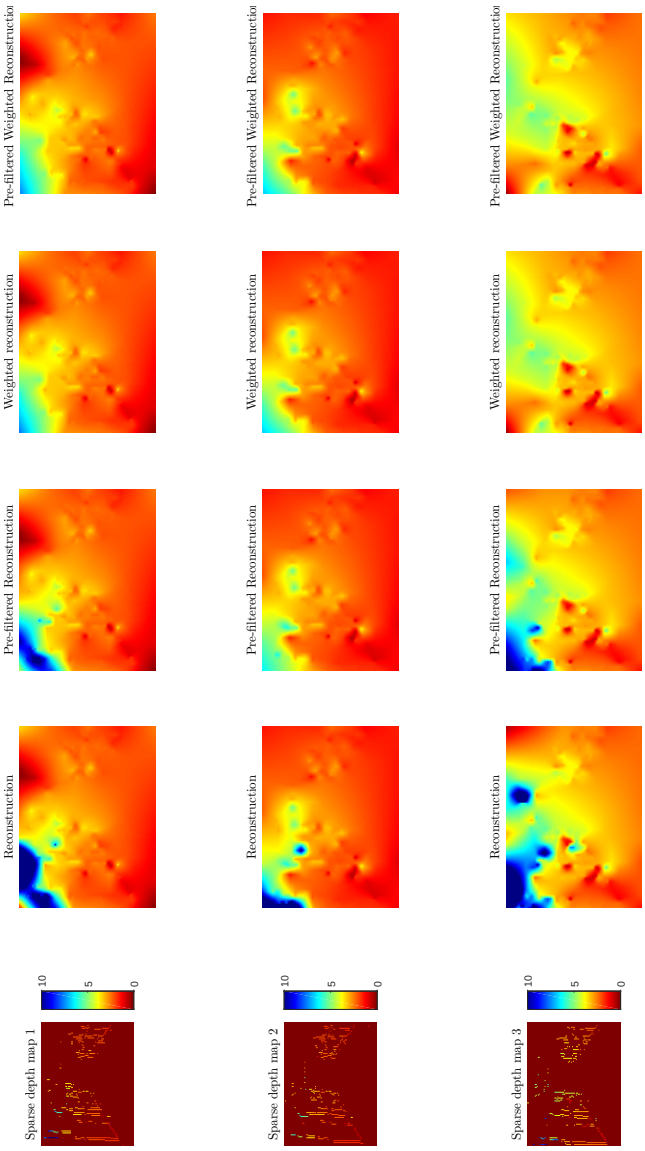


Figure 7-6: Reconstruction results of frame 985. First column, sparse depth map is shown. Second to fifth column show, standard reconstruction, pre-filtered, weighted and combined approach respectively.

Bibliography

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Susstrunk, S. (2010). SLIC Superpixels. *EPFL Technical Report 149300*(June), 15.

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(11), 2274–2281.

Allen, R., & Pavone, M. (2015). Toward a real-time framework for solving the kinodynamic motion planning problem. *International Conference on Robotics and Automation*, 928–934.

Allen, R., & Pavone, M. (2016). A Real-Time Framework for Kinodynamic Planning with Application to Quadrotor Obstacle Avoidance. {*AIAA*} *Conf. on Guidance, Navigation and Control*(January), 1–18.

Allen, R. E., Clark, A. A., Starek, J. A., & Pavone, M. (2014). A Machine Learning Approach for Real-Time Reachability Analysis Ross. *International Conference on Intelligent Robots and Systems*(Iros), 2202–2208.

Alvarez, H., Paz, L., Sturm, J., & Cremers, D. (2016). Collision Avoidance for Quadrotors with a Monocular Camera. *Experimental Robotics*, 149–163.

Bachrach, A., Prentice, S., He, R., Henry, P., Huang, A. S., Krainin, M., et al. (2012, sep). Estimation, planning, and mapping for autonomous flight using an RGB-D camera in GPS-denied environments. *The International Journal of Robotics Research*, *31*(11), 1320–1343. Available from `http://ijr.sagepub.com/cgi/doi/10.1177/0278364912455256`

Bajracharya, M., Howard, A., Matthies, L. H., Tang, B., & Turmon, M. (2009, jan). Autonomous off-road navigation with end-to-end learning for the LAGR program. *Journal of Field Robotics*, *26*(1), 3–25. Available from `http://doi.wiley.com/10.1002/rob.20269`

Bakolas, E., & Tsiotras, P. (2008). Multiresolution Path Planning Via Sector Decompositions Compatible to On-Board Sensor Data. In *Proceedings of the aiaa guidance, navigation, and control conference and exhibit.*

Barry, A. J., & Tedrake, R. (2014). Pushbroom Stereo for High-Speed Navigation in Cluttered

Environments. *3rd Workshop on Robots in Clutter: Perception and Interaction*, 2–8. Available from http://arxiv.org/abs/1407.7091

Beyeler, A., Zufferey, J. C., & Floreano, D. (2009). Vision-based control of near-obstacle flight. In *Autonomous robots* (Vol. 27, pp. 201–219).

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning* (Vol. 4) (No. 4). Available from http://www.library.wisc.edu/selectedtocs/bg0137.pdf

Bouktir, Y., Haddad, M., & Chettibi, T. (2008). Trajectory planning for a quadrotor helicopter. *2008 16th Mediterranean Conference on Control and Automation*, 1258–1263.

Brockers, R., Fragoso, A., & Matthies, L. (2016, may). Stereo vision-based obstacle avoidance for micro air vehicles using an egocylindrical image space representation. In T. George, A. K. Dutta, & M. S. Islam (Eds.), *Spie defense+ security* (p. 98361R). International Society for Optics and Photonics.

Brockers, R., Kuwata, Y., Weiss, S., & Matthies, L. (2014). Micro air vehicle autonomous obstacle avoidance from stereo-vision. In *Spie defense + security* (p. 90840O). International Society for Optics and Photonics.

Chao, H., Gu, Y., & Napolitano, M. (2014). A survey of optical flow techniques for robotics navigation applications. *Journal of Intelligent and Robotic Systems: Theory and Applications*, *73*(1-4), 361–372.

Choset, H. M. (2005). *Principles of Robot Motion: Theory, Algorithms, and Implementation.*

Choudhury, S., Gammell, J. D., Barfoot, T. D., Srinivasa, S. S., & Scherer, S. (2016, may). Regionally accelerated batch informed trees (RABIT*): A framework to integrate local information into optimal path planning. In *2016 ieee international conference on robotics and automation (icra)* (pp. 4207–4214). IEEE. Available from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7487615

Conroy, J., Gremillion, G., Ranganathan, B., & Humbert, J. S. (2009). Implementation of wide-field integration of optic flow for autonomous quadrotor navigation. In *Autonomous robots* (Vol. 27, pp. 189–198).

Cowling, I. D., Yakimenko, O. a., Whidborne, J. F., & Cooke, A. K. (2007). A Prototype of an Autonomous Controller for a Quadrotor UAV. *European Control Conference*, 1–8.

Cowling, I. D., Yakimenko, O. a., Whidborne, J. F., & Cooke, A. K. (2010). Direct Method Based Control System for an Autonomous Quadrotor. *Journal of Intelligent & Robotic Systems*, *60*, 285–316.

Dey, D., Shankar, K. S., Zeng, S., Mehta, R., Agcayazi, M. T., Eriksen, C., et al. (2016). Vision and learning for deliberative monocular cluttered flight. In *Springer tracts in advanced robotics* (Vol. 113, pp. 391–409).

Donald, B., Xavier, P., Canny, J., & Reif, J. (1993, nov). Kinodynamic motion planning. *Journal of the ACM*, *40*(5), 1048–1066. Available from http://portal.acm.org/citation.cfm?doid=174147.174150

Droeschel, D., Nieuwenhuisen, M., Beul, M., Holz, D., Stückler, J., & Behnke, S. (2015). Multilayered Mapping and Navigation for Autonomous Micro Aerial Vehicles. *Journal of Field Robotics*, n/a—-n/a. Available from http://dx.doi.org/10.1002/rob.21603

Engel, J., Sturm, J., & Cremers, D. (2013). Semi-dense visual odometry for a monocular camera. In *Proceedings of the ieee international conference on computer vision* (pp. 1449–1456).

Forster, C., Pizzoli, M., & Scaramuzza, D. (2014). SVO: Fast semi-direct monocular visual odometry. In *Proceedings - ieee international conference on robotics and automation* (pp. 15–22).

Foucart, S., & Rauhut, H. (2013). *A mathematical introduction to compressive sensing* (Vol. 1). Springer.

Fraundorfer, F., Heng, L., Honegger, D., Lee, G. H., Meier, L., Tanskanen, P., et al. (2012). Vision-based autonomous mapping and exploration using a quadrotor MAV. In *Ieee international conference on intelligent robots and systems* (pp. 4557–4564).

Frazzoli, E., Dahleh, M. A., & Feron, E. (2002). Real-time motion planning for agile autonomous vehicles. *AIAA Journal of Guidance and Control*, *25*(1), 116–129.

Gammell, J. D., Srinivasa, S. S., & Barfoot, T. D. (2014, sep). Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In *2014 ieee/rsj international conference on intelligent robots and systems* (pp. 2997–3004). IEEE. Available from `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6942976`

Gammell, J. D., Srinivasa, S. S., & Barfoot, T. D. (2015). BIT *: Batch Informed Trees for Optimal Sampling-based Planning via Dynamic Programming on Implicit Random Geometric Graphs. *International Conference on Robotics and Automation*, *abs/1405.5*, 3067–3074. Available from `http://arxiv.org/pdf/1405.5848.pdf`

Goldberg, S. B., & Matthies, L. (2011). Stereo and IMU assisted visual odometry on an OMAP3530 for small robots. In *Ieee computer society conference on computer vision and pattern recognition workshops.*

Grant, M., & Boyd, S. (2008). *Graph implementations for nonsmooth convex programs.* Springer-Verlag Limited.

Grant, M., & Boyd, S. (2014, March). *CVX: Matlab software for disciplined convex programming, version 2.1.* `http://cvxr.com/cvx`.

Hausman, K., Weiss, S., Brockers, R., Matthies, L., & Sukhatme, G. S. (2016, may). Self-calibrating multi-sensor fusion with probabilistic measurement validation for seamless sensor switching on a UAV. In *2016 ieee international conference on robotics and automation (icra)* (pp. 4289–4296). IEEE. Available from `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7487626`

Hecke, K. van, De Croon, G. C. H. E., Maaten, L. van der, Hennes, D., & Izzo, D. (2016). Persistent self-supervised learning principle: from stereo to monocular vision for obstacle avoidance. , 1–17. Available from `http://arxiv.org/abs/1603.08047`

Hirschmüller, H., Innocent, P. R., & Garibaldi, J. (2002). Real-time correlation-based stereo vision with reduced border errors. *International Journal of Computer Vision*, *47*(1-3), 229–246.

Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., & Burgard, W. (2013). Octomap: An efficient probabilistic 3d mapping framework based on octrees. *Autonomous Robots*, *34*(3), 189–206.

Hrabar, S., Sukhatme, G. S., Corke, P., Usher, K., & Roberts, J. (2005). Combined optic-flow and stereo-based navigation of urban canyons for a UAV. In *2005 ieee/rsj international conference on intelligent robots and systems, iros* (pp. 302–309).

Hsu, D., Kindel, R., Latombe, J.-C., & Rock, S. (2002). Randomized Kinodynamic Motion Planning with Moving Obstacles. *The International Journal of Robotics Research*, *21*(3), 233–255.

Hu, X., & Mordohai, P. (2012). A quantitative evaluation of confidence measures for stereo vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(11), 2121–2133.

Janson, L., Schmerling, E., Clark, A., & Pavone, M. (2015). Fast marching tree:

A fast marching sampling-based method for optimal motion planning in many dimensions. *The International Journal of Robotics Research*. Available from http://ijr.sagepub.com/content/early/2015/05/04/0278364915577958.abstract

Kabanava, M., & Rauhut, H. (2015). Cosparsity in compressed sensing. In *Compressed sensing and its applications* (pp. 315–339). Springer.

Karaman, S., & Frazzoli, E. (2011, jun). Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research*, *30*(7), 846–894. Available from http://ijr.sagepub.com/content/30/7/846.short http://ijr.sagepub.com/cgi/doi/10.1177/0278364911406761

Kavraki, L. E., Vestka, P., Latombe, J. C., & Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, *12*(4), 566–580.

Kendoul, F. (2012). Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *Journal of Field Robotics*, *29*(2), 315–378.

Keshavan, J., Gremillion, G., Alvarez-Escobar, H., & Humbert, J. S. (2015). Autonomous Vision-Based Navigation of a Quadrotor in Corridor-Like Environments. *International Journal of Micro Air Vehicules*, *7*(2), 111–124.

Khan, a. M. (2013). Image Segmentation Methods: A Comparative Study. *International Journal of Soft Computing and Engineering (IJSCE)*, *3*(4), 84–92.

Kuffner, J., & LaValle, S. M. (2000). RRT-connect: An efficient approach to single-query path planning. *Proc. IEEE International Conference on Robotics and Automation ICRA '00*, *2*(Icra), 995—-1001 vol.2.

Ladd, A. M., & Kavraki, L. E. (2004). Measure theoretic analysis of probabilistic path planning. {*IEEE*} *Trans.\ on Robotics and Automation*, *20*(2), 229–242.

Lavalle, S. M. (2006). Planning Algorithms. *Cambridge*, 842. Available from http://ebooks.cambridge.org/ref/id/CBO9780511546877

LaValle, S. M. (2011). Motion Planning. Part II: Wild Frontiers. *IEEE Robotics & Automation Magazine*, *18*(June), 108–118. Available from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5876226

LaValle, S. M., & Kuffner, J. J. (2001, may). Randomized Kinodynamic Planning. *The International Journal of Robotics Research*, *20*(5), 378–400. Available from http://ijr.sagepub.com/cgi/doi/10.1177/02783640122067453

Lee, T., Leok, M., & Mcclamroch, N. H. (2013). Nonlinear robust tracking control of a quadrotor UAV on SE(3). *Asian Journal of Control*, *15*(2), 391–408.

Levinshtein, A., Stere, A., Kutulakos, K. N., Fleet, D. J., Dickinson, S. J., & Siddiqi, K. (2009). TurboPixels: Fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(12), 2290–2297.

Li, Y., Littlefield, Z., & Bekris, K. E. (2016, apr). Asymptotically optimal sampling-based kinodynamic planning. *The International Journal of Robotics Research*, *35*(5), 528–564. Available from http://ijr.sagepub.com/cgi/doi/10.1177/0278364915614386

Liu, M. Y., Tuzel, O., Ramalingam, S., & Chellappa, R. (2011). Entropy rate superpixel segmentation. In *Proceedings of the ieee computer society conference on computer vision and pattern recognition* (pp. 2097–2104).

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, *60*(2), 91–110.

Luders, B. D., Karaman, S., Frazzoli, E., & How, J. P. (2010). Bounds on

Tracking Error using Closed-Loop Rapidly-Exploring Random Trees. *American Control Conference (ACC), 2010*(1), 5406 – 5412. Available from http://acl.mit.edu/papers/Luders10_ACC.pdf

Ma, F., Carlone, L., Ayaz, U., & Karaman, S. (2016). Sparse Sensing for Resources-Constrained Depth Reconstruction. *Int. Conf. on Intelligent Robots and Systems (IROS)*.

Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings eighth ieee international conference on computer vision. iccv 2001* (Vol. 2, pp. 416–423).

McGuire, K., Croon, G. de, Wagter, C. de, Remes, B., Tuyls, K., & Kappen, H. (2016). Local histogram matching for efficient optical flow computation applied to velocity estimation on pocket drones. *arXiv preprint arXiv:1603.07644*.

Mellinger, D., & Kumar, V. (2011). Minimum snap trajectory generation and control for quadrotors. *International Conference on Robotics and Automation*, 2520–2525.

Moore, A. P., Prince, S. J. D., & Warrell, J. (2010). "Lattice cut" - Constructing superpixels using layer constraints. In *Proceedings of the ieee computer society conference on computer vision and pattern recognition* (pp. 2117–2124).

Moore, A. P., Prince, S. J. D., Warrell, J., Mohammed, U., & Jones, G. (2008). Superpixel lattices. In *26th ieee conference on computer vision and pattern recognition, cvpr.*

Nam, S., Davies, M. E., Elad, M., & Gribonval, R. (2013). The cosparse analysis model and algorithms. *Applied and Computational Harmonic Analysis*, *34*(1), 30–56.

Nieuwenhuisen, M., & Behnke, S. (2016). Layered Mission and Path Planning for MAV Navigation with Partial Environment Knowledge. *Intelligent Autonomous Systems*, *13*, 307–319. Available from http://www.scopus.com/inward/record.url?eid=2-s2.0-84945968485&partnerID=tZOtx3y1

Nuske, S., Choudhury, S., Jain, S., Chambers, A., Yoder, L., Scherer, S., et al. (2015). Autonomous exploration and motion planning for an unmanned aerial vehicle navigating rivers. *Journal of Field Robotics*, *32*(8), 1141–1162.

Otte, M. W., Richardson, S. G., Mulligan, J., & Grudic, G. (2009). Path planning in image space for autonomous robot navigation in unstructured environments. *Journal of Field Robotics*, *26*(2), 212–240.

Ren, X., & Malik, J. (2003). Learning a classification model for segmentation. *Proceedings Ninth IEEE International Conference on Computer Vision*, *1*(c), 10–17 vol.1.

Richter, C., Bry, A., & Roy, N. (2013). Polynomial Trajectory Planning for Quadrotor Flight. *International Conference on Robotics and Automation*. Available from http://www.michigancmes.org/papers/roy7.pdf

Richter, C., Bry, A., & Roy, N. (2016). Polynomial Trajectory Planning for Aggressive Quadrotor Flight in Dense Indoor Environments. *Robotics Research*(Isrr), 649—-666. Available from http://groups.csail.mit.edu/rrg/papers/ISRR13_Richter.pdf

Ross, I. M., & Fahroo, F. (2006). Issues in the real-time computation of optimal control. *Mathematical and Computer Modelling*, *43*(9-10), 1172–1188.

Ross, S., Melik-Barkhudarov, N., Shankar, K. S., Wendel, A., Dey, D., Bagnell, J. A., et al. (2013a). Learning monocular reactive UAV control in cluttered natural environments. In *Proceedings - ieee international conference on robotics and automation* (pp. 1765–1772).

Ross, S., Melik-Barkhudarov, N., Shankar, K. S., Wendel, A., Dey, D., Bagnell, J. A., et al. (2013b). Presentation of the Paper Learning Monocular Reactive UAV Control in Cluttered Natural Environments . , 1765–1772.

Schmerling, E., Janson, L., & Pavone, M. (2015). Optimal Sampling-Based Motion Planning under Differential Constraints: the Drift Case with Linear Affine Dynamics. *Conference on Decision and Control (CDC)*(Cdc), 2574–2581.

Schmid, K., Lutz, P., Tomic, T., Mair, E., & Hirschmüller, H. (2014). Autonomous Vision-based Micro Air Vehicle for Indoor and Outdoor Navigation. *Journal of Field Robotics*, *7*(PART 1), 81–86.

Schops, T., Enge, J., & Cremers, D. (2014). Semi-dense visual odometry for AR on a smartphone. In *Ismar 2014 - ieee international symposium on mixed and augmented reality - science and technology 2014, proceedings* (pp. 145–150).

Schouwenaars, T., Moor, B. D., Feron, E., How, J., DeMoor, B., Frazzoli, E., et al. (2001). Mixed Integer Programming for Multi-Vehicle Path Planning. *In European Control Conference 2001*, *1*(3), 2603–2608.

Shen, S., Michael, N., & Kumar, V. (2011). 3D Indoor Exploration with a Computationally Constrained MAV. In *Rss workshop* (pp. 1–3).

Shen, S., Mulgaonkar, Y., Michael, N., & Kumar, V. (2013). Vision-based state estimation for autonomous rotorcraft MAVs in complex environments. *Proceedings - IEEE International Conference on Robotics and Automation*, 1758–1764.

Shen, S., Mulgaonkar, Y., Michael, N., & Kumar, V. (2014). Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft MAV. *Proceedings - IEEE International Conference on Robotics and Automation*, 4974–4981.

Stipanović, D. M., Hwang, I., & Tomlin, C. J. (2004). Computation of an over-approximation of the backward reachable set using subsystem level set functions. *Dynamics of Continuous, Discrete and Impulsive Systems Series A: Mathematical Analysis*, *11*(2-3), 399–411.

Sunberg, Z. N., Kochenderfer, M. J., & Pavone, M. (2016). Optimized and Trusted Collision Avoidance for Unmanned Aerial Vehicles using Approximate Dynamic Programming. *International Conference on Robotics and Automation*, 1–8. Available from `http://arxiv.org/abs/1602.04762`

Tijmons, S., Croon, G. de, Remes, B., De Wagter, C., & Mulder, M. (2016). Obstacle Avoidance Strategy using Onboard Stereo Vision on a Flapping Wing MAV. , 1–13. Available from `http://arxiv.org/abs/1604.00833`

Tijmons, S., Croon, G. de, Remes, B., De Wagter, C., & Mulder, M. (2016). Obstacle avoidance strategy using onboard stereo vision on a flapping wing mav. *arXiv preprint arXiv:1604.00833*.

Tijmons, S., De Croon, G. C. H. E., Remes, B., De Wagter, C., & Mulder, M. (2016). Obstacle Avoidance Strategy using Onboard Stereo Vision on a Flapping Wing MAV. , 1–13. Available from `http://arxiv.org/abs/1604.00833`

Urmson, C., & Simmons, R. (2003). Approaches for heuristically biasing RRT growth. *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, *2*, 1178–1183 vol.2.

Van den Bergh, M., Boix, X., Roig, G., & Van Gool, L. (2015). SEEDS: Superpixels Extracted Via Energy-Driven Sampling. *International Journal of Computer Vision*, *111*(3), 298–314.

Veksler, O., Boykov, Y., & Mehrani, P. (2010). Superpixels and supervoxels in an energy op-

timization framework. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* (Vol. 6315 LNCS, pp. 211–224).

Wang, S., Lu, H., Yang, F., & Yang, M.-H. (2011). Superpixel tracking. *2011 International Conference on Computer Vision*, 1323–1330.

Webb, D. J., & Berg, J. van den. (2013, may). Kinodynamic RRT*: Asymptotically optimal motion planning for robots with linear dynamics. In *2013 ieee international conference on robotics and automation* (pp. 5054–5061). IEEE. Available from `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6631299`

Yu, H., & Beard, R. (2013). A vision-based collision avoidance technique for micro air vehicles using local-level frame mapping and path planning. *Autonomous Robots*, *34*(1-2), 93–109.

Zhang, Y., Hartley, R., Mashford, J., & Burn, S. (2011). Superpixels via pseudo-Boolean optimization. In *Proceedings of the ieee international conference on computer vision* (pp. 1387–1394).

Zingg, S., Scaramuzza, D., Weiss, S., & Siegwart, R. (2010). MAV navigation through indoor corridors using optical flow. *Proceedings - IEEE International Conference on Robotics and Automation*, 3361–3368.