3D structure estimation from image stream in urban environment

Mohamad Motasem Nawaf

3D STRUCTURE ESTIMATION FROM IMAGE STREAM IN URBAN ENVIRONMENT

Mohamad Motasem Nawaf

A thesis submitted for partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Speciality: image, vision, signal

European Doctorate Label

Defended on December 5, 2014

Approved by:

Reviewers:	Paolo Favaro	Professor	University of Bern, Switzerland
	Joaquim Salvi	Professor	University of Girona, Spain
Examiners:	Christine Solnon	Professor	INSA de Lyon, France
	Andrew Wallace	Professor	Heriot Watt University, United Kingdom
Advisor:	Alain Trémeau	Professor	Université de Saint Etienne, France
Co-Advisor:	Dro Désiré Sidibé	Associate Professor	Université de Bourgogne, France

Estimation de la Structure 3D d'un Environnement Urbain à Partir d'un Flux Video

Mohamad Motasem Nawaf

Thèse présentée pour obtenir le grade de

Docteur de l'Universite Jean Monnet

Label Européen

Specialité: image, vision, signal.

Soutenue le 5 Décembre, 2014

Devant le jury:

Rapporteurs:	Paolo Favaro	Professeur	University of Bern, Suisse
	Joaquim Salvi	Professeur	University of Girona, Espagne
Examinateurs:	Christine Solnon	Professeur	INSA de Lyon, France
	Andrew Wallace	Professeur	Heriot Watt University, Royaume-Uni
Directeur:	Alain Trémeau	Professeur	Université de Saint Etienne, France
Co-Directeur:	Dro Désiré Sidibé	Maître de conférences	Université de Bourgogne,France

Abstract

In computer vision, the 3D structure estimation from 2D images remains a fundamental problem. One of the emergent applications is 3D urban modelling and mapping. Here, we are interested in street-level monocular 3D reconstruction from mobile vehicle. In this particular case, several challenges arise at different stages of the 3D reconstruction pipeline. Mainly, lacking textured areas in urban scenes produces low density reconstructed point cloud. Also, the continuous motion of the vehicle prevents having redundant views of the scene with short feature points lifetime. In this context, we adopt the piecewise planar 3D reconstruction where the planarity assumption overcomes the aforementioned challenges.

In this thesis, we introduce several improvements to the 3D structure estimation pipeline. In particular, the planar piecewise scene representation and modelling. First, we propose a novel approach that aims at creating 3D geometry respecting superpixel segmentation, which is a gradient-based boundary probability estimation by fusing colour and flow information using weighted multi-layered model. A pixel-wise weighting is used in the fusion process which takes into account the uncertainty of the computed flow. This method produces non-constrained superpixels in terms of size and shape. For the applications that imply a constrained size superpixels, such as 3D reconstruction from an image sequence, we develop a flow based SLIC method to produce superpixels that are adapted to reconstructed points density for better planar structure fitting. This is achieved by the mean of new distance measure that takes into account an input density map, in addition to the flow and spatial information.

To increase the density of the reconstructed point cloud used to perform the planar structure fitting, we propose a new approach that uses several matching methods and dense optical flow. A weighting scheme assigns a learned weight to each reconstructed point to control its impact to fitting the structure relative to the accuracy of the used matching method. Then, a weighted total least square model uses the reconstructed points and learned weights to fit a planar structure with the help of superpixel segmentation of the input image sequence. Moreover, the model handles the occlusion boundaries between neighbouring scene patches to encourage connectivity and co-planarity to produce more realistic models. The final output is a complete dense visually appealing 3D models. The validity of the proposed approaches has been substantiated by comprehensive experiments and comparisons with state-of-the-art methods.

Résumé

Dans le domaine de la vision par ordinateur, l'estimation de la structure d'une scène 3D à partir d'images 2D constitue un problème fondamental. Parmi les applications concernées par cette problématique, nous nous sommes intéressés dans le cadre de cette thèse à la modélisation d'un environnement urbain. Plus spécifiquement, nous nous sommes intéressés à la reconstruction de scènes 3D à partir d'images monoculaires générées par un véhicule en mouvement. Dans ce cas particulier, plusieurs défis se posent à travers les différentes étapes de la chaine de traitement inhérente à la reconstruction 3D. L'un de ces défis vient du fait de l'absence de zones suffisamment texturées dans certaines scènes urbaines, d'où une reconstruction 3D (un nuage de points 3D) trop éparse. De plus, du fait du mouvement du véhicule, d'une image à l'autre il n'y a pas toujours un recouvrement suffisant entre différentes vues consécutives d'une même scène. Dans ce contexte, et ce afin de lever les verrous ci-dessus mentionnés, nous proposons d'estimer, de reconstruire, la structure d'une scène 3D par morceaux en se basant sur une hypothèse de planéité.

Dans cette thèse, nous proposons plusieurs améliorations à la chaine de traitement associée à la reconstruction 3D. Tout d'abord, afin de structurer, de représenter, la scène sous la forme d'entités planes nous proposons une nouvelle méthode de reconstruction 3D, basée sur le regroupement de pixels similaires (*superpixel segmentation*), qui à travers une représentation multi-échelle pondérée fusionne les informations de couleur et de mouvement. Cette méthode est basée sur l'estimation de la probabilité de discontinuités locales aux frontières des régions calculées à partir du gradient (*gradient-based boundary probability estimation*). Afin de prendre en compte l'incertitude liée à l'estimation du mouvement, une pondération par morceaux est appliquée à chaque pixel en fonction de cette incertitude. Cette méthode génère des regroupements de pixels (superpixels) non contraints en termes de taille et de forme. Pour certaines applications, telle que la reconstruction 3D à partir d'une séquence d'images, des contraintes de taille sont nécessaires. Nous avons donc proposé une méthode qui intègre à l'algorithme SLIC (*Simple Linear Iterative Clustering*) l'information de mouvement. L'objectif étant d'obtenir une reconstruction 3D plus dense qui estime mieux la structure de la scène. Afin d'atteindre cet objectif, nous avons également introduit une nouvelle distance qui, en complément de l'information de mouvement et de données images, prend en compte la densité du nuage de points.

Afin d'augmenter la densité du nuage de points utilisé pour reconstruire la structure de la scène sous la forme de surfaces planes, nous proposons une nouvelle approche qui mixte plusieurs méthodes d'appariement et une méthode de flot optique dense. Cette méthode est basée sur un système de

pondération qui attribue un poids pré-calculé par apprentissage à chaque point reconstruit. L'objectif étant de contrôler l'impact de ce système de pondération, autrement dit la qualité de la reconstruction, en fonction de la précision de la méthode d'appariement utilisée. Afin d'atteindre cet objectif, nous avons appliqué un processus des moindres carrés pondérés aux données reconstruites pondérées par les calculés par apprentissage, qui en complément de la segmentation par morceaux de la séquence d'images, permet une meilleure reconstruction de la structure de la scène sous la forme de surfaces planes. Nous avons également proposé un processus de gestion des discontinuités locales aux frontières de régions voisines dues à des occlusions (*occlusion boundaries*) qui favorise la coplanarité et la connectivité des régions connexes. L'objectif étant d'obtenir une reconstruction 3D plus fidèle à la réalité de la scène. L'ensemble des modèles proposés permet de générer une reconstruction 3D dense représentative à la réalité de la scène. La pertinence des modèles proposés a été étudiée et comparée à l'état de l'art. Plusieurs expérimentations ont été réalisées afin de démontrer, d'étayer, la validité de notre approche.

Acknowledgements

It has been more than three years when I started my PhD as being an old dream. Since then, I had a pleasant journey full of learning and gaining experience. This journey would not have been possible without the help and support of many people who accompanied me during this period.

First of all I would like to thank my supervisor Professor Alain Trémeau for his excellent guidance, support and the space of freedom I can move without restrictions. He was available from the initial until the final level of my PhD. His guidance drew the main guidelines of my PhD which enabled me to achieve the plans on time.

During my PhD I was fortunate enough to spend several months at Heriot-Watt University in the UK. Here, my great thanks to the Professor Andrew Wallace to give me this opportunity, and for the experience I gained while working there. I would like also to thank my co-supervisor Dr. Désiré Sidibé who helped me to improve the quality of my publications and manuscript. Also for the his valuable comments and suggestions.

I would like to thank the Professors Paolo Favaro and Joaquim Salvi for accepting to review this manuscript. Their comments on the manuscript have helped me to clarify the concepts and expand the target audience of this work. In addition, I would like to thank Professor Christine Solnon for accepting to be the president of the thesis committee.

My thanks also go to members of the Hubert Curien laboratory, both labmates and staff for their kindness and making my PhD life an enjoyable and memorable one.

Last but not least, this thesis would not have been possible without the unconditional love and support of my family. I dedicate this thesis to them.

Contents

Table des matières

List of Figures

List of Tables

List of Abbreviations

BM	Block Matching
BA	Bundle Adjustment
CRF	Conditional Random Field
DoF	Degrees of Freedom
DLT	Direct Linear Transformation
EM	Expectation Maximization
FM	Fundamental Matrix
GBP	Generalized Boundary Probability
GPS	Global Positioning System
HSI	Hue-Saturation-Intensity
IMU	Inertial Navigation System
LiDAR	Laser scanner
LF	Low Level Features
LK	Lucas-Kanade method
MRF	Markov Random Field
MVS	Multi View Stereo
PMVS	Patch-based Multi View Stereo
RF	Random Forest
RANSAC	Random Sample Consensus
SLIC	Simple local iterative clustering
SVD	Singular Value Decomposition.
SFM	Structure from Motion
VRML	Virtual Reality Modeling Language

1 Introduction

Contents

1.1 Structure Estimation From 2D Images

Nowadays, cameras are becoming one of the favourite personal devices that we use in our daily life. Either in their standalone form or embedded in other devices such as mobile phones and tablet PCs. This popularity has as a consequence a huge amount of photos and videos being captured and stored worldwide. Some of those photos and videos are being taken in an organized way for more useful purposes, and here is our interest. For instance, image sequences in urban scenes.

When a picture is taken by a camera, depth information about the scene is lost. Indeed, one of the early topics of computer vision is recovering the three-dimensional (3D)

structure of a scene using two-dimensional (2D) images. Inspired by human stereo vision system, the early techniques focused on performing depth estimation using stereo pair of vertically aligned cameras, where the computed points disparity is directly related to their depth. Later on, approaches for multi-view scene reconstruction start to appear and widely attract researchers from early nighties. Motivated by the increase availability of digital images and the computational speed from one side, and the interest in the digital 3D modelling from another side. Such approaches are dedicated to estimate the depth using information from several overlapped images of a scene [Hartley 2004]. This results in establishing the fundamentals of 3D reconstruction in the general case. Most of the successive works in this domain aim at increasing both the density and the reconstruction quality of the obtained 3D models [Furukawa 2010]. Some went further by performing surface reconstruction and 3D meshing [Wu 2012, Lhuillier 2013].

In the meanwhile, other specific cases such as 3D reconstruction from monocular image sequence have been treated as a multi-view problem, as it follows the same assumptions on which the multi-view fundamentals are based. However, applying the common multi-view vision techniques in this context faces several challenges, which will be explained later on in details. The resulting 3D models are generally less dense and have lower quality. Although several works have been dedicated to deal with the aforementioned context [Vedaldi 2007, Micusik 2009], this special case remains challenging and did not receive the same attention as for the stereo and multi-view vision. Here come our *motivations* to target the monocular vision setup in this thesis. We demonstrate that there are several cues that can be exploited to improve the output quality and density. Among several possible contexts, we target specifically building in-city 3D models from monocular image sequence. The *motivation* of using a single camera is for its cheap price and simple setup which is available to everyone, compared to other sophisticated image acquisition configurations, such as omni-directional camera and rolling-shutter camera rigs. The common acquisition setup in this case is a camera installed on a mobile vehicle and pointing forward. This setup already exist in many mobile vehicles as a part of their surveillance (and recently safety) systems [McCall 2006].

Regarding the assumed context, we differentiate between two kinds of targeted scene

Figure 1.1: Examples of rural (first raw) and urban (second raw) scenes. Source: Google Earth 2014.

types: rural and urban environment. The rural environment is known to be rich in texture with less planar structures. It is also more open so the majority of the scene is located out of the range of vision system (Except for aerial images where several successful methods for 3D reconstruction exist). In contrary, urban environment is poor with texture, while it tends to have more planar structures, mostly vertically aligned. Figure 1.1 shows typical examples for rural and urban environments which confirm the above mentioned properties. Given these facts, some adopted approaches for one scene's kind may not be the best for the other kind. We dedicate our thesis to the latter case, motivated by the increasing interest in building 3D city models. More precisely, the building façades, road structure and other stationary objects.

A general requirement that needs to be discussed in most of computer science related methods is the computational time. In the assumed context, having the reconstruction algorithm in real time is an advantage, as this will extend the number of applications where the method can be applied. For instance, driving assistant systems. However, as our focus is on the output quality, in our specific context, there is more information that can be extracted in off-line processing than real-time. We explain this point in simple words, when the acquisition system is far from a certain object, details are less clear, and when it is getting closer, everything in the scene is becoming clearer (except for boundary objects which will become out of the view). This means that at

any frame within the sequence, complementary information from both frames before and after can improve the reconstruction quality, which prevents providing real-time output. Overall, we adopt a general policy in this thesis; we worry about computation time except if there is a processing time-quality trade-off, where in this case we go for the reconstruction quality.

1.2 Context & Problem Statement

Most of the methods proposed for 3D reconstruction from 2D images use the camera motion as a cue (Temporal cue) to reveal the 3D structure. The common pipeline of such methods, which are so-called Structure from Motion (SFM), mainly relies on feature points detection, matching and 3D triangulation [Hartley 2004]. This process is followed by a global bundle adjustment to minimize the re-projection error and refine both the structure and the relative camera motion [Triggs 2000]. The density of the obtained 3D model in this case is directly related to the number of matches. In case of stereo vision, a semi dense matching could be obtained using a stereo pair that allows forming fairly good 3D model [Hirschmuller 2007]. Similarly, in case of having a set of unstructured images for a scene, where there exist several redundant laterally shifted views, up to semi-dense reconstruction could be obtained using Multi View Stereo (MVS) approaches, which mainly extend the sparse feature points correspondence to patch matching (Sometimes called PMVS, being patch based MVS) [Furukawa 2010]. This results in quite dense and visually appealing models.

Now if we come to the case of camera forward motion in urban environment, several challenges arise at different stages of the traditional 3D reconstruction pipeline. Mainly, lacking textured areas in urban scenes, which results in less feature points, and consequentially, less 3D reconstructed points. Additionally, the continuous motion of the vehicle prevents having redundant views of the scene with short feature points lifetime. Mathematically speaking, depth recovery from points correspondence of two images taken in forward camera motion (Epipole inside the image) is noisier, and even it is subject to ill-posed problems [Vedaldi 2007] for some points, than laterally shifted images. This is because in the latter case, the depth is directly proportional to disparity, while in the first case the disparity is a function of points spatial position

and depth together (this will be discussed later in details). Applying the standard MVS in this context is difficult, or results in non-dense unrecognisable 3D models. Which is also due to the fact that most of MVS methods rely on good feature matching methods such as; SIFT [Lowe 2004], SURF [Bay 2006], and recently ORB [Rublee 2011]. The quality here is defined by two features, first is to be sufficiently discriminative, that is, being distinguished from other features, and second, to be invariant to similarity transformations, such as rotation, translation and scale changes, as well as illumination variations. However, these methods have a drawback that they provide relatively small number of matches (which is not a problem when redundant views are available). In contrast, extending the number of matches by allowing more tolerant feature point's quality or using denser matching methods affects the quality of the 3D reconstruction and the relative motion estimation. From another side, extending the sparse feature point to match patches is more difficult in the assumed context because image pixels in forward motion undergo an affine transformation, while in the laterally shifted case is up to similarity transformation, whereas the photometric information are due to less change than the first case.

In the area of 3D reconstruction, assumptions about the scene rigidity has to be defined in advance. Non-rigid scenes can be encountered when we have a deformation of objects in the scene such as human faces and bodies, or due to mobile objects in the scene. Considering non-rigid scenes requires major modification in the traditional 3D reconstruction procedure since matched points are no longer static in the scene, whereas conventional stereo vision fundamentals do not apply here. In our work, as we focus on 3D urban reconstruction, possible mobile objects in the scene are pedestrians and vehicles. For both of them, there is no interest to be reconstructed. Hence, in this dissertation we consider only the rigid case.

Apart from the camera motion cue, other cues can be also employed to achieve the goal of 3D reconstruction. Here, we mainly emphasis on the spatial cues that exist in a single image, which helps to infer (some) depth information. Several cues have been exploited in research such as depth from defocus [Favaro 2008], vanishing points [Wang 2009], horizontal line [Alvarez 2010], patterns and structure [Hoiem 2007], shading variations and lighting [Alvarez 2010]. Unfortunately, most of these cues are not present in all kinds of images, and generally not robust. However, since the

last decade a new generation of methods have been developed to perform 2D to 3D conversion based on a single image using machine learning techniques. They are based on the use of exhaustive feature extraction and probabilistic models to learn depth using some priors. Although this is not the main topic in this dissertation, the idea here is to use such kind of techniques together with SFM to improve depth estimation.

Hence, this thesis mainly aims at addressing the 3D reconstruction of urban scenes while the camera is undergoing forward motion, which is still considered a challenging problem given the above mentioned points. *The goal* is to provide a complete scene 3D model that is as dense as possible and visually appealing 3D models. As we have seen, the main issue faced in the assumed context is the lack of feature points matches among the image sequence (Temporal dimension) which leads to non-dense 3D model.

1.3 Improved Structure Estimation Pipeline

In this thesis, we propose a solution that deals with the mentioned challenges by exploiting visual information in two dimensions;

1.3.1 Spatial image dimension

We exploit the spatial image information by taking advantage of appearance similarity to assume spatial belonging to same object in the 3D scene. For this aim, we are inspired by computer graphics applications where the virtual world is represented by a mesh composed of small planar patches, mostly triangles. Each of these patches has a unique colour. For a given 2D view capture of such virtual world, the relationship between every two adjacent patches is either connection (hinge or coplanar) or occlusion. By transferring this concept to real world, the 3D scene can be represented when projected to 2D by small planar patches. Now, when there are some known 3D location for a set of points in the same scene, and if it is known that those points belong to the scene structure. *i.e.* are contained in the planar patches, the plane parameters for each of the planar patches can be obtained if there is enough number of known 3D

points associated to each of them. For instance, a point cloud obtained using SFM technique. In practice, several homogeneous regions in urban scene, such as building windows, columns, side-walks and roads, are poor with feature points, whereas they can be well fitted to planar patches. The proposed approach in this case is expected to perform well. From another side, trees and other greenery areas that may be present in urban scenes violate the planar assumption, so the proposed representation may not be the best option. We *propose a solution* to deal with the later problem through an adaptive patches size in the scene based on the level of texture. Going back to the assumed representation, the 2D decomposition of the scene into planar patches is obtained using an over-segmentation of the input image, that is called superpixel segmentation, which tends to group pixels based on their appearance properties, and hence to increase their probability to belong to one surface in the 3D scene. However, there are some issues related to superpixel generation that need to be addressed:

- How much should the image be over-segmented (number of superpixels).
- What is a good superpixel segmentation method specifically adapted to our defined purpose of 3D representation.
- Are there any specific constraints on the size, regularity and boundaries of the formed superpixels.

In this thesis, we allocate *Chapters 3 and 4* to propose efficient methods and address these problems in details.

1.3.2 Temporal image dimension

We also exploit the temporal dimension. Particularly, to increase the number of feature points matches, and hence, to increase the density of the obtained 3D models. As mentioned earlier, one of the drawbacks of the good feature points matching methods is that they produce relatively small number of feature points. The *main idea proposed* in this context is to combine several feature matching methods to increase the number of matches. Furthermore, to consider a noisy dense optical flow as being a part of the matches involved in the 3D reconstruction. The main issue when using such mixture of matches is that it affects the output quality, because the number of matches

obtained using some feature matching methods (*e.g.* dense optical flow) is large and of low quality compared to such of good methods (*e.g.* obtained using SIFT). So the impact of lower quality matches will have more impact on the obtained model than the better quality ones. Therefore, our proposed solution takes this issue into account by a kind of weighted impact of each point when used to fit the scene structure. More precisely, a weighted plane fitting model. This approach raises another issue about how to define the weights, which is not trivial. Although we know that those weights should be in some sort proportional to the accuracy of the used feature matching method. Hence, we also exploit this issue in details and a learning based approach is proposed to compute the proper weights (more details in *Chapter* 5). Furthermore, by using the concept of variable weights, other factors can be also considered while allocating the weights to points. Commonly, those factors are known a priori to affect the accuracy of the 3D reconstructed points. For instance, the feature point which is matched/tracked in five frames is likely to produce more accurate 3D reconstructed point than a point reconstructed using a feature point that is only trackable in two frames . Another important point that affects the reconstruction quality is the distance to the camera, from stereo vision fundamentals we know that the error introduced in the depth estimation grows non-linearly with the distance to the camera. Moreover, at certain depth the vision based system becomes blind so that all points located after certain distance will be assigned same depth value (in the best case). Indeed, we extend the weighting model to consider other factors that we believe they affect the accuracy. The learning procedure validates the necessity of any added factor.

1.3.3 Dataset

In this thesis, we mainly rely on the image sequences provided in KITTI dataset [Geiger 2012]. The main reasons beyond this choice are;

- It provides the same assumed configuration (beside other configurations such as colour/grayscale stereo).
- Relatively high resolution images (375 × 1242 Pixels). For instance, compared to MIT DARRA urban challenge dataset [Huang 2010] (376x240) and The CMU Visual Localization Data Set (256 × 192) [Badino 2011].

- The sequences cover most of the scenarios and contains most of the various objects we see in urban environment (28 sequences in the raw data section, 21 sequences in the odometery section. In total there are around 30K images).
- Large range laser scanner data, one point cloud per image (~ 100k points per frame). We rely on the provided laser scanner data as a ground truth for any learning related method we propose.
- Precise trajectory obtained using Inertial Navigation System (GPS/IMU), which is used to validate the visual odometrey.
- Provides all necessary calibration parameters (Camera, Camera-to-GPS/IMU, Camera-to-Laser scanner, etc).
- It is being widely adopted by recent and high quality works in several domains of computer vision, for instance, [Vogel 2013, Sengupta 2013, Yamaguchi 2013, Ros 2015].

1.4 Contributions

The main contributions of this thesis are the following: (Logically ordered)

- We propose a procedure to evaluate superpixel segmentation for the goal of 3D scene representation. This procedure provides a measure that shows if a given superpixels segmentation respects the 3D geometry of a scene, which is achieved by the mean of computing the error introduced when converting a dense depth map into a triangular mesh based on superpixels. This allows selecting and improving the existing general-purpose state-of-the-art superpixel generation method to be used in any piecewise 3D reconstruction pipeline. *(This work is published as a part of [Nawaf 2014a], and detailed in Chapter 4).*
- A novel approach that aims at creating 3D geometry respecting superpixels. The superpixel generation is based on a generalized boundary probability estimation using colour and flow information similar to [Leordeanu 2012]. However, we propose a pixel-wise weighting in the fusion process which takes into account the variable uncertainty of computed dense depth using optical flow. This

method is applicable to any active/passive 3D reconstruction application. In particular, a method that models the output (*e.g.* point cloud or dense depth map) as a mesh with minimum loss of precision. *(This work is published in [Nawaf 2014a], and detailed in Chapter 3).*

- Adaptive simple local iterative clustering (SLIC) superpixel segmentation method. The original method is extended to be adaptive to the sparse feature points density for more balanced 3D structure fitting. This is achieved by the mean of new distance measure that takes into account the feature points density. And also we initialize the clustering with feature points density adapted seeds instead of the originally regular seeds. The superpixels obtained in this case are regular and limited by size, so this method is suitable to be applied if the reconstruction method requires establishing superpixels correspondence between several consecutive frames in a sequence (unlike the previous method, where superpixels are not constrained). *(This works is published as a part of [Nawaf 2014b], and detailed in Chapter 4)*

- Closed-form plane parameters estimation scheme that involves using 3D points obtained using several feature points matching techniques including a noisy dense optical flow. We use a weighted total least squares model to handle the uncertainty of each depth source. This model is employed to perform a weighted fitting of the slanted-planes structure, and hence to form the 3D model. *(This work is published in [Nawaf 2014b], extension is submitted to [Nawaf 2014-1], and detailed in Chapter 5)*

- We exploit using depth learning from single image approach together with SFM to improve the 3D structure estimation. Based on the depth estimation method from single image presented in [Saxena 2009b], we extend the proposed Markov Random Field model to include new potential functions related to 3D reconstructed points using SFM technique, and also constrained by the limited planar motion of the vehicle. The obtained results are improved with respect to the depth computed using single image. However, the method proposed in the previous point provides better outputs. *(This work is published in [Nawaf 2012], extended in [Nawaf 2013] and detailed in Appendix B)*

1.5 Thesis outline

This thesis is structured as follows:

- **Chapter 2** surveys the current state-of-the-art and contrasts the proposed 3D reconstruction approach with respect to previous works. We also provide a literature review on superpixel segmentation methods as being an essential part that we rely on in our framework.

- **Chapter 3** presents the first proposed superpixel segmentation method, which is non-constrained gradient based aims at generating superpxiels for the goal of creating mesh representation that respects the 3D scene geometry. Unlike the method proposed in *Chapter 4*, there are no constraints on the size, shape and number of superpixels. This method uses a new fusion framework which employs both dense optical flow and colour images to compute the probability of boundaries. The main contribution of this approach is that we introduce a new colour and optical flow pixel-wise weighting model that takes into account the non-linear error distribution of the depth estimation from optical flow. We also introduce the evaluation procedure we are based on to assess the quality of superpixel segmentation in terms of respecting the 3D geometry.

- **Chapter 4** presents the second proposed superpixel segmentation method, which is based on adaptive simple local iterative clustering (SLIC). This method differs from the one proposed in *Chapter 3* that it aims at producing constrained size superpxiels, which is an important property when the used 3D modelling approach involves establishing explicit/implicit superpixel correspondence between views. The superpixels shape and size is locally controlled based on an input density map. We adapt the method for the application of 3D planar structure fitting by using the feature point density as an input to control locally the size of superpixels for balanced feature points distribution over superpixels.

- **Chapter 5** explains the proposed planar 3D reconstruction pipeline from monocular image sequence. The focus is on the following components: first, a sparse 3D reconstruction scheme using several feature matching methods. Second, a frame-to-frame superpixel correspondence method. This method is essential in

the proposed pipeline as it is used to integrate the temporal information along the image sequence. Third, a boundary probability map computation based on colour and flow information. The boundary information are used in the planar fitting procedure to integrate the spatial depth information. Fourth, the weighted structure fitting scheme which is based on a total least square model. Then we present the experiments to evaluate the proposed method, with a detailed discussion that highlights both the advantages and the limitations.

- **Chapter 6** Draws the conclusions, and possible future directions that are aimed to address the limitations and the further improvements.

- **Appendix A** list the publications which are related to the topics presented in this thesis.

- **Appendix B** This chapter presents a work which is not in main track of thesis thesis. It presents a novel approach to improve 3D structure estimation from an image stream in urban scenes. The idea is to introduce the monocular depth cues that exist in a single image, and add time constraints to improve the 3D structure estimation with respect to structure from motion traditional techniques. The scene is also modelled as a set of small planar patches obtained using over-segmentation, and the goal is to estimate the 3D positioning of these planes. We propose a fusion scheme that employs Markov Random Field (MRF) model to integrate spatial and temporal depth features. The proposed MRF model is then solved using convex optimization techniques.

2 Background

Contents

In this chapter, we give an overview of related works. First, we review available techniques for urban reconstruction, which can be divided into four categories [Musialski 2013] depending on the used acquisition technique; image-based or active-sensing (Laser scanner). For each case it could be applied in the context of aerial imaging or ground-level acquisition setup. Being focused on image-based techniques, we review the general dense 3D reconstruction from 2D images techniques. We explain the difficulties faced with these methods when applied to our context. Next, we move to the particular case of monocular image sequence and the related techniques.

As many of successful methods in literature adopt the superpixel representation of the scene, we also review the state-of-the-art methods for superpixels generation, in particular, the methods which are employed in 3D world modelling. We also discuss their advantages and disadvantages, and the ideas and constraints we considered in our proposal in order to implement an improved solution.

Finally, we review depth estimation from single image approaches, specifically, the general methods based on machine learning techniques. Also we mention the existing methods which fuse both spatial and temporal information to improve depth estimation.

2.1 Urban Reconstruction Techniques

Several techniques have been used in order to perform urban 3D reconstruction. Here in this section we mention the modalities used for this purpose rather than the previously assumed configuration (Mobile vehicle with forward motion). The main purpose is to show their advantages and current drawbacks. These techniques include aerial imaging and the laser scanner (LiDAR).

Urban reconstruction from aerial imaging methods uses the same concepts of multi-view geometry, although these methods rely more on matching line segments over multiple views instead of the common point matching procedure in traditional multi-view approaches [Baillard 1999]. The triangulated lines in 3D are used to compute the roof planes orientation. Hence, piecewise planar rooftops are reconstructed. A successful method for automatic 3D reconstruction proposed in [Zebedin 2008] requires each building to be segmented, and then converted into simplified models composed of planes and surfaces of revolution. We also use a similar concept of planar reconstruction, although we have a different aim by targeting street-level reconstruction rather than top-view reconstruction.

Laser scanners (LiDAR) represent an alternative technique to computer vision based methods to perform sparse depth reconstruction. LiDARs are based on push-broom shaped pulse emitters/receivers, which can be static or rotating depending on the acquisition configuration. This allows to compute the depth for a set of points in the scene by measuring the phase shift between the emitted and the received pulse. In general, they provide more accurate measures compared to image-based techniques, in addition to its larger range (modern ground-level LiDARs has a range of 80 meters).

Similar to image-based techniques, urban reconstruction LiDAR related techniques are also used in two contexts; aerial and ground-level acquisition. In the aerial con-

text, the point could obtained by LiDAR is fitted to geometric primitives in order to reconstruct surface models [Zhou 2013]. Other techniques employ both aerial images and LiDAR data to model the surfaces. For instance, the method proposed in [Sohn 2007] fuses segmented buildings and range measurements in order to refine the reconstruction. However, similar to aerial image-based techniques, the obtained reconstruction contains mostly rooftops, whereas vertically oriented surfaces (such as building façades) are occluded. Moreover, the overall obtained model has lower resolution. Nevertheless, it has some advantages such as the greater coverage especially in non-accessible areas and requires relatively smaller number of images. Commercial products of big companies such as Apple, Google and Acute3D, provide successful reconstructed city 3D models produced using semi-automatic techniques relying on the above mentioned approaches. Again, the available created maps deliver no street-level detail. In the context of ground level LiDAR, there exist recently several attempts [Früh 2004, Pandey 2011, Geiger 2012, Smith 2009] to provide 3D point clouds of urban environment using mobile vehicle equipped with 360° rotating LiDAR scanner. This allows to provide sparse depth map of the scene. Further processing is then needed to provide 3D models through surface reconstruction techniques (discussed in details in the next section). Modern LiDARs, such as the one used in [Geiger 2012, Pandey 2011][1], provide up to 100K points per measuring cycle which covers 360°. The measures are arranged in horizontal lines (Modern LiDARs provides 64 lines). To create colourful 3D models, RGB cameras are used together with LiDARs to provide colour information. Based on the configuration used by [Geiger 2012], from a looking forward RGB camera which is 0.5 MB resolution, the depth can be obtained for around 16K points. This represents 0.032% of the pixels in the image. Which is another significant weaknesses related to this approach beside the issues we discussed earlier that are related to cost and system implementation. Hence, image-based 3D reconstruction is still an open problem in 3D reconstruction domain. Related works for urban reconstruction from monocular image sequence will be reviewed after introducing the literature review of 3D reconstruction from 2D ground-level images in the next section.

[1] Both implementations use Velodyne HDL-64E LiDAR system

2.2 Structure Estimation From 2D Images

Structure from motion (SFM) methods aim at creating 3D point cloud from 2D images. Given the ongoing development in feature detection, description and matching, efficient state-of-the-art SFM techniques are capable of providing high quality sparse 3D point clouds [Agarwal 2009, Frahm 2010, Crandall 2011, Pollefeys 2008]. The common SFM pipeline consists of feature matching, robust relative motion estimation, which mostly employs RANSAC procedure, and finally simultaneous structure and motion bundle adjustment [Lourakis 2009, Wu 2011] to minimize the re-projection error and refine the structure. As mentioned earlier, the density of the reconstructed point cloud is limited by the number of matched features per frame. Also, the quality of the reconstructed point cloud is related to feature points lifetime (the number of frames a feature point is tracked). This later fact is limited by the context (*e.g.* the point leaves the viewed scene). However, most of the 3D reconstruction and modelling techniques rely on SFM output as initialization. In this thesis, we do not aim at improving SFM pipeline, whereas we use it as a tool.

Towards more detailed 3D reconstruction, many methods have been proposed to extend the sparse SFM to provide denser representation of a scene. In particular the Multi-View Stereo (MVS) algorithms that aims at producing very dense point clouds or surface meshes [Vogiatzis 2007, Furukawa 2010, Hiep 2009] by employing the photo-consistency constrain across multiple views. For instance, the patch based MVS [Furukawa 2010] uses SFM as an initial solution, then it matches small rectangular patches in the scene by minimizing a photometric discrepancy function. This allows to grow the initial key points to neighbouring points under the photometric criteria and hence to have a denser reconstruction. This results in a semi-dense cloud of patches which is later employed in surface reconstruction. As mentioned earlier, this approach performs poorly in low textured areas, and requires redundant views [Mičušík 2010]. The 3D models we obtained using MVS applied to KITTI dataset [Geiger 2012] are not recognizable due to low density point cloud. Hence, is out of the scope of this thesis to provide a full review for other general MVS methods. However, we refer the reader to [Musialski 2013] for a complete survey of MVS and urban reconstruction.

According to our knowledge, no prior work uses several feature matching methods

together in a weighting scheme as we propose in our approach. Nevertheless, some works use a combination of several tolerance levels of salient points quality. For instance, in [Hiep 2009], after generating the SIFT matches and estimating the pose, a denser point cloud is generated by extracting more matches based on quality tolerant DOGs and Harris detectors, and using block matching procedure. At the end, the matches are categorized as false or true, and the mesh generation is based on true matches and deals with them equally. In our approach, each matching point is associated with a learned weight that controls its impact in fitting the structure.

In the context of urban reconstruction, a manifold surface reconstruction [Lhuillier 2013] is based on the sparse 3D point cloud obtained using SFM. The 3D model generation is an independent post processing that considers only the obtained point cloud. The goal is to generate a mesh representation of the scene, then to use the colour information to produce textured 3D model by interpolation. The keypoint is to perform 3D Delaunay triangulation by ray tracing and iterative region growing while maintaining the manifold property. Here, we could also mention the surface reconstruction approaches which also can be also applied in this context. For instance, the Poisson reconstruction [Kazhdan 2006] has been widely applied to object reconstruction. In this method, an implicit function is derived from a Poisson equation, providing the best match between the gradient of such function and the normals of the input point cloud. However, this approach tends to produce smooth surfaces which is not suitable for urban environments. Other concurrent solution is the triangulation based surface reconstruction such as the method proposed in [Marton 2009]. Which is based on incremental surface growing by incrementally selecting k-neighbourhood in a sphere, and projecting them on a plane that is approximately tangential to the surface formed by the neighbourhood. The points are then pruned by visibility and connected to consecutive points by edges to form the triangles. All mentioned triangulation and surface reconstruction methods [Lhuillier 2013, Kazhdan 2006, Marton 2009] do not take into account the image colour and appearance information, and more important the object boundaries. The reconstructed models using such methods are generally deformed in low density feature points, whereas occluded objects are fused with the background. Another method for urban scenes reconstruction which employs images spatial information is the 3D piecewise planar dense reconstruction approach [Mičušík 2010], which uses images from Google Street-view. This method

adopts the superpixels representation to deal with the problem of correspondence ambiguities in low texture areas. It assumes a predefined geometry of the scene as each superpixel is assigned a depth and one of the three urban scene normals. A Markov Random Field (MRF) model is formed to handle the geometric properties and the neighbouring coherence of superpixels. In our approach, we also adopt the superpixel representation. However, we use a closed-form model to fit the structure and to enforce smooth depth transitions between coplanar superpixels. Beside the piecewise planar representation, other approaches use simplified geometric assumptions of the scene. For instance, the 3D reconstruction method proposed in [Cornelis 2008] models the urban scene as a set of vertical folded planes that could be reconstructed using vertical lines matching. Folds detection relies on a stereo camera pair. The method employs object recognition to detect some objects, such as cars, to help refining the trajectory, and also treat such objects independently. This approach does not generalize well to all standalone objects that are present in urban scenes. Meanwhile, in our approach we control the scene geometry (*i.e.* number of independent planes) through a co-planarity enforcing parameter. In the street-side reconstruction context, the method proposed in [Xiao 2009] provides urban reconstruction of building's façades. The method uses a classifier based on colour and texture to segment buildings from ground, vegetation and sky. The buildings are reconstructed using a rectilinear model. A similar method that is based on the rectilinear assumption is presented later in [Vanegas 2010] which assumes a Manhattan world 3D structure. The 3D model is generated based on constrained shape and multi-view photoconsistency grammar. In both of the aforementioned methods, the authors consider the architecture of a specific building and assume more redundant views. Whereas our proposition aims at full scene reconstruction.

2.3 Superpixels For 3D Scene Representation

A wide range of methods aiming at 3D modelling employ a piecewise representation of the scene using superpixels. This includes the areas of stereo matching [Yamaguchi 2012], optical flow [Vogel 2013], monocular optical flow [Yamaguchi 2013], 3D scene modelling [Saxena 2009b, Mičušík 2010, Bódis-Szomorú 2014], occlusion boundaries detection [Sun 2014, He 2010]. These methods use a variety of superpix-

els generation techniques which provides different output in terms of size, shape, regularity and number of superpixels. Apart from speed, there is no other explicit justification for the choice of the superpixels generation algorithm. In this thesis, we consider this issue by proposing a superpixels evaluation method for the purpose of 3D scene modelling.

Several superpixels generation methods use colour, texture and position information as features. An early example for colour based superpixels method is the graph-based model [Felzenszwalb 2004]. In this method, the pixels are represented as nodes and the edges are computed as the similarity between nodes. Then, superpixels are obtained by applying the minimum spanning tree algorithm. In this method, there is no restriction on the shape/number/alignment of the resulting superpixels. In contrast, a remarkable property in several other superpixels methods is that they consider the regularity and the arrangement of the superpixels. For instance, the simple linear iterative clustering (SLIC) method [Achanta 2012] introduces a new distance measure that involves the position of the pixel as well as colour. This distance measure is taken into account when a label is assigned to each pixel. Hence, there is a limit on the size of the formed superpixels subject to a regularization parameter.

Among the methods that tend to produce a grid aligned superpixels are SEEDS (Superpixels Extracted via Energy-Driven Sampling) [Van den Bergh 2012a] and Turbopixels [Levinshtein 2009]. SEEDS uses a fixed number of uniformly distributed seeds (rectangular shape clusters) for initialization, and then refines the boundaries between them based on minimizing an energy function that consists of a colour distribution and boundary terms. Opposite way, Turbopixels method starts with rectangular grid distributed clusters centers and then it grows them based on a *Boundary Velocity* which is computed from local image gradients. These regularity aware methods produce superpixels with relatively similar size and regular shape, and the output is more or less aligned to a grid. For 3d modelling purposes, having such property is not necessary as it does not reflect the real world structure. Moreover, it produces unnecessary additional number of superpixels which will result in more mesh faces in the reconstructed 3D model (*e.g.* an extreme case if we have a single homogeneous surface in the image). However, constrained superpixel size and shape is needed in case it is necessary to establish superpixels correspondence between overlapping

views. Otherwise, a large error may be introduced by mismatches. *In Chapter 3 and 4*, we propose two superpixel generation methods, regularity aware and regularity aware-less. In the first method, the number of superpixels varies based on the nature of scene, whereas it is controllable in the second method . For more details about colour-based superpixels methods we refer the reader to the review in [Achanta 2012].

Recently, due to the increasing use of depth information (obtained from different types of sensors) for computer vision tasks, superpixels generation including depth appears as an important issue. Specially, it is important to know how to incorporate/fuse depth, and what superpixels method to use. Several methods [Jebari 2012, Van den Bergh 2012b] already exist for this purpose. For instance, the method proposed in [Jebari 2012] fuses depth information in a watershed based superpixels algorithm. The fusion approach takes the maximum of the Laplacian computed from a grayscale and depth image. In [Van den Bergh 2012b] the authors proposed a SLIC [Achanta 2012] based method where they incorporate depth in the distance function. Both methods use global weight for each channel/layer (*e.g.* colour and depth). That means all pixels of a particular channel have the same weight. Our both proposed methods are inspired by these methods in including optical flow. Further more, we introduce a new distance measure for the regularity aware SLIC based method, and local depth weighting for the gradient based method (as been justified earlier in Chapter 1).

2.4 Depth Learning From Single 2D Images

In computer vision, structure from motion (SFM) has taken a great attention by researchers, it is considered as one of the well-studied problems. However, most of the efforts are focused on a certain number of aspects. For instance, improving feature points matching [Lowe 2004], formalizing better constraints to improve relative camera pose estimation [Pollefeys 2008], robust methods for outliers rejection [Raguram 2008], linear/non-linear re-projection error optimization and bundle adjustment [Triggs 2000], formalizing a set of constraints on more than two frames [Hartley 2004]. Most of these contributions do only consider temporal information that results from image stream variation with respect to time, without trying to analyse the monocular depth cues that are present in every single image.

From another side, several monocular cues that exist in a single image have been exploited by researchers, that includes; vanishing points [Hoiem 2006], shades and shadows [Savarese 2007], haze, patterns and structure [Lindeberg 1993]. Unfortunately, most of these cues are not present in all kinds of images, and they require specific settings. Meanwhile, a new generation (since last decade) of methods that perform 2D to 3D conversion using a single image have been proposed. Generally these methods have no constraints and are based on the use of exhaustive feature extraction and probabilistic models to learn depth. An early approach that attempts to estimate general depth of an image is proposed in [Torralba 2002], which employs Fourier spectrum to compute a global spectral signature of a scene to estimate the average depth of the image scene. Later on, an innovative attempt to perform 3D reconstruction from one image was proposed in [Hoiem 2005]. In this approach, the image is first over-segmented into superpixels, then each superpixel is classified as sky, vertical (objects) or planar (ground). It employs a wide set of colour, texture, location, shape and edge features for training. Finally, the vertical region is "cut and folded" in order to create a rough 3D model. Although this method has been improved later by considering some geometric subclasses (centre, left, right, etc.) in [Hoiem 2007], the "ground-vertical" world assumption does not apply for wide range of images. A similar concept is proposed in [Pfeiffer 2012], which is extended to motion classes such as "right headed, left headed, oncoming and static background". Such a medium-level representation of 3D scenes named "stixel world" allows the extraction of multiple objects in complex inner city scenarios, including pedestrian recognition and detection of partially hidden moving objects. The best class assignment and the dependencies between neighbouring stixel labels are dynamically defined from prior knowledge about the current local 3D environment and temporal information using a conditional MRF. A more general method is proposed in [Liu 2010] which estimates the depth from a single image based on some predicted semantic labels (sky, tree, road, etc.) using multi-class pixel-wise image labelling model. Then, the computed labels guide the 3D estimation by establishing a possible order and positioning of image objects. In [Alvarez 2012], a convolutional neural network based method is proposed to learn features from noisy labels to recover the 3D scene layout of a road image. It combines colour planes to provide a statistical description of road or side-walk areas (*i.e.* horizontal ground), that exhibits maximal texture uniformity. In [Sturgess 2009],

11 object classes (road, building, sky, tree, side-walk, car, etc.) are used for labelling. Motion and structure features (height above the camera, distance to the camera path, projected surface orientation, feature track density and residual reconstruction error, inferred from 3D point clouds) and appearance features (textons, colour, location and HOG descriptors) are combined thanks to a Conditional Random Field model. Another general approach has been proposed in [Saxena 2009b] which does not have initial assumption about scene's structure. It proceeds by over-segmenting the image similar to [Hoiem 2005]. The absolute depth of each image patch is estimated based on learning a MRF model, where a variety of features that capture local and contextual information is employed. As an extension, the authors proposed a model to create 3D reconstruction from sparse views. We see later in Appendix B that a part of our work is inspired by this method. The idea that we propose here as contribution is to adapt (and train) our method to road scenes and forward motion. The added constraints have been defined to improve relative motion. Also the optical flow based SFM provides approximate but denser feature points as an alternative to points triangulation since the later tends to fail near image plane axes. Another improvement in the proposed work is that we compute occlusion boundaries based on the motion between two frames which is more robust and accurate than a multi-segmentation based approach using a single image.

In the context of combining both spatial and temporal depth information, a method that combines SFM with a simultaneous segmentation and object recognition is proposed in [Sturgess 2009], it targets road scene understanding. The task is achieved through a conditional random field model which consists of pixel-wise potential functions that incorporate motion and appearance features. The author claims that it overcomes the effect of small baseline variations. In our method, we adopt direct depth estimation rather than object recognition. However, similar to [Sturgess 2009], our method is also supervised and learning oriented, we benefit from computed features to capture contextual information and to learn depth. In comparison with our approach, we use small planar patches to model the world rather than the pixel-wise approach used in [Sturgess 2009] as we think they better describe the world around us. This idea is also supported by the experimental results in [Saxena 2009a]. The method in [Li 2008] proposes to combine sparse reconstruction using SFM with a surface

reconstruction using MRF optimization. The main difference with our approach is that in this work they do not use the superpixels segmentation to model the depth of objects but a 2D tetrahedral mesh segmentation to fit objects surfaces. In our study we use superpixels as we think they preserve more neighbouring relationships between uniform 2D surfaces and temporal-consistency. Another approach in the same context is to use the semantic structure from motion approach [Bao 2011] which is based on a probabilistic model. The proposed model incorporates object recognition with 3D pose and location estimation tasks. Also it involves potential functions that represent the interaction between objects, points and regions. Another approach that combines both spatial and temporal information is proposed in [Cigla 2012]. A stereo matching algorithm with ground plane and temporal smoothness constraints is proposed for vehicle control and surveillance applications. In this paper, the authors exploit the geometry of the scene (the road plane geometry) and a vertical damping scale in order to enforce temporal consistency. This method enables to relax the smoothness of disparity maps along vertical axis and to prevent disparity resolution loss due to lack of texture or occlusions due to motion. Spatial and temporal information are aggregated via permeability filter and guided filter. Compared to other aggregation methods, this approach does not exploit contextual information nor the intrinsic complementarity of spatial and temporal information.

2.5 Discussion and Conclusion

To summarize, the main innovations that we propose in this thesis, in comparison with the state-of-the-art are:

- Although many methods use a piecewise representation for 3D modelling, no criteria have been defined for the selection of the superpixel segmentation for this purpose. We consider this lack in our proposal and a new measure is defined for the goal of 3D modelling.
- Existing superpixel methods are for general purpose. Improvements are made in terms of computation speed and respecting hand-made ground truth segmentation. None of them consider the goal of 3D modelling and meshing. Here, we propose two methods towards this goal, each method is adapted to certain

scenario:

- The first method produce non-constrained superpixel segmentation which respect the 3D geometry of the scene. This method is applicable to any active/passive 3D reconstruction application. In particular, a method that models the output (*e.g.* point cloud or dense depth map) to a mesh with minimum loss of precision. In this method, there is no constraints on the shape/size of superpixels. The superpixel generation is based on a generalized boundary probability estimation using colour and flow information similar to [Leordeanu 2012]. However, we propose a pixel-wise weighting in the fusion process which takes into account the variable uncertainty of computed dense depth using optical flow.
- In some 3D modelling approaches, it is necessary to establish superpixels correspondence between two or several consecutive frames in a sequence. In this case, non-regular superpixels such as the output of the first proposed method makes finding such correspondence difficult. Moreover, in some approaches, the superpixels are used for planar fitting to create 3D models. This fitting is based on a 3D point cloud which is computed from overlaid feature points on the superpixels. A balanced feature points distribution over superpixels is an advantage in this case (more details in *Chapter 4*). Here, we propose an adaptive simple local iterative clustering (SLIC) based segmentation that deals with both aforementioned issues. First, the original method is extended to be adaptive to the feature points density for more balanced 3D structure fitting. This is achieved by the mean of new distance measure that takes into account the sparse points density. And also we initialize the clustering with feature points density adapted seeds instead of the originally regular seeds. Second, the superpixels obtained using clustering based approach are regular and limited by size, so the superpixels correspondences can be established efficiently.

- As we have seen in 3D reconstruction from monocular image sequence related works, this area remains an open problem. The solutions that have been proposed assuming same configuration remains relatively few and went in the direction of surface reconstruction rather than benefiting from appearance

similarity. In our proposition, we benefit from the appearance similarity by adopting the piecewise representation. For fitting the planar structure, we propose a closed-form plane parameters estimation scheme that involves using 3D points obtained using several feature points matching techniques including a noisy dense optical flow. We use a weighted total least squares model to handle the uncertainty of each depth source. This model is employed to perform a weighted fitting of the slanted-planes structure, and hence to form the 3D model.

- Independent from the main track of this thesis, we exploit improving 3D structure estimation by fusing SFM sparse output with monocular depth estimation learned from single image as in the approaches we reviewed already. Based on the monocular depth estimation method proposed in [Saxena 2009b], we extend the proposed MRF model to include new potential functions related to 3D reconstructed points using SFM technique, and also constrained by the limited planar motion of the vehicle.

In this chapter we have explained what are the main argues which motivate the approaches selected/proposed in this study.

3 Superpixel Segmentation for 3D Scene Representation and Meshing

Contents

In this chapter, we propose a gradient based superpixel segmentation method for the goal of creating mesh representation that respects the 3D scene geometry. Unlike the method proposed in *Chapter 4*, there are no constraints on the size, shape and number of superpixels. In this method, we propose a new fusion framework which employs both dense optical flow and colour data to compute the probability of boundaries. The main contribution of this approach is that we introduce a new colour and optical flow pixel-wise weighting model that takes into account the non-linear error distribution of the depth estimation from optical flow. Experiments show that our method is better

than the other state-of-the-art methods in terms of smaller error in the final produced mesh.

3.1 Introduction

Superpixels can be defined as an over-segmentation of an image which is obtained by dividing the image into small homogeneous colour/texture regions so that each of them belongs to only one object/surface. Superpixels has been widely used in many computer vision tasks such as object detection [Shu 2013], depth estimation and 3D scene modelling [Saxena 2009b, Mičušík 2010, Bódis-Szomorú 2014], stereo vision [Yamaguchi 2012], optical flow [Vogel 2013], monocular optical flow [Yamaguchi 2013], occlusion boundaries detection [He 2010] and scene segmentation [Jebari 2012, Silberman 2012, Van den Bergh 2012b]. In the aforementioned works, the choice of the used superpixel segmentation algorithm is not explicitly justified. However, the clustering based algorithm [Achanta 2012] is being increasingly adopted in recent works due to its real-time performance. We observe also that the graph-based segmentation method [Felzenszwalb 2004] comes second in its popularity for the purpose of 3D modelling. Both mentioned methods provide an output that largely vary in terms of size, shape, regularity and number of superpixels. Here arises the question about what method is better for the purpose of 3D modelling, and what is the criteria to make such decision. In this chapter, we consider this issue by proposing a superpixels evaluation method for the purpose of 3D scene modelling. This allows evaluating and comparing the existing superpixel generation methods. Moreover, we propose a *new superpixel generation pipeline* which provides better superpixels for 3D representation and meshing.

In this work, we provide superpixel segmentation method that can be used as a tool to decompose a scene into piecewise planes. In particular, to reconstruct a scene using triangular mesh based on the obtained superpixels, so that the mesh respects the 3D geometry of the scene. A triangle mesh comprises a set of triangles that are connected by their common edges or corners. We are motivated by the fact that many graphics softwares represent 3D world structures by meshes. Moreover, modern software packages and hardware devices can operate more efficiently on meshes compared

to the massive cloud of points. Also, meshes have an advantage of being compact to represent continuous structures. Therefore, *our aim* is to propose a method that is applicable to any active/passive 3D reconstruction application. In particular, a method that models the output (*e.g.* point cloud) to a mesh with minimum loss of precision.

Superpixels are mainly computed using colour information of an image [Achanta 2012, Felzenszwalb 2004, Levinshtein 2009]. Recently, depth and/or flow information have been used with colour [Van den Bergh 2012b, Leordeanu 2012, Silberman 2012]. We believe that flow information is an essential source based on the fact that spatially uniform regions have continuous flow whereas occlusion boundaries are often associated with flow disturbances. Hence, flow information can be used to detect boundaries. Moreover, in combining colour and flow, false boundaries in colour based segmentation can be identified.

In our method, we mainly target outdoor scenes. In this case, the popular structured light based depth sensors fail due to sunlight, which makes depth computation difficult. As an alternative solution, we rely on stereo vision to obtain depth information computed from optical flow using a pair of images of the target scene. Hence, we propose a fusion scheme that incorporates a dense optical flow and colour images to compute superpixels and generate the mesh. In contrary with the methods proposed in the literature [Leordeanu 2012, Van den Bergh 2012b], our fusion method takes into account (a) the non-linear error distribution of the depth estimation obtained using optical flow; (b) the fact that it has a limited range; and (c) that it could not be computed in parts of the image due to the view change between two images. To incorporate such information, we introduce a pixel-wise weighting to be used while fusing boundary information using optical flow and colour. Hence, our contribution is a novel locally adaptive weighting approach.

3.2 Superpixels Evaluation Method

In image segmentation area, evaluating the performance of the proposed approaches is often based on testing the output segmentation with a ground truth. The obtained score in this case is computed based on how accurate the boundaries provided by the

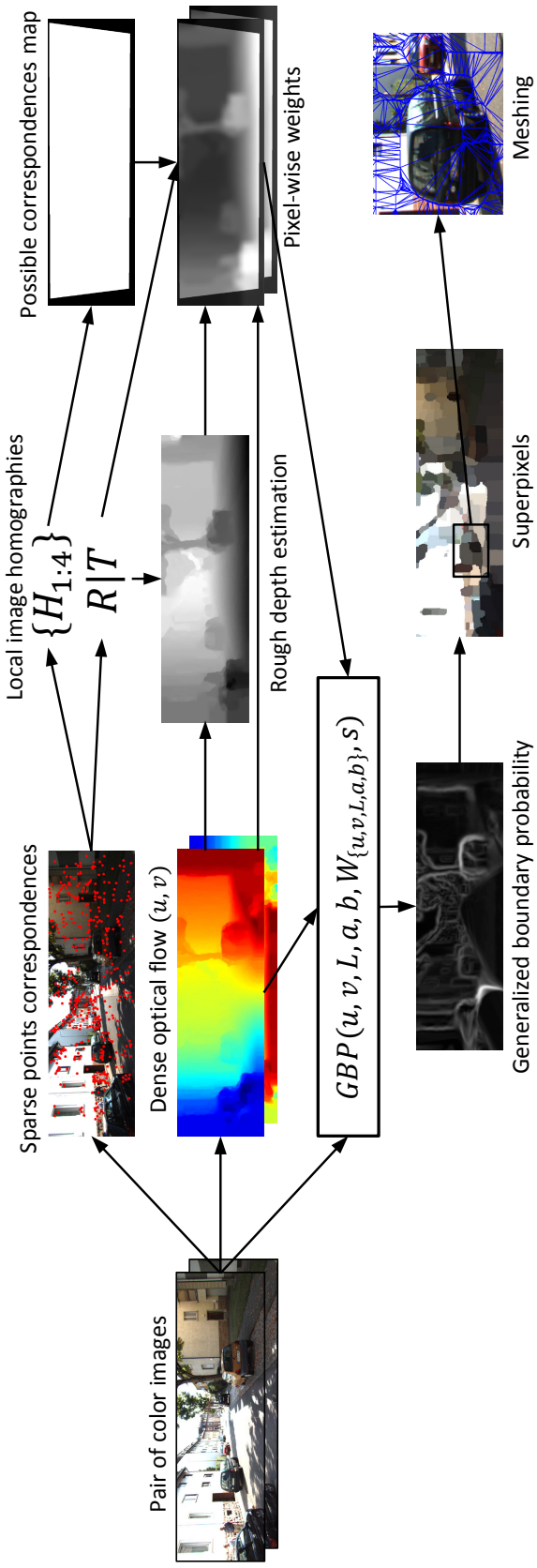


Figure 3.1: Overview of the proposed superpixel method.

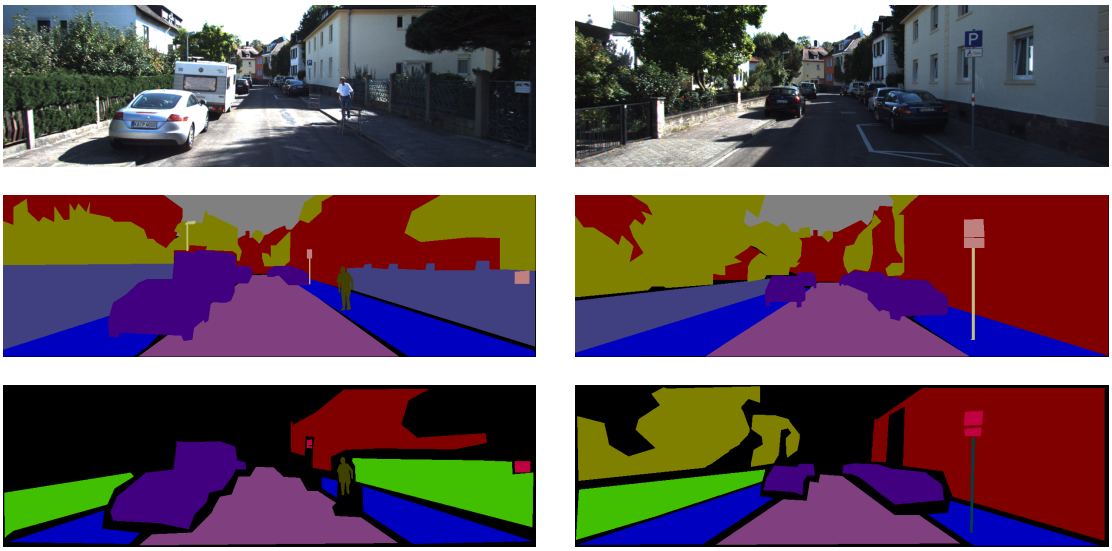


Figure 3.2: Two frames from KITTI dataset [Geiger 2012] (first row), and the hand-made ground truth segmentation as provided in [Sengupta 2013] (second row) and provided in [Ros 2015] (third row). The idea is to show the significant difference in terms of number of labels, level of details and localization of boundaries.

algorithm coincide with the ground truth. In practice, this ground truth is hand-made and can vary based on many aspects. For instance, The same user may not be able to produce the same segmentation again. In KITTI dataset, state-of-the-art methods involve a hand-made validation for the segmentation method. Figure 3.2 shows two examples of two frames. For each, we provide two different segmentations which are treated as a ground truth by the methods proposed in [Ros 2015, Sengupta 2013]. Furthermore, the well known segmentation dataset that we encounter in most of the segmentation methods (as well as superpixel segmentation methods) is The Berkeley Segmentation Dataset and Benchmark (BSDB) [Martin 2001], where it is mentioned that *"The human segmented images provide our ground truth boundaries. We consider any boundary marked by a human subject to be valid"*.

Given these facts, we seek a new evaluation method that is not subject to human assessment from one side, while it is specifically dedicated for the proposed application which is 3D modelling from another side. For this aim, we propose to assess the quality of superpixel segmentation for the goal of 3D modelling by analysing the error introduced by the 3D mesh generated based on such segmentation, with respect to the original depth map provided with the used dataset. Figure 3.1 shows an example

of an input image pair, obtained superpixels and corresponding 3D mesh (last row in the Figure). The mesh is obtained by dividing the image into a set of triangles that covers the whole image, and each triangle lies completely in one superpixel (more details in Section 3.3.5). Our method is evaluated and compared to state-of-the-art superpixels methods using the KITTI dataset [Geiger 2012] which is provided with depth ground truth.

3.3 Superpixels Generation Scheme

The block diagram of our proposed method is illustrated in Figure 3.1. Starting from a pair of colour images, we compute a sparse feature points correspondences. These sparse feature points are used to recover the relative motion of the two images, and to compute local homographies that are used to define a mask of the overlap between the two images. At the same time, based on the pair of the input images, we compute a dense optical flow, which is used to obtain a rough dense estimation of the scene. Then, we use the relative motion estimated parameters and the mask of overlap to compute the pixel-wise weights for each optical flow channel. Then, we employ a global boundary probability generator that takes as input: (a) the two channels of the optical flow; (b) the three layers of one input colour image (in CIELAB colour space) and (c) the pixel-wise and layer-wise learned weights. This step is followed by watershed segmentation to generate the superpixels. Finally, a mesh representation is obtained based on the superpixels. Each of these steps is described in the following subsections.

3.3.1 Relative Motion Recovery

An accurate relative motion $[R|T]$ is needed to compute a depth map, to estimate the pixel-wise weights and also to perform a minor outliers correction of the optical flow. For this purpose, we use a traditional approach by first performing SIFT feature points matching [Lowe 2004] on the image pair, and estimating the Fundamental matrix [1] using RANSAC procedure. Then, given the camera intrinsic parameters[1], we compute the Essential matrix[1] that encodes the rotation and translation between the

[1] For detailed explanation about epipolar geometry fundamentals we refer to [Hartley 2004]

(a) (b)

Figure 3.3: Two depth maps computed using two pairs of images that shares the same first image. (a) The image pair are shifted horizontally (stereo pair); (b) The image pair are obtained with dominant forward motion (Epipole near the center, borders problem).

two images. Before extracting $[R|T]$ we perform rank correction on the essential matrix by forcing the two eigen values to be equal by taking their mean, and setting the third eigen value to zero. Now, $[R|T]$ can be extracted using SVD according to the method proposed in [Hartley 2004]. Note that the translation at this step is computed up to scale. Which is enough for the proposed method (see Equation 3.4 for clarification).

3.3.2 Dense Optical Flow and Depth Map Estimation

The usage of optical flow in this work is essential. It helps to identify the spatial uniformity in the scene and hence it works as a complement to colour images. We adopt the dense optical flow underlying median filtering method proposed in [Sun 2010b] (We use the publicly available code [Sun 2010a]). Among the proposed variations, we use Classic-C method which involves minimizing the classical optical flow objective function:

$$\begin{aligned} E(\mathrm{u},\mathrm{v}) = \sum_{i,j} & \rho_D(I_1(i,j) - I_2(i+u_{i,j}, j+v_{\mathrm{i},j})) \\ & + \lambda[\rho s(u_{i,j} - u_{i+1,j}) + \rho s(u_{i,j} - u_{i,j+1}) + \\ & \rho s(v_{i,j} - v_{i+1,j}) + \rho s(v_{i,j} - v_{i,j+1})], \end{aligned} \tag{3.1}$$

where u and v are the horizontal and vertical components of the optical flow field to be estimated from images I_1 and I_2, λ is a regularization parameter, and ρs and ρ_D are the data and spatial penalty functions. The Classic-C method uses a Charbonnier penalty term $\rho(x) = \sqrt{x^2 + \epsilon^2}$ and 5×5 median filtering window size. This method showed to have better occlusions handling and flow de-noising. Additionally, we perform a minor

outliers detection and correction based on the recovered fundamental matrix. Given the dense points correspondences obtained by the optical flow, we compute a simple first-order geometric error (Sampson distance) for each point. We allow more relaxed (3-5 times) distance threshold compared to the average distance of the selected *inliers model* computed using the sparse SIFT features (in the previous Section). The flow vectors that exceed this threshold are replaced by linearly interpolated new values.

For dense depth map computation, we apply the Direct Linear Transformation (DLT) triangulation method followed by *structure only* bundle adjustment, which involves minimizing a geometric error function as described in [Hartley 2004]. We use the Levenberg-Marquardt based framework proposed in [Lourakis 2009]. In the special case of *close to degenerated* configurations (*e.g.* epipole inside the image), computing the depth map in the epipole's neighbouring is difficult. In this particular case we calculate a rough relative depth map by removing spatial correlation from the magnitude of the optical flow. This correlation results from the presence of x and y in the optical flow equation (see Equation 3.3). To remove this correlation, we first search for the correlation centre $(\hat{c}_x, \hat{c}_y)$ by maximizing the following pairwise correlation formula:

$$\underset{\hat{c}_x \hat{c}_y}{\arg\max} \sum_{ij} \sqrt{(i-\hat{c}_x)^2 + (j-\hat{c}_y)^2} . \sqrt{u_{ij}^2 + v_{ij}^2} \tag{3.2}$$

where i and j are the image coordinates, u and v are the optical flow components. Then, we divide each point in the optical flow magnitude by the euclidean distance to image centre shifted by $[\hat{c}_x\ \hat{c}_y]$. Figure 3.3b shows an example of an approximation of the depth map computed using this method. The input image pair in this case are taken with the same camera moving forward. Applying traditional triangulation approach to obtain the depth in this case results in undefined depth in the epipole point's [2] neighbourhood. Having such undefined depth for some points in the image prevents integrating the estimated depth in the gradient based method proposed here. Hence, we believe that the depth map obtained by this approach is good enough to extract boundary information compared to the laterally shifted images (*e.g.* Figure 3.3a).

[2] Triangulation of *close to parallel* lines. See figure 12.6 in [Hartley 2004] for illustration.

3.3.3 Pixel-Wise Optical Flow Channels Weighting

The desired pixel-wise weighting should reflect the uncertainty in depth information obtained from the optical flow. The weights are computed based on: (a) the error distribution of depth estimation as a function of the optical flow error; and (b) handling the occlusions on the boundaries of the image. There are several error sources that disturb the depth estimation from images. In the scope of this study, we only consider the error made during computing pixels correspondences (or flow vector) which is assumed to be uniform in the image (assuming we have undistorted images). Our aim is to establish an uncertainty measure of the depth based on the aforementioned error. We assume the application targeted have relatively larger translational shift than rotational between the image pair. By this assumption we do not loose generality as it is the case in most realistic configurations. The optical flow $(u,\ v)$ for a point $P(X,Y,Z)$ in the three dimensional world, in case of translational displacement $T(T_X,T_Y,T_Z)$ between two views, is given by:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{s}{Z} \begin{bmatrix} T_Z x - T_X f \\ T_Z y - T_Y f \end{bmatrix} \tag{3.3}$$

here s is a constant related to camera intrinsics. f is the focal length. (x,y) is the projection of P in the image plane. Z axis is normal to image plane and pointing forward. Based on this equation, we can compute the error in the estimated depth as a function of error in optical flow as:

$$\begin{bmatrix} r_u \\ r_v \end{bmatrix} = \begin{bmatrix} \frac{\partial Z}{\partial u} \\ \frac{\partial Z}{\partial v} \end{bmatrix} = s \begin{bmatrix} \frac{-f T_X Z^2}{(x T_Z - f T_X)^2} \\ \frac{-f T_Y Z^2}{(y T_Z - f T_X)^2} \end{bmatrix} \tag{3.4}$$

This equation shows that the estimated depth error is non-linear. Also, note that the depth computed from larger optical flow introduces less error compared with small one. We use this fact to establish our uncertainty measure. Therefore, we assign an uncertainty value for the optical flow inversely proportional to the estimated depth in that point according to Equation 3.4. However, due to the discretized configuration (pixels array representation), this is only valid up to a certain distance limit where

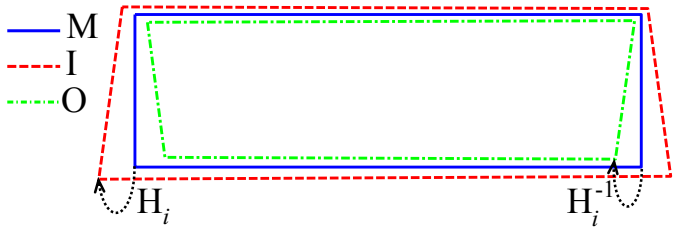


Figure 3.4: An example of possible correspondences frame O computed using local homographies. The rest of the image $(M-O)$ is projected to outside the the second image as $(I-M)$.

differences in depth beyond a given point are not recognizable by the computer vision system. Formally, by considering that the flow vector is defined by a linear system composed of two types of components; Z dependent and non Z dependent terms (See the general optical flow Equation B.3). We define a *blind zone* as the set of points where the Z dependent terms contribute to the optical flow less than one pixel. Hence, we build a pixel-wise uncertainty map for each optical flow channel based on Equation 3.4 and by considering the aforementioned remark by assigning zero weight for pixels in the blind zone. Computationally, the depth Z is computed as an average of Gaussian window centred at the related pixel in the depth map. This helps to handle the noisy depth specially on the occlusion boundaries. Note that it is enough to have the translation up to scale, as the weights will be normalized later.

Another issue we consider is that in each of the input images, and due to the change of view point, some parts at the boundaries in one image does not exist in the other (no correspondence). The optical flow computed in those parts is obtained by data propagation, which is generally erroneous (see the noisy boundaries in Figure 3.3b). Hence, we proceed to find these parts in order to take them into account in the computed uncertainty map. For this purpose, based on the sparse feature points (obtained in Section 3.3.1), we compute the correspondences of the four image corners in the other image. Hence, we calculate four local 2D to 2D points homographies for each of the four corners using n nearest feature points such that:

$$\mathbf{p}_2^i = \mathbf{H}_i \mathbf{p}_1^i \; \{i = 1:4\} \tag{3.5}$$

here $\mathbf{p}_1^i$ is the feature point homogeneous coordinates in the first image, which belongs

to the set of n nearest points ($n \sim 50$) to the corner i. $\mathbf{H}_i$ is the corresponding 3×3 homography. We compute the homographies by using RANSAC with DLT simple fitting [Hartley 2004]. We assume that the selected points have small depth variations. However, using RANSAC here helps to reject the points whose depth is far from the mean depth. We estimate a frame of possible correspondences by applying the inverse of the computed homographies on each corner. All the points that belong to this frame are projected in both images, Figure 3.4 illustrates this step. We generate a binary mask $\mathbf{C}$ (which has the same size of the image) based on the computed frame so that a pixel value is equal to one if it is within the possible correspondence frame.

Now based on the depth error analysis and the binary mask we can write overall pixel-wise weighting function for optical flow channels as:

$$\begin{bmatrix} \mathbf{W}_u \\ \mathbf{W}_v \end{bmatrix} = \begin{bmatrix} (\mathbf{C} + \alpha\bar{\mathbf{C}})\mathbf{R}_u \\ (\mathbf{C} + \alpha\bar{\mathbf{C}})\mathbf{R}_v \end{bmatrix} \tag{3.6}$$

where α controls the impact of the pixels that do not belong to possible correspondences area defined by $\bar{\mathbf{C}}$ (the compliment of $\mathbf{C}$), and $\mathbf{R}_.$ is a unit normalized error matrix computed as $1/r_.$ (given in Equation 3.4). This proposed function assigns the weights inversely proportional to the depth error introduced by the each flow component.

In order to allow contributions from colour channels to fulfil the parts with high uncertainty in flow channels, we assign the pixel-wise weight for colour channels as:

$$\mathbf{W}_{\text{LAB}} = 1 - \beta\sqrt{\mathbf{W}_u^2 + \mathbf{W}_v^2} \tag{3.7}$$

here β is a normalizer that imposes $\mathbf{W}_{\text{LAB}} \in [0..1]$.

3.3.4 Generalized Boundary Probability

In order to compute the boundary probability, we extend the generalized boundary detection method proposed in [Leordeanu 2012]. We select this method due to several advantages such as: (a) significantly lower computational cost with respect to the state-of-the-art methods and (b) ability to combine different types of information

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

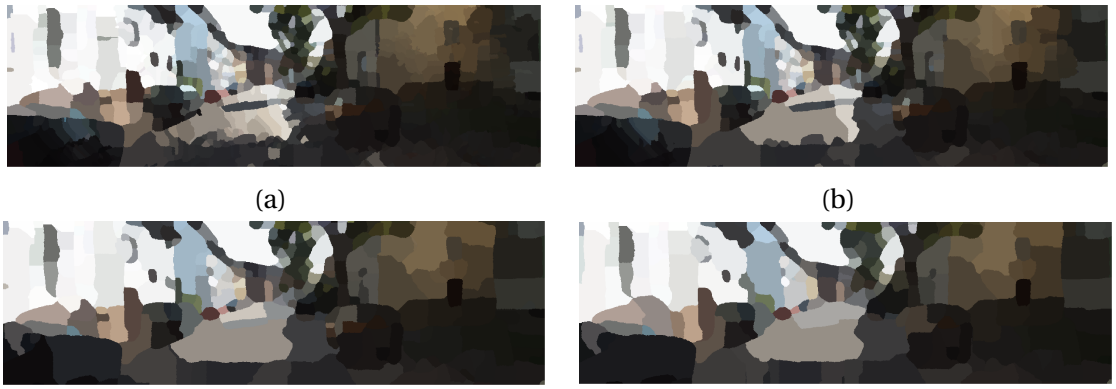
Figure 3.5: Original image (a), and the obtained boundary probability based on optical flow (b), colour (c), both colour and optical flow (d).

(*e.g.* colour and depth) through easily adaptable layer-wise integration. Most importantly, the closed form formulation of this method allows us to easily incorporate our proposed locally adaptive weights.

Consider that we have an image with K layers where each layer has an associated boundary. Now for each layer, let us denote $\mathbf{n} = [n_x, n_y]$ the boundary normal, $\mathbf{b} = [b_1, ..., b_K]$ the boundary heights and $\mathbf{J} = \mathbf{n}^\top \mathbf{b}$ the rank-1 $2 \times K$ matrix. Then, the boundary detection is formulated as computing $\|b\|$ which defines the boundary strength. The closed form solution [Leordeanu 2012] computes $\|b\|$ as the square root of the largest eigenvalue of a matrix $\mathbf{M} = \mathbf{J}\mathbf{J}^\top$, where the unknown matrix $\mathbf{J}$ is computed from known values of two matrices $\mathbf{P}$ and $\mathbf{X}$ as: $\mathbf{J} \approx \mathbf{P}^\top \mathbf{X}$. The matrix $\mathbf{P}$ associates the position information and the matrix $\mathbf{X}$ associates each layer information. Therefore, we can redefine the matrix $\mathbf{M}$ for a pixel p as: $\mathbf{M}_p = \left(\mathbf{P}^\top \mathbf{X}_p\right)\left(\mathbf{P}^\top \mathbf{X}_p\right)^\top$. Note that we can compute $\|b\|$ for an image (using $\mathbf{P}$ and $\mathbf{X}$) only if the layers are properly scaled. Usually, the scale for each layer s_i is learned [Leordeanu 2012] from annotated images. However, we also include the pair-wise weighting matrix $\mathbf{W}_i$ (Equations 3.6, 3.7). Therefore, we construct the matrix

$$\mathbf{M}_p = \sum_i s_i \mathbf{W}_{i,p} \mathbf{M}_{i,p} \tag{3.8}$$

where $\mathbf{M}_i$ defines the matrix for the i_{th} layer. In our approach we use the following layers: L^*, a^* and b^* (CIELAB colour components) and optical flow channels u and v.



(a) (b)

(c) (d)

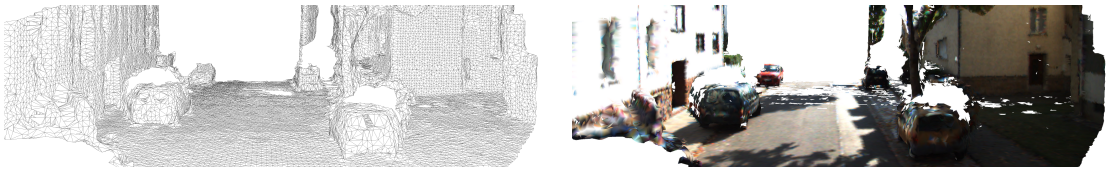
Figure 3.6: Multiple Superpixel segmentations generated from the generalized boundary probability (Figure 3.5d) using watershed approach. (a) without post-filtering. (b,c,d) iterative median filter is applied before watershed segmentation, more iterations for less number of superpixels.

Figure 3.5d shows an illustration of a boundary probability map estimated using our method. It shows the advantage of the pixel-wise weighing such as removing strong false boundaries originated from colour (Figure 3.5c). It also allows to complete far details that are not present in the blind zone of the flow-based boundary probability (Figure 3.5b).

3.3.5 Superpixels Formation and Mesh Generation

We apply watershed algorithm [Szeliski 2011] on the boundary probability in order to produce superpixels. The number of the resulting superpixels could be roughly controlled by applying variable window size median filtering on the boundary probability map. In Figure 3.6, we show some examples of generated superpixels at several over-segmentation levels. The output shown in Figure 3.6a is generated without applying median filtering, it corresponds to the maximum number of superpixels that can be obtained. Whereas there is no minimum number as this can be controlled by the filtering iterations.

One concern which may arise in this procedure is the effect of texture on producing false superpixels boundaries. Indeed, in colour images, boundaries in one layer often coincide with boundaries in other layers, which will produce large boundary



(a) (b)

Figure 3.7: Exemplar 3D mesh (a) and the corresponding textured 3D model of the scene (b).

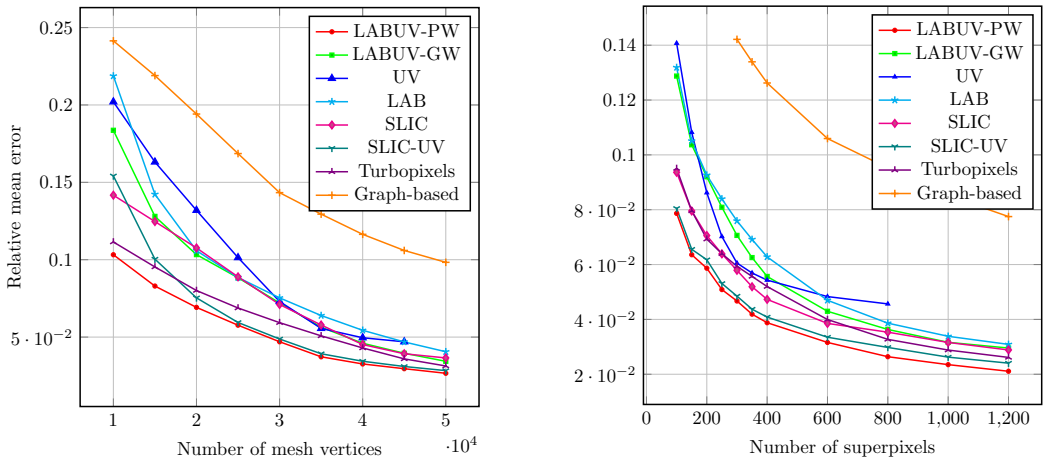
probability. However, when combining colour images with optical flow, the textured areas provide generally good flow estimation due to the density of extracted feature points. Hence, when optical flow channels are combined with colour in the proposed scheme, boundary response at textured areas will be weakened.

After obtaining the superpixels we convert them to a standard mesh representation (VRML). To this aim, we apply the following procedure on a binary edge map formed from superpixels; First we detect all the segments that form a straight line in the edge map. Then, for each of the detected segments we only keep the two ends. The remaining edges are the vertices of the mesh. The mesh faces are then formed by the known Delaunay triangulation manifesting on each superpixel's vertices. This way guarantees that a triangle is contained in only one superpixel. Converting the 2D mesh to 3D mesh is then straightforward by knowing the 3D locations of all vertices. Figure 3.7 shows a 3D mesh example computed using the superpixels shown in Figure 3.6a. Whereas Figure 3.7b shows a textured version where colour information are obtained by back projecting the input image based on the depth map.

3.4 Experiments and Results

To evaluate our method we use the KITTI dataset [Geiger 2012], which contains outdoor scenes obtained using a mobile vehicle. The dataset provides depth data obtained using laser scanner ($\sim 80m$). This enables us to test our fusion model, and in particular the efficiency of the pixel-wise weighting. We select our test images[3] to cover most possible camera configurations (stereo, forward motion, rotation, etc.).

[3]Raw data section, sequences # 0001-0013, 0056, 0059, 0091-0106.



(a) Depth error vs. number of vertices

(b) Depth error vs. number of superpixels

Figure 3.8: Detailed relative mean error.

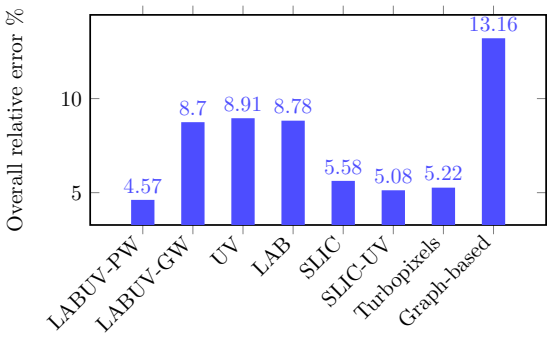


Figure 3.9: Overall relative mean error.

We evaluate the performance of our method (LABUV-PW) compared to the following methods: SLIC [Achanta 2012], SLIC-UV (an extended SLIC [4] that includes optical flow), Turbopixels [Levinshtein 2009] and graph-based [Felzenszwalb 2004]. Moreover, to show the impact of the pixel-wise weighting we test a variation of our method that uses a global weight (learned according to [Leordeanu 2012]) per layer (LABUV-GW). Additionally, we include individual results for colour only (LAB) and optical flow (UV).

The evaluation is carried out by producing multiple segmentations with variable numbers of superpixels that covers a certain range ($\sim 25 - 2000$) for each test image. Each segmentation is converted into 2D mesh according to the method shown in Section 3.3.5. Then, based on the ground truth depth map, we obtain the 3D location

[4] Implemented based on the new measure proposed in [Van den Bergh 2012b]

of the mesh vertices, hence it becomes a 3D mesh. Next, we calculate a relative depth error $|\hat{Z} - Z|/Z$ between the ground truth depth Z and the depth obtained from the 3D mesh $\hat{Z}$. Hence we compute a detailed mean error versus the number of superpixels/vertices. The obtained results are illustrated in Figures 3.8a and 3.8b. It is shown that LABUV-PW performs the best among all tested methods for any number of superpixels/vertices. We notice that the extended SLIC-UV approach provides close performance to LABUV-PW, however it has remarkably large error for small number of superpixels. We attribute this to the regularity aware behaviour embedded in SLIC which enforces segmenting large uniform regions. Figure 3.9 shows the overall mean error for the evaluated methods. Here we notice the large improvement when considering the pixel-wise weighting LABUV-PW compared to the global weighing LABUV-GW.

Concerning computation time, our implementation runs on an Intel Xeon 3.20 GHz (up to 3.6 GHz) with 8 GB of RAM memory. Most of the processing time is allocated to the optical flow computation (1 minute for a 0.46MP frame). Using other GPU-assisted or accelerated optical flow methods caused the performance to drop down (due to less quality of occlusions boundaries). In the rest of the pipeline, for SLIC-UV we use a modified SLIC implementation in C (vl_feat library [Vedaldi 2010]), and for LABUV-PW we use the generalized boundary probability (GBP) [Leordeanu 2012] (MATLAB code) and watershed transform (MATLAB built-in function [Meyer 1994]). For KITTI dataset, the average computational time for around 1K superpxiels is around 2.7 seconds for SLIC, against 1.9 seconds for the GBP+watershed. These results change slightly in the RGB case. Moreover, we notice that SLIC computational time increases (at least linearly) with the increase of number of superpixels, while it is not the case with GBP+Watershed. We refer to [Achanta 2012] for a computational time comparison of colour based methods.

3.5 Discussion and Conclusion

We conclude this chapter by summarizing the major contributions and the implemented ideas in our proposed superpixel segmentation method.

- The output superpixels using the proposed method can be very useful for 3D

modelling and meshing since it respects the structure of 3D scene.

- The proposed evaluation method measure the error made with respect to the 3D geometry, which is more representative than using the hand-made subjective ground truth segmentation.

- The idea of introducing the pixel-wise weighting represents a key advantage compared to global weighting. Because it assigns a representative value for each depth measure which reflects its accuracy. The considered accuracy is a function of depth and spatial position within the 2D image.

- The linear fusion scheme allows a smooth integration of colour and flow information with blind zone handling to produce an efficient generalized boundary probability.

- The boundary probability could be directly converted to superpixels using a simple watershed algorithm. The mesh is generated based on superpixels so that mesh's faces do respect superpixels edges, and hence the scene structure.

- The experiments showed that our method achieved lower error compared to other state-of-the-art (general purpose) algorithms especially for small number of superpixels. Also, including flow information gave better performance than using only colour (SLIC-UV vs SLIC, and LABUV-GW vs LAB).

The main limitation of this approach (and also other non-constrained superpixel generation methods, such as the graph-based method [Felzenszwalb 2004]) is that it cannot be applied in the case when superpixels correspondence is needed. For instance, in the piecewise stereo matching [Yamaguchi 2012] and Multi-View 3D Reconstruction [Bódis-Szomorú 2014, Nawaf 2014b]. This is one of the motivations for our second superpixel generation method that we propose in *Chapter 4*.

Another property/drawback for this algorithm (holds also for all gradient based methods) is that the number of superpixels cannot be controlled. In particular, there is a limitation of the maximum number of superpixels which cannot be exceeded. In contrary, clustering based methods such as SLIC can control the number of superpixels to some extend. Although, when increasing the number of clusters, the ratio of merged

clusters increases. However, the upper limit of the number of superpixels remains in practice quite larger than our proposed method (~ 30% more). From another side, the inability of controlling the number of superpixels may not be considered a disadvantage in some applications when it is needed to leave the number of superpixels as a function of the complexity of the scene.

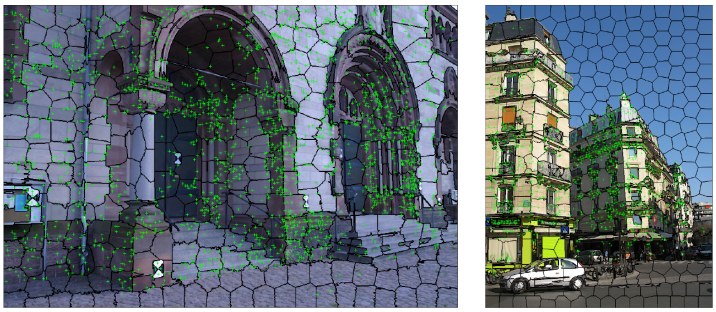
Note that this chapter is based on the published article [Nawaf 2014a].

4 Constrained Superpixel Segmentation for 3D Scene Representation

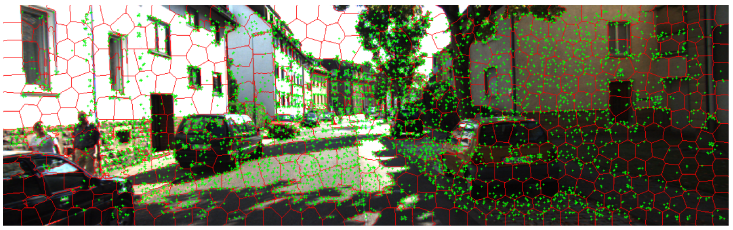
Contents

In this chapter we present an adaptive simple local iterative clustering (SLIC) based superpixel segmentation method for the goal of 3D representation. This method differs from the one proposed in *Chapter 3* that it aims at producing constrained size superpxiels, which is an important property when the used 3D modelling approach involves establishing explicit/implicit superpixel correspondences between views.

The original SLIC method [Achanta 2012] is extended to allow local control of the size of superpixels by the mean of an input density map which reflects the desired size locally. Here, we consider the application of planar patches fitting. So we consider the input density such of the 2D projection of 3D reconstructed points on the image plane. This option is efficient to balance the 3D structure fitting such as in the method proposed in *Chapter 5* and also in other piecewise planar based methods, such as [Bódis-Szomorú 2014]. The proposed extension is achieved by the mean of new



(a)



(b)

Figure 4.1: Original SLIC superpixels with overlaid 3D reconstructed points. (a) From Herz-Jesu-P8 and Mirbel datasets as presented in [Bódis-Szomorú 2014]. (b) From KITTI dataset.

distance measure that takes into account the input density map. Also, we initialize the clustering with input density adapted seeds instead of the originally regular seeds. The superpixels obtained in this case have roughly regular size. Similar to the method proposed previously in *Chapter 3*, the distance measure also involves using flow information which embed the scene discontinuities. This aims at producing more 3D geometry respecting superpixels.

4.1 Introduction

The piecewise representation approach aims at representing the scene structure by small slanted-planes, so each of them belongs to only one object/surface in the scene. Towards this goal, the image is over-segmented into small homogeneous colour/texture regions, which defines the superpixels. Many recent computer vision approaches adopt a piecewise representation for the purpose of 3D scene modelling [Saxena 2009b, Mičušík 2010, Bódis-Szomorú 2014, Vogel 2013, Yamaguchi 2012, Yamaguchi 2013]. These methods use several approaches to obtain the initial superpixel segmentation. For instance, the graph-based segmentation method [Felzen-

szwalb 2004] has been applied in the multi-view stereo method [Mičušík 2010] and the monocular depth estimation from single image [Saxena 2009b], whereas the local clustering based method [Achanta 2012] is applied in the monocular flow estimation method [Yamaguchi 2013], the stereo estimation method[Yamaguchi 2012], and the multi-view 3D reconstruction [Bódis-Szomorú 2014]. The common reason to use the piecewise representation is mainly to overcome the problem of lack of feature points in the scene. Based on few reconstructed feature points per patch/superpxiel, it is possible to approximate other points in that patch by planar fitting. The mentioned methods use several variations of approaches to handle the intra-patch occlusion relations based on energy minimization of empirical potential functions. However, all the aforementioned methods use a superpixel segmentation technique that treat all image parts equally, and does not take into account the distribution of feature points in the image. This may produce ill posed plane fitting problem if the number of points that falls within one patch is less than a certain number, which is three points in theory. However, in practice more points are needed to compensate for the noise and the outliers. Our aim here is to develop a method that considers the distribution of feature points towards generating superpixels that have close numbers of feature points. One more aspect that has to be considered here is that the superpixels should have a size constraint, so they can be employed in the reconstruction methods that require establishing superpixels correspondence between several consecutive frames in a sequence, such as the method proposed in *Chapter 5*, or in stereo vision. Figure 4.1 shows three examples of SLIC superpixels with overlaid feature points. We notice clearly that many superpixels does not contain any feature points (only buildings are considered in Figure 4.1a), so the plane parameters of such superpixels cannot be computed. In our proposition, we do not claim that we completely solve this problem. Instead, we propose a global assessment criteria which is the standard deviation of the number of feature points per superpixel. Minimizing this value leads towards balanced distribution of feature points. The remained ill-posed fitting problems can be handled in the 3D reconstruction pipeline by using other cues such as occlusion boundaries and depth propagation.

We have seen in the previous chapter that we proposed a method that is more efficient for general purpose 3D representation. Unfortunately, it is difficult to adapt gradient-based segmentation approach to provide regular size superpixels. Whereas clustering

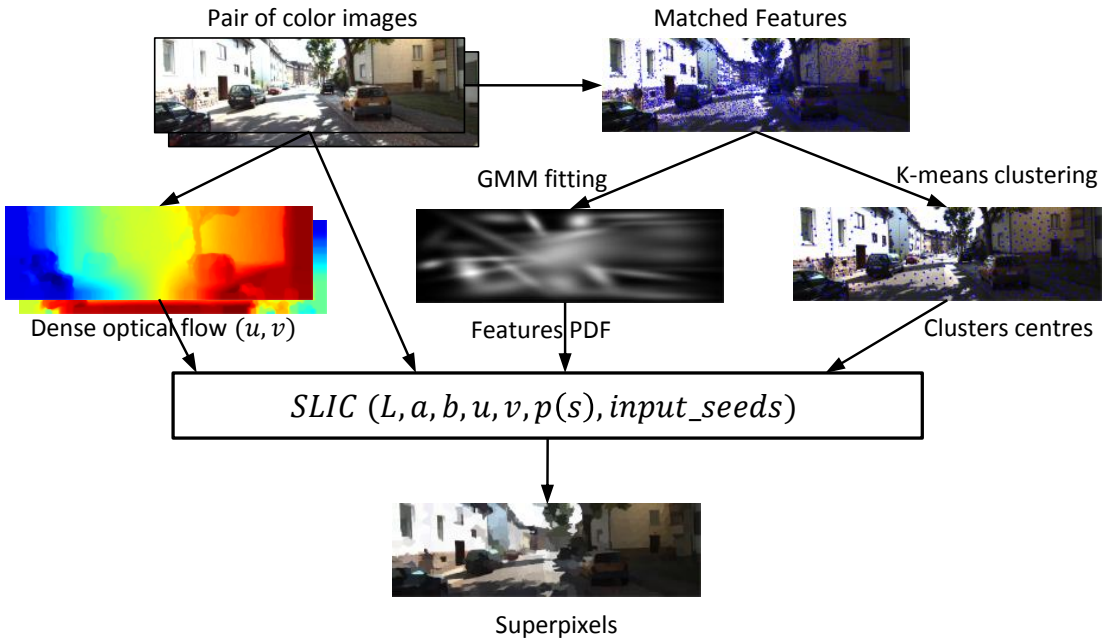


Figure 4.2: Superpixels formation pipeline.

based-methods (such as SLIC) naturally provide this property. This is the reason why we propose our method here based on a clustering technique. We justify our selection of SLIC to develop our approach by the experimental study carried out in *Chapter 3*, where SLIC-UV (SLIC with extended flow aware distance) performance came second after the proposed method LABUV-PW (gradient based). Hence, the method proposed here is feature points density aware extended SLIC-UV.

4.2 Constrained Superpixel Generation

As mentioned earlier, there are two main constraints for the superpixel segmentation method, one is being size/shape aware, and the other is to take into account an input density map to control superpixels size. We clarify the explicit contradiction that may be seen between these assumed two constraints as follows; for the first constraint, we consider the size/shape regularity constraint compared to gradient based segmentation methods such as [Felzenszwalb 2004] and the method proposed in *Chapter 3*. In these methods, superpixels area ratios can be very large ($upto100\times$), whereas the desired ratio is to be within the range ($\sim 1-4\times$). This is because for larger ratios

(*i.e.* large difference in superpixel sizes) establishing frame-to-frame superpixel correspondences become ambiguous with dominant many-to-many correspondences relationships. Our proposed method to match superpixels is based on local holographies so that it fails in this case. Moreover, some features such as colour and shape cannot be used for the matching. For the second constraint, we mean to vary the size of the within small range ($\sim 1-4\times$, in practice, the average is less than 1.5 times) based on the input density map. Although we assume here that the density is such of feature points, it could be any input density map. For instance, another possible application is to generate superpixels from RGB-D images while using the depth as the input density, this will encourage forming small superpixels in far depth areas and vice-versa. Hence, the obtained superpixels have a kind of equal size in real world dimension. However, here we will only focus on the first case as it will be used in the 3D reconstruction pipeline proposed in *Chapter 5*.

Based on our previous study in Section 3.2 on evaluating superpixel segmentation for the purpose of 3D scene representation, the most efficient superpixel regularity aware method is a modified flow-based version of [Achanta 2012], which we consider here with further adaptation. As we target piecewise 3D representation, an important aspect to take into account is the size of the superpixels; with a larger number of superpixels, the planar assumption for each superpixel is more satisfied (similar to image resolution concept). In contrast, if the average number of sparse feature points per superpixels becomes less, that affects the fitting quality or even lead to ill-posed problem. To deal with this issue, we develop an adaptive superpixel generation scheme (see Figure 4.2 for superpixels generation pipeline) by extending the simple linear iterative clustering SLIC algorithm [Achanta 2012] as follows; first, similar to [Van den Bergh 2012b], we add a flow difference term to the distance measure to include the optical flow. The idea is to encourage the segmentation to respect flow discontinuities. Second, instead of initializing the segmentation by uniformly distributed seeds, we use the cluster centres that result from applying k-means procedure to spatial feature points position. This latter step reflects the necessity to have more superpixels in higher density feature points areas which correspond to more detailed parts of the image. In the same way, to encourage the superpixels to be smaller in higher density area, we weight the spatial distance measure by certain value computed based on the local density of feature points.

4.2.1 Proposed Method Inputs

We use the following information as input to the proposed extended SLIC method:

- **Colour image in CIELAB colour space** L^***,**a^***,**b^* : Which is the same as in the original SLIC method.

- **Dense optical flow channels** u**,** v : we use the dense optical flow underlying median filtering method [Sun 2010b] (Classical objective function with Charbonnier penalty term), which shows to have better occlusions handling and flow de-noising, an example is shown in Figure 4.2. Additionally, similar to the method proposed in *Chapter 3*, we perform a minor outliers detection and correction based on computing a first-order geometric error using recovered fundamental matrix and applying linear interpolation to replace the outliers.

- **Density map to control local superpixels size** p : We estimate a pixel-wise probability density based on a Gaussian mixture model ($K \sim 100$) fitted to the spatial coordinates of the feature points using iterative Expectation Maximization (EM) approach (see Figure 4.3 for illustration). Having the output parameters $\{\mu_i, \sigma_i\}$, where $i = 1..K$. We can compute a density value for each pixel location $\mathbf{s}$ as

$$p(\mathbf{s}) = \sum_{i=1}^{K} \phi_i \mathcal{N}(\mu_i, \sigma_i) \tag{4.1}$$

  where ϕ_i is the weight associated to the normal distribution $\mathcal{N}(\mu_\mathbf{i}, \sigma_\mathbf{i})$. $p(\mathbf{s})$ represents the local feature points density. It is necessary to spatially normalize the probability density to be integrated in the colour-spatio-temporal distance measure. Let $\tilde{p}(\mathbf{s})$ denotes the [0..1] normalized representation of $p(\mathbf{s})$.

- **Initial spatial clusters centres** (x, y) : The clusters centres resulting from applying k-means procedure on the sparse feature points. The number of k-means cluster centres has to be slightly more (10%) than the desired number of superpixels due to the merging that may occur during the clustering procedure and also of small superpixels in the post-processing phase. In the general case of an input density map (not such of feature points), the centres can be obtained as the means of a Gaussian mixture model fitted to the density map.

4.2.2 Clustering Algorithm

Here the original SLIC algorithm is used, it has been modified to incorporate the changes we have made in the inputs and also the distance measure. We explain briefly the algorithm in the following paragraph. For more details we refer to [Achanta 2012].

The first step is to move the initial clusters centres (x, y) away from possible noisy pixels or edges by computing the gradient of the image and then moving the centres to the lowers gradient value within its 3×3 neighbourhood. The clustering procedure begins with initializing clusters centres in a seven dimensional space $Ci = [l_i\ a_i\ b_i\ x_i\ y_i\ u_i\ v_i]T$ where (x_i, y_i) is the input spatial centres explained before. An initial rough assignment step associates each pixel with the nearest cluster centre (spatial distance only). This is followed by an updated assignment using the distance measure D, which will be introduced in the next section.The search for similar pixels is done within some limited range around the superpixel centre. This limit defines actually the maximum desired superpixel size. Next, new clusters centres C_i values are calculated as the mean (for seven dimensions) of all the pixels that belongs to the cluster. An error is computed between the new and previous cluster centres. A loop of assignment and update is repeated until the error converge.

The post processing step enforces connectivity by merging disjoint pixels with nearby superpixels. Moreover, we observed that this algorithm may results in very small superpixels (<200 pixels) which may not be desired. We merge such superpixels with a neighbouring superpixel based on the D distance computed between clusters centres.

4.2.3 Clustering Distance Measure

The distance measure D is used as a metric to assign the belonging of a pixel to a cluster. The main problem here is to combine the differences in a way that handles the inconstancy between the various data in colour, location and optical flow spaces. For two pixels $\mathbf{s}_j$ and $\mathbf{s}_k$, we formulate the distance measure as

$$D = d_{lab} + \eta.\Psi_p.d_{xy} + \xi.d_{uv} \tag{4.2}$$

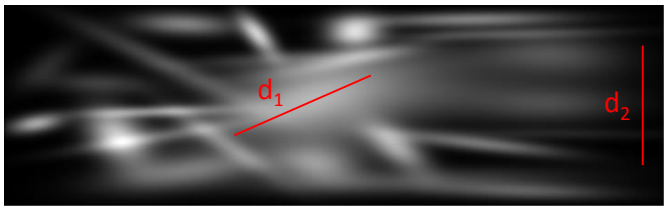


Figure 4.3: Feature points density map. The two equal Euclidean distances d_1 and d_2 are weighted with the local density so that $\Psi_p^{d_1} > \Psi_p^{d_2}$.

where d_{lab}, d_{uv} and d_{xy} are the Euclidean distances in CIELAB colour space, optical flow and pixels spatial coordinates as follows,

$$\begin{aligned} d_{lab} &= \sqrt{(a_j - a_k)^2 + (b_j - b_k)^2 + w_l(l_j - l_k)^2} \\ d_{xy} &= \sqrt{(x_j - x_k)^2 + (y_j - y_k)^2} \\ d_{uv} &= \sqrt{(u_j - u_k)^2 + (v_j - v_k)^2} \end{aligned} \tag{4.3}$$

where w_l is a weight associated to colour intensity. This weight is not considered in the original SLIC, however, in [Van den Bergh 2012b] the authors claimed that assigning lower value helps to decrease the effects of shadows (w_l is set to 0.5). This is true in theory since reducing this weight reduce the brightness value and gives more impact to the colour. However, we did not observe a remarkable change in the output.

Going back to Equation 4.2, ξ controls the temporal compactness, and it is related to the quality of the obtained optical flow, which varies remarkably from side motion to forward motion, with the latter more noisy. Note that our assumption in assigning pixels to clusters while involving optical flow is valid only when computing optical flow based assuming general smoothness constraint (which is the case with the used method [Sun 2010b]), where each surface bounded with occlusion boundaries converges to the same value of optical flow even if it is not parallel to image plane. Otherwise, the slanted planes have to produce gradually increasing/decreasing flow in theory. Another aspect here is the baseline distance between the two views used to calculate the optical flow. Making ξ as a function of time delta between the two frames as proposed in [Van den Bergh 2012b] does not give a good estimate for two reasons; first, the magnitude of optical flow is related to the displacement direction and not only displacement quantity. For instance, the magnitude of optical flow obtained from

lateral motion is larger than such from forward motion with the same displacement. Second, the velocity is not necessarily constant but it depends on the scenario. In the used dataset we empirically set $\xi = 2$ for optical flow computed in lateral motion (*e.g.* stereo), and $\xi = 0.8$ for optical flow computed in forward motion.

The parameter η controls the spatial compactness, which is responsible of superpixels shape. Using large values encourage circular shape (Honeycomb like output), whereas small values allow non regular extensions and less smooth edges. Indeed, the parameter η is application and dataset dependant. In the used dataset, we set $\eta = 5$ based on analysing the colour variance per cluster, and also the variance of the number of feature points per superpixel (this will be discussed further in Section 4.3). The smaller variance means more equally distributed feature points, which is one of the quality assess criteria we consider here. Finally, the weighting function Ψ_p is associated with the spatial term. It is the responsible of involving the input density map to control superpixels size locally. This weighting function is given by

$$\Psi_p = \frac{1}{(|j-k|)} \sum_{i=j}^{k} \tilde{p}(\mathbf{s}_i),\ \mathbf{s}_i \in \overline{\mathbf{s}_j \mathbf{s}_k} \tag{4.4}$$

which weights the distance with the mean density along the line segment between the two points $\mathbf{s}_j$ and $\mathbf{s}_k$. For example, the distances $d1$ and $d2$ shown in Figure 4.2 are weighted by the local density, as a result, $d1$ is more weighted than $d2$. This encourages forming larger superpixels in areas with lower density. Note that this weighting method and the density-aware initial clusters centres have to be jointly performed to achieve the desired goal.

4.3 Experiments and Results

To evaluate the proposed superpixels method we use the KITTI dataset [Geiger 2012] which contains outdoor scenes obtained using a mobile vehicle[1]. The dataset provides depth data obtained using laser scanner ($\sim 80m$). This enables us to test our method for the claimed advantages. We leave the application of 3D representation for the experimental section in *Chapter 5*.

[1] We use the raw data sequences # 0001-0013, 0056, 0059, 0091-0106.

Table 4.1: Analysis of feature points distribution over superpixels. In SLIC-UV-D, the spatial compactness parameter η is set to 5.

NB. Superpixels		SLIC	Graph-Based	LABUV-PW	SLIC-UV-D
200	Mean	31.5			
	STD	26.4	36.7	32.2	**17.3**
400	Mean	15.7			
	STD	13.3	25.3	19.4	**11.7**
600	Mean	10.5			
	STD	10.6	19.4	16.5	**7.3**
800	Mean	7.8			
	STD	8.2	12.1	12.7	**6.1**
1000	Mean	6.3			
	STD	7.1	9.4	9.1	**5.4**
All	Mean	14.36			
	STD	13.12	20.58	17.98	**9.56**

The first assessment is to analyse the feature points distribution over superpixels. As mentioned earlier, it is desirable to have balanced distribution of feature points so that it is possible to perform 3D reconstruction of the scene while each planar patch is fitted with a number of points close to the mean number of points for all patches. Therefore, the criteria we use here to assess such property is the standard deviation (STD) computed for a given superpixel segmentation and the overlaid feature points. We perform this evaluation on several methods including; SLIC [Achanta 2012], Graph-based[2] [Felzenszwalb 2004], LABUV-PW [Nawaf 2014a] and SLIC-UV-D (which we refer to the method proposed here). Table 4.1 shows the mean and the standard deviation of the number of feature points per superpixel. The results are obtained at several over-segmentation levels. We notice the large improvement for SLIC-UV-D compared to SLIC. While the other gradient based methods LABUV-PW and graph-based have remarkably large STDs values, which is unwanted for piecewise scene modelling. This is expected due to the large variance of superpixels sizes in those methods. Note that these results are obtained when the spatial compactness parameter η is set to 5. Using larger values for η causes the mean STD to become less (But not linearly with increasing η). Whereas setting η to small values, the obtained superpixels are less regular in shape and their boundaries are more rough. Also the computed STD

[2]The results obtained for graph-based and LABUV-PW methods are an approximation only since it is not possible to control the number of superpixels

goes larger. However, the *no free lunch theorem* applies also here. Indeed, there exists a trade-off between geometry/boundaries respecting and larger values for spatial compactness. This will be more detailed ahead.

For visual comparison, we highlight 3 pairs of adjacent superpixels obtained using the proposed method, and the superpixels obtained using the original SLIC method and the graph-based method as illustrated in the Figures 4.4, 4.5 and 4.6 respectively. For each case we show the area of the obtained superpixel and also the number of feature points contained inside. We can notice clearly that the original SLIC method produces more regular superpixels with similar size. However, there is a large variance in the number of feature points, unlike the proposed method where the variance of the number of feature points is small. Finally the graph-based has no constraints on the size so this explains the non-equal feature points distribution over superpixels.

Area (Pixels)	1666	620	2280	667	873	1072
NB. Matches	18	20	16	22	24	21

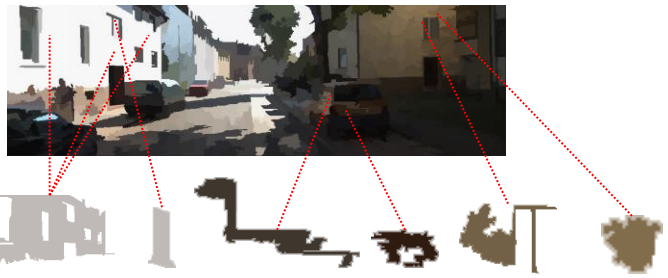
Figure 4.4: Example of superpixels obtained using the proposed SLIC-UV-D method.

The second assessment we perform here is to study the effect of the aforementioned improvement on the reconstructed 3D scene geometry, *i.e.* respecting the boundaries. Here we perform two studies. First, analysing the effect of the spatial compactness parameter η on the boundaries quality. Second, we evaluate the boundaries quality in comparison with other methods. For this purpose we perform the experiments we carried out in Section 3.4 on the proposed method here. We evaluate the performance of our method compared to the other methods mentioned in Section 3.4.

Now, we give a short reminder for the evaluation procedure, based on each superpixel method, we produce multiple segmentations with variable numbers of superpixels

Area (Pixels)	1213	1201	1102	901	1010	932
NB. Matches	7	16	22	32	24	21

Figure 4.5: Example of superpixels obtained using original SLIC method [Achanta 2012].



Area (Pixels)	**60900**	707	1658	638	1668	360
NB. Matches	435	8	33	18	32	4

Figure 4.6: Example of superpixels obtained using the graph-based method [Felzenszwalb 2004].

that covers a certain range ($\sim 100-1200$) for each test image. Each segmentation is converted into 2D mesh according to the method shown in Section 3.3.5. Then, based on the ground truth depth map, we obtain the 3D location of the mesh vertices, hence it becomes a 3D mesh. Next, we calculate a relative depth error $|\hat{Z}-Z|/Z$ between the ground truth depth Z and the depth obtained from the 3D mesh $\hat{Z}$. Hence we compute a detailed mean error versus the number of superpixels/vertices. Both metrics represent how much a given superpixel method respects the 3D geometry of the scene.

In the first study, we start by setting the spatial compactness parameter $\eta = 1$. Note

that for smaller values we get very rough boundaries and non regular superpixels. Next, we increase η with a shift of three. At each value we evaluate the error introduces by the segmentation, and also we analyse the distribution of feature points using the STD criteria as before. The obtained results are illustrated in the Figures 4.8a and 4.8b. The bottom curve (Which is close to SLIC-UV) is the best for respecting the boundaries, however, it is the worst in terms of feature point distribution. We stop increasing η as the boundaries quality became poor. To choose the best parameter value we are empirically based on three aspects; the STD of the number of feature points per superpixel, the delta size for the decreasing STD (we stop when it is small), and finally the overall relative mean error which represent the boundaries quality. For the given dataset we set $\eta = 5$ (the results shown in Table 4.1 and the Figures 4.8c and 4.8d (to be discussed later) are produced using this value). Note that the chosen value have to be reset for new scene types/datasets.

Now, our second study is to compare the proposed method (at the chosen trade-off parameters) with other methods as mentioned earlier. The obtained results for this case are illustrated in Figures 4.8c and 4.8d. It is shown that the proposed method performs slightly less than the original SLIC (and obviously than our gradient based method LABUV-PW). However, we argue that with this drop down in boundaries quality we have better feature points distribution. Figure 4.7 shows the overall mean error for the all evaluated methods (with other methods explained in *Chapter 3*) where the proposed method has around 1.2% more error than SLIC, and 1.7% more compared to SLIC-UV. However, the gain in feature points distribution as a difference in STD is 3.56 and 4.12 points respectively.

Concerning the computation time, our implementation runs on an Intel Xeon 3.20 GHz (up to 3.6 GHz) with 8 GB of RAM memory. Note that for any method that uses dense optical flow (UV components), most of the processing time is allocated to the optical flow computation (1 minute for a 0.46MP frame). Using other GPU-assisted or accelerated optical flow methods results in noisy flow, specially at the boundaries. This caused the performance of the segmentation algorithm to drop down since the optical flow is involved in the distance measure. In the rest of the pipeline, for SLIC-UV-D we use a modified SLIC implementation in C (vl_feat library [Vedaldi 2010]).For KITTI dataset, the average computational time for around 1K superpixels is around

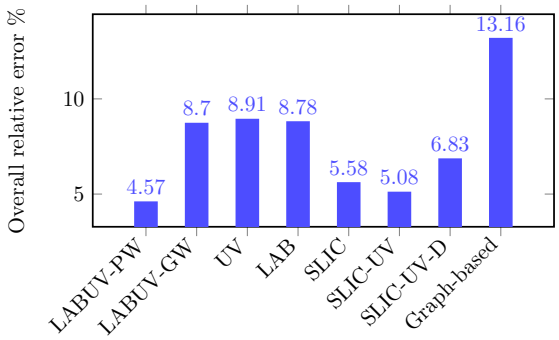


Figure 4.7: Overall relative mean error. In SLIC-UV-D $\eta = 5$, the detailed feature points STD is as given in Table 4.1

2.7 seconds for SLIC, against 3.1 seconds for the modified SLIC-UV-D (given that the optical flow is precomputed). We refer to [Achanta 2012] for a timing comparison of other colour based methods.

4.4 Discussion and Conclusion

We proposed a constrained superpixel segmentation method that can be very useful for 3D representation and modelling where it allows controlling the size of the output superpixels locally. In our proposition, the size is controlled based on feature points density map so that the produced superpixels have roughly an equal number of overlaid feature points. The generated superpixels have constrained shape and size as being based on clustering that involves spatial distance. The size limitation allows the method to be applied in any 3D modelling method that involves establishing superpixels correspondence between views.

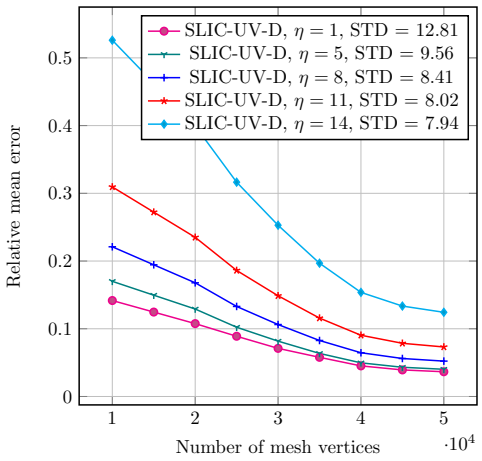
The proposed method is an extension of the Simple Local Iterative Clustering (SLIC). We propose a *new colour-spatio-temporal* function that includes the optical flow to produce a segmentation that respects the flow discontinuities. Also it takes into account the input density map which allows controlling the size of superpixels locally by the mean of weighted distance function. As the produced superpixels are constrained with size and location, the originally regular distributed seeds are not appropriate. Therefore, clusters centres are initialized with non-regular seeds computed based on the input density map so that they are more consistent with the distance measure that

involves using such density map.

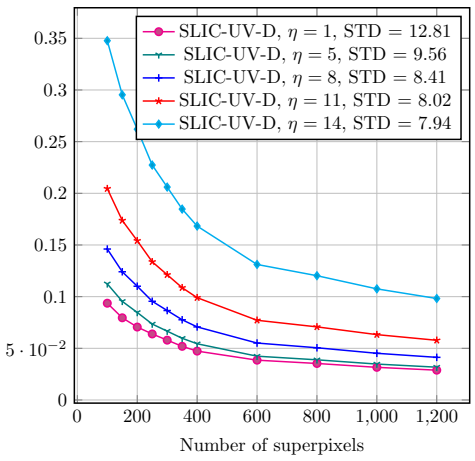
The experiments showed that our method achieved fair distribution of feature points over the generated superpixels compared to other general-purpose state-of-the-art algorithms. We used the standard deviation (STD) of the number of feature points per superpixel. Also, including flow information gave better performance than using only colour. However, the cost that we cannot avoid here is a small drop in boundaries quality, with this latest represent a trade-off together with the mentioned STD criteria minimization.

Note that instead of using the CIELAB colour space used in the distance measure, an illumination invariant colour space such as hue-saturation-intensity (HSI)-based algorithm could be applied accordingly to the used dataset. However, we did not notice remarkable improvement while using the KITTI dataset.

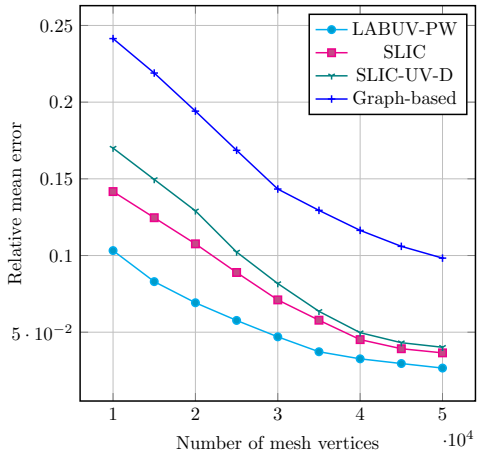
Note that this chapter is based on a part of the published article [Nawaf 2014b]



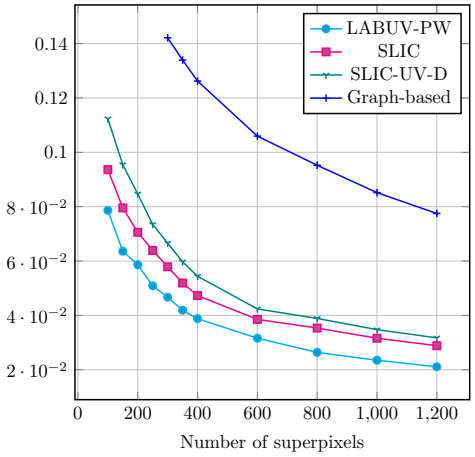
(a) Depth error vs. number of vertices



(b) Depth error vs. number of superpixels



(c) Depth error vs. number of vertices



(d) Depth error vs. number of superpixels

Figure 4.8: Detailed Experimental results for respecting the 3D geometry (boundaries quality). (a) and (b) show the effect of varying the spatial compactness parameter η on the boundaries quality. For each η value we provide the error introduced by the segmentation and also the STD of the number of feature points per superpixel. (c) and (d) show a detailed comparison for the proposed method SLIC-UV-D with the state-of-the-art methods LABUV-PW, SLIC and the Graph-based. In SLIC-UV-D, the parameter η is set to 5, while the mean STD is 9.56, the details are as given in Table 4.1.

5 Planar Structure Estimation From Monocular Image Sequence

Contents

In this chapter, we present a complete planar 3D reconstruction pipeline from monocular image sequence. We start with a brief introduction that summarizes the objective and the related works. Then, we move to explain in details the proposed structure estimation pipeline. First, we explain our sparse 3D reconstruction scheme using several feature matching methods. We also provide an experimental study on the quality of each feature matching method. Based on this study, we introduce the weighting scheme which associates each matching method with a learned weight that represents its desired impact within the planar fitting model. Second, we present a

frame-to-frame superpixel correspondence method. This method is essential in the proposed pipeline as it is used to integrate the temporal information along the images sequence. Third, we explain how we compute a boundary probability map based on colour and flow information. The boundary information are used in the planar fitting procedure to integrate the spatial depth information. Fourth, we introduce the weighted structure fitting scheme which is based on total least square model. Finally, we provide various experimental results for intermediate and final 3D models and we discuss those results.

5.1 Introduction

Recovering the 3D structure from an image sequence is a main focus in computer vision. Several structure from motion (SFM) approaches have been proposed to recover a 3D point cloud using sparse feature matching, triangulation and bundle adjustment [Snavely 2006, Geiger 2011]. Towards better reconstruction, multi-view stereo (MVS) [Pollefeys 2004] methods provide denser point cloud based on several redundant and laterally shifted views. Now, several available tools provide (quasi-) dense 3D models of an object of focus or building façades [Furukawa 2010, Vergauwen 2006]. Here, we emphasis on our interest in applications of mobile vehicle in urban environment. Among the several image acquisition setups, such as panoramic images [Mičušík 2010], omni-directional camera [Lhuillier 2013] and stereo rig [Cornelis 2008], we target specifically building in-city 3D models from monocular image sequence. In this particular case, several challenges arise at different stages of the 3D reconstruction pipeline (We detailed these challenges in *Chapter 1*, here we give a brief reminder). Mainly, lacking textured areas in urban scenes, which results in less feature points, and consequentially, less 3D reconstructed points. Additionally, the continuous motion of the vehicle prevents having redundant views of the scene with short feature points lifetime. This makes the standard MVS difficult [Mičušík 2010], or results in non-dense unrecognisable 3D models. Which is also due to the fact that most of MVS methods [Furukawa 2010, Snavely 2006, Pollefeys 2004] rely on good feature matching methods such as; SIFT [Lowe 2004], SURF [Bay 2006], and recently ORB [Rublee 2011]. These methods have a disadvantage that they provide relatively small number of obtained matches (which is not a problem when redundant views are available). In contrast,

extending the number of matches by allowing more tolerant feature point's quality or using denser matching methods, such as in [Geiger 2011], affects the quality of reconstruction and the relative motion estimation.

In this work, we provide a solution to increase the point cloud density by fusing 3D reconstructed points obtained using several feature points matching methods without letting numerous but less accurate points dominate fewer but more accurate ones when performing the planar fitting of the 3D structure. Hence, we propose a method that provides a complete scene reconstruction from monocular image sequence. Similar to other works [Yamaguchi 2013], we take advantage of appearance similarity in consecutive colour frames to assume spatial belonging to same object in the 3D scene. Hence, the scene is represented by slanted-planes that are either connected or occluded with their neighbourhood. These planes correspond to image patches obtained using adaptive flow-based superpixel segmentation that respect the scene discontinuities. By estimating the planes parameters, we obtain a fully dense scene reconstruction that utilizes all pixels colour information in the image sequence. Our main contribution is a closed-form plane parameters estimation scheme that involves using 3D points obtained using several feature points matching techniques including a noisy dense optical flow. We use a weighted total least squares model to handle the uncertainty of each depth source. This uncertainty is due to several aspects, in our work we take into account:

- The accuracy of the matching method used to reconstruct the point, which is obtained using a learning based approach;
- The number of matches along the image sequence (lifetime) for a certain feature point, which reflects the accuracy of the reconstructed 3D point;
- The baseline distance between two camera poses, since it is relative to the accuracy of the reconstructed point according to stereo reconstruction fundamentals.

The aim of estimating those uncertainty measures is to perform a weighted fitting of the slanted-planes structure using 3D reconstructed point cloud. This point cloud is

computed using the best combination of efficiency proved methods at different stages of the SFM pipeline.

From another side, having chosen the piecewise scene representation, we adopt the superpixel segmentation method proposed in *Chapter 4* to take into account the spatial distribution of feature points for more balanced plane fitting. As mentioned earlier, the aim is to minimize the variance of the number of points used to fit each plane in the 3D structure. Additionally, the usage of the dense optical flow restricts the segmentation to respect both image and flow discontinuities.

One more point, during the planar fitting, we constraint softly the connectivity between superpixels based on occlusion probability map. This leads to propagate depth information between neighbouring patches, which helps to complete missing depth information by encouraging piecewise co-planarity and results in more realistic models. The occlusion probability map is computed based on the generalized boundary probability method as proposed in *Chapter 3*, which is estimated using optical flow and colour information.

5.2 Structure Estimation Pipeline

The reconstruction pipeline that we propose starts by applying a common SFM in order to obtain a 3D point cloud using an efficient combination of several feature matching methods. Then, the point cloud is used to fit the slanted-planes structure which is established based on superpixel segmentation for each of the frames in the sequence, and also the superpixels correspondences. The plane fitting involves using point-wise learned weights, and also occlusion boundary information in order to constraint depth transitions between neighbouring planes. Finally, the 3D model is reconstructed by back-projecting the image texture to 3D space from all visible frames. In the following we will provide the details of each of these steps. A simplified overview of the reconstruction pipeline is provided in Figure 5.1.

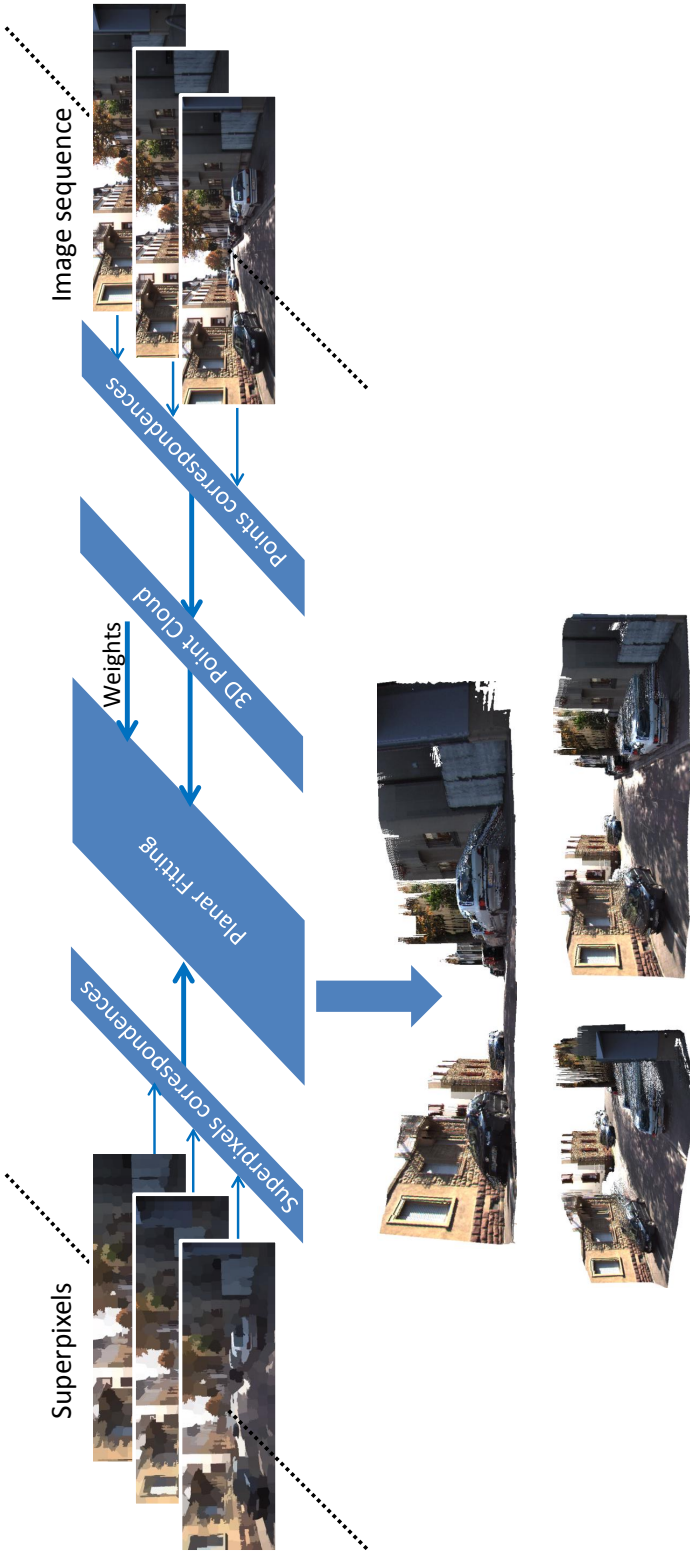


Figure 5.1: Proposed 3D structure estimation pipeline.

Method	Avg. Nb. points	First-order geometric error	Depth error	First-order geometric error (BA)	Depth error (BA)
SIFT	460	5e -5	1.47	47e -6	1.36
SURF	650	17e-5	1.87	16e-5	1.76
ORB	500	7e-5	1.51	65e-6	1.42
ORB	1000	29e-5	1.92	26e-5	1.82
LF/BM	3650	46e-5	2.02	42e-5	1.88
LK (Q = 0.01)	3490	284e-5	3.12	270e-5	2.95
LK (Q = 0.03)	1940	126e-5	2.98	118e-5	2.76
Dense Optical Flow	466K	1642e-5	4.77	1442e-5	4.15

Table 5.1: Comparison between some selected feature matching methods based on KITTI dataset [Geiger 2012] (BA refers to the results after performing global Bundle Adjustment).

5.2.1 Joint Feature Matching

Our approach is generic; it integrates several feature detection and matching techniques. We are motivated by the experimental results indicated in Table 5.1, which shows a comparison between some selected feature matching methods such as, SIFT [Lowe 2004], SURF [Bay 2006], ORB (500 and 1000 feature points) [Rublee 2011], Lucas-Kanade (LK (Q denotes feature quality)), low level features detection (blobs and corners) with block matching (LF/BM) [Geiger 2011], and finally dense optical flow [Sun 2010b]. These results have been obtained using the KITTI dataset [Geiger 2012] (image size is 0.42 mega pixels).

In the experiments, two measures have been used to evaluate the feature matching quality, which are : *a*) the mean first-order geometric error (Sampson distance); *b*) the mean absolute depth error between the reconstructed 3D points and the ground truth[1]. We use the provided pose estimation to compute the fundamental matrix and perform the triangulation. For each of the given measures, results are shown before and after running the global bundle adjustment. However, there is noticeable correlation for both measures and hence any results of them can be used in the learning procedure (explained later). Overall, there is a trade-off between the number of matched features and their quality. For instance, SIFT provides around 460 matches

[1]Since the ground truth (laser scanner data) is very sparse, there is a low chance for a detected feature point to coincide with an existing depth measure. For this reason, we use distance interpolation, however, with constrained allowed maximum distance.

per frame (1242 × 375 Pixels), with an average $5e-5$ of geometric error, while the low level features with block matching LF/BM method provides around 3650 matches, and $284e-5$ average geometric error.

After performing the 3D triangulation and obtaining the point cloud, the 3D points obtained from SIFT matching are more accurate than such obtained from the low level features matching. However, for the purpose of fitting the planar structure the judgement on the methodology is not trivial. In other words, using less but more accurate points for planar fitting against using more points with less accuracy. Here, no decision about the accuracy can be made, nor in the case of mixing both points together. However, inspired by the obtained statistics, in our method we take into account the variable reliability of each matching method by the mean of a weight associated to each 3D point that controls the impact of such point in our plane fitting scheme. Obviously this weight depends on the used feature matching method. As it is difficult to provide a theoretical methodology to calculate such weights, moreover, the existence of many accuracy assessment measures that could be computed for each matching methods (for instance, the two measures presented here), these measures are not necessarily linearly correlated, therefore, we propose to use a learning based approach to find these weights based on the given ground truth.

An obvious point which may arise here is the redundancy of feature points when combining several matching methods; the same feature point (or same after rounding to nearest pixel) is more likely to be detected by multiple feature detectors. In the matching phase, the correspondence may be (not) the same. As a consequence, this may create identical points in 3D when the match is the same, or several non identical points where at most only one is correct. To cope with this problem, we follow an empirical reasoning inspired from experiments as follows:

- Same redundant feature point and same [2] matched point. In this case, the match has higher probability to be correct and accurate. Hence, we keep the redundancy so that the reconstructed 3D points will have more impact in fitting the planar structure.
- Same redundant feature point and different matches. Here, we differentiate

[2] Rounded to sub-pixel accuracy

between several cases. If the matches are obtained using;

- Only global/brute force matching methods (*e.g.* SIFT, SURF, ORB), we follow voting based solution to decide which match to keep. However, the match is removed in case of equality.
- Only local matching methods (*e.g.* LF/BM, LK), the match obtained using the method with higher accuracy (according to the learned weights which will be explained in Section 5.4.1) is kept.
- Mixture of global and local matching method, here the match is removed since no qualitative reasoning could be established.

5.3 Pose Estimation and 3D Reconstruction

To estimate a frame-to-frame camera relative pose, we use a common approach [Hartley 2004] by estimating the fundamental matrix using SIFT matches and the RANSAC procedure. Then, given the camera intrinsic parameters, we compute the rotation and translation (up to scale). To find the translation scales, we propose a solution which is specifically suitable to fixed and known camera setups, *i.e.* camera pose with respect to ground plane, which it is the case in the KITTI dataset.

We find the ground plane by locating the feature points that belong to a predefined region in the image (located at the middle bottom of the image), which are more likely to belong to the ground plane for the given mobile vehicle configuration. This region is learned based on analysing the depth variance of all 3D points obtained from the laser scanner. The points that belong to ground plane show generally very small depth variation, so the desired region can be selected by empirically thresholding the obtained variance and then forming a closed region. Alternatively, road detection techniques can be applied to detect the ground plane [Alvarez 2012]. Based on the 3D reconstruction of the obtained feature points, we perform a robust plane fitting using RANSAC procedure, the scene can be scaled accordingly to match the fixed camera configuration.

Note that the obtained odometry using this approach is more accurate (with respect to the provided Inertial Navigation System IMU data) than using the linear closed form

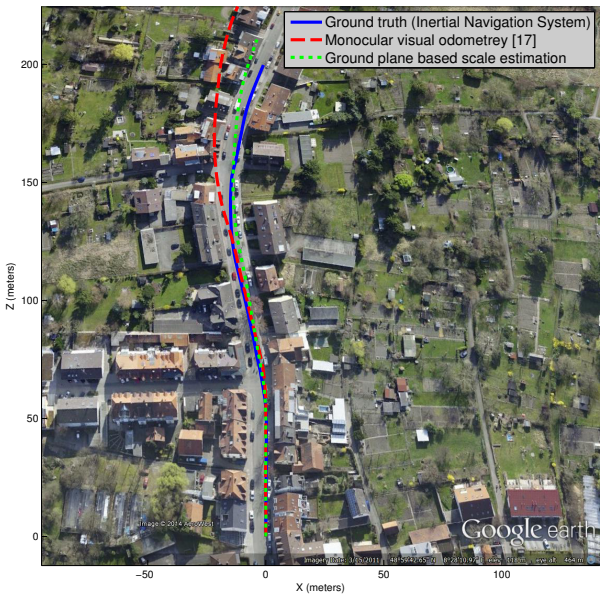


Figure 5.2: Estimated trajectory using fixed configuration assumption and Monocular visual odometry [Esteban 2010] compared to Inertial Navigation System (GPS/IMU) data superimposed onto a Google Earth image of KITTI dataset sequences 0095.

1-point algorithm [Hartley 2004], or the monocular visual odometry [Esteban 2010]. However, both methods remain an alternative in the general case. Figure 5.2 shows an example of obtained trajectory using the proposed method and the general visual odometry method [Esteban 2010] compared to IMU ground truth.

The 3D reconstruction is then straight forward, based on all tracked matches we apply the direct linear transformation (DLT) triangulation method followed by two-stages of bundle adjustment, which involves minimizing a geometric error function as described in [Hartley 2004]. We use the Levenberg-Marquardt based framework proposed in [Lourakis 2009]. In the first stage, we perform combined *structure and motion* bundle adjustment using only SIFT matched features. In the latter stage, we do *structure only* bundle adjustment, where we fix the obtained relative motion from the first stage, and we refine the structure using the matches of all methods. Note that using two bundle adjustment stages provides more accurate results (both structure and motion) due to the variable accuracy of matching methods as discussed before. We remind that the dense optical flow is not considered at this step.

5.3.1 Frame-to-Frame Superpixels Correspondence

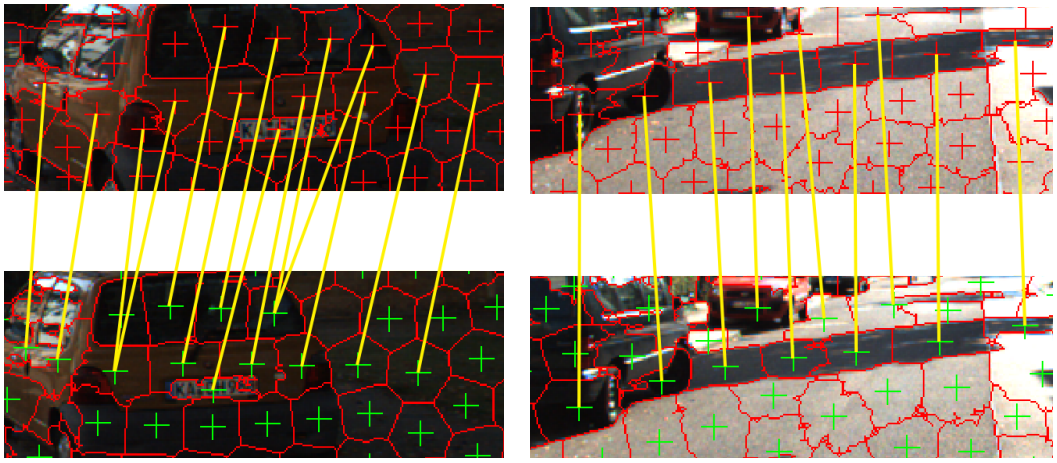


Figure 5.3: Example of frame-to-frame superpixels correspondence.

The proposed method is based on fitting a planar structure using information from several images. Maybe the closest solution to ours which we find in the literature is the piecewise planar reconstruction from multi-view stereo [Bódis-Szomorú 2014]. In this method, a planar structure fitting is performed based on superpixels and overlaid SFM point cloud. One of the main differences we mention here compared to our proposed solution is that the texture is taken from only one image, whereas in our solution we use all texture information in image sequence. To make this feasible, it is necessary to establish a frame-to-frame superpixels correspondence to be used in the proposed planar structure fitting scheme. In other words, for a given superpixel in one frame, to find the position of such superpixel in the next frame, and so forth for next frames (if a match exists). This sequence of tracked superpixels (original and the matches) are assumed to belong to the same surface in 3D. From another side, the depth information assigned to each individual superpixel, more precisely, the obtained SFM points which belongs to a certain superpixel vary from frame to frame. And hence, a proper depth fusion for all tracked superpixels can improve the reconstructed surface in 3D.

Now, we explain the procedure to find frame-to-frame superpixels correspondence. Formally, given the superpixel segmentations of two consecutive frames, let us say f and f', we search for a mapping $\mathscr{H} : \mathbf{S} \rightarrow \mathbf{S}'$, which assigns each superpixel, $\mathbf{S} \in f$ to a superpixel, $\mathbf{S}' \in f'$. For this aim, we use the matched feature points obtained using

SIFT to estimate the spatial motion of a superpixel between the two frames by the mean of local homography (as being a projection of planar patches). Having $\mathbf{S}$, we use the contained feature points $\{\mathbf{p} \in \mathbf{S}\}$ to compute a homography $\mathbf{H_S} \in \mathbb{R}^{3\times3}$ using simple DLT fitting as

$$\mathbf{p}' = \mathbf{H_S}\mathbf{p} \tag{5.1}$$

where $\mathbf{p}' \in f'$ and $(\mathbf{p}, \mathbf{p}')$ is a matched pair. Then, the homography $\mathbf{H_S}$ is used to map the pixels of $\mathbf{S}$ to a new set of locations in f', denoted $\hat{\mathbf{S}}'$. In practice, the obtained $\hat{\mathbf{S}}'$ is not necessarily continuous over its covered pixels, also, it may be not mapped inside a single target superpixel. Hence, we chose $\mathbf{S}'$ as the superpixel that has maximum overlap and colour similarity with $\hat{\mathbf{S}}'$. Figure 5.4 illustrates the proposed procedure, and Figure 5.3 shows two examples of some superpixel correspondences for two consecutive frames. The left side figure shows the case where two superpixels are mapped to one. We found experimentally in most of many-to-one mapping cases, that the superpixels are coplanar. So it does not affect the reconstruction procedure. The colour similarity constraint insures that if an error is made during this step, the superpixel will not be assigned to another surface. This fact is demonstrated in the obtained 3D models presented in Section 5.4.2.

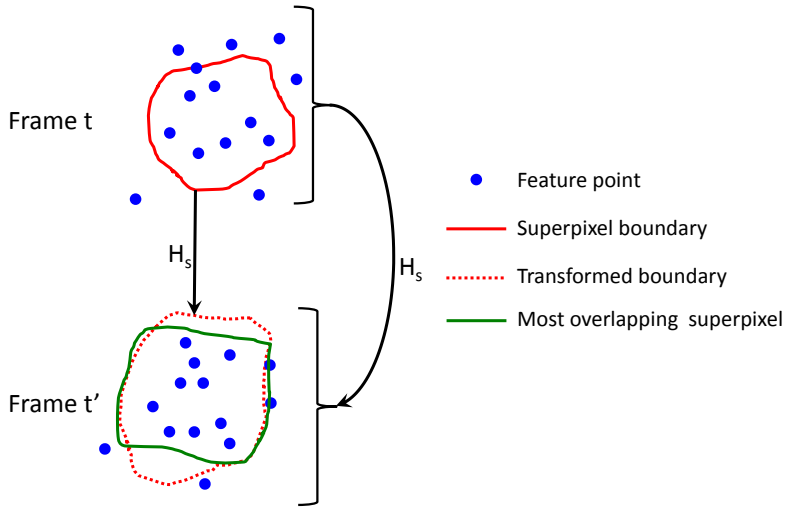


Figure 5.4: Illustration of finding superpixels correspondence using local homographies.

5.3.2 Weighted Total Least Squares for Planar Structure Fitting

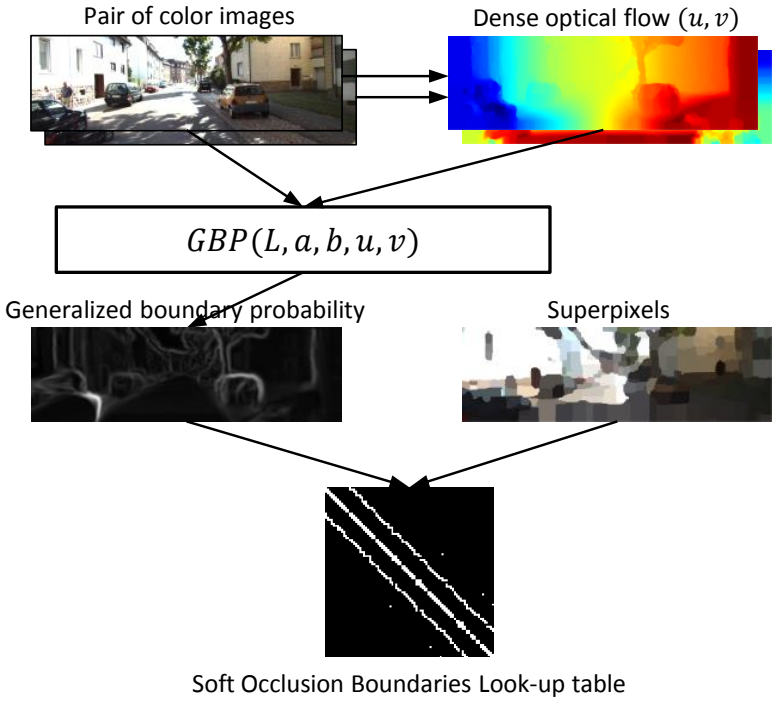


Figure 5.5: Boundary probability computation.

Our goal is to produce a 3D model by fitting the planar patches (based on the obtained superpixels) to the reconstructed feature points and optical flow. As mentioned earlier, feature matching methods have various accuracies (as shown in Table 5.1). All of the solutions we encountered in literature, for instance [Mičušík 2010, Bódis-Szomorú 2014, Vogel 2013, Yamaguchi 2012, Yamaguchi 2013], treat 3D points equally during the plane fitting procedure. This approach is unsuitable here given the experimental study and the conclusions we made in the previous section. Hence, we propose to use the weighted least square where the error contribution of a data point following certain model (here it is plane equation) is controlled via a weight associated to it. Therefore, by using the weighted least square for plane fitting we can treat depth information obtained using each method based on some learned weight that reflects its accuracy. For instance, this allows to fuse sparse but more accurate depth information (*e.g.* obtained using SIFT) with a noisy dense depth obtained from optical flow without having dominant impact of the dense depth.

This adopted weighting concept allows considering other aspects that affect a priori the accuracy of a reconstructed point. Here we propose two more aspects;

Algorithm 1 Planar Structure Fitting Pipeline

Input: Superpixel segmented n image frames, sparse SFM point cloud, learned weights

Output: A set of planes parameters θ

```
for t = 1 to n − 1 do
    for all S_i ∈ f_t do
        Calculate H_S ∈ ℝ^{3×3} using Equation 5.1
        Find Ŝ'_i by mapping S_i according to H_S
        Search for a correspondence S'_i ∈ f_{t+1} using the algorithm explained in Section 5.3.1
        if Found(S'_i) then
            Keep a track of matched superpixel
        else
            Compute the centroid using Equation 5.6
            Form the matrix as in Equation 5.5 {Using the 3D points whose projection appears inside S_i and also inside all of its previously tracked superpixels, also by including the 3D points selected by the methodology in 5.3.3 }
            Find θ_i by solving Equation 5.3 using SVD
        end if
    end for
end for
```

- We take into account the fact that the longer feature point's lifetime, the more accurate is the 3D reconstruction. This fact is valid if all feature point's matches along the sequence are taken into account. There are two reasons that support this fact. First, the impact of erroneous match in one frame is decreased when having more matches, and second, the overall baseline between the first and last match is becoming larger, and hence the accuracy is larger based on stereo vision fundamentals. Experimentally, based on SIFT points matches, the mean depth error (in meters) of feature points that have 2,3 and 4 frames lifetime is 1.93, 1.52 and 1.24 respectively [3]

- The accuracy of the reconstructed point as a function of the baseline distance, which is the frame-to frame camera translation. Because larger translations allows larger disparity limits, and hence higher accuracy. This point is taken into account in our model as a frame-wise weight.

One may argue the significance of the above considered aspects when they are taken

[3] All results are provided with by fixing all other configurations

into account together. For example, is the choice of the matching method has negligible effect against the feature point's lifetime, or the distance to the camera. The answer to this issue comes during the learning process. The weight change for a given combination reflects the importance of distinguishing 3D reconstructed points based on the used criteria. For instance, we tried to extend the weight by adding a term related to the distance of the 3D point to the camera, this is based on the fact that the accuracy of the reconstructed point is a function (without considering the blind zone) of its distance to the camera. However, in practice no dominant effect of such criteria is shown. *i.e.* learned weights do not changed noticeably. The reason may be due to the fact that depth differences for 3D points involved in fitting a planar patch is small. Therefore, we do not consider the point's depth in the weighting scheme.

Now, after presenting the fundamental elements, we move to explain the structure estimation procedure, which is as follows; a frame-to-frame superpixels correspondences is applied and a tracking record is established for all frames according to the procedure explained in Section 5.3.1. Again, for every frame f_t, we estimate the plane parameters which correspond to each superpixel $\mathbf{S}_i \in f_t$ that does not have correspondences in frame f_{t+1}. This means that we delay the planar patch fitting until the last frame where this patch appears (in the next frame it will go out of the view). The reason is that the patch is assumed to the closest to the camera (under the forward motion assumption) so that the 3D points related to such patch are reconstructed with the highest accuracy. Hence, the plane parameter estimation is based on the 3D points (N denotes their number) whose projection appears inside $\mathbf{S}_i$ (in frame f_t), and also inside all superpixels in frames $f_{u<t}$ whose tracking ends with $\mathbf{S}_i$. Additionally, we use a uniformly picked samples of the 3D points obtained using the dense optical flow of f_t. This later step does not show a significant change in the obtained results. Actually, although it allows the reconstruction of the patches which do not have enough sparsely matched 3D points, the noisy flow in some areas affects the reconstruction quality. Weights learning does not provide a solution to this problem because the accumulated overall improvement resultant from introducing the optical flow dense depth prevents decreasing the weight associated to it. Empirically, we found by visual assessment of small details of the obtained 3D models that decreasing the learned weight by $20\% - 30\%$ is a compromise for this issue. Let us note that the learned weights for the optical flow are dependant on the sampling ratio. As a result,

within a major range of the sampling ratio of the dense depth, the weights are being modified accordingly, whereas the output 3D model quality remains mostly the same.

Now, we present formally the plane parameters estimation. Let us denote the plane parameters as $\theta = [\tilde{\mathbf{n}}^\top \; d]^\top$ where $\tilde{\mathbf{n}}^\top$ is the normalized normal and d is the 3D euclidean distance to the origin. According to this definition, the orthogonal distance between a 3D point $\mathbf{x}$ and the plane is given as

$$\mathrm{D}(\mathbf{x}, \theta) = \mathbf{x}\tilde{\mathbf{n}}^\top + d \tag{5.2}$$

we formalize the plane fitting problem as

$$\sum_{t=j-k}^{j} w_f(t) \sum_{i=1}^{N} w_{m,l}.\mathrm{D}(\mathbf{x}_{i,t}, \theta)^2 \tag{5.3}$$

here, $w_{m,l}$ is the learned weight associated to a reconstructed point $\mathbf{x}_{i,t} = [x \; y \; z]$ obtained using the feature matching method m based on l frames (point's tracking lifetime), and t is the largest frame index where the points projection appears (assuming increasing index with time). Note that each 3D point is used only once (not to confuse with redundant 3D points). The difference $j-k$ is the index of the last frame that has at least one trackable superpixel until f_j. The function $w_f(t)$ provides the frame-wise weighting. Analytically, this weight gives more bias to last frames for two reasons; first, the chance to have wrong superpixel correspondence increases for longer tracking. Second, as last frames are closer to the scene (forward motion assumption), the 3D reconstruction is more accurate. Hence, to give less weight while moving away from the scene, we formulate

$$w_f(t) = \mathrm{e}^{-\|T_t\|/\beta} \tag{5.4}$$

here T_t is the relative motion translation of the frame f_t with respect to f_j, and β is a parameter that controls the weight decreasing rate. For instance, setting $\beta = 3$ suppresses the impact of more than 3 frames away.

For simplification, by considering $w_n = w_f(t)\, w_{m,l}$ the weight associated to a data point $x_n = [x_n \; y_n \; z_n]$. The solution to Equation 5.3 is achieved by computing the

singular value that corresponds to the smallest eigenvalue, denoted σ, of a $(N \times k) \times 3$ matrix which takes the form

$$\begin{bmatrix} \sqrt{w_1}(x_1 - \bar{x}) & \sqrt{w_1}(y_1 - \bar{y}) & \sqrt{w_1}(z_1 - \bar{z}) \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \sqrt{w_{N\times k}}(x_{N\times k} - \bar{x}) & \sqrt{w_{N\times k}}(y_{N\times k} - \bar{y}) & \sqrt{w_{N\times k}}(z_{N\times k} - \bar{z}) \end{bmatrix} \tag{5.5}$$

and the centroid of the points is given by

$$\bar{\mathbf{x}} = [\bar{x}\ \bar{y}\ \bar{z}] = \frac{\sum_{t=j-k}^{j}(w_f(t)\sum_{i=1}^{N} w_{m,l}\mathbf{x}_{i,t})}{\sum_{t=j-k}^{j}(w_f(t)\sum_{i=1}^{N} w_{m,l})} \tag{5.6}$$

The entire 3D reconstruction procedure is presented as pseudo-code Algorithm 1. Let us note that it is possible to encapsulate the plane fitting in RANSAC procedure, in this case, the weighted sum-of-squares of residuals is given as

$$\frac{\sigma}{\sum_{t=j-k}^{j}(w_f(t)\sum_{i=1}^{N} w_{m,l})} \tag{5.7}$$

However, due to the large number of points obtained using the dense optical flow compared to the rest of points, RANSAC is not a good choice because there is higher chance to select dense depth points due to their large number. Nevertheless, it is more robust to be applied when the dense optical flow is not considered. In our method, we experienced slightly better results when using the dense optical flow as mentioned earlier.

5.3.3 Boundary Probability to Improve Connectivity

Integrating boundary information has a dramatical improvement over the reconstructed 3D model. Indeed, the available accuracy of 3D reconstructed points does not provide connected structure. Some patches remain floating in the scene, which are not visually appealing. Figure 5.6 shows an example of an obtained 3D model without integrating any boundary information. This shows the necessity for such

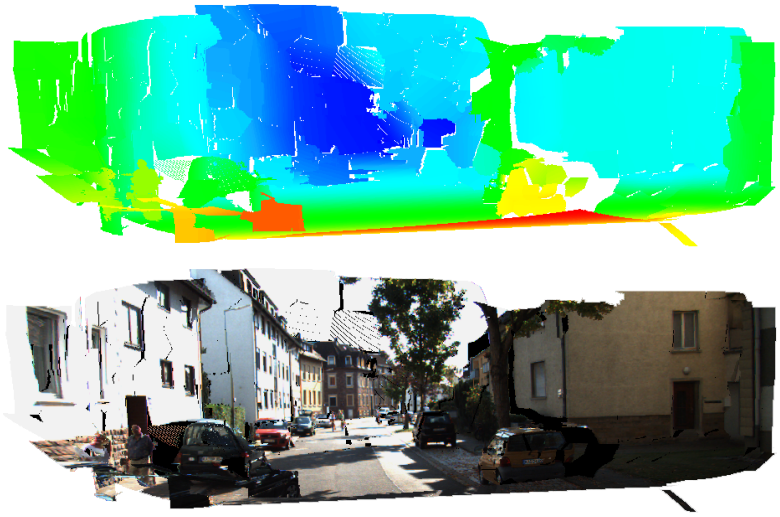


Figure 5.6: Example of 3D model without integrating boundary information, most of adjacent patches are not connected.

an update to the proposed method. Actually, all of piecewise based reconstruction methods takes into account the connectivity with neighbouring patches into account. The most popular way is to handle this relationship through a MRF/CRF. This is seen in the following works [Mičušík 2010, Bódis-Szomorú 2014, Vogel 2013, Yamaguchi 2012, Yamaguchi 2013], where a potential function is responsible for penalizing the dis-connectivity proportionally to an occlusion probability. The model is then solved using optimization techniques, none of them is a closed-form. Hence, we propose a solution that can be integrated into our weighted plane fitting model, while remaining closed-form, efficient and faster to resolve compared to other probabilistic models.

Realistic scenes are composed generally of connected structures and also have some occlusions. The majority of the boundaries that are obtained from superpixel segmentation are for connected structures (as it depends on colour information) and fewer for occlusions. However, they do not necessarily reflect the real occlusion boundaries (although there are much more falsely detected occlusion boundaries (false positives) than falsely undetected ones (false negatives)).

Indeed, real occlusions can be better inferred using both colour and flow as spatially uniform regions have continuous flow and homogeneous colour. In this work, we employ the closed-form generalized boundary probability method [Leordeanu 2012]

the same way we proposed in Section 3.3.4. The method combines low- and mid-level image representations in a single eigenvalue problem, which is then solved over an infinite set of putative boundary orientations. We compute a boundary probability map using the following layers; L^*, a^* and b^* (CIELAB colour space) and the two optical flow channels (u, v). The pipeline of computing the boundary probability is illustrated in Figure 5.5.

We use the obtained boundary probability to add some constraints that encourage connected structure in 3D as follows; For every two neighbouring superpixels, we compute a soft value of occlusion probability, denoted O_i, by taking the mean boundary probability for the pixels located on their common edge with two pixels of width. Hence, we form a sparse lookup table that contains all O_i for all superpixel combinations, so that for each two superpixels it returns a soft occlusion indicator. Next, for each superpixel, we select the n closest sparse feature points to the common edge (we exclude the dense depth), and we include their 3D reconstruction in fitting the neighbouring superpixel. However, we impose a modified weight

$$w'_{m,l} = \alpha \, w_{m,l} \, O_i \tag{5.8}$$

where α handles the inter-superpixel impact so that large values encourage co-planarity between neighbouring non-occluded superpixels. In our implementation we choose empirically $(n = 5, \alpha = 5)$. Using small values for both parameters results in many floating patches in the image, while using larger values leads to obliterate scene details and produce less independent planes. Generally, larger values are recommended in texture-less scenes.

5.4 Experiments and Results

5.4.1 Feature Matching Methods Selection

Learning the weights $w_{m,l}$ helps to identify good combinations of feature matching methods. Using Equation 5.3 for this purpose can be in practice intractable. Instead, we use a simplified formulation that does not consider the frame-wise weights. Hence,

method \ lifetime	2	3	4	5	>5
SIFT	1	1.64	1.82	1.91	1.97
ORB (1000)	0.92	1.51	1.62	1.71	1.84
LF/BM	0.81	1.23	1.40	1.48	1.54
Dense depth (1/10)	0.14	-	-	-	-

Table 5.2: Normalized learned weights associated to 3D points obtained using one combination of feature matching methods and the number of frames the feature point is tracked (point's lifetime).

weights learning is achieved by minimizing the formula

$$\sum_{i=1}^{N} w_{m,l}.\mathrm{D}(\mathbf{x}_{i,t},\theta)^2 \tag{5.9}$$

based on the given ground truth data and the reconstructed 3D points. We use the Nelder-Mead simplex method [Lagarias 1998] which provided faster convergence than gradient-decent approaches. Moreover, it does not require an analytic form of the cost and can be easily applied (fminsearch in MATLAB).

As mentioned earlier, given that the laser scanner data is quite sparse, there is a low chance for a detected feature point to coincide with an existing depth measure. For this reason, we use distance interpolation within some limits.

For faster convergence, the weights are initialized with values inversely proportional to the geometric error shown in Table 5.1. After testing several combinations of methods (list in Table 5.1), our first observation is that the obtained weights are nearly inversely correlated with the error induced by each method. Expectedly, reconstructed points from more than two frames are more weighted, also, the weights for the points reconstructed with 5 frames and more become steady. An important note here is that the weight $w_{m,l}$ given to a certain method is not independent from the used combination. *i.e.* One method can be more weighted than another in one combination while it can be less weighted when included within another combination. This can be explained by the variant redundancy that each method introduce to a given combination. As a result, it is not possible to provide a method-weight general results. We leave this point to be as a potential future work.

By analysing the obtained weights using several combinations, we can obviously

Figure 5.7: Original frame from the sequence 95 and a Dense 3D model obtained using the proposed method from several view points.

Figure 5.8: Original frame from the sequence 93 and a Dense 3D model obtained using the proposed method from several view points.

know the good combination of methods where the weights associated to all matching methods are significant while producing relatively small error based on the formula 5.9. Based on this strategy, we choose SIFT, LF/BM, ORB and the dense optical flow (sampled by 1/10) to be the best combination. Adding more points using other methods does not worth the slight improvement (few additional non-overlapping feature points). Table 5.2 shows the normalized weights obtained for this selection. The results shows the variable weights associated to each feature matching method. Also, for each method, the weight changes as a function of the number of the frames the feature point is tracked (lifetime). An important point to mention here is that the obtained learned weights depend on the number of feature points, which is related to the nature of the used dataset.

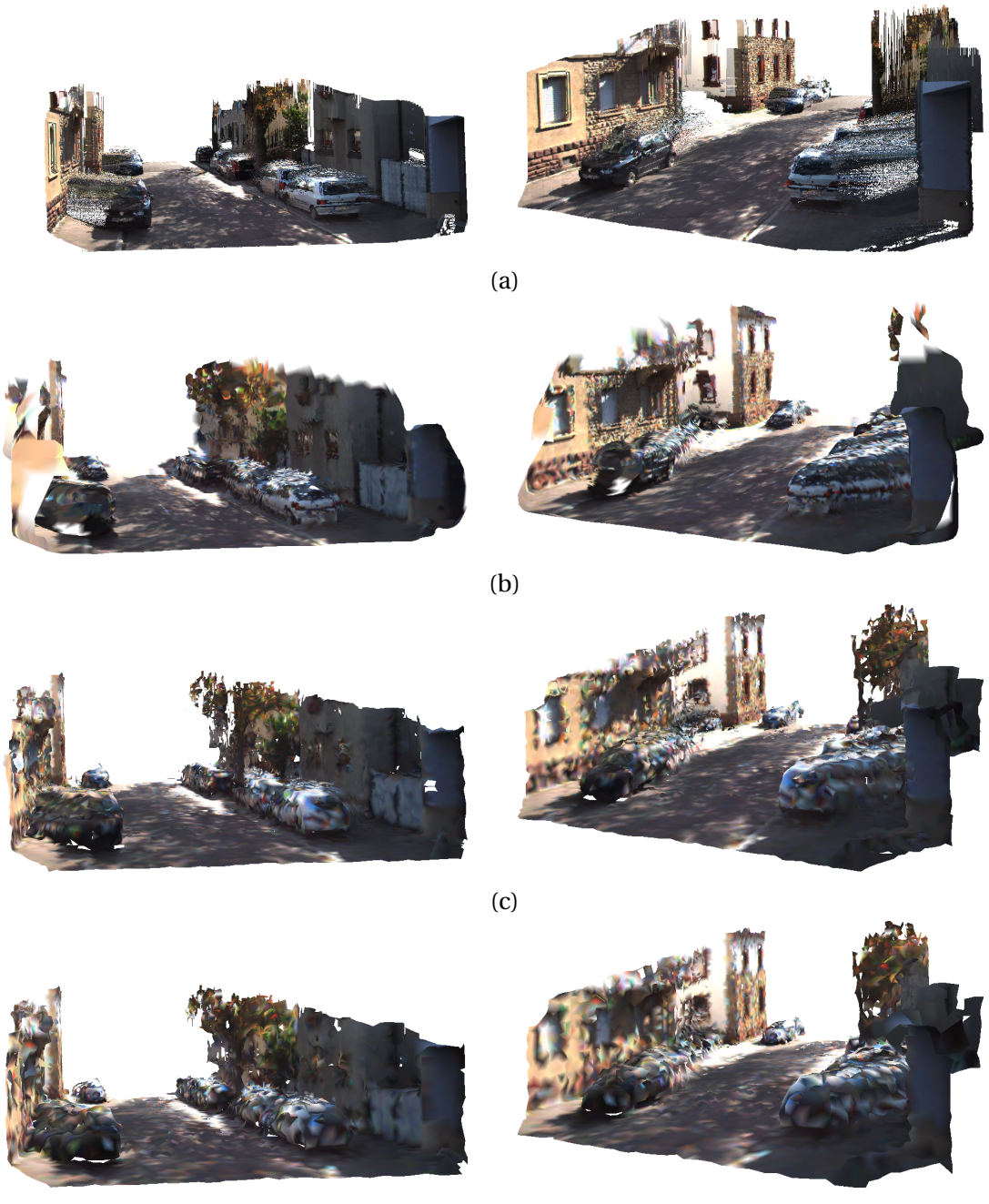


Figure 5.9: Comparison of 3D models created by different methods. Our proposed method (a), Poisson surface reconstruction [Kazhdan 2006] using dense optical flow and sparse points (b), surface reconstruction of sparse points using the greedy triangulation method [Marton 2009] (c), and Delaunay triangulation based manifold surface reconstruction [Lhuillier 2013] (d).

(a) (b)

Figure 5.10: 3D models to show the significance of integrating the boundary information, also show the robustness of the ground floor estimation. (a) boundary information are used. (b) boundary information are not used, many adjacent patches are not connected, some are floating.

5.4.2 3D Model Reconstruction

We tested our 3D reconstruction method on several sequences from KITTI dataset [Geiger 2012]. After obtaining the plane parameters which correspond to superpixels, we project the texture of the image sequence to 3D in order to obtain a dense reconstruction that makes use of all colour information included in the input images. Because the provided laser scanner data is sparse and covers only the lower 30% of the image, and due to the accumulated scale drift between frames, we did not investigate the reconstruction accuracy of the resulting 3D models. However, as our goal is to provide realistic scenes, we evaluate the quality of the results using a subjective measure of realism, which is commonly used in practice [Cornelis 2008, Lhuillier 2013, Mičušík 2010].

Two examples [4] of obtained 3D model are shown in the Figures 5.7 and 5.8. The scenes were chosen as they contain the common objects in urban environment (Building

[4] Sample result videos and images can be downloaded from http://perso.univ-st-etienne.fr/nam07924/Elsevier-IVC-2014.zip

façades, cars, trees). The 3D models are reconstructed using 12 frames, and it is cropped at the blind distance of the last frame, where the scene objects start to deform. We notice that the scene is well textured, in particular, the building façades and main road structure. The lack of visual information behind the stationing cars is due to the occlusion and missing scene information (the objects behind the cars are not revealed in any frame). We mention that our method is generally not robust to greenery as it violates the planar assumption.

We also provide comparisons between different methods using the same viewpoint. Figure 5.9a shows examples of 3D model obtained using our method. To emphasis the importance of some stages in our pipeline, mainly, the superpixels representation and the feature points fusion scheme, we provide the 3D models using two approaches ; first, by using dense optical flow and sparse point cloud. In this case, a smooth Poisson surface reconstruction [Kazhdan 2006] is necessary to provide visually recognizable models as illustrated in Figure 5.9b. The main problem with Poisson reconstruction is the bad handling of manifold junctions by forcing curved shape. Also the reconstruction is unpredictable in case of lack of 3D points (see the prominent object at the left side of the scene, which is due to the impact of noisy optical flow). Second, we use the sparse point cloud obtained using the selected feature matching methods treated equally. Also, to deal with the lack of depth estimate in many areas we perform surface reconstruction using two approaches; the greedy triangulation method [Marton 2009] which assumes locally smooth surfaces and performs incremental triangular mesh decoupling. Also we investigate the manifold surface reconstruction based on Delaunay triangulation approach presented in [Lhuillier 2013]. The obtained 3D models for both cases are shown in Figure 5.9c and 5.9d. It is slightly noticeable that the reconstructed trees are more visually appealing. However, the obtained 3D models still show to be remarkably less detailed than our approach.

Another essential step in our pipeline is integrating the occlusion boundary information. Let us recall the 3D model we presented in Figure 5.6 and also the 3D model in Figure 5.7. Both models are for the same scene, Figure 5.6 is the obtained model without integrating the boundary information, whereas Figure 5.7 is after being integrated. The difference between both models is obvious. The first one suffers from floating patches and dis-connected neighbouring patches, whereas this problem is

solved in the second case. In general, this problem is due to the noisy 3D points as a result of several factors during the 3D reconstruction phase. This problem becomes larger when there is a lack of sparse feature points whereas the dense optical flow is noisy as explained earlier.

Next, in Figure 5.10 we give a close-up views of a 3D model to show two points; first, as another example supporting the latest point related to occlusion boundaries. The model shown in 5.10a is produced without integrating the occlusion boundary information. Floating patches can be spotted clearly on the top right and top left corners as well as on the side of the vehicle. Second, to show the perfect quality of reconstructed ground plane using the procedure explained in Section 5.3.

Regarding time complexity, our implementation (mixed MATLAB and C++) runs on an Intel Xeon 3.GHz (up tp 3.6 GHz) with 8 GB of RAM memory. The dense optical flow still occupies most of the computational time (1 minute for a 0.46MP frame). Using other GPU-assisted or accelerated optical flow methods produced more noise, which affects the output quality. The plane fitting model takes around 30 seconds for 10 frames model, which is much faster than probabilistic models based methods [Gallup 2010], where the complete model takes around 1.5-2.5 minutes in total.

5.5 Discussion and Conclusion

We presented an efficient monocular 3D reconstruction pipeline for urban scenes. The extended flow, colour and feature density aware superpixel segmentation provides a meaningful representation for the slanted-planes assumption. The weighted total least square model allows fusing several feature matching methods while it prevents larger number but lower accuracy matches to have dominant impact when used with higher accuracy matches. Also, we propose a solution to handle the neighbouring relationship between planar patches using the same total least squares model. The obtained 3D models show the impact of the chosen scene representation and the fusion model on the output quality.

As we have seen, using the boundary probability improves remarkably the reconstructed structure. Using both colour and flow information to compute the boundary

probability increase the chance of detecting occlusions in the scene (although it provides much more false positives than false negatives). The error made by this approach causes to leave some connected structure without being constrained, whereas it does not force unconnected structure, in reality, to be connected in the 3D model. Which we think it is a better behaviour than the inverse case.

One of the problems that arises while fitting the planes using several frames is the sensitivity to relative motion estimation, as this produces an increasing drift when proceeding further away (shadowed plans of points). Another issue we could mention in the current framework is the assumption of fixed weights for fusing reconstructed points. Whereas the dense monocular optical flow suffers from unstable performance even in the same sequence. This can be a future perspective issue.

In this work, we considered that the weights associated to reconstructed points are learned, whereas learning the weights can be extended in several aspects. Mainly by forming the weights as a combination of learned and non-learned variables. Given that the learned weights depends on the number of feature points. An empirical function that takes as input the number of feature points can compensate for the weighting change due to the number of feature points, based on the fact, that the weights generally tends to decrease with the increasing of the feature points obtained by certain algorithm. In the same way, the number of frames used to reconstruct the point can be also excluded from learning.

As we have seen, the main limitation of the proposed method is the poor reconstruction of the objects that violates the planar assumption (although they are not common in urban scenes). A possible solution for this issue is two folds. First, by integrating a robust recognition system to detect such objects to be treated differently. This is already a used practice in Google Maps 3D maps, where the output 3D model is stored as hybrid low/ high-level vectorized representation. Some examples of high level representation are : finite set of tree shapes, greenery areas. Another example is the method proposed in [Cornelis 2008] which replaces on street vehicles by a predefined 3D models. Second possible improvement, is to use the available prior knowledge about the scene to form additional constraints. In our method we benefit from this fixed configuration prior to estimate the ground plane, which makes the method more robust. However, some other priors can be used such as the vertical alignment of

façades, windows, etc.

To summarize, the major contributions and ideas that we proposed in this chapter are the following:

- Using several feature matching methods together to increase the density of the obtained 3D reconstructed point cloud. The learned weight associated to each reconstructed 3D point represents its prior accuracy.
- The total weighted least square model estimates the plan parameters based on a set of input points and the associated weights so that the impact that each point has is proportional to its weight.
- We take into consideration the temporal dimension. When estimating the plane parameters for a certain patch, we give more impact to the closest frame to the scene as the accuracy is higher. This idea depends on the proposed frame-to-frame superpixels correspondence method, and it is integrated within the same weighted total least squares model.
- Occlusion boundaries information controls the depth propagation among neighbouring planar patches. The proposed methodology encourages softly the connectivity and co-planarity with neighbouring patches based on occlusion boundary map. This is done using the same model which can be efficiently solved through SVD.
- The proposed method provides dense 3D models that are more visually appealing than other comparable 3D reconstruction and surface reconstruction methods. The obtained performance is due to the proposed reconstruction pipeline, where all the aforementioned ideas play role, also the usage of the superpixel generation method proposed in *Chapter 4*.

Note that this chapter is based on the published article [Nawaf 2014b].

6 Conclusions and Future Directions

In this thesis we have described several innovative ideas and improvements to the current state-of-the-art in the context of structure from motion using images. The research presented in this context has focused on the specific application of improving the 3D reconstruction from a monocular image sequence taken using a mobile vehicle in urban environment, with a forward looking camera. We overcome the issues produced by the lack of redundant views and the poorly textured regions by adopting the piecewise planar 3D reconstruction. In which the planarity assumption allows to provide a complete dense structure estimation using a set of sparse reconstructed point cloud using SFM technique. In the presented research, we introduce several improvements to the 3D structure estimation pipeline. In particular, the planar piecewise scene representation and modelling.

6.1 Summary and Discussion

Our main contributions and ideas to improve the 3D structure estimation were made at different stages of the pipeline, namely : the piecewise scene representation, sparse 3D reconstruction and the planar structure fitting. We provide in the following a brief summary for each.

Piecewise scene representation : In this perspective, two superpixel segmentation methods have been proposed for the scene representation. Both methods can be used as an independent tool. Mainly, by the applications that adopt a piecewise

representation of 3D scene. The first developed approach aims at creating 3D geometry respecting superpixel segmentation. The superpixel generation is based on a generalized boundary probability estimation using colour and dense optical flow information in a multi-layer gradient based model. Our contribution in introducing the pixel-wise weighting to the flow channels represents a key advantage compared to global weighting. Which provides a solution to the noisy flow at image boundaries, and also takes into account the error of the computed optical flow as a non-linear function of the disparity. This method produces non-constrained superpixels in terms of size and shape.

Some applications imply a constrained size superpixels, such as the methods that track superpixels over an image sequence. Hence, our second developed superpixels method is based on the simple local iterative clustering approach where it produces regular size superpixels. The method uses flow and colour information to provide superpixels that respect the scene discontinuity. More importantly, we add a new input that allows controlling the size of the obtained superpixels locally. This is achieved by the mean of a new distance measure that takes into account this input density. And also we initialize the clustering with input density adapted seeds instead of the originally regular seeds. In our application for planar fitting, we use the density of the sparse feature points for this input to produced more balanced superpixels for better 3D structure fitting. The obtained superpixels in this case are relatively regular and limited by size, so this method is suitable to our 3D reconstruction pipeline which requires establishing superpixels correspondence between consecutive frames in a sequence.

Additionally, we proposed a new procedure to evaluate superpixel segmentation for the goal of 3D scene modelling. This procedure provides a measure that shows if a given superpixels segmentation respects the 3D geometry of a scene, which is achieved by the mean of computing the error introduced when converting a dense depth map to a triangular mesh based on superpixels. This allowed to evaluate and test both proposed methods against the existing general-purpose state-of-the-art approaches.

Sparse 3D reconstruction : To increase the density of the reconstructed point cloud that is used to perform the planar structure fitting, we proposed a new approach that uses a combination of several matching methods and dense optical flow. In order to

control the impact that each reconstructed point has in the planar fitting procedure, we proposed to learn a weight by the mean of a dataset provided with ground truth. This did not only help to assign weights to all reconstructed points, but also to select the best combination of feature matching methods with minimum redundancy.

Planar structure fitting : The obtained point cloud is used to fit a piecewise planar structure, which is based on the second proposed superpixel method. For planar parameters estimation, we developed a weighted total least squares model that uses the reconstructed points and the learned weights to fit a planar structure with the help of superpixel segmentation of the input image sequence. Also, the model handles the occlusion boundaries between neighbouring scene patches to encourage connectivity and co-planarity to produce more realistic models. The validity of the proposed methods has been substantiated by comprehensive experiments by considering several criteria and a large variety of combinations. The experiments have been carried out mainly by using KITTI dataset which compromises a large number of realistic real-world sequences so the obtained results became steady.

Independent from our presented research, we exploited fusing depth learned from single image together with SFM to improve the structure estimation. Based on the depth estimation method proposed in [Saxena 2009b], we extended the Markov Random Field model to include new potential functions related to 3D reconstructed points using SFM technique, and also constrained by the limited planar motion of the vehicle. The obtained results are improved with respect to the depth computed using single image.

6.2 Contributions

A summary of the main contributions of this thesis are the following:

- 3D geometry respecting superpixel method based on a generalized boundary probability estimation using colour and flow information. The key advantage is a pixel-wise weighting in the fusion process takes into account the variable uncertainty of computed dense depth using optical flow.
- Superpixels evaluation method for the goal of 3D scene representation. This

procedure provides a measure that shows if a given superpixel segmentation respects the 3D geometry of a scene. This allows to evaluate and compare the the existing general-purpose state-of-the-art superpixel generation method.

- An extended simple local iterative clustering (SLIC) superpixel segmentation method to be adaptive to the sparse feature points density for more balanced 3D structure fitting. This is achieved through a new spatio-colour-temporal distance measure.

- Improved piecewise planar structure estimation pipeline from monocular image sequence. The point cloud density is increased by using a combination 3D points obtained from several feature points matching techniques including a noisy dense optical flow. A Weighted total least squares model is proposed to handle the uncertainty of each depth point. This uncertainty is provided by the mean of a learned weight.

- We exploit using depth learning from single image approach together with SFM to improve the 3D structure estimation. Based on the depth estimation method from single image presented in [Saxena 2009b], we extend the proposed Markov Random Field model to include new potential functions related to 3D reconstructed points using SFM technique, and also constrained by the limited planar motion of the vehicle. The obtained results are improved with respect to the depth computed using single image. However, the method proposed in the previous point provides better outputs.

6.3 Future Perspectives

Despite the numerous advances made by the research presented in this thesis towards structure estimation and piecewise scene representation, this area of research is by no mean finished. Further advances could be made in several directions, we list some of them in the following.

Applications of superpixels : We proposed an efficient superpixel generation method that respects the 3D scene structure and we introduced the application of 3D meshing. However, superpixel nowadays are used in many other applications such as object

recognition, tracking, 3D modelling. The current proposed methods in these domains use mostly the graph-based [Felzenszwalb 2004] and SLIC [Achanta 2012] superpixels. Based on the experimental study which show that our LABUV-PW provides better representation of the scene. Our next short term perspective is to investigate applying it to those applications.

Depth aware superpixel size : In clustering based superpixel methods such as SLIC, superpixels size is regular in the 2D image due to the spatial location component in the distance measure. In this way, the size of the back projection of these superpixels to 3D objects is a function of the distance to the camera. When the depth is available (for instance in RGB-D images, or when the depth is computed from optical flow), the depth component D can play the same role as the density map we used to control the superpixels size. Our goal will be to produce more superpixels at large distance than close distance so the scene is divided into roughly equal patches in 3D. Whereas other methods do so but only in 2D. The benefit of such application is that it provides a uniform planar approximation of a scene. Moreover, in the context of finding superpixels correspondence among an image sequence, having this property is realistic as objects projection size changes with depth changes. Having the size of superpixel constrained with its depth allows providing similar semantic segmentation over the image sequence. We consider this idea as a future perspective to explore.

Parameters setting : A possible future direction to improve the clustering based superpixel method arises from the encountered parameters that have to be set such as η, ξ and w_l. So far these parameters are fixed based on learning or empirically set. However, some aspects can be further exploited, such as the accuracy of the dense optical flow which can be quantified so the weight associated can be written as a function. Same applies to the lighting conditions of the scene, so that the weight associated to colour information can be set. This problem emerges as another future work perspective.

Implementation : In the planar structure estimation method, although we have obtained good 3D models by processing up to 30-40 frames [1], our structure estimation method still cannot analyse longer video sequences at once, because of several challenges in the modelling, odometery, and the implementation. Because we keep all

[1] Frames in KITTI dataset are captured with 1 meter intervals

colour information included within the image sequence, the point cloud size grow up rapidly (~ 15 mega points). Handling larger sizes (We use MESHLAB and Point Cloud Library (PCL)) is difficult and computationally consuming. Whereas down-sampling or converting to mesh leads to loose some details. We plan, as future research, to continue trying to further improve the computational complexity of the proposed pipeline, as well as a complete C++ single run implementation.

Dataset : As we have seen, in most of our work we use the KITTI for learning/testing the proposed approaches. Although this dataset is becoming widely popular (maybe because it is the best so far), it has been taken in one city with quite unique theme which repeats so often (simple houses, stationing cars on both sides, trees). This remains not up to modern big cities, which may not be the best scenario for testing, neither to be the best motivating application. We would like to have access to other datasets than KITTI to further test our methods.

Planar assumption : Perhaps the most important limitation of the developed approach for the structure estimation is the poor reconstruction of the objects that violates the planar assumption. Various future improvements could be sought. One is to divide the scene into planar and non planar regions based on object recognition or semantic segmentation system. A similar approach has been already seen in stereo vision such as the solution proposed in [Gallup 2010]. Non-planar objects tends generally to have more texture (*e.g.* trees) so the point cloud is supposed to be denser. These objects can be better reconstructed using surface reconstruction techniques rather than the piecewise representation. Our proposed framework can be extended by adding a recognition system to spot the nature of different surfaces, so an appropriate procedure can be applied then.

Conclusion général et perspectives

Dans cette thèse, nous avons présenté plusieurs nouvelles idées et améliorations par rapport à l'état de l'art afin de reconstruire la structure d'une scène 3D à partir de l'information de mouvement et d'images 2D monoculaires. Notre étude a porté sur la modélisation d'un environnement urbain perçu par une caméra embarquée dans un véhicule qui se déplace le long d'une route. Notre objectif a été de surmonter certains verrous, comme l'absence de texture ou le manque de redondance entre vues consécutives, grâce à une approche de reconstruction 3D par morceaux en surfaces planes. L'hypothèse de planéité permet d'obtenir, à partir d'un ensemble d'un nuage de points reconstruits épars, une estimation de la structure dense. Pour obtenir une reconstruction complète du nuage de points 3D nous avons utilisé la technique d'estimation de la structure par le mouvement (en anglais *Structure From Motion – SFM*). Dans cette thèse, nous avons introduits plusieurs améliorations dans la chaîne de traitements qui conduit à l'estimation de la structure d'une scène 3D à partir d'une modélisation et d'une représentation sous la forme de surfaces planes. Les améliorations apportées concernent les processus de traitement ci-dessous décrits.

(i) **Processus de représentation d'une scène 3D par morceaux** (en anglais *Piecewise scene representation*). Afin de modéliser, représenter, une scène 3D en surfaces planes, deux méthodes de regroupement de pixels similaires (en anglais *superpixel segmentation*) ont été proposées. La première méthode est basée sur l'estimation de la probabilité des discontinuités locales aux frontières des régions calculées à partir du gradient (en anglais *gradient-based boundary probability estimation*). Elle s'appuie sur une représentation multi-échelle pondérée qui fusionne les informations de couleur et de mouvement. L'idée d'introduire une pondération locale par morceaux à l'information de mouvement constitue

un avantage comparé à une pondération globale. Cela permet, non seulement d'obtenir une solution pour réduire l'influence du bruit dû au mouvement aux frontières des régions calculées, mais également de compenser les erreurs liés au calcul du flot optique grâce à l'introduction d'une pondération non linéaire fonction de la disparité. Cette méthode permet de générer des *superpixels* non contraints en termes de taille et de forme. Dans certaines applications, telles que le suivi de certains *superpixels* dans une séquence vidéo, il est nécessaire de contraindre la taille des *superpixels*. Nous avons donc développé une seconde méthode de segmentation en *superpixels* qui cette fois-ci est basée sur une technique simple, itérative, de regroupement local qui génère des *superpixels* de taille régulière. Cette méthode utilise d'une part les informations de mouvement et de couleur afin de générer des *superpixels* qui respectent les discontinuités locales, et d'autre part utilise une nouvelle mesure de densité qui prend en compte la densité des points au sein du nuage de points 3D. Cette méthode, basée sur le principe de l'algorithme SLIC (en anglais *Simple Linear Iterative Clustering*), a comme principal atout d'intégrer l'information de mouvement à la méthode de regroupements considérée, ce qui la différentie des aux autres techniques de l'état de l'art.

Nous avons également proposé une nouvelle technique d'évaluation de la qualité d'une segmentation en *superpixels* dédiée à la modélisation d'une scène 3D. Cette technique mesure si la segmentation obtenue respecte la géométrie 3D de la scène. Cette mesure évalue l'erreur d'estimation de la carte de profondeur quand celle-ci est générée par maillage triangulaire dense à partir des *superpixels*. Nous avons ainsi pu évaluer la qualité des deux méthodes de segmentation en *superpixels* proposées et les comparer par rapport aux autres méthodes de l'état de l'art.

(ii) **Processus de reconstruction 3D éparse** (en anglais *Sparse 3D reconstruction*). Afin d'augmenter la densité du nuage de points reconstruit, utilisé pour modéliser la structure de la scène sous forme de surfaces planes, nous avons proposé une nouvelle approche qui combine plusieurs méthodes d'appariement de descripteurs image (e.g. SIFT and SURF) et le flot optique dense. Afin de contrôler l'impact que peut avoir chaque point reconstruit sur le processus de modélisation d'une scène 3D en surfaces planes, nous avons proposé d'estimer par

apprentissage, le poids que l'on va associer à chaque point à reconstruire, à partir d'une base de données pour lesquelles on connait la vérité terrain. Ceci nous permet non seulement d'assigner un poids à chaque point à reconstruire, mais également de sélectionner la meilleure combinaison de méthodes d'appariement de descripteurs avec une redondance minimale.

(iii) **Processus de modélisation de la structure d'une scène par des surfaces planes** (en anglais *Planar structure fitting*). L'objectif ici est d'utiliser le nuage de points obtenu afin de modéliser par morceaux la structure d'une scène 3D sous forme de surfaces planes, lesquelles sont calculées à partir de la seconde méthode de segmentation en *superpixels* ci-avant mentionnée. Afin d'estimer les paramètres qui caractérisent ces surfaces planes, nous avons appliqué un processus des moindres carrés pondérés aux données reconstruites pondérées par les poids calculés par apprentissage, qui en complément de la segmentation par morceaux de la séquence d'images, permet une meilleure reconstruction de la structure de la scène sous la forme de surfaces planes. Nous avons également proposé un processus de gestion des discontinuités locales aux frontières de régions voisines dues à des occlusions (en anglais *occlusion boundaries*) qui favorise la coplanarité et la connectivité des régions connexes. L'objectif étant d'obtenir une reconstruction 3D plus fidèle à la réalité de la scène.

L'ensemble des modèles proposés permet de générer une reconstruction 3D dense représentative à la réalité de la scène. La pertinence des modèles proposés a été étudiée et comparée à l'état de l'art. Plusieurs combinaisons de méthodes d'appariement de descripteurs et plusieurs critères d'étude ont été analysés. Plusieurs expérimentations ont été réalisées afin de démontrer, d'étayer, la validité de notre approche. Ces expérimentations ont été menées en utilisant la base KITTI, dont l'une des particularités est de disposer d'un grand nombre de séquences urbaines acquises dans des conditions réelles et pour lesquelles on dispose d'une vérité terrain.

Indépendamment des travaux de recherche ci-dessus mentionnés, nous avons également cherché à fusionner l'information de profondeur estimée à partir d'une image monoculaire avec les informations extraites par la SFM, et ce afin d'améliorer l'estimation de la structure d'une scène 3D. Pour cela, nous avons introduit une

nouvelle méthode d'estimation de la profondeur qui, contrairement à la méthode proposée par Saxena en 2009, prend à la fois en compte l'information extraite par la SFM et la contrainte selon laquelle dans l'application visée un véhicule ne peut avoir qu'un mouvement plan. Cette méthode est basée sur l'utilisation les champs de Markov. Les résultats expérimentaux obtenus ont permis de quantifier l'amélioration apportée par la méthode proposée.

A la fin de chaque chapitre de cette thèse, nous avons récapitulé l'ensemble de nos contributions, mis en perspectives, discuté, les principaux atouts de nos propositions et éventuels inconvénients, puis dressé quelques perspectives.

Pour finir, nous proposons plusieurs pistes de recherche afin : - soit d'améliorer la performance des algorithmes développés (e.g. les temps de traitement et les ressources mémoires nécessaires); - soit d'améliorer la prise en compte de l'information de profondeur (e.g. afin de contraindre la taille d'un *superpixel* en fonction de sa profondeur) ; - soit d'aller plus loin dans la prise en compte, la combinaison, d'information supplémentaires (e.g. afin de prendre en compte les surfaces non planes ou texturées ou afin d'améliorer la paramètrisation des pondérations utilisées); - soit d'étendre les méthodes proposées à d'autres domaines d'application ou d'autres bases de vidéos, ou d'autres champs d'investigation (e.g. *object recognition, tracking, 3D modelling*).

A List of Publications

1. Nawaf, Mohamad Motasem and Trémeau, Alain . "Monocular 3D Structure Estimation for Urban Scenes". Submitted to Elsevier Image and Vision Computing (Under review since 06/2014).

2. Nawaf, Mohamad Motasem and Trémeau, Alain . "Monocular 3D Structure Estimation for Urban Scenes". IEEE International Conference on Image Processing (ICIP), 2014.

3. Nawaf, Mohamad Motasem and Md Abul, Hasnat and Sidibé, Désiré and Trémeau, Alain . "Color and Flow Based Superpixels for 3D Geometry Respecting Meshing." IEEE Winter Conference on Applications of Computer Vision (WACV), 2014.

4. Nawaf, Mohamad Motasem, and Trémeau, Alain. "Fusion of Dense Spatial Features and Sparse Temporal Features for Three-Dimensional Structure Estimation in Urban Scenes." IET Computer Vision 7.5 : 302-310, 2013.

5. Nawaf, Mohamad Motasem, and Trémeau, Alain. "Joint Spatio-Temporal Depth Features Fusion Framework for 3D Structure Estimation in Urban Environment." European Conference on Computer Vision (ECCV) 2012. Workshops and Demonstrations. 526-535. 2012.

B Spatio-Temporal Depth Fusion for Monocular 3D Reconstruction

Contents

In this chapter we present a novel approach to improve 3D structure estimation from an image stream in urban scenes. **The work presented here is independent from what we have already presented in other chapters**. However, this work, which is complementary to the research presented in this thesis, was proposed earlier and published in [Nawaf 2012, Nawaf 2013]. Here, we also consider the particular setup where the camera is installed on a forward moving vehicle. Our idea is to introduce the monocular depth cues that exist in a single image, and add time constraints to improve the 3D structure estimation with respect to structure from motion traditional techniques. As in our previous work, the scene is also modelled as a set of small planar

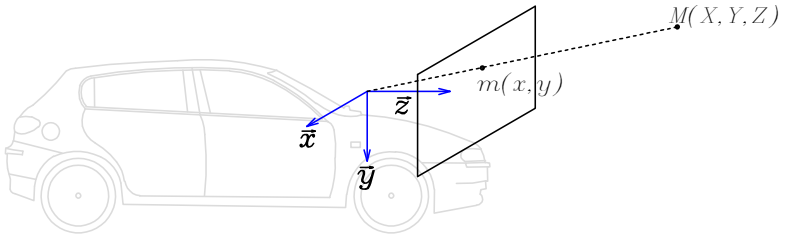


Figure B.1: Acquiring geometry: Camera installed on a moving vehicle with Z axis coincides with forward motion direction.

patches obtained using over-segmentation, and the goal is to estimate the 3D positioning of these planes. We propose a fusion scheme that employs Markov Random Field (MRF) model to integrate spatial and temporal depth features. Depth from spatial features is obtained by learning a set of global and local image features. Temporal depth is obtained via sparse optical flow based structure from motion approach. That allows decreasing the estimation ambiguity by forcing some constraints on camera motion. The proposed MRF model is then solved using convex optimization techniques. The experiments show that the joint spatio-temporal method overcomes the performance of the depth estimation from single image. Also, it provides a dense depth estimation which is an advantage over SFM.

B.1 Introduction

Estimating the 3D structure of a scene from 2D image stream is one of the most popular problems within computer vision. It is referred to as structure from motion (SFM). SFM has been applied in several applications [Aanæs 2003] such as robot navigation, obstacle avoidance, entertainments, driver assistance, reverse engineering and modelling, etc.

In this work, we focus on the problem of estimating the 3D structure from a video taken by a camera installed on a moving vehicle in urban environments. This setup leads possibly to create 3D maps of our world. However, the dominant forward motion of the camera from one side, and the texture-less scenes that are present generally in urban environment produce an erroneous depth recovery. The forward camera motion could result degenerated configurations for a naturally ill-posed problem,

or mathematically, a large number of local minima during the minimization of the re-projection error [Vedaldi 2007]. That results in inaccurate camera relative motion estimation. Moreover, the limited lifetime of tracked feature points prevents using general optimization methods such as in traditional SFM. Additionally, forward motion restricts features matching due to non-homogeneous scale changes of image objects, especially those aligned parallel to camera movement.

Here, we suggest to benefit from the monocular cues (*e.g.* spatial depth information) to improve depth estimation. We believe that such spatial depth information is complementary to temporal information. For instance, given a blue patch located at the top of an image, a SFM technique will probably fail to compute the depth due to the difficult matching problem from one side, and being in the blind zone of the vision system in the other side, while the monocular depth estimation method (supervised learning) will assign it the largest defined depth value as it will be considered as a sky with high probability.

Similar to other works [Saxena 2009b, Liu 2010], we consider that the urban world is made up of small planar patches, and the relationship between each two patches is either connected, planar or occluded. Based upon these considerations, the goal is to estimate the plane parameters where each patch lies. These patches are obtained from the image using over-segmentation method [Felzenszwalb 2004] or what is called superpixels segmentation. In order to fuse both temporal and monocular depth information, and also to handle the interactive relationship between superpixels, we propose to use a MRF model similar to the one used in [Saxena 2009b]. However, we extend the model by adding new terms to include temporal depth information computed using a modified SFM technique. Moreover we benefit from the limited Degrees of Freedom (DoF) of camera motion (which is such of the vehicle) to improve relative motion estimation, and in return, the depth estimation.

Spatial depth information is obtained using an improved version of the method proposed in [Saxena 2009b], which estimates the depth from a single image. The method employs a MRF model that is composed of two terms; one integrates a broad set of local and global features, while the other handles the neighbouring relationship between superpixels based on occlusion boundaries. In our method, we compute occlusion boundaries from motion [Humayun 2011] to obtain more reliable results

than using a single image as in the aforementioned method. Therefore, it is expected to have better reconstruction, even before integrating the temporal depth information.

To perform SFM, which represents temporal depth information, we use optical flow based technique that allows forcing some constraints on camera motion (which has limited DoF). Moreover, it is proved to have better depth estimation for small baseline distances and forward camera motion [Forsyth 2002]. Here, we compute a sparse optical flow using an improved method of Lucas-Kanade with multi-resolution and sub-pixel accuracy. Based on the famous optical flow equation [Ma 2004], we obtain the depth for a set of points in the image. Hence we can add some constraints on the position of scene patches to whom these points belong.

The remaining of this chapter is organized as follows. In Section B.2, we introduce the MRF model that integrates SFM with the monocular depth estimation, and we explain its potential functions, parameters learning and inference. Section B.3 presents our experiments and the results of evaluating our method. And finally, in Section B.4 we conclude our work and we discuss the advantages of the proposed method.

B.2 Spatio-Temporal Depth Fusion Framework

In this section, we first introduce some notations. Then we explain how we compute spatial and temporal depth features. After that, we discuss how to estimate occlusion boundaries, which play an important role in the proposed model. Next, we introduce the proposed framework as an MRF model that incorporates several terms related to spatial and temporal depth features. Finally we show how we estimate the parameters from a given dataset and perform the inference for a new input.

B.2.1 Image Representation

As mentioned earlier, we assume that the urban world is composed of planar patches, and the obtained superpixels are their *one-to-many* 2D projection. This assumption represents a good estimate if the number of computed superpixels is large enough. We obtain the superpixels from an image by using the over-segmentation algorithm [Felzenszwalb 2004], which is based on graph-cuts. The pixels are represented as

nodes and the edges are computed as the similarity between nodes. Then, superpixels are obtained by applying the minimum spanning tree algorithm. At this step, there are two parameters that controls the superpixel formation, which have to be defined. First, the standard deviation σ of a preprocessing smoothing Gaussian. Although this parameter aims at de-noising the image, it also prevents forming small superpixels caused by sharp patterns or noise. Therefore, it is preferable to set this variable to large values here for more efficient learning and also to have a larger number of overlaid SFM points. In our experiments we set $\sigma = 1.6$. The other parameter k controls the size of the formed superpixels. So it controls (approximately) the number of obtained superpixels. Due to the limited laser data resolution available as a ground truth for spatial depth learning (which is 55×305 in Make3D dataset [Saxena 2007]), the number of superpixels has to be limited so that there is enough depth information available to each superpixel, so it depends on image resolution. Here, we use $k = 1000$ for Make3D dataset, $k = 1500$ for our own acquired dataset and $k = 700$ for KITTI dataset [Geiger 2012].

Formally, we represent the image as a set of superpixels $\mathbf{S}^t = \{S_1^t, S_2^t, ..., S_n^t\}$, where S_i^t defines superpixel i at time (frame) t. We define $\alpha_i^t \in \mathbb{R}^3$ the plane parameters associated to S_i^t such that for a given point $x \in \mathbb{R}^3$ on the plane satisfies $\alpha_i^t x = 1$. Our aim is to find the plane parameters for all superpixels in the image stream. Figure B.3b shows an example of an original image and the corresponding superpixels.

B.2.2 Spatial Depth Features

Spatial features for supervised depth estimation have not achieved much success compared to other computer vision domains such as object recognition and classification. Although the problem of monocular vision had been well studied in human vision (even before computers appear) and many monocular depth cues that human uses have been identified, however, it was not possible to obtain explicit depth representative measurements such as in stereo vision. Recently, there were several attempts to infer image 3D structure using spatial features and supervised learning [Saxena 2009b, Liu 2010, Sturgess 2009]. In our method, we proceed in similar way, in order to capture texture information, the input image is filtered with a set of texture energies and gradient detectors ($\sim$ 20 filters) [Saxena 2009a]. Then by using superpixel

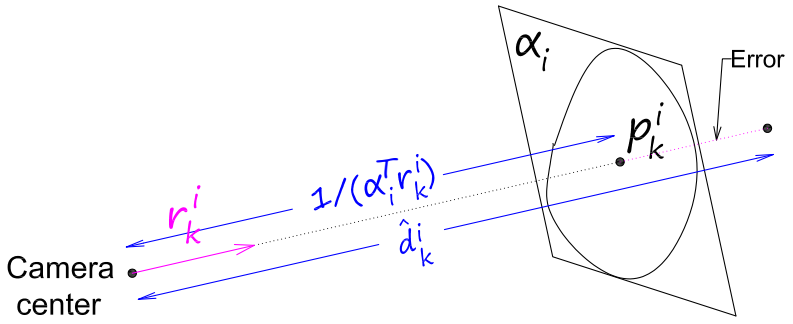


Figure B.2: Illustration for how to compute the error in depth between the estimated value and the depth for a given α_i.

segmentation image as a mask, we compute the filter response for each superpixel by summing its pixels in the filtered image. We refer the reader to [Saxena 2009a] for more details. In order to capture general information, the aforementioned step is repeated for multiple scales of the image. Also, to add contextual information, *e.g.* texture variations, each superpixel feature vector includes the features of its neighbouring superpixels. Additionally, the formed feature vector includes colour, location, and shape features as they provide representative depth source for fixed camera configuration and urban environment. For instance, recognizing the sky and the ground. These features are computed as shown in table 1 in [Hoiem 2005]. We denote $\mathbf{X_i^t}$ the feature vector for superpixel S_i^t.

B.2.3 Temporal Depth Features

In this subsection, we first describe some mathematical foundations and camera model. Then we explain how to perform sparse depth estimation which will be integrated in the probabilistic model given in subsection B.2.5.

We use a monocular camera mounted on a moving vehicle. We assume that the Z axis of the camera coincides with the forward motion of the vehicle as shown in Figure B.1. Based on pin-hole camera model and camera coordinate system, a given $3D$ point $\mathbf{M}(X,Y,Z)$ is projected onto the $2D$ image as $\mathbf{m}(x,y)$ by a perspective projection:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{B.1}$$

When the vehicle moves, which is also equivalent to fixed camera and moving world, the relationship between the velocity of a 3D point $[\dot{X}\,\dot{Y}\,\dot{Z}]^T$ and the velocity of its 2D projection $[\dot{x}\,\dot{y}]^T$ is given as the time derivative of equation B.1. Then, based on the well-known optical flow equation

$$\dot{\mathbf{M}} = -\mathbf{T} - \Omega \times \mathbf{M} \tag{B.2}$$

and assuming a rigid scene, the 3D velocity is decomposed into translational **T** and rotational Ω velocities [Ma 2004]. Hence we obtain equation B.3 which is the essence of most optical flow based SFM methods.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix} . \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} + \begin{bmatrix} xy/f & -f-(x^2/f) & -y \\ f+(y^2/f) & -xy/f & x \end{bmatrix} \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \tag{B.3}$$

Based on this equation, we proceed in computing a sparse depth. We estimate the relative camera motion between two adjacent frames by first performing SIFT feature points matching [Lowe 2004]. Next we estimate the fundamental matrix using RANSAC [Raguram 2008]. Then, given camera intrinsic parameters, we can obtain the Essential matrix that encodes the rotation and translation (which is up to scale) between the two scenes. This represents also the relative camera motion parameters $[\mathbf{T}\,\Omega]$. To reveal the scale ambiguity we employ the re-projection based method proposed in [Esteban 2010]. We track feature points over frames, then by using a shifting 3 frames window we compute a frame to frame translation scale by projecting the trackable points on a reference frame after introducing a scale factor between two frames. The scale factor is then computed by minimizing a least square set of equations using Singular Value Decomposition (SVD). Hence we compute a correct frame to frame scale for the sequence of images. However, having first frame scale set to $[\mathbf{I}|\mathbf{0}]$, we have an overall unknown scale. In our case, given that we are dealing with fixed configuration we could set this scale using metric measures.

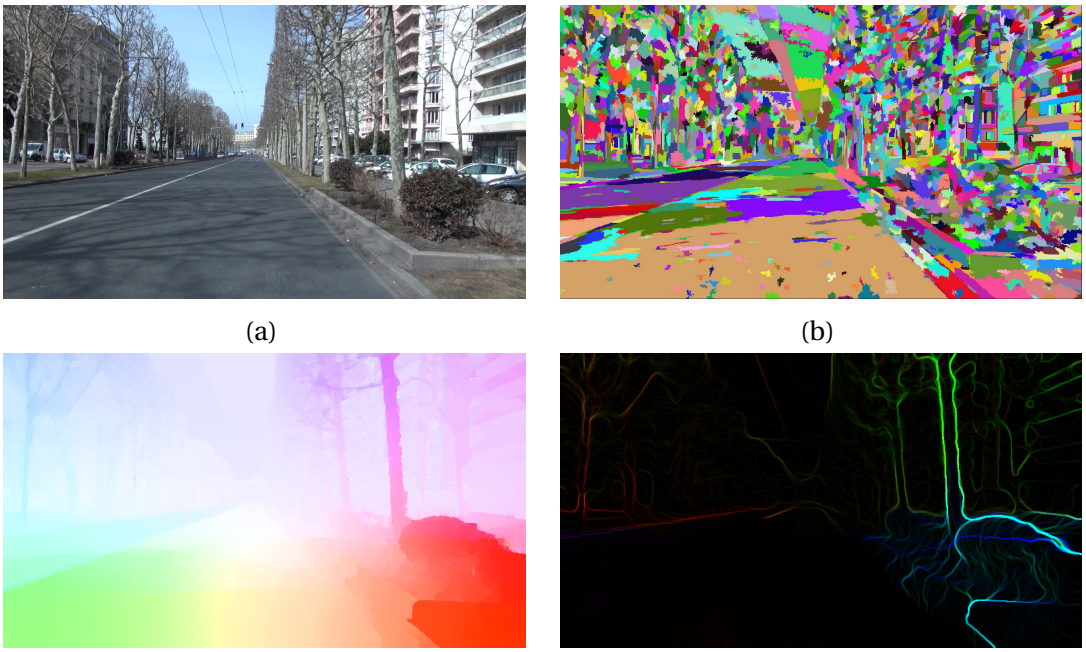
The left hand side of equation B.3 is basically the optical flow computed between two frames. In our implementation it is obtained using the well-known Lucas-Kanade with multi resolution and sub-pixel accuracy. Moreover, we benefit from the estimated Fundamental matrix to reject outliers in the optical flow. At this point, we could

compute an approximate depth for the selected feature points. Specifically, we set a threshold for the difference between x and y disparities. In case of large difference (which means the pixel is close to image axes but far from the centre) we compute the depth using only the larger component. We think this is an advantage over traditional 3D triangulation method where both x and y are treated equally. However, this additional step is applied only when we spot dominant forward motion, in which our assumption is only true.

Besides, given the specific camera setup as shown in Figure B.1, the motion of the camera is not totally free in the 3D space (motion of a vehicle). Therefore, we could add some constraints that express the feasible relative camera motion between two frames. For instance, limitation in T_y and Ω_z velocities. However, due to the absence of essential physical quantities, precise constraints on camera (or vehicle) motion could not be established theoretically. Instead, we evaluate experimentally possible camera motion estimated from a set of video sequences acquired in different scenarios. As a result, we can establish some roles to spot outliers in the newly computed values for relative camera motion $[\mathbf{T}\,\Omega]$. This way we improve the relative camera motion estimation in our case as we regularly have degenerated configurations (due to small baseline variations and dominant forward motion as mentioned earlier).

B.2.4 Occlusion Boundaries Estimation

When the camera translates, close objects move faster than far objects, and hence this causes to change the visibility of some objects in the scene. Although this phenomenon is considered as a problem in computer vision, it provides an important source of information about 3D scene structure. In our approach, we benefit from motion to infer occlusion boundaries. We use the method proposed in [Humayun 2011] to generate a soft occlusion boundary map from two consecutive image frames. The method is based on supervised training of an occlusion detector thanks to a set of visual features selected by a Random Forest (RF) based model. Since occlusion boundaries lie close to surfaces edges, we use the classifier output as an indicator to the relationship between two superpixels if they are connected or occluded. Hence we add a penalty term in our MRF that forces the connectivity between superpixels. This term is inversely-proportional to the obtained occlusion indicator. Figure B.3c shows



(a) (b)

(c) (d)

Figure B.3: (a) Original image. (b) Superpixels segmentation. (c) Occlusion surfaces. (d) Estimated occlusion boundary map (colour coded from green (strong boundary) to red (weak boundary)).

occlusion surfaces where pixels follow common motion, while Figure B.3d shows the estimated occlusion boundary map.

B.2.5 Markov Random Field for Depth Fusion

Markov Random Field (MRF) is becoming increasingly popular for modelling 3D world structure due to its flexibility in terms of adding appearance constraints and contextual information. In our problem, we formulate the depth fusion as an MRF model that incorporates certain constraints with variable weights so that they are jointly respected. Furthermore, we preserve the convexity of our problem such as in [Saxena 2009b] to allow solving it through a linear program rather than probabilistic approaches for less computational time.

We have seen earlier how to obtain temporal depth information, monocular depth features and occlusion boundaries. Figure B.4 shows a simplified process flow for the proposed framework. This flow is implemented within one MRF model, which we

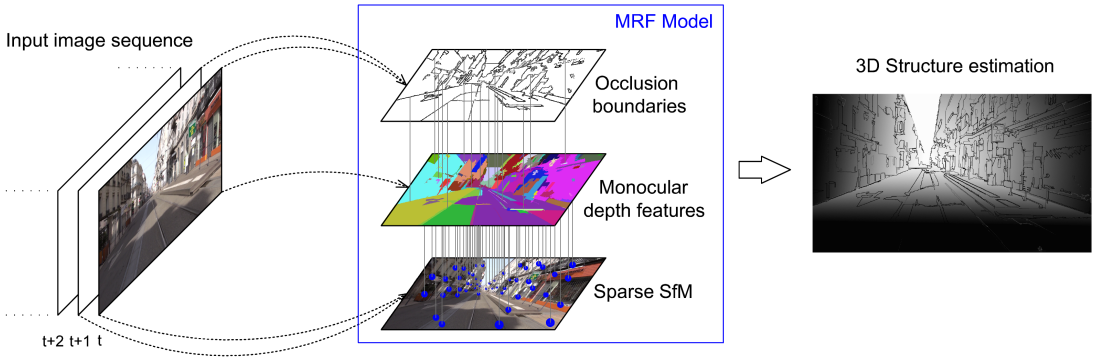


Figure B.4: Graphical representation of our MRF; for a given input of image sequence, occlusion boundaries and sparse SFM are estimated from two frames t and $t+1$, while monocular depth features are extracted from the current frame t, the MRF model integrate this information in order to produce a joint result of 3D structure estimation

formulate to includes all of aforementioned information as:

$$E(\boldsymbol{\alpha}^t|\mathbf{X^t},\mathbf{O},\hat{\mathbf{D}},\boldsymbol{\alpha}^{t-1};\boldsymbol{\theta}) = \underbrace{\sum_i \psi_i(\alpha_i^t)}_{\substack{\text{spatial depth}\\ \text{term}}} + \underbrace{\sum_{ij} \psi_{ij}(\alpha_i^t,\alpha_j^t)}_{\substack{\text{connectivity}\\ \text{term}}} + \underbrace{\sum_{ik} \phi_{ik}(\alpha_i^t,\hat{d}_k^i)}_{\substack{\text{temporal}\\ \text{depth term}}} + \underbrace{\sum_i \phi_i(\alpha_i^t,\alpha_i^{t-1})}_{\substack{\text{time}\\ \text{consistency}\\ \text{term}}} \tag{B.4}$$

where the superscripts t and $t-1$ refer to current and previous frames. $\mathbf{X}$ is the set of superpixels feature vectors. $\mathbf{O}$ is a map of occlusion boundaries computed from the frames t and $t-1$. The estimated sparse depth is $\hat{\mathbf{D}}$, while $\hat{d}_k^i$ is the estimated depth value for pixel k in superpixel i. α_i is superpixel i plane's parameters and $\boldsymbol{\alpha}$ is the set of parameters for all superpixels. $\boldsymbol{\theta}$ are the learned monocular depth parameters. We now proceed to describe each term in this model (In the first three terms we will drop down the superscript of frame indicator t for simplicity as they are in the same frame).

Spatial Depth Term

This term is responsible for penalizing the difference between the computed plane parameters and the ones estimated from spatial depth features (based on the learned parameters $\boldsymbol{\theta}$). It is given by the accumulated error for all pixels in the superpixel. See

[Saxena 2009a] P36-37 for details. For simplification, let's define a function $\delta(d_k^i, \hat{d}_k^i)$ that represents one point fractional depth error between an estimated value $\hat{d}_j^i$ and actual value d_j^i given plane parameters α_i. This potential function is given as

$$\psi_i(\alpha_i) = \beta_1 \sum_j v_k^i \delta(d_k^i, \hat{d}_k^i) \tag{B.5}$$

where v_k^i is a learned parameter that indicates the reliability of a feature vector $\mathbf{X}_k^i$ in estimating the depth for a given point p_k^i, see [Saxena 2009b] for more details. β_1 is a weighting constant.

Connectivity Prior

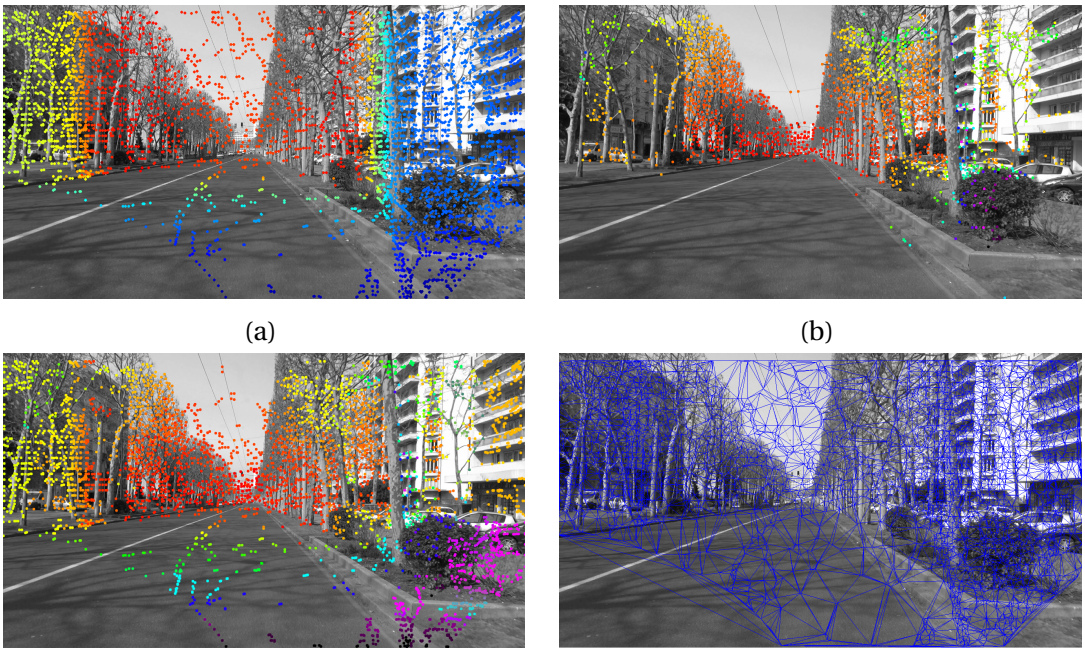
This term is based on the map of occlusion boundaries **O** explained earlier. For each two adjacent superpixels, we compute an occlusion boundary indicator by summing up all pixels located at the common border in the estimated map. The obtained occlusion indicators are normalized so that they are in the range [0..1]. We refer o_{ij} to the indicator between superpixels i and j. The potential function is computed for each two neighbouring superpixels by choosing two adjacent pixels from each. The function penalizes the difference in distance between each of them to the camera. We have

$$\psi_{ij}(\alpha_i, \alpha_j) = \beta_2 \, o_{ij} \sum_{k=l=1}^{2} \delta(d_k^i, d_l^j) \tag{B.6}$$

where β_2 is a weighting constant. This potential function forces neighbouring superpixels to be connected only if they are not occluded with the help of occlusion indicator o_{ij}. In comparison with the original method [Saxena 2009b], we drop down the co-planarity constraint as we believe that the included temporal information and estimating occlusion boundaries indicator for motion provide an important source of depth information about plane orientation. Therefore, we do not mislead the estimation procedure with such approximation.

Temporal Depth Term

This term enforces some constraints that are established from the set of points where the depth is known. It is evident that with three non-collinear points we can obtain



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

Figure B.5: (a) Depth estimation from single image. (b) Depth estimation using SFM technique. (c) The estimated depth using the combined method. (d) The triangulations associated with the depth estimation shown in (c).

plane parameters α_i. However, to consider less or more number of points, we formulate this potential function to penalize the error between the estimated depth $\hat{d}_k^i$ for a point $p_k^i \in S_i$, and the computed depth given plane parameters α_i. Figure B.2 shows how this error is computed. Hence we have

$$\phi_{ik}(\alpha_i, \hat{d}_k^i) = \beta_3 |\hat{d}_k^i - 1/{\alpha_i}^\top r_k^i| \tag{B.7}$$

where r_k^i is a unit vector that points from camera centre to the point p_k^i. And β_3 is a weighting constant. We compute absolute depth error rather than fractional error since SFM is more confident than spatial depth estimation.

Time Consistency Term

In case of more than two frames, the quality of the 3D structure estimation varies from one frame to another, and it depends highly on the relative camera motion components (larger T_x and T_y translational motions results in better 3D structure

estimation). Therefore we add some penalty in order to guide depth estimation at time t given the estimation at time $t-1$. This smooths the overall estimated structure variations in time. Hence, for each superpixel S_i^{t-1} we find its correspondence S_i^t based on the motion parameters and the size of common area. Additionally, we consider some visual features such as colour and texture. Eventually some superpixels will not have correspondence due to changing the field of view. We select the point p_k^i at the centre of the S_i^{t-1} and we form a ray from camera centre to this point. This ray intersects with superpixel S_i^t at point $p_k^{i\prime}$. The formulated potential function penalizes the distance across the ray between the two points

$$\phi_i(\alpha_i^t, \alpha_i^{t-1}) = \beta_4 \delta(d_k^{i\prime}, \hat{d}_k^i) \tag{B.8}$$

here β_4 is a smoothness term. We intend to only use one point to leave some freedom in plane orientation and for better 3D reconstruction refinement.

B.2.6 Parameters Learning and Inference

In our MRF formulation we preserve the convexity as all terms are linear or L_1 norm, which is solved using linear programming. To learn the parameters, we first proceed with the first two terms of equation B.4. We assume unity value for the parameters β_1 and β_2. The two parameters θ and ν are learned individually [Saxena 2009a] using the Make3D dataset wich comes with ground-truth. For the rest of the parameters, β_1 and β_2 defines how the method is spatially oriented, while large β_3 turns the method into conventional SFM. β_4 allows previous estimation to influence the current one. Hence the weighting constants $\beta_{1..4}$ depends on the context, although they could be learned through cross-validation.

B.3 Experiments and Results

It exists few datasets and benchmarks to evaluate 3D reconstruction methods from image sequences [Meister 2012, Geiger 2012, Pandey 2011]. In our experimental evaluation, we use the “KITTI vision benchmark suite” [Geiger 2012] which comprises

	Error ratio	STD
SIE/SFM	1.82	0.43
SFM/Combined	2.24	0.21
SIE/Combined	2.81	0.38

Table B.1: Experimental results of spatial (SIE), temporal (SFM) and combined methods

various sets of image sequences taken from a moving vehicle as in the assumed setup[1]. This is mainly due to its higher image and laser scanner resolution. Also, it has more diverse scenes taken in many scenarios.

As our aim is mainly to evaluate the proposed fusion scheme, we perform a 3D reconstruction on the given dataset using the three methods; 3D structure estimation from single image (SIE), the optical flow based structure from motion (SFM) explained in section B.2.3, and finally the proposed combined method. Thanks to the provided laser scanner data we could compute the error for each case as direct differences between the estimation and the ground truth. To be turned into representative measures, we computed the ratio between the error for each of the two baseline and the combined methods, here we convert the sparse SFM to dense by using weighted average to compute the depth for an intermediate point (unlike the results shown in Table B.2 where we consider sparse SFM). Table B.1 shows the error ratios between these methods averaged over the used image sequences, this way we can evaluate the performance of the combined method with respect to spatial or temporal component. The table also shows the standard deviation associated with each ratio which gives an idea about the stability of the results for different scenarios. From these results we could conclude that both spatial and temporal depth estimations are partially complementary as the combined method has better performance than each individual method. Figure B.5 shows an example of the results obtained using each method, the triangulations shown in B.5d helps to compute dense depth estimation.

From another side, we evaluate the error distribution for each of the three methods as a function of depth. Table B.2 shows the relative error $|\frac{\hat{d}}{d} - 1|$ between the estimated

[1] Raw data section, sequences # 0001-0013, 0048, 0056, 0059, 0091-0106

depth $\hat{d}$ and the laser scanner[2] measure d. In case of dense depth we only compute the error for the points where laser scanner data is available. While in case of sparse depth, we look for the nearest neighbouring point within small distance, if no such point exists, the estimated point is not considered in the computation. The second column in Table B.2 is for the results of SFM points, while the third column is only for the points used to compute the fundamental matrix (FM) (100-150 points in average). As expected, the sparse SFM points tend to be more accurate for close distances, while the large error for distances larger than 50(m) ensures the fact that SFM is blind for large distances. The depth estimation from single image shows similar depth error for all depth ranges. While the combined method gains an improvement over SIE over all ranges. Another remark that we notice here is the improvement in the combined method with the large depth range with respect to SFM (sparse case) which is ~ 12%. In total, although the error in SFM is slightly smaller than the combined method, as a return the combined method provides a dense depth estimation.

Depth range (meters)		0-10	10-20	20-30	30-40	40-50	50-80	All
Error %	SIE (dense)	23.52	29.05	29.44	23.21	22.02	24.74	**26.41**
	SFM (sparse)	11.82	9.06	14.92	18.81	23.69	40.69	**16.65**
	SFM (sparse, FM)	5.10	4.82	8.64	8.87	13.48	30.15	**9.14**
	Combined (dense)	14.94	16.86	17.88	22.22	21.97	28.37	**18.65**

Table B.2: Relative error distribution as a function of depth.

We found it interesting to study the effect of the number of matching points in SFM on the final relative error of the combined method. Figure B.6 shows the results obtained for 180 matching frames. Each couple of matching frames is associated with one point that relates the number of matching points (a) or the number of inliers used to compute the Fundamental matrix using RANSAC (b) against the relative error in the combined method. It is clear that in both figures there is an improvement in the results when we have more matching points since being more reliable depth source. We also evaluate the robustness of the trajectory estimation and compare its accuracy to the ground truth that is provided by an Inertial Navigation System (GPS/IMU). Figure B.7 shows two examples (sequences 0009 and 0095) of the computed trajectory and the provided ground truth superimposed onto a Google Earth maps. The

[2]We set the maximum distance to 80 meters which is the limit of the used LIDAR

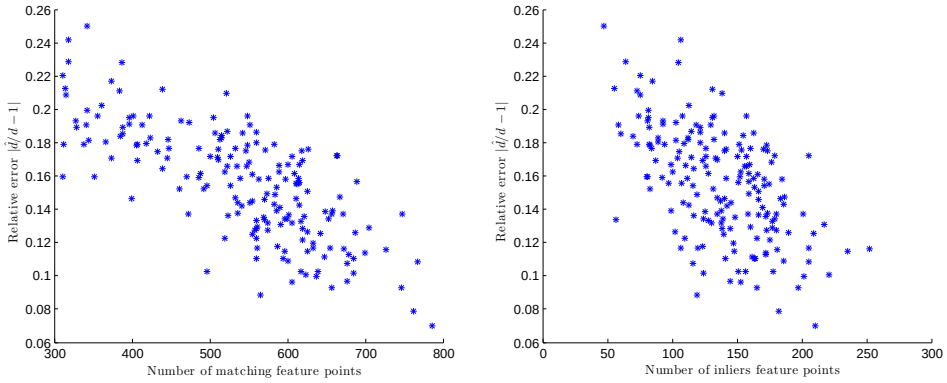


Figure B.6: Estimated depth relative error $|\frac{\hat{d}}{d}-1|$ versus (left) Number of matching feature points (frame to frame) (right) Number of inliers feature points used to compute the Fundamental matrix using RANSAC.

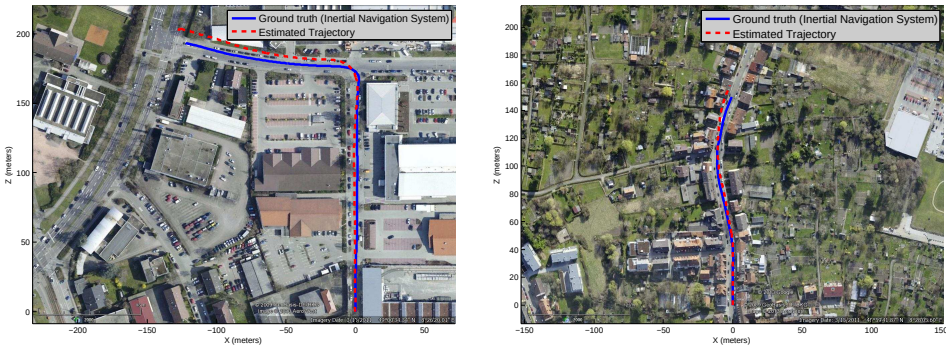


Figure B.7: Estimated trajectory (dashed red) and ground truth (blue) obtained from Inertial Navigation System (GPS/IMU) superimposed onto a Google Earth image of KITTI dataset sequences 0009 (left) and 0095 (right)

estimated trajectories gave an average translation error of 6.8% and rotation error 0.0187 $[deg\backslash m]$. Compared to non-constrained trajectory estimation we had an overall average improvement of translation error by 0.9% and rotation error by 3%. As expected, this improvement mainly applies to y direction (vertical), while it is equal for rotations.

Our implementation requires 85 seconds to perform the 3D estimation for 3 frames on a multi-core Linux PC (Intel I7, 8 GB Ram). Most of this time is allocated for the intensive spatial features extraction, while the feature points extraction and matching runs in parallel. For longer sequences the time increases linearly since our method performs local refinement.

B.4 Discussion and Conclusion

We have presented a novel framework to perform 3D structure estimation from an image sequence, which combines both spatial and temporal depth information to provide more reliable reconstruction. Temporal depth features are obtained using a sparse optical flow based structure from motion technique. The spatial depth features are obtained through a broad global and local feature extraction phase that tries to capture monocular depth cues. Both depth features are fused by the mean of an MRF model to be solved jointly. The experiments show that the joint method overcomes the performance of the estimation from single image. Also, it provides a dense depth estimation which is an advantage over SFM. By analysing the depth estimation relative error with respect to depth range we conclude that both used depth features are complementary to each other. Monocular depth features are independent from depth range, and SFM is blind for large distances. We also conclude that the joint method provides better performance than computing dense depth map using sparse SFM without taking colour consistency into account.
Although it is not our primary objective, trajectory estimation proved to be robust and accurate after introducing the constraints which are adapted to vehicle motion. Based on the results published in KITTI visual odometry benchmark [Geiger 2012], the proposed framework provides odometry estimation that is close to stereo based visual odometry methods.

The main limitation of the proposed approach is due to the possible failure of the monocular depth features. We encountered poor performance in estimating the depth in some cases such as: uncommon shape/colour/texture, lightning conditions, which affects the overall performance. This is the main reason that we went in a different direction by proposing the method in *Chapter 5* which is more robust, reliable and provides better outputs, moreover, easier to solve. However, the domain of depth estimation is still promising and new methods are being proposed to improve the current state-of-the-art. We think that to have better results for depth estimation from single images, is to go from general to specific, *i.e.* some geometrical constraints have to be made on the scene to benefit from the prior we have about urban environment.

To summarize, the major contribution that we proposed in this chapter is we improve 3D structure estimation by fusing SFM sparse output with monocular depth estimation learned from single image, so we obtain a dense 3D estimation. We extend the Markov Random Field model proposed in [Saxena 2009b] by integrating two potential functions that includes sparse SFM output. Moreover, the model is adapted to a looking forward camera installed on a mobile vehicle. We use the fixed configuration to estimate more accurate visual odometery.

Note that this chapter is based on the published articles [Nawaf 2012, Nawaf 2013]

Bibliography

[Aanæs 2003] H. Aanæs. *Methods for structure from motion.* PhD thesis, Danmarks Tekniske Universitet, 2003.

[Achanta 2012] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua and Sabine Susstrunk. *SLIC superpixels compared to state-of-the-art superpixel methods.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, pages 2274–2282, 2012.

[Agarwal 2009] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz and R. Szeliski. *Building Rome in a day.* In Computer Vision, 2009 IEEE 12th International Conference on, pages 72–79, Sept 2009.

[Alvarez 2010] J.M. Alvarez, T. Gevers and A.M. Lopez. *3D Scene priors for road detection.* In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 57–64. IEEE, 2010.

[Alvarez 2012] J.M. Alvarez, T. Gevers, Y. LeCun and A.M. Lopez. *Road scene segmentation from a single image.* In ECCV 2012, Part VII, LNCS 7578, pages 376–389, 2012.

[Badino 2011] H Badino, Huber D and Kanade T. *The CMU Visual Localization Data Set.* Website, 2011. http://3dvis.ri.cmu.edu/data-sets/localization,.

[Baillard 1999] Caroline Baillard, Cordelia Schmid, Andrew Zisserman, Andrew Fitzgibbon*et al. Automatic line matching and 3D reconstruction of buildings from multiple views.* In ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, volume 32, pages 69–80, 1999.

[Bao 2011] S.Y. Bao and S. Savarese. *Semantic structure from motion.* In Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, pages 2025–2032. IEEE, 2011.

[Bay 2006] H. Bay, T. Tuytelaars and L. Van Gool. *Surf: Speeded up robust features.* European Conference on Computer Vision, pages 404–417, 2006.

[Bódis-Szomorú 2014] András Bódis-Szomorú, Riemenschneider Hayko and Van Gool Luc. *Fast, Approximate Piecewise-Planar Modeling Based on Sparse Structure-from-Motion and Superpixels.* In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conferenceon, pages 57–64. IEEE, 2014.

[Cigla 2012] C. Cigla and A.A. Alatan. *An improved stereo matching algorithm with ground plane and temporal smoothness constraints.* In ECCV 2012 Ws/Demos, Part II, LNCS 7584, pages 134–147, 2012.

[Cornelis 2008] Nico Cornelis, Bastian Leibe, Kurt Cornelis and Luc Van Gool. *3d urban scene modeling integrating recognition and reconstruction.* International Journal of Computer Vision, vol. 78, no. 2-3, pages 121–141, 2008.

[Crandall 2011] David Crandall, Andrew Owens, Noah Snavely and Dan Huttenlocher. *Discrete-continuous optimization for large-scale structure from motion.* In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 3001–3008, 2011.

[Esteban 2010] Isaac Esteban, Leo Dorst and Judith Dijk. *Closed form solution for the scale ambiguity problem in monocular visual odometry.* In International Conference on Intelligent Robotics and Applications, pages 665–679, Berlin, Heidelberg, 2010.

[Favaro 2008] Paolo Favaro, Stefano Soatto, Martin Burger and Stanley J Osher. *Shape from defocus via diffusion.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 30, no. 3, pages 518–531, 2008.

[Felzenszwalb 2004] P.F. Felzenszwalb and D.P. Huttenlocher. *Efficient graph-based image segmentation.* IJCV, vol. 59, no. 2, pages 167–181, 2004.

[Forsyth 2002] David A Forsyth and Jean Ponce. Computer vision: a modern approach. Prentice Hall Professional Technical Reference, 2002.

[Frahm 2010] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik*et al. Building Rome on a cloudless day*. In Computer Vision–ECCV 2010, pages 368–381. Springer, 2010.

[Früh 2004] Christian Früh and Avideh Zakhor. *An automated method for large-scale, ground-based city model acquisition*. International Journal of Computer Vision, vol. 60, no. 1, pages 5–24, 2004.

[Furukawa 2010] Yasutaka Furukawa and Jean Ponce. *Accurate, dense, and robust multiview stereopsis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 8, pages 1362–1376, 2010.

[Gallup 2010] David Gallup, J-M Frahm and Marc Pollefeys. *Piecewise planar and non-planar stereo for urban scene reconstruction*. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 1418–1425, 2010.

[Geiger 2011] Andreas Geiger, Julius Ziegler and Christoph Stiller. *Stereoscan: Dense 3d reconstruction in real-time*. In IEEE Intelligent Vehicles Symposium, pages 963–968, 2011.

[Geiger 2012] A. Geiger, P. Lenz and R. Urtasun. *Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite*. In IEEE Conference on Computer Vision and Pattern Recognition, pages 3354–3361, Providence, USA, 2012.

[Hartley 2004] R. I. Hartley and A. Zisserman. Multiple view geometry in computer vision. Cambridge University Press, second édition, 2004.

[He 2010] Xuming He and Alan Yuille. *Occlusion boundary detection using pseudo-depth*. In ECCV, pages 539–552. Springer, 2010.

[Hiep 2009] Vu Hoang Hiep, Renaud Keriven, Patrick Labatut and J-P Pons. *Towards high-resolution large-scale multi-view stereo*. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 1430–1437. IEEE, 2009.

[Hirschmuller 2007] H. Hirschmuller and D. Scharstein. *Evaluation of cost functions for stereo matching*. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, pages 1–8. IEEE, 2007.

[Hoiem 2005] D. Hoiem, A.A. Efros and M. Hebert. *Automatic photo pop-up*. ACM ToG, vol. 24, no. 3, pages 577–584, 2005.

[Hoiem 2006] D. Hoiem, A.A. Efros and M. Hebert. *Putting objects in perspective*. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, volume 2, pages 2137–2144. IEEE, 2006.

[Hoiem 2007] D. Hoiem, A.A. Efros and M. Hebert. *Recovering surface layout from an image*. IJCV, vol. 75, no. 1, pages 151–172, 2007.

[Huang 2010] Albert S Huang, Matthew Antone, Edwin Olson, Luke Fletcher, David Moore, Seth Teller and John Leonard. *A high-rate, heterogeneous data set from the darpa urban challenge*. The International Journal of Robotics Research, vol. 29, no. 13, pages 1595–1601, 2010.

[Humayun 2011] A. Humayun, O. Mac Aodha and G.J. Brostow. *Learning to find occlusion regions*. In CVPR'11, IEEE Conference on, pages 2161–2168. IEEE, 2011.

[Jebari 2012] Islem Jebari and David Filliat. *Color and Depth-Based Superpixels for Background and Object Segmentation*. Procedia Engineering, vol. 41, pages 1307–1315, 2012.

[Kazhdan 2006] Michael Kazhdan, Matthew Bolitho and Hugues Hoppe. *Poisson surface reconstruction*. In Proceedings of the fourth Eurographics symposium on Geometry processing, 2006.

[Leordeanu 2012] Marius Leordeanu, Rahul Sukthankar and Cristian Sminchisescu. *Efficient closed-form solution to generalized boundary detection*. European Conference on Computer Vision, pages 516–529, 2012.

[Levinshtein 2009] Alex Levinshtein, Adrian Stere, Kiriakos N Kutulakos, David J Fleet, Sven J Dickinson and Kaleem Siddiqi. *Turbopixels: Fast superpixels using geometric flows*. PAMI, vol. 31, no. 12, pages 2290–2297, 2009.

[Lhuillier 2013] Maxime Lhuillier and Shuda Yu. *Manifold surface reconstruction of an environment from sparse Structure-from-Motion data.* Computer Vision and Image Understanding, vol. 117, no. 11, pages 1628–1644, 2013.

[Li 2008] P. Li, Gunnewiek R.K. and P.H.N. de With. *Scene reconstruction using MRF optimization with image content adaptive energy functions.* In Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems, ACIVS'08, pages 872–882, 2008.

[Lindeberg 1993] T. Lindeberg and J. Garding. *Shape from texture from a multi-scale perspective.* In Computer Vision. Fourth International Conference on, pages 683–691. IEEE, 1993.

[Liu 2010] B. Liu, S. Gould and D. Koller. *Single image depth estimation from predicted semantic labels.* In CVPR, IEEE Conference on, pages 1253–1260. IEEE, 2010.

[Lourakis 2009] Manolis IA Lourakis and Antonis A Argyros. *SBA: A software package for generic sparse bundle adjustment.* ACM Transactions on Mathematical Software, vol. 36, no. 1, page 2, 2009.

[Lowe 2004] D.G. Lowe. *Distinctive image features from scale-invariant keypoints.* International Journal of Computer Vision, vol. 60, no. 2, pages 91–110, 2004.

[Ma 2004] Yi Ma. An invitation to 3-d vision: from images to geometric models, volume 26. springer, 2004.

[Martin 2001] D. Martin, C. Fowlkes, D. Tal and J. Malik. *A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics.* In Proc. 8th Int'l Conf. Computer Vision, volume 2, pages 416–423, July 2001.

[Marton 2009] Zoltan Csaba Marton, Radu Bogdan Rusu and Michael Beetz. *On Fast Surface Reconstruction Methods for Large and Noisy Datasets.* In IEEE International Conference on Robotics and Automation, pages 3218–3223, Kobe, Japan, May 12-17 2009.

[McCall 2006] Joel C McCall and Mohan M Trivedi. *Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation.* Intelligent Transportation Systems, IEEE Transactions on, vol. 7, no. 1, pages 20–37, 2006.

[Meister 2012] S. Meister, B. Jähne and D. Kondermann. *Outdoor stereo camera system for the generation of real-world benchmark data sets.* Optical Engineering, vol. 51, no. 02, page 021107, 2012.

[Meyer 1994] Fernand Meyer. *Topographic distance and watershed lines.* Signal processing, vol. 38, no. 1, pages 113–125, 1994.

[Micusik 2009] Branislav Micusik and Jana Kosecka. *Piecewise planar city 3D modeling from street view panoramic sequences.* In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pages 2906–2912. IEEE, 2009.

[Mičušík 2010] Branislav Mičušík and Jana Košecká. *Multi-view superpixel stereo in urban environments.* International journal of computer vision, vol. 89, no. 1, pages 106–119, 2010.

[Musialski 2013] Przemyslaw Musialski, Peter Wonka, Daniel G Aliaga, Michael Wimmer, L Gool and Werner Purgathofer. *A survey of urban reconstruction.* In Computer Graphics Forum. Wiley Online Library, 2013.

[Nawaf 2012] Mohamad Motasem Nawaf and Alain Trémeau. *Joint spatio-temporal depth features fusion framework for 3d structure estimation in urban environment.* In Computer Vision–ECCV 2012. Workshops and Demonstrations, pages 526–535. Springer, 2012.

[Nawaf 2013] Mohamad Motasem Nawaf and Alain Trémeau. *Fusion of dense spatial features and sparse temporal features for three-dimensional structure estimation in urban scenes.* IET Computer Vision, vol. 7, no. 5, pages 302–310, 2013.

[Nawaf 2014a] Mohamad Motasem Nawaf, Hasnat Md Abul, Desiré Sidibé and Alain Trémeau. *Color and Flow Based Superpixels for 3D Geometry Respecting Meshing.* In Proceedings of the 2014 IEEE Winter Conference on Applications of Computer Vision. IEEE Computer Society, 2014.

[Nawaf 2014b] Mohamad Motasem Nawaf and Alain Trémeau. *Monocular 3D Structure Estimation for Urban Scenes.* In IEEE International Conference on Image Processing (ICIP), pages 526–535. Springer, 2014.

[Pandey 2011] Gaurav Pandey, James R. McBride and Ryan M. Eustice. *Ford campus vision and lidar data set.* International Journal of Robotics Research, vol. 30, no. 13, pages 1543–1552, November 2011.

[Pfeiffer 2012] D. Pfeiffer, F. Erbs and U. Franke. *Pixels, Stixels, and Objects.* In Computer Vision–ECCV 2012. Workshops and Demonstrations, pages 1–10. Springer, 2012.

[Pollefeys 2004] Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops and Reinhard Koch. *Visual modeling with a hand-held camera.* International Journal of Computer Vision, vol. 59, no. 3, pages 207–232, 2004.

[Pollefeys 2008] Marc Pollefeys, David Nistér, J-M Frahm, Amir Akbarzadeh, Philippos Mordohai, Brian Clipp, Chris Engels, David Gallup, S-J Kim, Paul Merrell*et al. Detailed real-time urban 3d reconstruction from video.* International Journal of Computer Vision, vol. 78, no. 2-3, pages 143–167, 2008.

[Raguram 2008] R. Raguram, J.M. Frahm and M. Pollefeys. *A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus.* Computer Vision–ECCV 2008, pages 500–513, 2008.

[Ros 2015] G. Ros, S. Ramos, M. Granados, A. Bakhtiary, D. Vazquez and A.M. Lopez. *Vision-based Offline-Online Perception Paradigm for Autonomous Driving.* In WACV, 2015.

[Rublee 2011] Ethan Rublee, Vincent Rabaud, Kurt Konolige and Gary Bradski. *ORB: an efficient alternative to SIFT or SURF.* In International Conference on Computer Vision, pages 2564–2571, 2011.

[Savarese 2007] Silvio Savarese, Marco Andreetto, Holly Rushmeier, Fausto Bernardini and Pietro Perona. *3D reconstruction by shadow carving: Theory and practical evaluation.* International journal of computer vision, vol. 71, no. 3, pages 305–336, 2007.

[Saxena 2007] A. Saxena, S.H. Chung and A.Y. Ng. *Make3D Range Image Data.* Website, 2007. http://make3d.cs.cornell.edu/data.html.

[Saxena 2009a] A. Saxena. *Monocular depth perception and robotic grasping of novel objects*. PhD thesis, Stanford University, 2009.

[Saxena 2009b] A. Saxena, M. Sun and A.Y. Ng. *Make3d: Learning 3d scene structure from a single still image*. PAMI, vol. 31, no. 5, pages 824–840, 2009.

[Sengupta 2013] Sunando Sengupta, Eric Greveson, Ali Shahrokni and Philip HS Torr. *Urban 3d semantic modelling using stereo vision*. In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 580–585. IEEE, 2013.

[Shu 2013] Guang Shu, Afshin Dehghan and Mubarak Shah. *Improving an Object Detector and Extracting Regions Using Superpixels*. In CVPR, pages 3721–3727, 2013.

[Silberman 2012] Nathan Silberman, Derek Hoiem, Pushmeet Kohli and Rob Fergus. *Indoor segmentation and support inference from RGBD images*. In ECCV, pages 746–760. Springer, 2012.

[Smith 2009] M. Smith, I. Baldwin, W. Churchill, R. Paul and P. Newman. *The new college vision and laser data set*. The International Journal of Robotics Research, vol. 28, no. 5, pages 595–599, 2009.

[Snavely 2006] N. Snavely, S.M. Seitz and R. Szeliski. *Photo tourism: exploring photo collections in 3D*. In ACM Transactions on Graphics, volume 25, pages 835–846, 2006.

[Sohn 2007] Gunho Sohn and Ian Dowman. *Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction*. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 62, no. 1, pages 43–63, 2007.

[Sturgess 2009] Paul Sturgess, Karteek Alahari, Lubor Ladicky and Philip Torr. *Combining appearance and structure from motion features for road scene understanding*. In Proceedings of the British Machine Vision Conference, BMVC'09, pages 1226–1238, 2009.

[Sun 2010a] D. Sun, Roth S. and M. J. Black. *Dense Optical Flow Code*, 2010. http://www.cs.brown.edu/people/dqsun/.

[Sun 2010b] Deqing Sun, Stefan Roth and Michael J Black. *Secrets of optical flow estimation and their principles*. In IEEE Conference on Computer Vision and Pattern Recognition, pages 2432–2439, 2010.

[Sun 2014] Deqing Sun, Ce Liu and Hanspeter Pfister. *Local Layering for Joint Motion Estimation and Occlusion Detection*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), volume 2, pages 399–406, 2014.

[Szeliski 2011] Richard Szeliski. Computer vision: algorithms and applications. Springer, 2011.

[Torralba 2002] A. Torralba and A. Oliva. *Depth estimation from image structure*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, no. 9, pages 1226–1238, 2002.

[Triggs 2000] B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon. *Bundle adjustment: a modern synthesis*. Vision algorithms: theory and practice, pages 153–177, 2000.

[Van den Bergh 2012a] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani and Luc Van Gool. *SEEDS: superpixels extracted via energy-driven sampling*. In ECCV, pages 13–26. Springer, 2012.

[Van den Bergh 2012b] Michael Van den Bergh and Luc Van Gool. *Real-time stereo and flow-based video segmentation with superpixels*. In IEEE Workshop on the Applications of Computer, pages 89–96, 2012.

[Vanegas 2010] Carlos A Vanegas, Daniel G Aliaga and Bedrich Benes. *Building reconstruction using manhattan-world grammars*. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 358–365, 2010.

[Vedaldi 2007] A. Vedaldi, G. Guidi and S. Soatto. *Moving forward in structure from motion*. In Computer Vision and Pattern Recognition, CVPR'07. IEEE Conference on, pages 1–7. IEEE, 2007.

[Vedaldi 2010] Andrea Vedaldi and Brian Fulkerson. *VLFeat: An open and portable library of computer vision algorithms*. In Proceedings of the international conference on Multimedia, pages 1469–1472. ACM, 2010.

[Vergauwen 2006] Maarten Vergauwen and Luc Van Gool. *Web-based 3d reconstruction service*. Machine vision and applications, vol. 17, no. 6, pages 411–426, 2006.

[Vogel 2013] Christoph Vogel, Konrad Schindler and Stefan Roth. *Piecewise Rigid Scene Flow*. In Computer Vision (ICCV), 2013 IEEE International Conference on, pages 1377–1384, 2013.

[Vogiatzis 2007] George Vogiatzis, Carlos Hernández, Philip HS Torr and Roberto Cipolla. *Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 12, pages 2241–2246, 2007.

[Wang 2009] J. Wang, F. Dong, T. Takegami, E. Go and K. Hirota. *A 3D Pseudo-Reconstruction from Single ImageBased on Vanishing Point*. Journal ref: Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 13, no. 4, pages 393–399, 2009.

[Wu 2011] Changchang Wu, Sameer Agarwal, Brian Curless and Steven M Seitz. *Multicore bundle adjustment*. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 3057–3064, 2011.

[Wu 2012] Changchang Wu, Sameer Agarwal, Brian Curless and Steven M Seitz. *Schematic surface reconstruction*. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 1498–1505. IEEE, 2012.

[Xiao 2009] Jianxiong Xiao, Tian Fang, Peng Zhao, Maxime Lhuillier and Long Quan. *Image-based street-side city modeling*. ACM Transactions on Graphics (TOG), vol. 28, no. 5, page 114, 2009.

[Yamaguchi 2012] Koichiro Yamaguchi, Tamir Hazan, David McAllester and Raquel Urtasun. *Continuous markov random fields for robust stereo estimation*. In European Conference on Computer Vision, pages 45–58. Springer, 2012.

[Yamaguchi 2013] Koichiro Yamaguchi, David McAllester and Raquel Urtasun. *Robust Monocular Epipolar Flow Estimation*. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR '13, pages 1862–1869, Washington, DC, USA, 2013. IEEE Computer Society.

[Zebedin 2008] Lukas Zebedin, Joachim Bauer, Konrad Karner and Horst Bischof. *Fusion of feature-and area-based information for urban buildings modeling from aerial imagery.* In Computer Vision–ECCV 2008, pages 873–886. Springer, 2008.

[Zhou 2013] Qian-Yi Zhou and Ulrich Neumann. *Complete residential urban area reconstruction from dense aerial LiDAR point clouds.* Graphical Models, vol. 75, no. 3, pages 118–125, 2013.