Exploiting Semantic Information for Accurate Segmentation, Localization in Dynamic Environments

Thesis submitted in partial fulfillment
of the requirements for the degree of

MS by Research
in
Computer Science and Engineering

by

N Dinesh Reddy
201307689
dineshreddy.n@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
December 2015

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Exploiting Semantic Information for Accurate Segmentation, Localization in Dynamic Environments" by N Dinesh Reddy, has been carried out under my supervision and is not submitted elsewhere for a degree.

_____ _____

Date Adviser: Prof. K Madhava Krishna

To All and None.

Acknowledgments

First and foremost I would like to thank Dr.Madhava Krishna for giving me an opportunity to joining his lab and for his support and guidance through out my thesis. Many thanks to all my lab mates for their valuable suggestions and discussions. I would also thank the institute and the lab for creating such a wonderful research environment and giving me freedom to work on projects. And above all I would like to thank my family and all my friends who have stood with me and have served as a source of motivation and moral support.I would also like to thank Dr. Visesh Chari and Prateek Singhal for thier valuable inputs during the course of the thesis. I specifically thank Sarthak, Arun, Bharath, Siva Karthik, Sudhashu, Tourani, Akhil, Falak, Harit, Sheetal for their support during my stay at the lab.

Abstract

Image based reconstruction of urban environments is a challenging problem that deals with optimization of large number of variables, and has several sources of errors like the presence of dynamic objects. Since most large scale approaches make the assumption of observing static scenes, dynamic objects are relegated to the noise modelling section of such systems. This is an approach of convenience since the RANSAC based framework used to compute most multiview geometric quantities for static scenes naturally confine dynamic objects to the class of outlier measurements. However, reconstructing dynamic objects along with the static environment helps us get a complete picture of an urban environment. Such understanding can then be used for important robotic tasks like path planning for autonomous navigation, obstacle tracking and avoidance, and other areas.

While the literature has been fairly dense in the areas of scene understanding and semantic labeling there have been few works that make use of motion cues to embellish semantic performance and vice versa. we address the problem of semantic motion segmentation, and show how semantic and motion priors augments performance. We propose an algorithm that jointly infers the semantic class and motion labels of an object. Integrating semantic, geometric and optical flow based constraints into a dense CRF-model we infer both the object class as well as motion class, for each pixel. We found improvement in performance using a fully connected CRF as compared to a standard clique-based CRFs. For inference, we use a Mean Field approximation based algorithm.

We also propose a system for robust SLAM that works in both static and dynamic environments. To overcome the challenge of dynamic objects in the scene, we propose a new model to incorporate *semantic constraints* into the reconstruction algorithm. While some of these constraints are based on multi-layered dense CRFs trained over appearance *as well as motion cues*, other proposed constraints can be expressed as additional terms in the bundle adjustment optimization process that does iterative refinement of 3D structure and camera / object motion trajectories. Our method outperforms recently proposed motion detection algorithms and also improves the semantic labeling compared to the state-of-the-art Automatic Labeling Environment algorithm on the challenging KITTI dataset especially for object classes such as pedestrians and cars that are critical to an outdoor robotic navigation scenario. We show results for accuracy of motion segmentation and reconstruction of the trajectory and shape of moving objects relative to ground truth. We are able to show average relative error reduction by 41

% for moving object trajectory reconstruction relative to state-of-the-art methods like TriTrack[23], as well as on standard bundle adjustment algorithms with motion segmentation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Vision based SLAM (vSLAM) is becoming an widely researched problem, partly because of its ability to produce good quality reconstructions with affordable hardware, and partly because of increasing computational power that results in computational affordability of huge optimization problems. While vSLAM systems are maturing and getting progressively complicated, the two main components remain camera localization (or camera pose estimation) and 3D reconstruction. Generally, these two components precede an optimization based joint refinement of both camera pose and 3D structure, called bundle adjustment.

In urban environments, vSLAM is challenging particularly because of the presence of *dynamic objects*. Indeed, it is difficult to capture videos of a city without observing moving objects like cars or people. However, dynamic objects are a source of error in vSLAM systems, since the basic components of such algorithms make the fundamental assumption that the world being observed is static. While optimization algorithms are designed to handle random noise in observations, dynamic objects are a source of *structured* noise since they do not conform to models of random noise distributions (like Gaussian distributions, for example). To overcome such difficulties, RANSAC based procedures for camera pose estimation and 3D reconstruction have been developed in the past, which treat dynamic objects as outliers and remove them from the reconstruction process.

Using object class and motion cues jointly provides for an enhanced understanding of the scene. We perceive better when we describe the scene in terms of a moving or stationary car (pedestrian) than in terms of presence of only few object classes. Motivated by this fact, we have formulated the problem of object class segmentation, which assigns an object label such as road or building to every pixel in the image, and motion segmentation, in which every pixel within the image is labelled as moving or stationary, jointly. Semantic motion segmentation has its application in robotics where an autonomous system will be in a better position to plan its path based on the joint knowledge. While successful attempts have been made to isolate and discard dynamic objects from such reconstruction processes, there are several recent applications that *benefit* from *reconstructions* of such objects. For example, reconstructing dynamic urban traffic scenes are useful since traffic patterns can be studied to produce autonomous vehicles that can better navigate such situations. Reconstructing dynamic objects

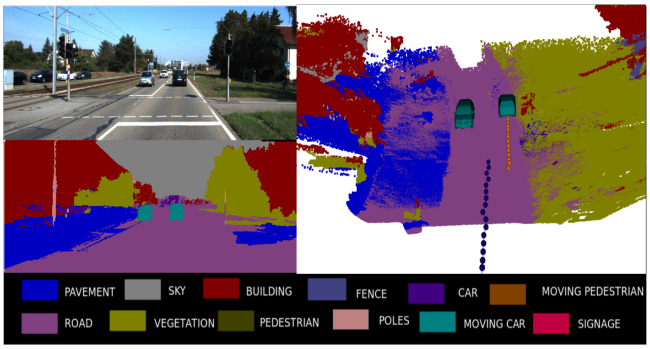

Figure 1.1: **Overview of our approach**: **Top left** A frame from highway sequence of the KITTI dataset. **Bottom left** Semantic Motion Segmentation to provide result. **Right** 3d reconstruction with overlaid semantic map and trajectories of the moving objects and camera. (Best viewed in color)

are also useful in indoor environments when robots need to identify and avoid moving obstacles in their path [42].

Reconstructing dynamic objects in videos present several challenges. Firstly, moving objects in images and videos have to be segmented and isolated, before they can be reconstructed. This in itself is a challenging problem in the presence of image noise and scene clutter. Degeneracies in camera motion also prevent accurate motion segmentation of such objects. Secondly, upon isolation, a separate vSLAM procedure must be initialized for *each* moving object, since objects like cars often move independent of each other and thus have to be treated as such. Often moving objects like cars occupy only a small portion of the image space in a video (Figure 1.1), because of which dense reconstructions are infeasible since getting long accurate feature correspondence tracks for such objects is difficult. Absence of large number of feature correspondences also hinders accurate estimation of the car's pose with respect to a world coordinate system. Finally, such objects cannot be reconstructed in isolation from the static scene, since optimization algorithms like bundle adjustment do not preserve contextual information like the fact that the car must move along a direction perpendicular to the normal of the road surface.

1.1 Related Work

Our system involves several components like semantic motion segmentation, dynamic body reconstruction using multi-body vSLAM, and trajectory optimization. Table 1.1 compares components of our approach with works in recent literature. In recent literature, TriTrack [23] is the closest approach to our method and we first explain it in detail as we compare our method to it in chapter 2.

TriTrack [23] is an approach for scene reconstruction, when a moving camera is observing a dynamic scene. It proceeds by first isolating and reconstrucing the trajectory of the camera using an odometry algorithm called VISO2 [9], with dense feature matching and stereo computation as key components. The computed camera motion is then passed over to a sparse scene flow segmentation algorithm to do motion segmentation in 3D, followed by independent trajectory optimization of the seg-

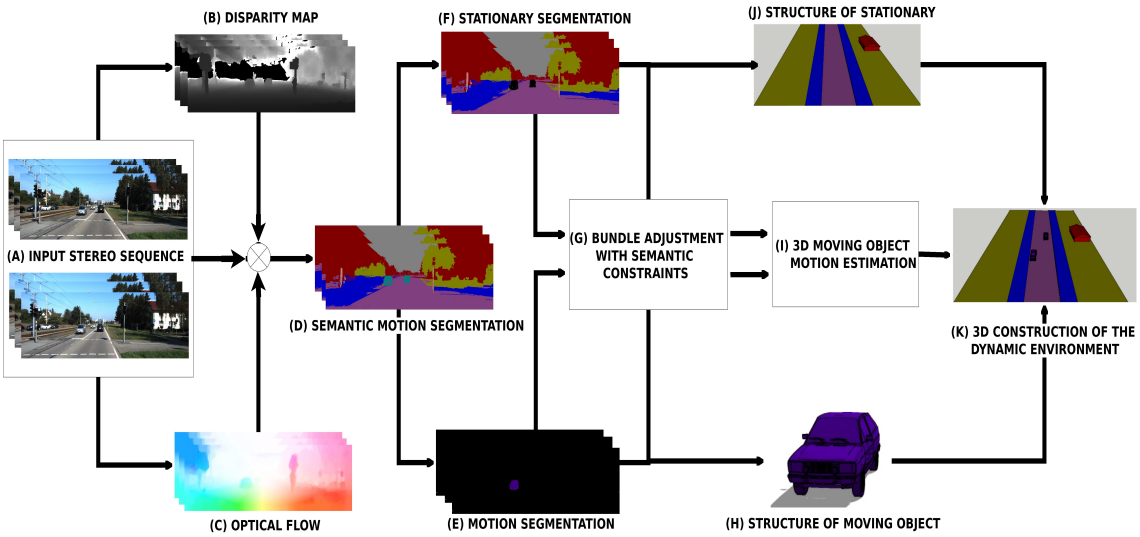

Figure 1.2: **Illustration of the proposed method**. The system takes a sequence of rectified stereo images (A). Our formulation computes the semantic motion segmentation (D) using the depth(B) and optical flow(C) information. We segment the moving objects (E) from the stationary background (F). We compute accurate structure of the static background (J) and the moving object (H) with the help of bundle adjustment (G). This leads to state-of-the-art 3d reconstruction of the dynamic environment(K) with the help of moving object trajectory estimation(I). (Best viewed in color.)

mented moving objects VISO2. We have extensively compared our results to TriTrack, as its a very good baseline for moving object localization and has been extensively tested in dynamic environments. Our approach improves over both motion segmentation and trajectory optimization using semantic constraints. We now focus on each one of the components of our algorithm and draw references to relevant works in the literature in this section.

There has been fairly large amount of literature in both Semantic and Motion segmentation. For semantic image segmentation, existing approaches use textonboost [32], in which weakly predictive features in a image like color, location and texton features are passed through a classifier to give the cost of a label for that particular pixel. These costs are combined in a contrast sensitive Conditional Random field [22] [16]. Most of the mid-level inferences do not use pixels directly, but segment the image into regions [14][15] [10] [19]. Substantial state of the art results for dense semantic image segmentation have been show using superpixel based hierarchical framework [20][36]. Recently a lot of the scene understanding research has gone into better understanding the scene using different parameters to get a better segmentation. In [21],[20],[36] Ladicky et al have used Object detectors, Co-occurrence statistics and Stereo disparity for improving the semantics. In [44], Yao et al have combined semantic segmentation, object detection and scene classification for understanding a scene as a whole.

Motion segmentation has been approached using geometric priors mostly from a video. General paradigm involves using Geometric constraints [25] or reducing the model to affine to cluster the trajectories into subspaces [7]. These methods have been shown not to work in complex environments

where the moving cars lie in the same subspace. We consider deviation of the trajectories based on the 3d motion of the camera estimated from the trajectories to provide us motion likelihood even in challenging scenarios. Semantics for motion detection is not new, Wedel et al [40] have used scene flow to segment motion in a stereo camera. There has been numerous work [25] [23][29] on segmenting moving object by compensating for the platform movement. Recently in [2] motion features have been learnt using deep learning to give better motion likelihood estimate. The applications for understanding motion semantics has been an emerging area and can be used to understand and model traffic pattern [46] [8].

Semantics have been used extensively for reconstruction [31] [37] [11] but haven't been exploited in motion segmentation till recently [27]. Generally, motion segmentation has been approached using geometric constraints [18] or by using affine trajectory clustering into subspaces [7]. In our approach we use motion *along* with semantic cues to segment the scene into static and dynamic objects, which allows us to work with fast moving cars, occlusions and disparity failure. We show a typical result of the motion segmentation algorithm in (Figure 1.1)(bottom left) where each variable is labelled for both multi-variate semantic class and binary motion class.

Dynamic body reconstruction is a relatively new development in 3D reconstruction with sparse literature on it. The few solutions in the literature can be categorized into decoupled and joint approaches. Joint approaches like [30] [33] use monocular cameras to jointly estimate the depth maps, do motion segmentation and motion estimation of multiple bodies. Decoupled approaches like [45] [18] [28] have a sequential pipeline where they segment motion and independently reconstruct the moving and static scenes. Our approach is a decoupled approach but essentially differs from other approaches, as we use a novel algorithm for semantic motion segmentation which is leveraged to obtain accurate localization of the moving objects through smoothness and planarity constraints to give an accurate dynamic semantic map.

Recent approaches to 3D reconstruction have either used semantic information in a qualitative manner [31], or have only proposed to reconstruct indoor scenes using such information [42] . Only Yuan et al. [45] propose to add semantic constraints for reconstruction. While our approach is similar to theirs, they use strict constraints for motion segmentation without regard to appearance information whereas our approach works for more general scenarios as it employs a more powerful inference engine in the CRF. We show a improved analysis of the algorithm compared to [26] for the localization of moving objects. We have evaluated the Semantic motion segmentation [27] on multiple new sequences of the KITTI dataset.

1.2 System Overview

We give an illustration of our system in Figure 1.2. Given rectified input images from a stereo camera, we first compute low level features like SIFT descriptors, optical flow (using DeepFlow [41]) and stereo [43]. These are then used to compute semantic motion segmentation, as explained in Chapter

Method	Outdoor	Stereo	MS	SR	MR	SBA
Sengupta *et al.*[31]	✓	✓		✓		
Hane *et al.*[11]	✓	✓		✓		
Jianxiong *et al.*[42]				✓		
kundu *et al.*[18]	✓		✓			
valentin *et al.*[37]	✓	✓		✓		
Vineet *et al.*[39]	✓	✓		✓	✓	
TriTrack [23]	✓	✓	✓		✓	
OURS	✓	✓	✓	✓	✓	✓

Table 1.1: Comparison with related work. MS=Motion Segmentation, SR=Semantic Reconstruction, MR=Motion Reconstruction, SBA=Semantic Bundle Adjustment

2. Once semantic segmentation is done per image, we isolate stationary objects from moving objects and reconstruct them independently. To do this, we connect moving objects across frames into tracks by computing SIFT matches on dense SIFT features [38].

Then we perform camera resectioning using EPnP [24] for stationary and ICP for moving objects, to register their 3D points across frames. This is then followed by bundle adjustment with semantic constraints (chapter 3), where we make use of the semantic and motion labels assigned to the segmented scene to obtain accurate 3D reconstruction. We then fuse the stationary and moving object reconstructions using an algorithm based on the truncated signed distance function (TSDF) [48]. Finally, we transfer labels from 2D images to 3D data by projecting 3D data onto the images, and using a winner-takes-it-all approach to assign labels to 3D data from the labels of the projected points.

1.3 Contribution

We present an end-to-end system that takes a video, segments the scene into static and dynamic components and reconstructs *both* static and dynamic objects separately. Additionally, while reconstructing the dynamic object, we impose several novel constraints into the bundle adjustment refinement that deal with noisy feature correspondences, erroneous object pose estimation, and contextual information. To be precise, we propose the following contributions in this paper

- We use a *new semantic motion segmentation* algorithm using multi-layer dense CRF which provides state-of-the-art motion segmentation and object class labelling.

- For the first time to our knowledge, we incorporate *semantic contextual information* like support relations between the road surface and object motion, which helps better localize the moving object's pose vis-a-vis the world coordinate system, and also helps in reconstructing them.

- We describe a *novel random sampling strategy* that enables us to maintain the feasibility of the optimization problem in spite of the addition of a large number of variables. Using this approach

we drastically reduce the size of our optimization problem *without* compromising on resultant accuracy.

We evaluate our system on 4 challenging KITTI Urban tracking datasets captured using a stereo camera. we get an improvement of 13.89 % relative to traditional bundle adjustment after using our novel semantic motion segmentation.

1.4 Organization of thesis

This Thesis is organized as follows. We cover introduction and related work in Chapter 1. We describe process of motion segmentation using object class semantic constraints in chapter 2. We track and initialize multiple moving bodies which we then optimize using a novel bundle adjustment in Chapter 3. Finally we show experimental results on challenging datasets in Chapter 4, whereas the future work and conclusion are mentioned in chapter 5.

Chapter 2

Semantic Motion Segmentation

2.1 Introduction

Using semantic contextual information helps in a better understanding of a scene. To exploit this information, we propose a method to model the whole image scene using a fully connected multi-label Conditional Random Field(CRF) with joint learning and inference. The problem of semantic scene understanding and motion segmentation can be solved jointly. As accurate labeling of object class can improve the inference of the motion labels for the corresponding pixels and motion information in a image improves the object labelling. Moreover, the class of the object provides a very important clue for motion analysis. For example, in a scene we can assume that the probability of a car or person moving is greater than the probability of a moving wall or a moving road. Further jointly solving these problems improves the overall accuracy of the classification problem by giving accurate boundaries. To provide some intuition behind this statement, note that the object class boundaries are more likely to occur at a sudden transition in motion and vice-versa.

We use sequential stereo pairs from three time instants to label the scene and estimate the motion, showing the robustness in our motion segmentation method. The interaction between the semantic labelling and motion likelihood is learnt, which helps us to efficiently segment the distant moving objects in few time instants. Each image pixel is labelled with both an object class and motion estimate. Various approximate methods for inference exist, such as maximum a posteriori methods (e.g graph-cuts), or variational methods, such as mean-field approximation, which allow us to approximately estimate a maximum posterior marginals solution (MPM). We have implemented mean field based inference algorithm proposed by [17] as it enables us to utilize efficient approximations for high-dimensional filtering, which reduce the complexity of message passing from quadratic to linear, resulting in inference that is linear in the number of variables and sub-linear in the number of edges.

Herein we show that joint labeling formulation is mutually beneficial for motion as well as semantic labeling. Our method is similar to [36] [47] [34] where they have used a multi layer multi-label CRF for joint estimation of scene reconstruction and attributes respectively. Specifically we show significant performance gain for motion labeling in the challenging KITTI street datasets in comparison to the state

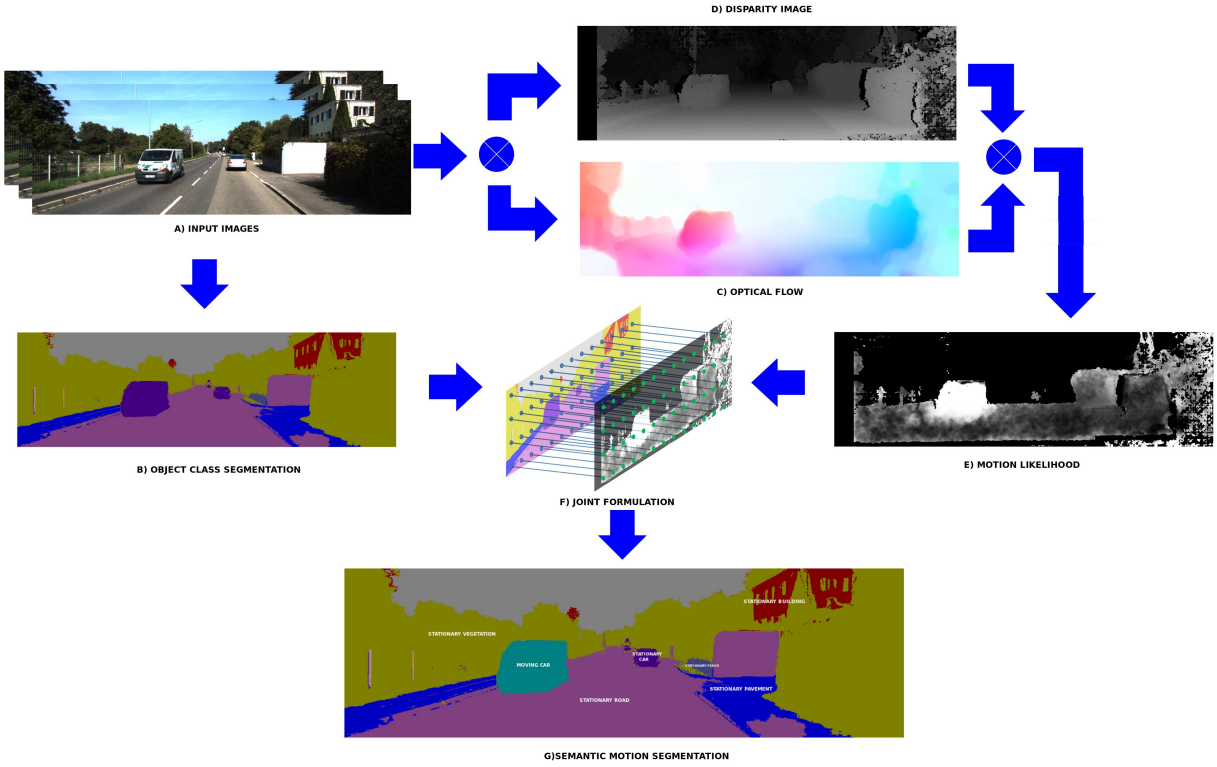

Figure 2.1: Illustration of the proposed semantic motion segmentation method .The system takes a sequence of rectified stereo images from the tracking dataset of KITTI (A). Our formulation computes the Object class probabilities (B) and motion likelihood (E) using disparity map(D) and optical flow(C).These are input into a joint formulation which exploits the object class and motion co-dependencies by allowing a interact between them (F).The inference is computed using the mean field approximation method to give a joint label to each pixel(G). Best viewed in color.

of the art methods in motion segmentation [25]. Concurrently we also improve the performance of ALE [19] by showing segmentation results closer to ground truth especially for pedestrians and cars. We accomplish this using motion likelihood estimates and incorporate semantics to get a better holistic understanding of the dynamic scene. These results show that this approach closely mimics perception by humans where semantic labels play an important role to identify motion.

In this section, we deal with the first module of our system. A sample result of our segmentation algorithm is shown in Figure 2.1. With input images from a stereo camera, we give an overview on how we perform semantic segmentation [19] to first separate dynamic objects from the static scene. We combine classical semantic segmentation with a new set of motion constraints proposed in [27] to perform semantic motion segmentation, that *jointly* optimizes for semantic and motion segmentation.

2.2 Problem Formulation

Our joint optimisation consists of two parts, object class segmentation and motion segmentation. We introduce the terms to be used in this chapter. We define a dense CRF where the set of random variables $Z = \{Z_1, Z_2, ..., Z_N\}$ corresponds to the set of all image pixels $i \in \mathcal{V} = \{1, 2, ..., N\}$. Let $\mathcal{N}$ be the neighbourhood system of the random field defined by the sets $\mathcal{N}_i \forall i \in \mathcal{V}$, where $\mathcal{N}_i$ denotes the neighbours of the variable Z_i as shown in left image of Figure 2.2. Any possible assignment of labels to the random variables will be called as labelling and denoted by z.

2.2.1 Dense Multi-class CRF

We formulate the problem of object class segmentation as finding a minimal cost labelling of a CRF defined over a set of random variables $X = \{x_i, x_2, ..., x_{\mathcal{N}}\}$ each taking a state from the label space $\mathcal{L} = \{l_1, l_2, ..., l_k\}$, where k represents the number of object class labels. Each label l indicates a different object class from 11 possible classes. These energies are:

$$E^O(x) = \sum_{i \in \mathcal{V}} \psi_i^O(x_i) + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \psi_{i,j}^O(x_i, x_j) \tag{2.1}$$

The unary potential $\psi_i^O(x_i)$ describes the cost of the pixel taking the corresponding label. The pairwise potential encourages similar pixels to have the same label. The unary potential term is computed for each pixel using pre-trained models of the color, texture and location features for each object [32] .In a typical graph topology, we consider a 4 or 8 neighbour connected network. With the mean field inference algorithm it is possible to use a fully connected graph, where all the pixels in the image are interconnected given certain forms of pairwise potential.Therefore, the pairwise potential takes the form of a potts model:

$$\psi_{i,j}^O(x_i, x_j) = \begin{cases} 0 & \text{if } x_i = x_j; \\ p(i,j) & \text{if } x_i \neq x_j. \end{cases} \tag{2.2}$$

For multi class image segmentation and labeling we use contrast-senstitive two-kernel potentials, defined in terms of the color vectors I_i and positions p_i

$$p(i,j) = \underbrace{exp(-\frac{|p_i - p_j|^2}{2\theta_\beta^2} - \frac{I_i - I_j}{2\theta_v^2})}_{appereance\ kernal} + \underbrace{exp(-\frac{|p_i - p_j|^2}{2\theta_p^2})}_{smoothness\ kernal} \tag{2.3}$$

Where p_i indicates the location of the ith pixel, I_i indicates the intensity of the i_{th} pixel, and θ_β,θ_p ,θ_v are the model parameters learned from the training data. the appearance kernel is inspired by the observation that nearby pixels with similar color are likely to be in the same class. the degrees of nearness and similarity are controlled by parameters θ_β and θ_v. The smoothness kernel removes small regions.

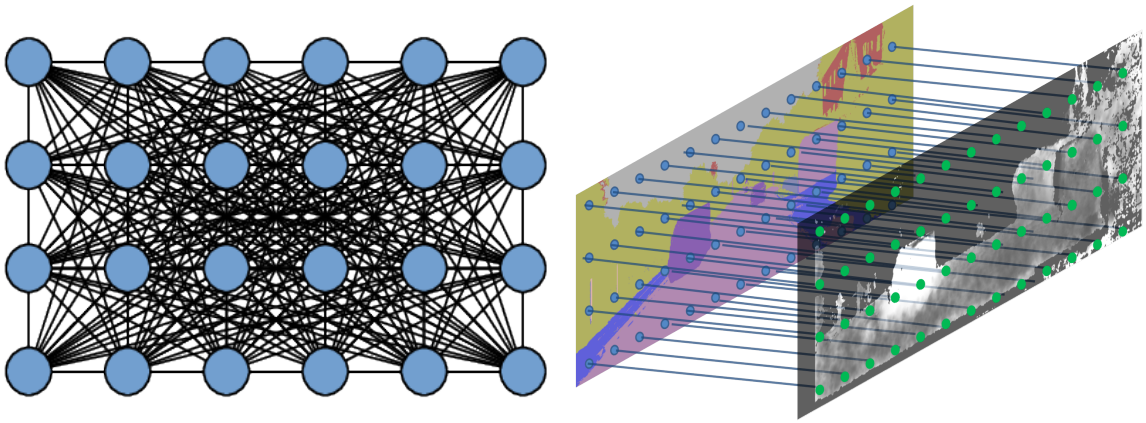

Figure 2.2: The left image depicts the Dense CRF model being used in the formulation and the right image depicts the joint CRF formulation of the object class and the motion class.

2.2.2 Dense Motion CRF

We use a standard dense CRF for formulating the semantic motion segmentation .The problem is posed as finding a minimal cost labelling of a CRF over a set of random variables $\mathcal{Y} = \{y_1, y_2, ..., y_{\mathcal{N}}\}$ which can take the label of moving or stationary i.e $\mathcal{M} = \{m_1, m_2\}$. where m_1 represents all the stationary pixels in the image and m_2 corresponds to the moving pixels. The formulation for motion is as follows:

$$E^{\mathcal{M}}(y) = \sum_{i \in \mathcal{V}} \psi_i^{\mathcal{M}}(y_i) + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \psi_{ij}^{\mathcal{M}}(y_i, y_j) \tag{2.4}$$

Where the unary potential $\psi_i^{\mathcal{M}}(x_i)$ is given by the motion likelihood of the pixel and is computed as the difference between the predicted flow and optical flow. The predicted flow is given by :

$$\hat{X}' = KRK'X + KT/z \tag{2.5}$$

where K is given as the Intrinsic camera matrix , R and T are the translation and rotation of the camera respectively, z is the depth of the pixel from camera. X is the location of the pixel in image coordinates and $\hat{X}'$ is the predicted flow vector of the pixel given from the motion of the camera. Thus unary potential is given as:

$$\psi_i^{\mathcal{M}}(x_i) = ((\hat{X}' - X')^T \Sigma^{-1} (\hat{X}' - X')) \tag{2.6}$$

Where Σ is called the covariance matrix which is the sum of covariance of optical flow and the covariance of measured optical flow. Here $\hat{X}' - X'$ represents the difference of the predicted flow and optical flow. The pairwise potential $\psi_{i,j}^{\mathcal{M}}(y_i, y_j)$ is given as the relationship between neighbouring pixels and encourages the adjacent pixels in the image to have similar flow. The cost of the function is defined

as:

$$\psi_{ij}^{\mathcal{M}}(y_i, y_j) = \begin{cases} 0 & \text{if } y_i = y_j; \\ g(i,j) & \text{if } y_i \neq y_j. \end{cases} \tag{2.7}$$

where $g(i, j)$ is a edge feature based on the difference between the flow of the neighbouring pixels:

$$g(i, j) = |f(y_i) - f(y_j)| \tag{2.8}$$

where f(.) is defined as the function which returns the flow vector from the optical flow of the corresponding pixel.

2.2.3 Joint CRF Formulation

In this section, we try to use object class segmentation and motion estimate to jointly estimate the label of the dynamic scene . Each random variable Z_i = [X_i,Y_i] takes a label z_i = [x_i,y_i], from the product space of object class and motion labels and correspond to the variable Z_i taking a object label x_i and motion y_i. In general the energy of the CRF for joint estimation is written as :

$$E^{\mathcal{J}}(z) = \sum_{i \in \mathcal{V}} \psi_i^{\mathcal{J}}(z_i) + \sum_{i \in \mathcal{V}, j \in \mathcal{N}_i} \psi_{i,j}^{\mathcal{J}}(z_i, z_j) \tag{2.9}$$

where $\psi_i^{\mathcal{J}}$, $\psi_{i,j}^{\mathcal{J}}$ are the sum of the previously mentioned terms ψ_i^{O} and $\psi_i^{\mathcal{M}}$, $\psi_{i,j}^{O}$ and $\psi_{i,j}^{\mathcal{M}}$ respectively. We include some extra terms which help in understanding the relation between the labels of X , Y . In the real world scenarios there is a relationship between the object class and corresponding motion likelihood for each pixel. We compute an interactive unary and pairwise potential terms so that a joint inference can be performed.

2.2.3.1 Joint unary potential

The unary potential $\psi_i^{\mathcal{J}}(z_i)$ can be defined as an interactive potential term which incorporates a relationship between the object class and the corresponding motion likelihood as shown in right image of Figure. 2.2. We can directly take the relationship between the object class and all the possible motion models as a measure to calculate the joint unary potential. As this requires large amount of training data to incorporate all motion models for each the class. We look at class and motion correlation function which incorporates the class-motion compatibility and can be expressed as :

$$\psi_{i,l,m}^{O\mathcal{M}}(x_i, y_j) = \lambda(l, m); \tag{2.10}$$

Here $\lambda(l, m) \in [-1, 1]$ is a learnt correlation term between the motion and object class label.The combined unary potential of the joint CRF is given as follows:

$$\psi_{i,l,m}^{\mathcal{J}}([x_i, y_i]) = \psi_i^{O}(x_i) + \psi_i^{\mathcal{M}}(y_i) + \psi_{i,l,m}^{O\mathcal{M}}(x_i, y_i) \tag{2.11}$$

where ψ_i^{O} and $\psi_i^{\mathcal{M}}$, are the unary potentials previously discussed for object class and motion likelihood of a pixel i given the image.

2.2.3.2 Joint pairwise potential

The joint pairwise potential $\psi_{ij}^{\mathcal{J}}(m_i, m_j)$ enforces the consistency of object class and motion between the neighbouring pixels. This potential term exploits the condition that, when there is a change in the motion layer, then there is a high chance for the label of the object class to change. Similarly, if there is a change in the label in the object class then it is more likely for the label in the motion layer to change. To include this behaviour in our formulation, we have taken the joint pairwise term as:

$$\psi_{ij}^{\mathcal{J}}([x_i, y_i], [x_j, y_j]) = \psi_{ij}^{O}(x_i, x_j) + \psi_{ij}^{\mathcal{M}}(y_i, y_j) \tag{2.12}$$

Here $\psi_{ij}^{O}(x_i, x_j)$ and $\psi_{ij}^{\mathcal{M}}(y_i, y_j)$ have been defined earlier as the pairwise terms of object class and motion respectively.

2.3 Inference

2.3.1 Mean Field Inference

Inference has been a challenging problem for large scale CRFs. We begin by reviewing the approach of [17], which provides filter-based method for performing fast approximate maximum posterior marginal (MPM) inference in multi-label CRF models with fully connected pairwise terms, where the pairwise terms have the form of a weighted mixture of Gaussian kernels. Given a simple form of Dense-CRF, [17] shows how fast approximate MPM inference can be performed using cross bilateral filtering techniques within a mean-field approximation framework. The mean-field approximation introduces an alternative distribution over the random variables of the CRF, $Q(X)$, where the marginals are forced to be independent, e.g. $Q(X) = \prod_i Q_i(x_i)$. The mean-field approximation then attempts to minimize the KL-Divergence $D(Q||P)$ between Q and the true distribution P. By considering the fixed-point equations that must hold at the stationary points $D(Q||P)$, the following update may be derived for $Q_i(x_i = l)$ given the settings of $Q_j(x_j)$ for all $j \neq i$

$$Q_i(x_i = l) = 1/Z_i \ \exp\{-\psi_i(x_i) - \sum_{l' \in \mathcal{L}} \sum_{i \neq j} Q_j(x_j = l').\psi_p(x_i, x_j) \tag{2.13}$$

where $Z_i = \sum_{x_i = l \in \mathcal{L}} \exp\{-\psi_i(x_i) - \sum_{l' \in \mathcal{L}} \sum_{i \neq j} Q_j(x_j = l').\psi_p(x_i, x_j)$ is a constant which normalizes the marginal at pixel i. If the updates in Eq. 2.13 are made in sequence across pixels $i = 1...N$, the KL-Divergence is guaranteed to decrease. In [17], it is shown that parallel updates for Eq.2.13 can be evaluated by convolution with a high dimensional gaussian kernel using any bilateral filter.

2.3.2 Inference in Product label space

Now we discuss how we provide an efficient inference method for jointly estimating per-pixel object class and motion labels. The mean-field approximation introduces an alternative distribution over the random variables of the CRF,$Q_i(z_i)$, where the marginals are forced to be independent $Q(z) = \prod_i Q_i(z_i)$. The mean-field approximation then attempts to minimize the KL-divergence between the Q and the true distribution P. We can therefore take $Q_i(z_i) = Q_i^O(x_i)Q_i^{\mathcal{M}}(y_i)$. Here Q_i^O is a multi-class distribution over the object labels , and $Q_i^{\mathcal{M}}$ is a binary distribution over moving or stationary given by $\{0, 1\}$.

$$\begin{aligned} Q_i^O(x_i = l) = 1/Z_i \ \exp\{-\psi_i^O(x_i) - \sum_{l' \in \mathcal{L}} \sum_{i \neq j} Q_i^O(x_j = l').\psi_{ij}^O(x_i, x_j) \\ - \sum_{m' \in \mathcal{M}} Q_i^{\mathcal{M}}(y_i = m').\psi_{i,l,m}^{O\mathcal{M}}(x_i, y_i)\} \end{aligned} \tag{2.14}$$

The inference for the Motion layer is similar to the object class layer and is given by:

$$\begin{aligned} Q_i^{\mathcal{M}}(y_i = m) = 1/Z_i \ \exp\{-\psi_i^{\mathcal{M}}(y_i) - \sum_{m' \in \mathcal{M}} \sum_{i \neq j} Q_j^{\mathcal{M}}(y_j = m')\psi_{ij}^{\mathcal{M}}(y_i, y_j) \\ - \sum_{l' \in \mathcal{L}} Q_i^O(x_i = l').\psi_{i,l,m}^{O\mathcal{M}}(x_i, y_i)\} \end{aligned} \tag{2.15}$$

Where Z_i is given as the normalization factor , and $m \in \{0, 1\}$. As proposed in [17], Using $n + m$ Gaussian convolutions we can efficiently evaluate the pairwise summations which are given as Potts model.

2.4 Learning

We learn the parameters for the label and motion in this section. We describe a piecewise method for training the label and motion correlation matrices. In the model described, we train for the matrix simultaneously by learning an $(n + 2)^2$ correlation matrix.

We use the modified adaboost framework implemented in [47]. For training we denote the training dataset of N instances of pixels or regions as $\mathcal{D} = \{(\mathbf{t_1}, \bar{z}_1), (\mathbf{t_2}, \bar{z}_2), ..., (\mathbf{t}_{\mathcal{N}}, \bar{z}_{\mathcal{N}})\)$. Here, t_i is a feature vector for the i-th instance and $\bar{z}_i = [\bar{x}_i, \bar{y}_i]$ is an indicator vector of length $n + 2$, where $\bar{x}_i(l) = 1$ implies that the class label is associated with the pixel or region instance of i and $\bar{x}_i(l) = -1$ represents that the class is not associated with the instance i and similarly for $\bar{y}_i(m) = 1$ and $\bar{y}_i(m) = -1$ represents the association of motion m for the instance i. Therefore, $\bar{z}_i$ represents the object class and motion ground truth information for the instance i.

In the following approach, we show how to compute $\lambda(l, m)$ .The boosting approach in [13] generates a strong classifier $H_{s,l}(t)$ for each object class l and each round of boosting $s = 1, 2, 3, ..., S$.These strong classifiers can be defined as:

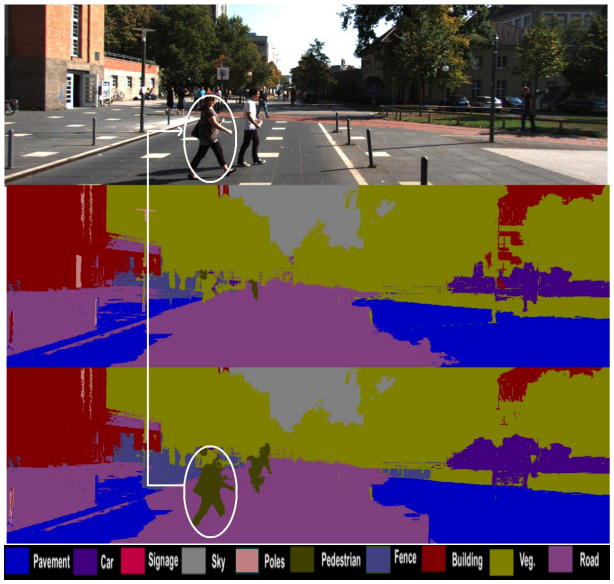

Figure 2.3: In the figure we do a comparative evaluation between the results of Full-C and our method.The original image(1) is taken from the KITTI dataset. The output of the Full-C(2) is depicted which shows a wrong labelling of the pedestrian pixels in the image. The Results of the proposed method(3) depict the improvement in the semantic segmentation.

$$H_{s,l}(t) = \sum_{s=1,2,\ldots,S} \alpha_{s,l} h_{s,l}(\mathbf{t}) \tag{2.16}$$

Here $h_{s,l}$ are weak classifiers , and $\alpha_{s,l}$ are the non-negative weights set by the boosting algorithm.As proposed in [13] , we use their joint learning approach, which generates a sequence of reuse weights $\beta_{s,l}(H_{s-1}, m)$ for each class and motion attributes l, m at each iteration s. These represent the weight given to the strong classifier for motion label m in round $s-1$ in the classifier for l at round s. Using the following reuse weights and the strong classifiers we can calculate the label correlation :

$$\lambda(l, m) = \sum_{s=2,\ldots S} \alpha_{s,l}(\beta_{s,l}(H_{s-1,m})) - \beta_{s,l}(-H_{s-1,m})) \tag{2.17}$$

This learning approach incorporates information about the motion likelihood and appearance relationship between motion and objects.

2.5 Conclusion

We have proposed a joint approach simultaneously to predict the motion and object class labels for pixels and regions in a given image. The experiments suggest that combining information from motion and objects at region and pixel-levels helps semantic image segmentation, An example result has been

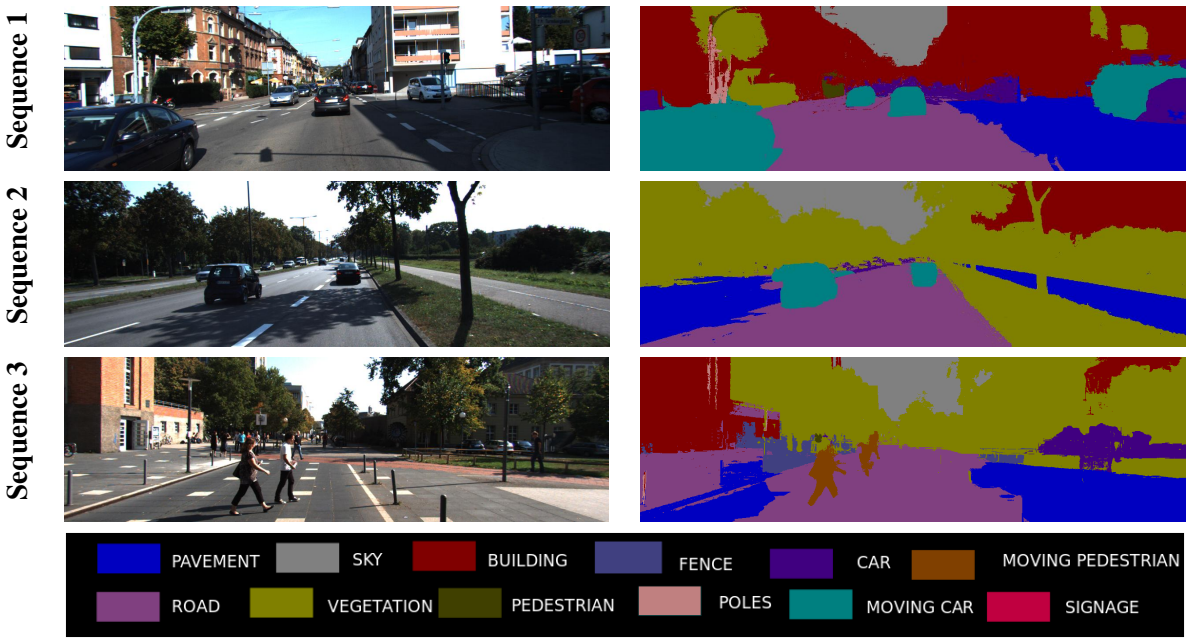

Figure 2.4: The pixel wise result of our method on the KITTI test dataset . Note, that we are able to segment degenerate motions in the sequences. We show our results on varying scenarios i.e in an urban setting(Sequence 1), highway setting(Sequence 2) and in case of moving pedestrian(Sequence 3). We achieve state-of-the art results for motion segmentation using our joint formulation. Best viewed in color

depicted in Fig. 2.3, where the semantic segmentation has been improved using the motion constraints. We show sample results of the algorithm in Fig 2.4, where we have evaluated the complete algorithm on 3 datasets of KITTI. Accurate motion segmentation is an important criterion for reconstruction and understanding of dynamic scenes. We show in chapter 3, the relevance of the semantic motion segmentation and its applications in dynamic scene understanding.

Chapter 3

Dynamic Body VSLAM with Semantic Constraints

3.1 Introduction

We have used semantic information for improvement of motion segmentation and detection in Chapter 2. In this chapter we exploit these constraints to improve the accuracy of the trajectory of the moving object and its reconstruction. While there have been many advances in semantic recognition as well, the state-of-the-art lacks far behind the robustness needed for most applications of robotic perception in dynamic environments. This system allows the semantic information and the reconstruction modules to help each other produce better overall results: SfM methods fail on moving, specular vehicles. In addition to improving 3D location estimates, the exact 3D shape and pose provided open the possibility for more sophisticated planning and control downstream in the autonomous vehicles processing pipeline. We use different semantic constraints to improve Dynamic body VSLAM. We formulated the problem using bundle adjustment based trajectory optimization.

3.2 Problem Formulation

We isolate pixels belonging to moving objects from static objects in the motion segmented images which are the output of our semantic motion segmentation algorithm in 2. Pixels belonging to each type of object (static or motion) are then used as input to localize and map each object independently. In this section, we propose a novel framework for trajectory computation for static or moving objects from a moving platform. The below process is carried out for all the moving objects and the camera mounted vehicle[1]. Let us introduce some preliminary notations for trajectory computation. The extrinsic parameters for frame $k = 1, 2, 3, 4...n$ are the rotation matrix R_k and the camera center C_k relative to a world coordinate system. Then the translation vector between the world and the camera coordinate systems is T_k = -$R_k C_k$.

[1] Henceforth referred as camera

We do a motion based clustering of the moving objects and segment out each moving object from the image. These moving objects are tracked in the image for the complete sequence. We use multiple complementary sensing modalities and cues acquired through an ensemble of detectors,this ensemble includes semantic-based object detector, depth based shape detector and motion detector. These detection algorithms are fused into a framework using a sampling based method i.e Reversible Jump Markov Chain Monte Carlo particle filter, we follow the formulation of [3] to track the moving objects in the image. Urban sequences have fast moving cars with large image motions which cant be tracked using standard trackers like KLT. Particle filters have the distinct ability to track each moving object independently using separate motion models for each particle which works robustly in outdoor scenarios.

3.2.1 Trajectory Initialization

We initialize the motion of each object separately using SIFT feature points. SIFT feature points are tracked using dense optical flow between consecutive pair of frames. Key points with valid depth values are used in a 3-point-algorithm within a RANSAC framework to find the robust relative transformation between pairs of frames. We obtain pose estimates of the moving object in the world frame by chaining the relative transformations together in succession. For moving objects the initial frame k where detection occurs is taken as the starting point. Trajectory estimates are then initialized for each object independently corresponding to the frame k assuming the camera is static.

3.2.2 3D Object Motion Estimation

Once 3D trajectories are estimated for each object independently, we need to map these trajectories onto the world coordinate system. Since, we are dealing with stereo data and for every frame we have 3D information, this mapping can be represented as simple coordinate transformations. Also, since we are not dealing with monocular images, the problem of relative scaling can be avoided.

Given the pose of the real camera in the k^{th} frame $((R_k^c, T_k^c))$ and virtual camera (R_k^v, T_k^v) [45] computed during trajectory initialization described earlier, we should be able to compute the pose of the b^{th} object (R_k^b, T_k^b) relative to its original position in the first frame in the world coordinate system. The object rotation R_k^b and translation T_k^b are given as

$$R_k^b = (R_k^c)^{-1} R_k^v, \qquad T_k^b = (R_k^c)^{-1}(T_k^v - T_k^c) \tag{3.1}$$

Thus we get the localization and sparse map of both the static and moving world. We found this approach to object motion estimation to be better on both small and long sequences than TriTrack [23].

3.2.3 Dynamic Object Trajectory Optimization

Once 3D object motion and structure initialization has been done, we need to refine the structure and motion using bundle adjustment (BA). In this section, we describe our framework for BA to refine the

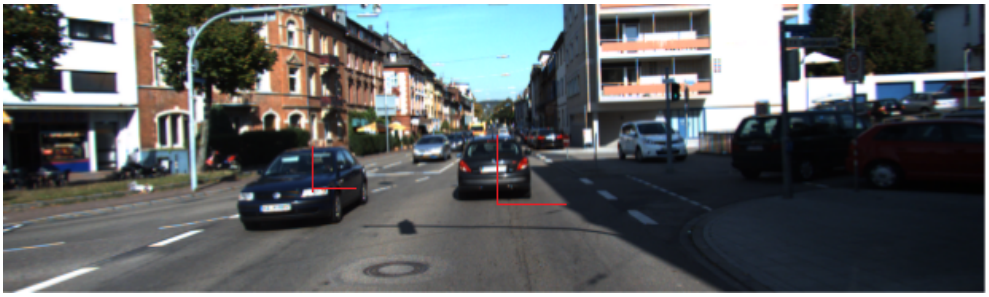

Figure 3.1: We depict the constraint to showcase the motion of the moving object to be perpendicular to the ground plane.

trajectory and sparse 3D point reconstruction of dynamic objects along with *several novel* constraints added to BA that increase the accuracy of our trajectories and 3D points. We term these constraints *semantic or contextual* constraints since they represent our *understanding* of the world in a geometric language, which we use to effectively optimize 3D points and trajectories in the presence of noise and outliers. These semantic constraints are a consequence of the semantic motion labels acquired from the semantic motion segmentation algorithm (chapter 2). The assumptions underlying these constraints derive from *commonly* observed *shape* and *motion* traits of cars in urban scenarios. For example the normal constraints follow the logic that the motion of a dynamic object like a vehicle is always on a plane (the road surface) and hence constrained by its normal. Similarly, the 3D points on a dynamic object are constrained to lie within a 3D "box" since dynamic objects like cars cannot be infinitely large. Finally, our trajectory constraints encode the fact that dynamic objects have smooth trajectories, which is often true in urban scenarios. In summary, we try to minimize the following objective function

$$min \sum_{\text{i}} \sum_{p \in V(i)} \texttt{BA2D} + \lambda \texttt{BA3D} + \lambda \texttt{TC} + \texttt{NC} + \texttt{BC} \tag{3.2}$$

where BA2D represents the 2D BA reprojection error ($\|\tilde{x}_p^i - K[R_i \mid T_i]X_p\|^2$), BA3D represents the 3D registration error common in optimization over stereo images ($\|\tilde{X}_p^i - [R_i \mid T_i]X_p\|^2$) and TC, NC, BC represent various optimization terms that can be seen as imposed constraints on the resulting shape and trajectories as explained below. Here i indexes into images, and˜represents variables in the camera coordinate system, with other quantities being expressed in the world coordinate system. Also, $p \in V(i)$ represents pixels visible in image i.

$$\texttt{NC2}: \quad \sum_{i=1}^{m} N_g^i \cdot (T_c^k - T_c^{k-1}) \tag{3.3}$$

3.2.3.1 Planar Constraint

We constrain motion to be perpendicular to the ground plane where the ground plane normal is found from the initial 3D reconstruction of the ground as shown in Fig 3.1.

$$\texttt{NC1}: \quad N_g \cdot (T_c^k - T_c^{k-1}) \tag{3.4}$$

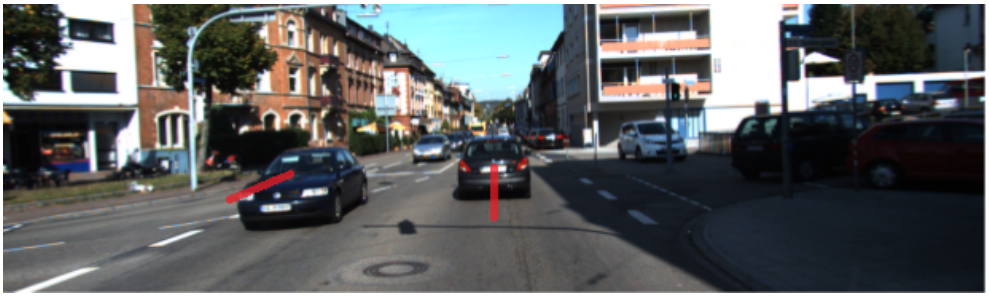

Figure 3.2: The image represents the smooth motion of cars in an urban environment used as a constaint in the problem formulation.

where N_g is the normal of the ground plane in the camera frame, T_c^k - T_c^{k-1} is the direction of camera motion in the local coordinate system. This local motion and normal estimation allows us to use the same constraint even on changing planes like up or down a slope. Since 3D reconstruction of the ground can be noisy, estimation of N_g is done using least squares. Alternatively, we could follow a RANSAC based framework of selecting m top hypotheses for the normal N_g^i $(i = 1 \ldots m)$, and allow bundle adjustment to minimize an average error of the form

3.2.3.2 Smooth Trajectory Constraints

We enforce smoothness in trajectory, a valid assumption for urban scenes as shown in fig 3.2, by constraining camera translations in consecutive frames as

$$\texttt{TC1}: \quad \|(T_c^{k+1} - T_c^k) \times (T_c^k - T_c^{k-1})\| \tag{3.5}$$

where $T_c^{k+1}, T_c^k, T_c^{k-1}$ are the 3d translations at frame k+1, k and k-1. Alternatively, we could also minimize the norm between two consecutive translations unlike TC1, which only penalizes direction deviations in translation.

$$\texttt{TC2}: \quad \|(T_c^{k+1} - 2 * T_c^k + T_c^{k-1})\|^2 \tag{3.6}$$

3.2.3.3 Box Constraints

Depth estimation of objects like cars are generally noisy because their surface is not typically Lambertian in nature, and hence violates the basic assumptions of brightness constancy across time and viewing angle. Furthermore, noise in depth infuses errors into the estimated trajectory through the trajectory initialization component. To improve the reconstruction accuracy in such cases, and to limit the destructive effect that noisy depth has on object trajectories, we introduce shape priors into the BA cost function that essentially constrains all the 3D points belonging to a moving object to remain with a "box" as shown in Fig 3.3. More specifically, let X_i^b & X_j^b be two 3D points on a moving object O^b. For every such pair of points on the object, we define the following constraint

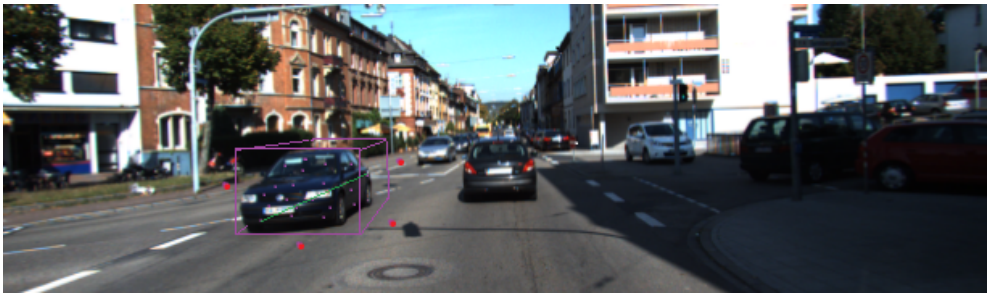

Figure 3.3: Depiction of box constraints on the motion of the car. The points lying inside the bounding box around the image are optimized using a soft box constraint in bundle adjustment

$$\texttt{BC1}: \quad \sum_{\forall X_i^b, X_j^b \in O^b} \|X_i^b - X_j^b - B(i,j)\|^2 \\ -\delta \leq B(i,j) \leq \delta \tag{3.7}$$

where $B(i,j)$ is a vector of bounds with individual components $(b_x(i,j), b_y(i,j), b_z(i,j))$ and δ is a vector of positive values.

Note that the above equation is defined for every pair of points on the object, which leads to a *quadratic explosion* of terms since $B(i,j)$ is a separate variable for each pair. x

3.2.3.3.1 Alternate Formulations One way to reduce the explosion would be to reduce the number of variables added because of the box constraints to the Bundle Adjustment. This could be done by alternatively minimizing the following terms instead of the constraint in equation (3.7)

$$\texttt{BC2}: \quad \sum_{\forall (X_i^b, X_j^b) \in O^b} \|X_i^b - X_j^b - b(i,j)\|^2, -\delta \leq b(i,j) \leq \delta \tag{3.8}$$

$$\texttt{BC3}: \quad \sum_{\forall (X_i^b, X_j^b) \in O^b} \|X_i^b - X_j^b - B\|^2, -\delta \leq B \leq \delta \tag{3.9}$$

$$\texttt{BC4}: \quad \sum_{\forall (X_i^b, X_j^b) \in O^b} \|X_i^b - X_j^b - b\|^2, -\delta \leq b \leq \delta \tag{3.10}$$

where $b(i,j)$ in equation (3.8) is a scalar common to all 3 dimensions, i.e we assume the cuboid to have the same lenght in all the three dimensions. Here, B (equation (3.9)) is a 3×1 vector common to all point pairs, and b (equation (3.10)) is a scalar common to all pairs and dimensions.

3.2.3.3.2 Alternate Minimization Strategies It is now known that a lot of information in terms like BC1, BC2, BC3, BC4 above are redundant in nature [4], and there is essentially a small "subset" of pairs which is sufficient to produce optimal or near-optimal results in such cases. However, it is not clear how to pick this small subset. Here, we take the help of the Johnson-Lindenstrauss theorem and its

variants [6, 5], to select a random set of pairs from the ones available, such that we closely approximate the BC error when all the point pairs are used.

More specifically, the terms expressed in BC1, BC2, BC3, BC4 can all be expressed in the form

$$\texttt{BCLin}: \quad \|AX - B\|^2, \text{ such that }, CB = D \tag{3.11}$$

where X is a concatenation of all 3D points, and B is a collection of all box bounds. The matrix A is constructed in such a way that each row of A consists of only two non-zero elements at the i^{th} and j^{th} positions with values 1 and -1 respectively, and they represent the difference $X_i^b - X_j^b$. The linear constraint $CB = D$ is useful to represent the fact that some elements of vector B are equal to others. While this is useful to represent BC2, BC3, BC4 (BC1 can be exactly represnted without this constraint) we temporarily "relax" this constraint, and enforce it post-optimization by taking the average of duplicate variables. Note that the dimensions of A are of the order $3^nC_2 \times 3n$, where n is the number of 3D points. Notice that for $n = 3000$, ${}^nC_2$ is approximately 4.5 million, and is highly slow to optimize! To reduce this computational burden, we embed the above optimization problem in a randomly selected subspace of considerably lower dimension, with the guarantee that the solution obtained in the subspace is close to the original problem solution with high probability. To do this, we draw upon a slightly modified version of the *affine embedding* theorem presented in [5] which states

Theorem 3.2.1 *For any minimization of the form $\|AX - B\|$, where A is of size $m \times n$ and $m \gg n$, there exists a* subspace embedding matrix $S : \mathbb{R}^m \mapsto \mathbb{R}^t$ *where $t = poly(n/\epsilon)$ such that*

$$\|SAX - SB\|_2 = (1 \pm \epsilon)\|AX - B\|_2 \tag{3.12}$$

Moreover, the matrix S of size $t \times m$ is designed such that each column of S has only 1 non-zero element at a randomly chosen location, with value 1 or -1 with equal probability.

Note that since elements of S are randomly assigned 1 or -1, the above transformation cannot be exactly interpreted as a random sampling of pairs of points. However for the sake of implementation simplicity, we "relax" S to a random selection matrix. As we show later, empirically we get very satisfying results.

Finally, there can be several strategies to select random pairs of points for box constraints. We experimented with the following in this paper.

- Strat1: Randomly select pairs from the available set.
- Strat2: Randomly select one point, and create its pair with the 3D point that is farthest from the selected point in terms of Euclidean distance.
- Strat3: Randomly select one point, and sort other points in descending order based on Euclidean distance with selected point. Pick the first point from the list that has not been part of any pair before.

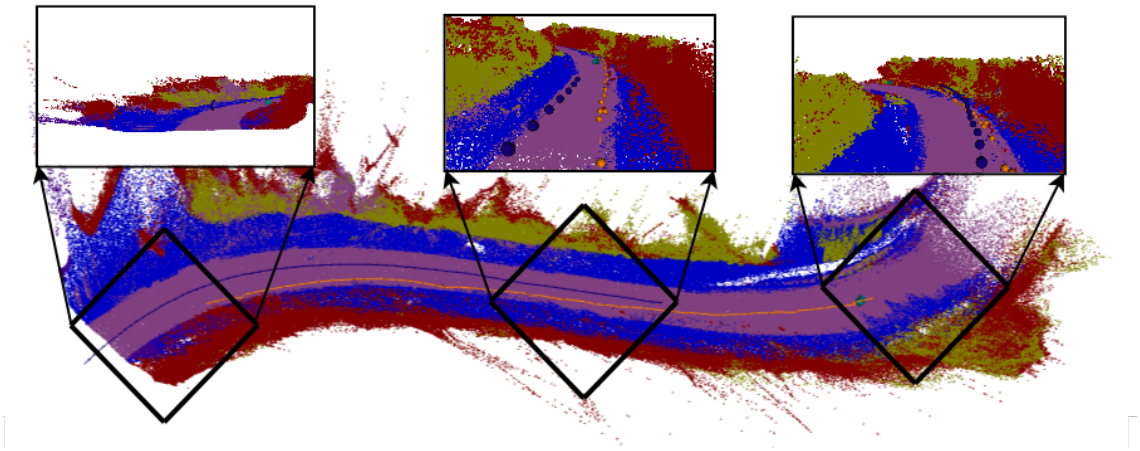

Figure 3.4: Reconstruction result for **KITTI 4** sequence with overlay of sematics. Note the accurate reconstruction of trajectories and of the car and the camera, in spite of curvilinear motion. Please see supplementary video for further details.

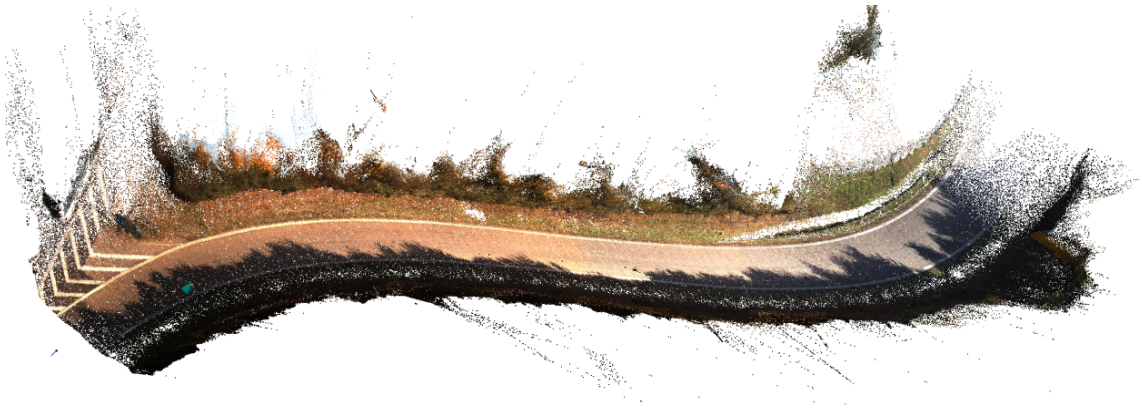

Figure 3.5: Reconstruction result for **KITTI 4** sequence overlayed with the image colour. Please see supplementary video for further details.

Once the proper set of constraints are selected from the above choices, the final objective function in equation 3.2 is minimized with L_2 norm using CERES solver [1].

3.3 3D RECONSTRUCTION

To create a dense semantic map, we use the trajectory of the camera and moving objects. We compute the depth map of the image to reconstruct the surfaces of objects. Using the trajectory of the camera we register these depth images using Truncated Signed Distance function (TSDF). The generated 3D volume is labelled using the labels obtained from semantic motion segmentation. The trajectory used for the reconstruction of a moving object is the trajectory before it is transformed into the world coordinate system.

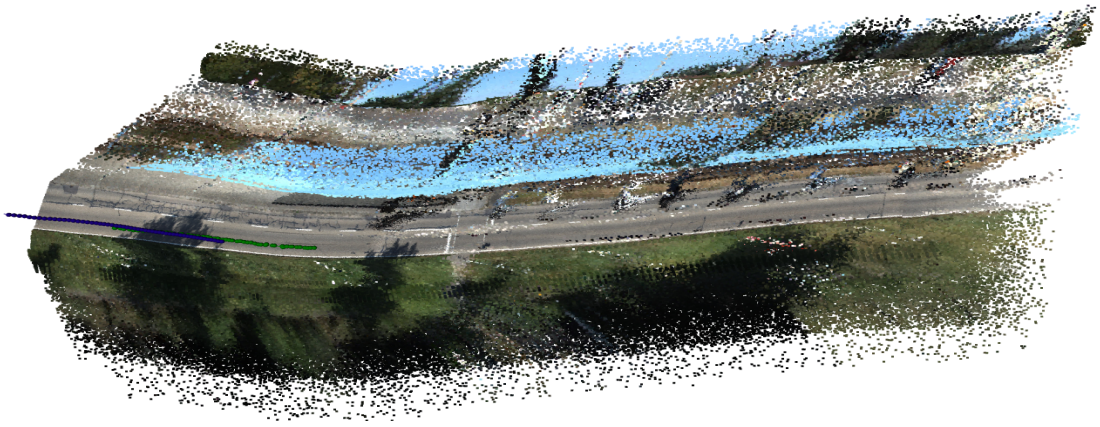

Figure 3.6: Reconstruction result for **KITTI 1** sequence overlayed with the image colour.

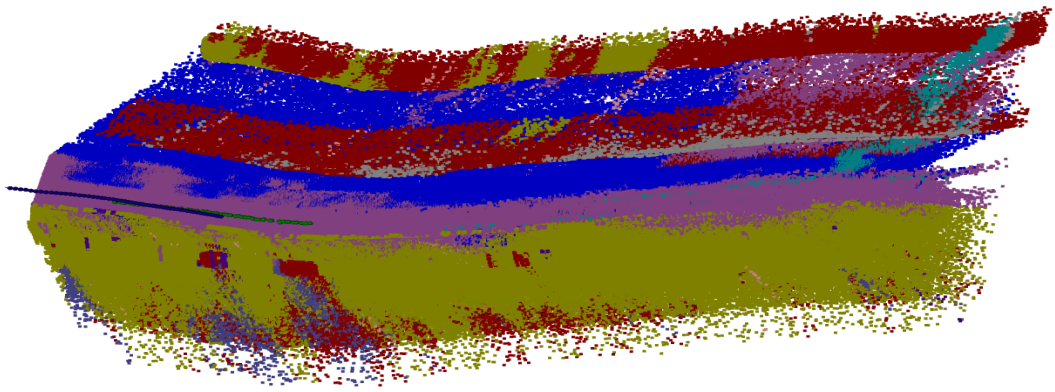

Figure 3.7: Reconstruction result for **KITTI 1** sequence with overlay of semantics and trajectories.

3.3.1 Depth Map Computation

Given a rectified stereo images we compute the disparity images using the semi-global block matching algorithm (SGBM). The disparity inconsistencies are smoothed out using a bilateral filter and the labels assigned from the semantic segmentation. The depth images from the disparity map are computed using :$z_i = B.f/d_i$, where z_i and d_i are the depth and the disparity corresponding to the i^{th} pixel respectively. The B and f are the baseline and the focal length of the stereo rig. To get a better estimate of the 3D volume we clipped the disparity values.

3.3.2 TSDF Volume

We fuse the depth map for moving objects as well as stationary background incrementally into a single 3D reconstruction using the volumetric TSDF representation. A signed distance function corre-

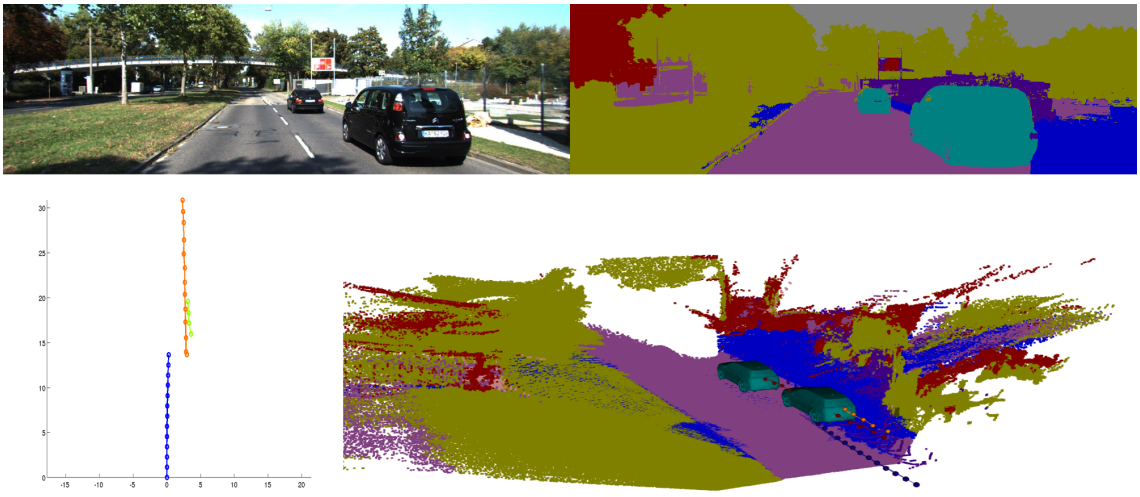

Figure 3.8: The overall algorithm output on KITTI 2 sequence, left top represents the Input Stereo image, The right top represnets the semantic motion segmentation , The botton left is the trajectories of the moving objects in the world coordinate frame. The bottom right shows the reconstruction of the dynamic objects in the world coordinate frame. Best viewd in color.

sponds to the closest surface interface, with positive values corresponding to free space, and negative values corresponding to points behind the surface. This representation allows for the efficient registration of multiple surface measurements, by globally averaging the distance measures from every depth map at each point in the object space. The estimate of the normal of the surface is computed using ray casting as proposed in [48].

3.3.3 TSDF Labelling

The label prediction for each triangulated face in the mesh is done through random sampling of the mesh. Each random sampled point is projected back into images using the estimated camera pose, resulting in a set of image points for each 3D point in the TSDF cloud. We compute each label score for the 3d point in all the projected images. The mesh is assigned the label based on the winner-take-all algorithm,i.e. the label which has the highest score is assigned to the mesh.

3.4 Conclusion

We have shown that semantic constraints have improved the reconstruction and localization of moving objects in dynamic environments. We have shown accurate reconstruction of a rotating dyanmic sequence with semantics in Fig. 3.4 and with image information in fig. 3.5. We have tested the algorithm on large sequence with multiple cars and the results of the reconstruction are displayed in 3.7 and with real worldreconstruction in 3.6. The output of the algorithm is depicted in Fig. 3.8, it shows

the input image, semantic motion segmentation, trajectories in world coordinate frame and the final dynamic environment reconstruction. We shown an extensive evaluation and comparision of the proposed method in chapter 4.

Chapter 4

Evaluation and Comparison

In this section we provide extensive evaluation of our algorithms on both synthetic and real data. For real datasets, we have used the KITTI tracking dataset for evaluation of the algorithm as the ground truth for localization of moving objects per camera frame is available. It consists of several sequences collected by a perspective car-mounted camera driving in urban, residential and highway environments, making it a varied and challenging real world dataset. We have taken four sequences consisting of 30, 212, 30 and 100 images for evaluating our algorithm. We choose these 4 sequences as they pose serious challenges to the motion segmentation algorithm as the moving cars lie in the same subspace as the camera. We have extensively compared with TriTrack as it has accurate localization of moving objects.These sequences also have a mix of m ultiple cars visible for short duration along with cars visible for the entire sequence which allows us to test the robustness of our localization and reconstruction algorithms on both short and long sequences. While we show qualitative results for all the four sequences, we show extensive quantitative evaluation for the longest sequence of 212 frames called **KITTI1** sequence.

We do extensive quantitative evaluation on synthetic dataset as well. We generated 1000 3D points on a cube attached to a planar ground to simulate a car and road. We then move the car over the road, while simultaneously moving the camera to generate moving images after projection of the 3D points. Finally we added Gaussian noise to both the 3D points on the car and the points on the road to simulate errors in measurement. Correspondences between frames are automatically known as a result of our dataset design.

4.1 Evaluation of Semantic Motion Segmentation

We have used a popular street-level dataset—*KITTI* for evaluation. It consists of several sequences collected by a car-mounted camera driving in urban,residential and highway environments, making it a varied and challenging real world dataset. We have taken 6 sequences from the tracking dataset of KITTI each containing 20 stereo images, each of size 1024 x 365. Firstly, these sequences were manually annotated with the 11 object classes containing the spectrum of classes. Secondly, each of the image was annotated with moving and non-moving objects using the tracking ground truth data.

Method	Pavement	Road	Sky	Car	Building	Vegetation	Poles	Pedestrian	Fence
Super-pixel CRF **(AHCRF)[19]**	74.2	93.2	95.2	77.9	94.2	84.5	**31.7**	32.2	50.4
Fully-Connected CRF **(Full-C)[17]**	73.5	93.4	95.3	77.4	**94.7**	84.6	27.3	31.3	50.2
AHCRF + Motion **(JAHCRF)**	**73.7**	93.7	95.2	81.2	94.5	84.7	31.5	37.2	**50.6**
Full-C + Motion **(OURS-O)**	73.6	**93.8**	**95.3**	**85.2**	94.5	**84.7**	27.1	**39.2**	50.4

Table 4.1: This table shows the image-based semantic Evaluation for all the sequences of the KITTI dataset.We compare our results with publicly available semantic segmentation .1) super-pixel Clique-based CRF(AHCRF) 2)Fully connected CRF (FULL-C) 3)Joint motion and object class segmentation using clique (AHCRF-Motion) 4) Our method for Semantic segmentation.The table shows a substantial improvements in the object class segmentation of the car and pedistrian.

These sequences are challenging as they contain multiple moving cars and the labels consisted of 11 classes *i.e Pavement, Car, Signal, Sky, Poles, Pedestrian, Fence, Building, Vegetation and Road*. We have selected KITTI dataset as it contains stereo image pairs with a wide baseline.We Learned the motion compatibility, as simple lookup table would not work due to instances where the semantic prior is wrong. The hard negatives provide us with the ability to categorically remove objects with wrong motion likelihood a common occurrence due to inconsistent disparity. This would allow us to test on a variety of datasets without needing to train for similar classes . The dataset was also chosen with the view to showcase the algorithms capability on degenerate cases which are not commonly addressed in other datasets.

We have used semi-global block matching [12] disparity map computation algorithm for the disparity computation in the stereo camera sequence. For the computation of the motion of the moving camera, we have used RANSAC based algorithm to solve for the Eq.(2.5). We have added the temporal consistency of motion across 3 images to the likelihood estimate which improves the results. As for the dense optical flow computation in the implementation, we have used the Deepflow algorithm from [41], which has given state-of-the art results for the KITTI evaluation benchmark. For object class segmentation we have used the publicly available Texton boost classifier to compute the unary potentials for each class.

4.1.1 Qualitative evaluation

We show our results in comparison to Ground truth, in semantic segmentation with FULL-C and in motion to GEO-M. FULL-C isn't able to segment cars as a whole and miss out on several patches while GEO-M fails in the case due to degeneracy in motion. KITTI1, has patches on the front car due

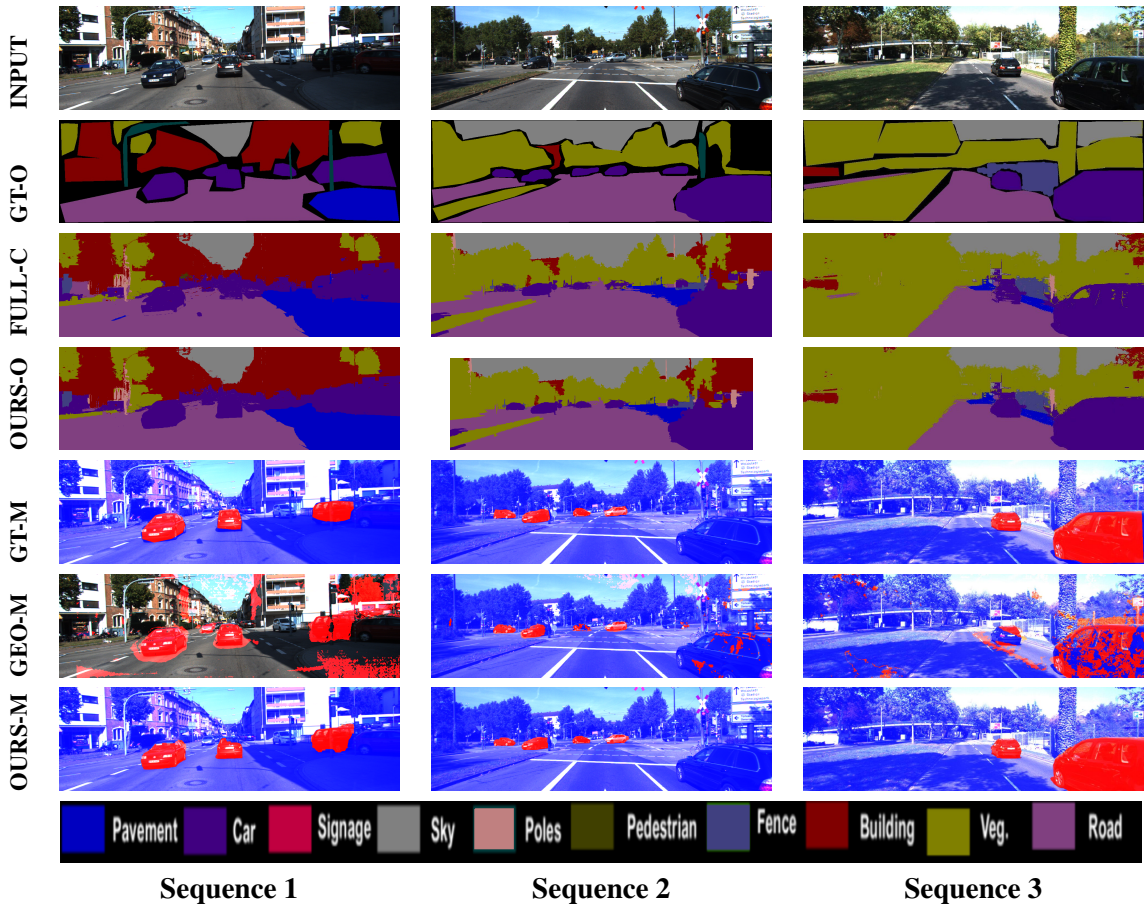

Figure 4.1: Qualitative object class and motion results for the KITTI dataset 1) Images of three sequences of KITTI dataset(INPUT) 2)Ground truth of object class segmentation (GT-O) 3)Object class segmentation results using fully connected CRF(FULL-C) 4)Object class segmentation using the Joint formulation of the proposed method (OURS-O) 5) Ground truth of the motion segmentation (GT-M) 6)Motion segmentation using geometric constraints (GEO-M)[25] 7) Proposed method dense motion segmentation(OURS-M).For motion segmentation blue depicts stationary and red pixels represent moving. best viewed in color

to failure of disparity computation, also the car's window is wrongly classified by ALE. These things are corrected by our method. The motion consistency helps in removing the window patch while in KITTI3 the degeneracy is handled by our motion estimator independent of such geometric constraints. The joint formulation captures the motion and improves the semantics of the image. We also show an improvement in the segmentation where the disparity computation has failed as show in the KITTI3. We take a specific example in Fig 2.3 showing Pedestrians, the image is smoothed out by the fully connected dense CRF leading to the wrong labelling while our method is able to correctly segment the whole pedestrian. This again reiterates the use of motion correlation for a better labelling. In all the above cases, we can see the effectiveness of our algorithm in handling the motion to generate a dynamic

	Error Type	MS	MS+NC1	MS+NC1 +TC1	MS+NC1+TC1+ BC1 (1000 constr)
BA2D	rmse	2.425649	2.362224	2.351205	**2.302849**
	mean	1.989408	1.955466	1.969793	**1.937154**
	median	1.669304	1.616398	1.685272	**1.640389**
BA3D	rmse	3.627977	3.587194	3.352087	**3.270264**
	mean	2.544718	2.527314	2.398578	**2.367702**
	median	2.000463	1.997689	1.941246	**1.928450**
BA23D	rmse	2.357187	2.305733	2.296139	**2.254192**
	mean	2.035764	1.986784	1.971698	**1.881728**
	median	1.877257	1.759010	1.760857	**1.756554**

Table 4.2: Dynamic scene of **KITTI** dataset of 212 frames. Note that adding box constraints over normal and trajectory lead to the best results.BA23D = BA2D + BA3D

	Error Type	Without MS	MS	MS+NC1	MS+NC1 +TC1
BA2D	rmse	1.416246	1.001566	**0.941971**	0.958505
	mean	1.212164	0.826189	**0.764188**	0.779054
	median	1.088891	**0.677419**	0.690825	0.716546
BA3D	rmse	1.476649	**0.959499**	0.975747	0.978197
	mean	1.272985	**0.786729**	0.822169	0.824090
	median	1.279508	**0.712513**	0.773672	0.769680
BA23D	rmse	1.472399	**0.958505**	0.958541	0.958541
	mean	1.269541	**0.779054**	0.779132	0.779132
	median	1.269238	**0.716546**	0.716967	0.716967

Table 4.3: Static scene of **KITTI** dataset. Note that adding Motion Segmentation (MS) drastically improves results, while normal constraints also help in some cases. BA23D = BA2D + BA3D

semantic model of the scene. We tried using motion cues as a feature in the object class unary. This couldnt be used as a discriminative feature for an object class, as objects can have different motions which can not be learned through textonboost. We have implemented the Fully connected CRF module as it was showing substantial improvement in results for specific classes like pedestrian compared to a superpixel clique based CRF model.

4.1.2 Quantitative evaluation

We quantitavely compare our approach against the other state-of-the-art image segmentation approaches, including pairwise CRF semantic segmentation approach with super-pixel based higher orders (AHCRF), Fully connected CRF (Full-C) and joint motion-object CRF with superpixel-clique consistency (JAHCRF). The quantitative evaluation of the object class segmentation of our joint optimiza-

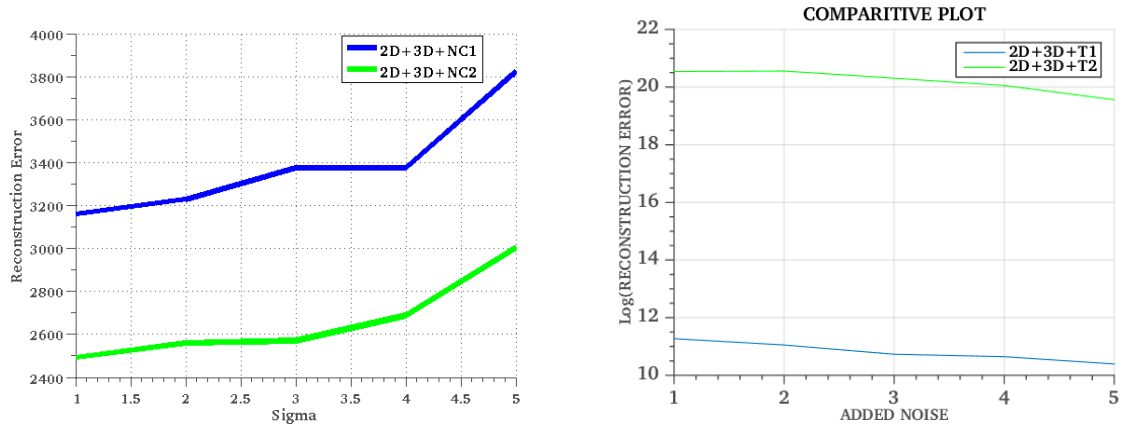

Figure 4.2: Synthetic results for Normal and trajectory constraints.

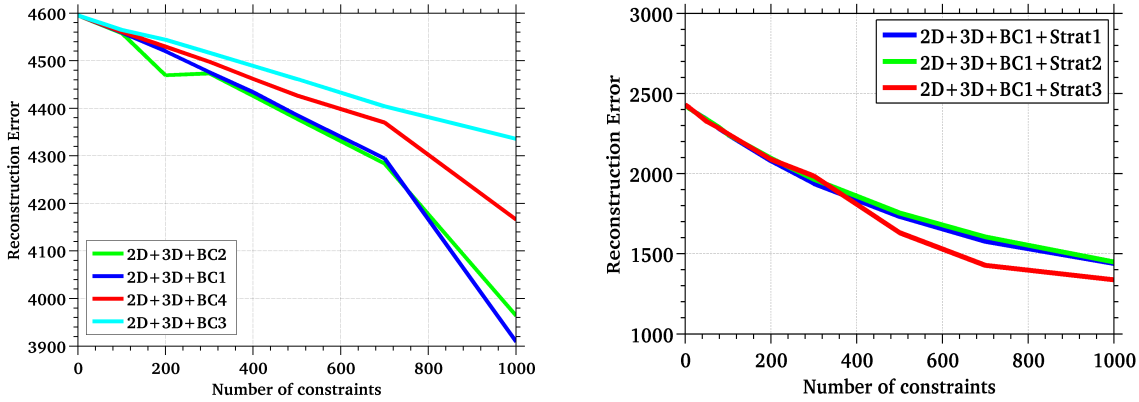

Figure 4.3: Synthetic results for box constraints. Note that in the two experiments we added a large amount of noise and picked 1000 constraints from around 500000 pairs of points, which means we use 0.2% of all available constraints. We infer that `BC1` in (a) and `Strat3` in (b) are the best performers.

tion method with respect to other approaches is summarized in Table 4.1. Evaluation is performed by cross verifying each classified pixel with the Ground truth .We choose the average intersection/union as the evaluation measure for both the image segmentation and the motion segmentation.It is defined as $TP/(TP+FP+FN)$, where TP represents the true positive ,FP the false positive and FN as the false negative. We observe an increase in performance for most of the classes in each of these measurements, mainly the object classes car and person have shown substantial improvements in accuracy. This is attributed to the fact that motion can be associated with specific classes and the pairwise connections in the motion domain respect the continuity in optical flow, while in the image domain,the connections between neighbouring pixels might violate the occlusion boundaries.

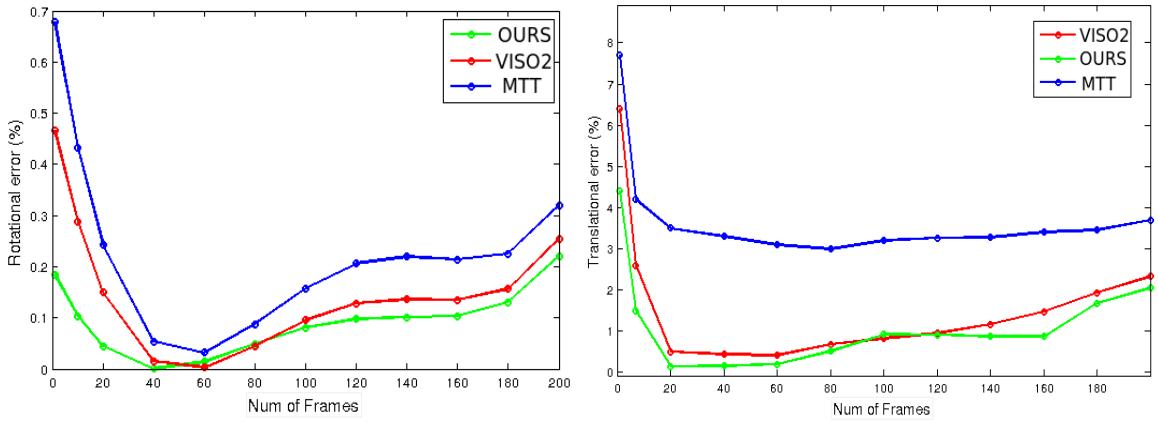

Figure 4.4: Comparative study of the translational and rotational error with respect to the number of poses for VISO2(TRITRACK), OURS and MMT.

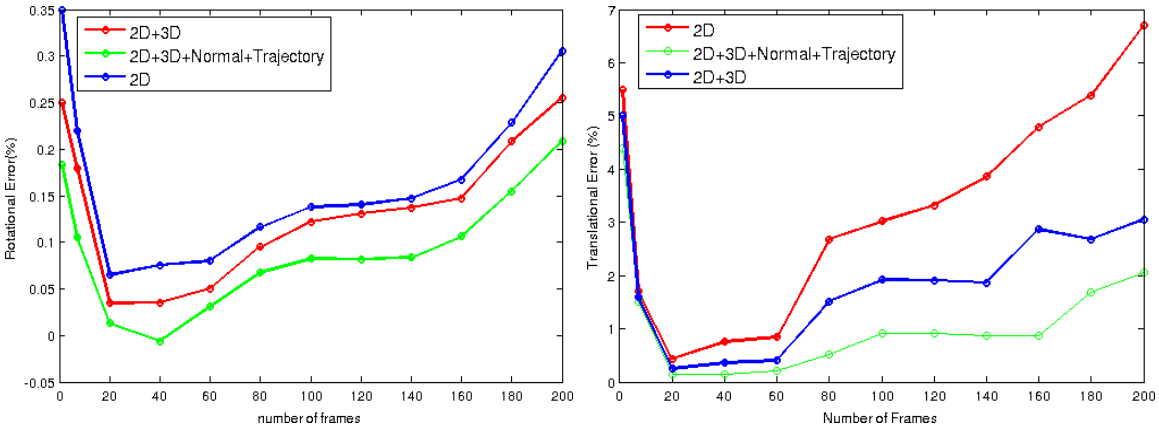

Figure 4.5: Comparative study of the translational and rotational error with respect to the number of poses for different Bundle adjustment constraints.2D represents image based BA, 2D+3D represents both images based and the depth based BA. 2D+3D+Normal+trajectory is our approach.

4.2 Quantitative Evaluation of Object Trajectory Optimization

In this section, we do an extensive evaluation of the different terms proposed in chapter 3. Note that we tried all the different terms and strategies proposed here on real data as well, and in all cases conclusions derived from synthetic data experiments are consistent with real data.

In the following section we present the results for evaluation of various terms and strategies.

4.2.1 Normal Constraint

This constraint is a contextual constraint in the sense that it enforces the fact that the moving object is usually attached to a planar ground in urban settings, and so any deviation of the object trajectory

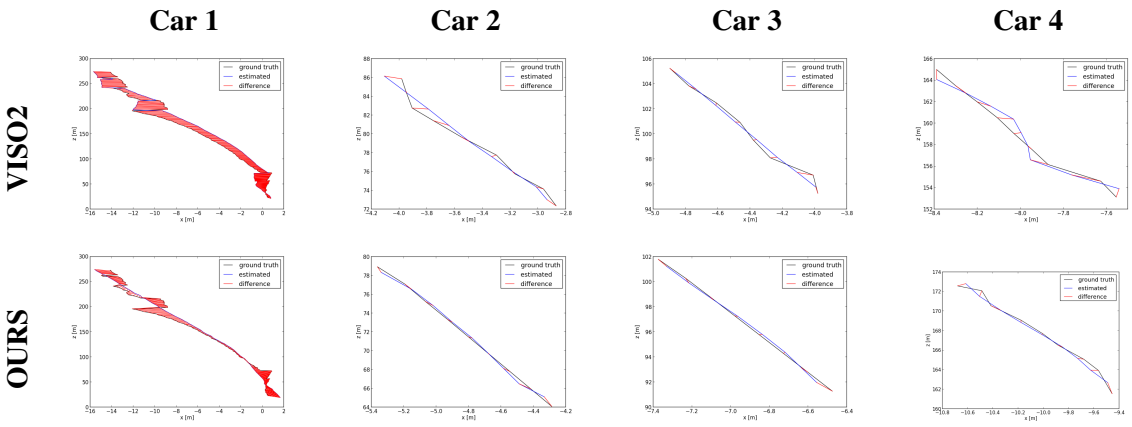

Figure 4.6: Comparison plots for 5 moving cars in the KITTI 1. The black plot represents the ground truth trajectory of the moving car in the world frame. Blue plot represents the estimated trajectory of the moving car. Red lines represent the error in the estimate with respect to ground truth for the trajectories. The error comparison is computed between OUR method and VISO2.

along the direction of the normal of the ground plane should be penalized. While `NC1` computes a least-squares estimate for the normal which is optimal under Gaussian noise, `NC2` computes several normal hypotheses using a RANSAC framework. Figure (4.2) shows the results comparing the two terms. We find that `NC1` normally performs better.

4.2.2 Trajectory Constraint

The trajectory constraint enforces smoothness in moving object trajectories, by either enforcing that the direction of motion should not change significantly between consecutive frames (`TC1`) or enforcing that both direction and magnitude must be constrained (`TC2`). Figure (4.2) plots comparative results, and we infer that `TC1` performs better.

4.2.3 Box Constraint

Box constraints enforce that the 3D reconstruction of the moving object in consideration must be *compact*. This is a useful constraint since gross errors in the depth of the object as estimated by the stereo algorithm [43] normally are not corrected by BA since it settles into a local minima. Thus, to "focus" the BA towards better optimizing the 3D structure, we add these constraints.

4.2.3.1 Box Sampling Strategies

Since box constraints lead to an explosion of terms added to BA, we experiment with 4 strategies to reduce this computational burden by random sampling [5]. Figure (4.3) show results for various terms

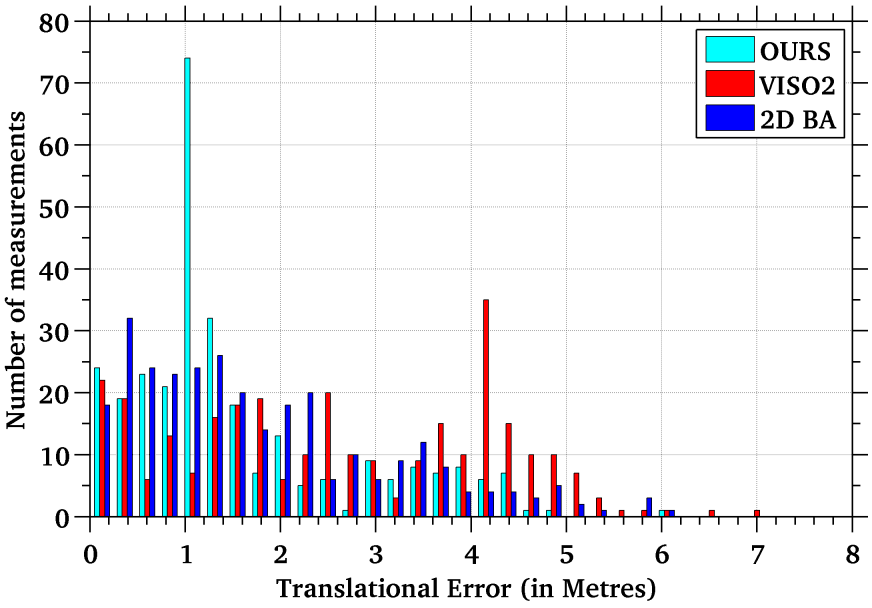

Figure 4.7: Comparison of trajectory errors of our algorithm to TriTrack [23] and standard BA after motion segmentation. The histogram plots RMSE magnitude on the x axis, and number of pose measurements that fall in each bin on the y axis. Note that most of our errors are concentrated on the left (low error), while TriTrack [23] and BA are more evenly spread. The total summed error: 2D-BA - 1.79, TriTrack - 2.62, Ours - 1.54.

of box constraints, and various strategies to optimize. Normally we find that `BC1` along with `Strat3` performs best.

4.3 Trajectory Evaluation

We compare the estimated trajectories of the moving objects and the camera to the TriTrack (Stereo) [23] for camera and TriTrack for the moving object trajectories. VISO2 S(Stereo) [9] has reported error of 2.44 % on the KITTI odometry dataset, making it a good baseline algorithm to compare with. As proposed by Sturm et al. [35], the comparison methodology is based on ATE for root mean square error (RMSE), mean, median. We use their evaluation algorithm which aligns the 2 trajectories using SVD. We show all the three statistics, as mean and median are robust to outliers, while RMSE shows the exact deviation from the ground truth.

Table (4.3) shows results for trajectory error estimation for the static part of the KITTI1 sequence, with dynamic objects removed. As can be seen, we get significant improvement in camera trajectory estimation *after* motion segmentation. This reinforces our claim that motion segmentation is essential for trajectory estimation in dynamic scenes. Since semantic constraints are tailored to dynamic bodies the best improvement using them are seen (across all rows) in table 4.2. Table (4.2) depicts the trajectory error for the moving object visible in all the 212 images of the sequence. We progressively show

how each constraint on the motion of the moving object complements its trajectory computation and reconstruction in successive columns. This further enhances our claim that our semantic constraint on dynamic bodies allows us to localize and reconstruct them more accurately. We have further evaluated the error accuracy with increasing number of frames for the KITTI1 sequence. The error of our algorithm was lesser compared to TriTrack with increasing distance.

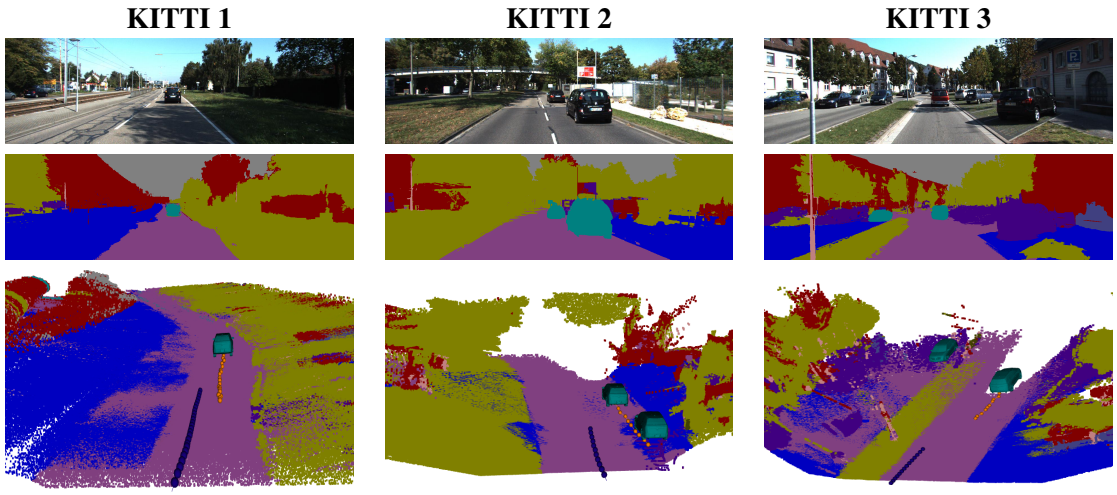

Figure 4.8: We show the **(INPUT)** image sequences for which we compute the semantic motion segmentation (**SMS**). We have depicted the reconstruction of moving objects with their trajectories **(3D-REC)**. Blue trajectories represent the camera capturing the scene. All segmentation color labels are consistent with Figure 1.1. (best viewed in color)

For quantitative evaluation of our method on the **KITTI1** sequence, we have computed the trajectories of all the moving objects. These trajectories are compared to their respective ground truth and the absolute position error of each pose is computed. We have done a histogram based evaluation of all the position error as depicted in Fig(4.7), where we compare the trajectories of our algorithm with TriTrack. We have evaluated the algorithm for a complete of 297 poses of moving objects and found that our approach outperforms TriTrack and standard 2D bundle adjustment. Qualitative results of the trajectories and reconstruction of some of the moving objects is depicted in the Fig(4.8, 3.4).

Chapter 5

Conclusions

Dynamic scene understanding is a challenging problem due to its complexity and scalability. Segmenting and Localizing a dynamic object opens-up a lot of possiblities for robot perception and automation. We have exploited the semantic and geometric constraints for improving the understaing of the dyanmic scene as a whole. Our framework has shown improvement compared to multiple state-of-the-art methods in scene understanding.

We have proposed a joint approach simultaneously to predict the motion and object class labels for pixels and regions in a given image. The experiments suggest that combining information from motion and objects at region and pixel-levels helps semantic image segmentation and motion segmentation. The algorithm has improved the semantic motion segmentation compared to the previous methods of scene understanding.

Furthur, We have proposed a joint labelling framework for semantic motion segmentation and reconstruction in dynamic urban environments. We modelled the problem of creating a semantic dense map of moving objects in a urban environment using trajectory optimization. The experiments suggest that semantic segmentation provide good initial estimates to aid generalized bundle adjustment based approach. This helps in improving the localization of the moving objects and creates an accurate semantic map.

We intend to extend the method by segmenting objects with different motion and segment each object as a different class. We also plan to achieve the GPU implementation for the proposed algorithm and generalize the current approach for scene understanding. We will continue expanding the annotations in the KITTI tracking dataset. The algorithm proposed is computed using a batch of images simultaneously. We are looking at the possibility of making the algorithm incremental. The Ground Truth annotation and evaluation criterion for moving object trajectory computation will be released.

Chapter 6

Related Publications

1. N Dinesh Reddy, Prateek Singhal and K Madhava Krishna **"Semantic Motion Segmentation Using Dense CRF Formulation"**. *In Proceedings of Indian Conference on Computer Vision, Graphics and Image Processing(ICVGIP)*,ORAL 2014.

2. N Dinesh Reddy, Prateek Singhal, Visesh Chari and K Madhava Krishna, **"Dynamic Body VSLAM using Semantic Constraints(IROS)**, 2015.

3. N Dinesh Reddy, Visesh Chari and K Madhava Krishna. **"Using Semantic Information for Segmentation, Localization and Tracking in Dynamic Environments**. Robotics and Autonomous Systems (RAS) - Elsevier Special Issue on Localization and Mapping in Challenging Environments, 2016 (Under Review).

Bibliography

[1] S. Agarwal, K. Mierle, and Others. Ceres solver. `http://ceres-solver.org`.

[2] chieh-chih Wang and T. han Lin. Deep learning of spatio-temporal features with geometric-based moving point detection for motion segmentation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[3] W. Choi, C. Pantofaru, and S. Savarese. Detecting and tracking people using an rgb-d camera via multiple detector fusion. In *CORP in conjunction with ICCV*, 2011.

[4] S. Choudhary, V. Indelman, H. I. Christensen, and F. Dellaert. Information-based reduced landmark SLAM. In *IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015*, pages 4620–4627, 2015.

[5] K. L. Clarkson and D. P. Woodruff. Low rank approximation and regression in input sparsity time. *CoRR*, abs/1207.6365, 2012.

[6] A. Dasgupta, R. Kumar, and T. Sarlós. A sparse johnson–lindenstrauss transform. *CoRR*, abs/1004.4240, 2010.

[7] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[8] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3d traffic scene understanding from movable platforms. 2014.

[9] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *In Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[10] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *International Conference on Computer Vision (ICCV)*, pages 1–8. IEEE, 2009.

[11] C. Hane, C. Zach, A. Cohen, R. Angst, and M. Pollefeys. Joint 3d scene reconstruction and class segmentation. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 97–104, 2013.

[12] H. Hirschmueller. Stereo processing by semiglobal matching and mutual information. 30:328–41, 2008.

[13] S.-J. Huang, Y. Yu, and Z.-H. Zhou. Multi-label hypothesis reuse. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 525–533, New York, NY, USA, 2012. ACM.

[14] P. Kohli, M. P. Kumar, and P. H. S. Torr. P & beyond: Move making algorithms for solving higher order functions. [14], pages 1645–1656.

[15] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. In *Conference on Computer Vision and Pattern Recognition (CVPR)* [15].

[16] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009.

[17] P. Krahenbuhl and V. Koltun. *Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials*. 2011.

[18] A. Kundu, K. M. Krishna, and C. V. Jawahar. Realtime multibody visual slam with a smoothly moving monocular camera. In *Computer Vision, 2011. ICCV 2011. IEEE International Conference on*, 2011.

[19] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical crfs for object class image segmentation. In *International Conference on Computer Vision (ICCV)* [19], pages 739–746.

[20] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Graph cut based inference with co-occurrence statistics. In *In European Conference on Computer Vision(ECCV)*, pages 239–253, 2010.

[21] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where and how many? combining object detectors and crfs. In *European Conference on Computer Vision(ECCV)*, pages 424–437, Berlin, Heidelberg, 2010. Springer-Verlag.

[22] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[23] P. Lenz, J. Ziegler, A. Geiger, and M. Roser. Sparse scene flow segmentation for moving object detection in urban environments. In *Intelligent Vehicles Symposium (IV)*, 2011.

[24] V. Lepetit, F.Moreno-Noguer, and P.Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal Computer Vision*, 81(2), 2009.

[25] R. K. Namdev, A. Kundu, K. M. Krishna, and C. V. Jawahar. Motion segmentation of multiple objects from a freely moving monocular camera. In *IEEE International Conference on Robotics and Automation (ICRA)* [25], pages 4092–4099.

[26] N. D. Reddy, P. Singhal, V. Chari, and K. M. Krishna. Dynamic body vslam with semantic constraints. *arXiv preprint arXiv:1504.07269*, 2015.

[27] N. D. Reddy, P. Singhal, and K. M. Krishna. Semantic motion segmentation using dense crf formulation. In *Proceedings of the 2014 Indian Conference on Computer Vision Graphics and Image Processing*, page 56. ACM, 2014.

[28] V. Romero-Cano, G. Agamennoni, and J. Nieto. A variational approach to simultaneous tracking and classification of multiple objects. In *Information Fusion (FUSION), 2014 17th International Conference on*, pages 1–8. IEEE, 2014.

[29] V. Romero-Cano and J. I. Nieto. Stereo-based motion detection and tracking from a moving platform. In *Intelligent Vehicles Symposium(IV)*, pages 499–504, 2013.

[30] A. Roussos, C. Russell, R. Garg, and L. Agapito. Dense multibody motion estimation and reconstruction from a handheld camera, 2012.

[31] S. Sengupta, E. Greveson, A. Shahrokni, and P. H. Torr. Urban 3d semantic modelling using stereo vision. *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[32] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *European Conference on Computer Vision(ECCV)*, pages 1–15, 2006.

[33] S. Song and M. Chandraker. Joint sfm and detection cues for monocular 3d localization in road scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3734–3742, 2015.

[34] B. soo Kim, M. Sun, P. Kohli, and S. Savarese. Relating things and stuff by high-order potential modeling. In *European Conference on Computer Vision(ECCV) Workshops*, pages 293–304, 2012.

[35] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[36] P. Torr, W. Clocksin, Y. Bastanlar, S. Sengupta, C. Russell, P. Sturgess, and L. Ladicky. Joint optimization for object class segmentation and dense stereo reconstruction. In *British machine vision conference(BMVC)*, 2010.

[37] J. P. C. Valentin, S. Sengupta, J. Warrell, A. Shahrokni, and P. H. S. Torr. Mesh based semantic modelling for indoor and outdoor scenes. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 2067–2074, 2013.

[38] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[39] V. Vineet, O. Miksik, M. Lidegaard, M. Nießner, S. Golodetz, V. A. Prisacariu, O. Kähler, D. W. Murray, S. Izadi, P. Perez, and P. H. S. Torr. Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[40] A. Wedel, A. Meißner, C. Rabe, U. Franke, and D. Cremers. Detection and segmentation of independently moving objects from dense scene flow. In *EMMCVPR*, pages 14–27, 2009.

[41] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *IEEE Intenational Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013.

[42] J. Xiao, A. Owens, and A. Torralba. SUN3D: A database of big spaces reconstructed using sfm and object labels. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 1625–1632, 2013.

[43] K. Yamaguchi, D. A. McAllester, and R. Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *European Conference on Computer Vision, In ECCV*, pages 756–771, 2014.

[44] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)* [44], pages 702–709.

[45] C. Yuan and G. Medioni. 3d reconstruction of background and objects moving on ground plane viewed from a moving camera. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 2261–2268, Washington, DC, USA, 2006. IEEE Computer Society.

[46] H. Zhang, A. Geiger, and R. Urtasun. Understanding high-level semantics by modeling traffic patterns. In *International Conference on Computer Vision (ICCV)*, 2013.

[47] S. Zheng, M.-M. Cheng, J. Warrell, P. Sturgess, V. Vineet, C. Rother, and P. H. S. Torr. Dense semantic image segmentation with objects and attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, Ohio, United States, 2014.

[48] Q.-Y. Zhou, S. Miller, and V. Koltun. Elastic fragments for dense scene reconstruction. In *Proc. of the International Conference on Computer Vision (ICCV)*, 2013.