Universitat Autònoma
de Barcelona

Optical Flow in Driver Assistance Systems

A dissertation submitted by **Naveen Onkarappa** at Universitat Autònoma de Barcelona to fulfil the degree of **Doctor en Informàtica**.

Bellaterra, September 2013

Director	**Dr. Angel D. Sappa** Computer Vision Center Autonomous University of Barcelona
Tutor	**Antonio M. López Peña** Dept. of Computer Science Autonomous University of Barcelona
Thesis Committe	**Dr. Michel Devy** Robotics, Action and Perception Group LAAS-CNRS **Dr. Miguel Ángel Sotelo Vázquez** Dept. of Automation University of Alcalá **Dr. Fadi Dornaika** Dept. of Computer Science and Artificial Intelligence University of Basque Country **Dr. Aura Hernàndez Sabaté** Computer Vision Center Autonomous University of Barcelona **Dr. Frederic Lerasle** Dept. of Electronics, Electrical Engineering and Automatic Control Paul Sabatier University
European Mention Evaluators	**Dr. Vitor M. Ferreira dos Santos** Dept. of Mechanical Engineering University of Aveiro **Dr. Georgios Tirantafyllidis** Dept. of Architecture, Design and Media Technology Aalborg University

This document was typeset by the author using LaTeX 2ε.

The research described in this book was carried out at the Computer Vision Center, Universitat Autònoma de Barcelona.

ISBN 978-84-940902-1-9

Printed by Ediciones Gráficas Rey, S.L.

To Appaji, Amma,
Prashu and Putti

Acknowledgments

Foremost, I am truly indebted and thankful to my adviser Dr. Angel D. Sappa first for providing me with an opportunity to do a PhD and for his guidance through out this research period without which this would not have come to this thesis. I would like to appreciate his patience, hard work, support and kind behavior.

I am grateful to my PhD scholarships funded by AGAUR of Generalitat of Catalunya and the project (CTP 2008ITT 00001), and to the mobility grant from the Spanish Ministry of Education. Also I would like to acknowledge other funding supports (Projects: TRA2007-62526/AUT, TRA2010-21371-C03-01, TIN2011- 25606 and research programme Consolider-Ingenio 2010: MIPRCV (CSD2007-00018)).

Thanks to the evaluation committee Dr. Michel Devy, Dr. Miguel Ángel Sotelo Vázquez, Dr. Fadi Dornaika, Dr. Aura Hernàndez Sabaté, Dr. Frederic Lerasle, and european mention evaluators Dr. Vitor M. Ferreira dos Santos and Dr. Georgios Tirantafyllidis for their time, comments and reports.

I extend my gratitude to Prof. Joachim Weickert for hosting me for the internship and Dr. Henning Zimmer for technical discussions. I would like to acknowledge Dr. Oisin Mac Aodha and all the people whose code I have used. I would like to thank all the editors and anonymous reviewers of my publications. Also I would like to acknowledge the help in big or small of Dr. Diego Cheda, Martin and also Patricia for translating the abstract to Catalan.

Thanks to all present and past members of ADAS group Dr. Antonio Lopez, Dr. Daniel Ponsa, Dr. Joan Serrat, Dr. Filipe Lumbras, Dr. David Geronimo, Dr. José M. Álvarez, Dr. Javier Marin, Dr. Jose Rubio, Dr. Mohammad Rouhani, Dr. Fernando Barrera, Monica and specially to Dr. David Vazquez and German Ros for all the discussions and support. I really thank the support of all the nice people in administrative and IT sections of CVC.

I would like to thank Dr. D. S. Guru back in India who introduced me to research. I would like to appreciate the wishes from my elderly friends Dr. Shivakumara and Bharathi madam.

Special thanks to friends in CVC specially Ariel, Bogdan, Murad, David Rojas, Antonio Clavelli for their help, coffee meetings, interesting discussions and for all the time we spent during these years. I would also like to thank my other friends outside CVC and flatmates for all the time we spent together.

I owe a great deal to my dear friends Mahesh, Dixon, Malli, Nagendra, Sharada, Elena, Gleb and Carlos who have always supported me, be there whenever I needed them and for all the time I enjoyed with them.

Last, but not the least, I am grateful and owe a lot to my family. Thanks to cousins and relatives for their wishes and blessings.

Abstract

Motion perception is one of the most important attributes of the human brain. Visual motion perception consists in inferring speed and direction of elements in a scene based on visual inputs. Analogously, computer vision is assisted by motion cues in the scene. Motion detection in computer vision is useful in solving problems such as segmentation, depth from motion, structure from motion, compression, navigation and many others. These problems are common in several applications, for instance, video surveillance, robot navigation and advanced driver assistance systems (ADAS). One of the most widely used techniques for motion detection is the optical flow estimation. The work in this thesis attempts to make optical flow suitable for the requirements and conditions of driving scenarios. In this context, a novel space-variant representation called reverse log-polar representation is proposed that is shown to be better than the traditional log-polar space-variant representation for ADAS. The space-variant representations reduce the amount of data to be processed. Another major contribution in this research is related to the analysis of the influence of specific characteristics from driving scenarios on the optical flow accuracy. Characteristics such as vehicle speed and road texture are considered in the aforementioned analysis. From this study, it is inferred that the regularization weight has to be adapted according to the required error measure and for different speeds and road textures. It is also shown that polar represented optical flow suits driving scenarios where predominant motion is translation. Due to the requirements of such a study and by the lack of needed datasets a new synthetic dataset is presented; it contains: i) sequences of different speeds and road textures in an urban scenario; ii) sequences with complex motion of an on-board camera; and iii) sequences with additional moving vehicles in the scene. The ground-truth optical flow is generated by the ray-tracing technique. Further, few applications of optical flow in ADAS are shown. Firstly, a robust RANSAC based technique to estimate horizon line is proposed. Then, an egomotion estimation is presented to compare the proposed space-variant representation with the classical one. As a final contribution, a modification in the regularization term is proposed that notably improves the results in the ADAS applications. This adaptation is evaluated using a state of the art optical flow technique. The experiments on a public dataset (KITTI) validate the advantages of using the proposed modification.

Resum

El moviment és un atribut perceptiu del cervell humà molt important. La percepció visual que fa el cervell del moviment és el procés d'inferir la velocitat i direcció dels elements d'un escenari mitjançant entrades visuals. Anàlogament, la visió per computador s'assisteix mitjançant informació del moviment de l'escena. En visió per computador, la detecció de moviment és útil per a resoldre problemes com per exemple segmentació, estimació de la profunditat, estimació de l'estructura a partir del moviment, compressió de dades o navegació entre d'altres. Aquests problemes són comuns a diferents aplicacions, com ara vídeo vigilància, navegació de robots i sistemes avançats d'assistència a la conducció (Advanced Driver Assistance Systems, ADAS). Una de les tècniques més utilitzades per a detectar moviment, és el càlcul d'*optical flow*. El treball tractat en aquesta tesi pretén que les formulacions d'*optical flow* siguin més apropiades als requeriments i condicions dels escenaris de conducció. En aquest context, es proposa una nova representació de l'espai-variant anomenada representació reverse log-polar, i es demostra que, quan s'utilitza per a ADAS, té un rendiment millor que la tradicional representació log-polar. La representació espai-variant redueix la quantitat de dades necessàries que han de ser processades. Una altra contribució important està relacionada amb l'anàlisi de la influència de les característiques específiques d'escenaris de conducció per a la precisió de l'*optical flow*. S'han considerat característiques tals com la velocitat del vehicle i la textura de la carretera. D'aquest estudi s'infereix que, el pes del terme de regularització s'ha d'adaptar segons una mesura d'error i per a diferents velocitats i textures de la carretera. També es mostra que la representació polar d'*optical flow* funciona molt millor per a escenaris de conducció on el moviment principal són translacions. Degut als requeriments d'aquest estudi, i per la manca de bases de dades es presenta una nova base de dades sintètica que conté: i) seqüències amb diferents velocitats i textures en un escenari urbà; ii) seqüències amb moviments complexos de la càmera col·locada al vehicle; i iii) seqüències amb altres vehicles en moviment dintre la mateixa escena. L'*optical flow* corresponent a cada seqüència s'obté mitjançant la tècnica de *ray-tracing*. A més a més, es presenten algunes aplicacions per a *optical flow* en escenaris ADAS. Per començar, proposem una tècnica robusta basada en RANSAC per estimar la línia de l'horitzó. Després, presentem una estimació de l'*egomotion* per a comparar la representació espai-variant proposada amb les representacions clàssiques. Com a contribució final, es proposa una modificació del terme de regularització que millora notablement els resultats per a aplicacions d'ADAS. Aquesta adaptació s'avalua mitjançant tècniques d'*optical flow* d'última generació. Els experiments realitzats amb una base de dades pública (KITTI) validen els avantatges d'utilitzar la modificació proposada.

Resumen

La percepción del movimiento es uno de los más importantes atributos del cerebro humano. La percepción visual del movimiento consiste en inferir velocidad y dirección de los elementos móviles que interactúan en una escena, mediante la interpretación de diferentes entradas visuales. Análogamente, la visión por computador hace uso de la información del movimiento en la escena. La detección de movimiento en visión por computador es útil para resolver problemas tales como: segmentación, estimación de profundidad, compresión, navegación, entre otros. Estos problemas son comunes en distintas aplicaciones, por ejemplo: video vigilancia, navegación de robots y sistemas avanzados de asistencia a la conducción (ADAS). Una de las técnicas más utilizadas para detectar movimiento es la estimación del flujo óptico. El trabajo abordado en esta tesis busca formulaciones del flujo óptico más adecuadas a las necesidades y condiciones de los escenarios de conducción. En este contexto, se propuso una novedosa representación del espacio, llamada representación *inversa log-polar*, la cual se demuestra que tiene un desempeño mejor que la tradicional representación log-polar para aplicaciones ADAS. Las representaciones de espacio-variante reducen la cantidad de datos a ser procesados. Otra contribución importante está relacionada con el análisis de la influencia de las características específicas de los escenarios de conducción en la precisión del flujo óptico estimado. Características tales como la velocidad del vehículo y la textura de la carretera son consideradas en el estudio. De este estudio, se infiere que el peso del término de regularización tiene que ser adaptado de acuerdo con la medida de error requerida y para diferentes velocidades y texturas de la carretera. También se concluye que la representación polar del flujo óptico es la más apropiada en escenarios de conducción, donde el movimiento predominante es la translación. Debido a las exigencias de tal estudio, y por falta de las bases de datos necesarias, se presenta un nuevo conjunto de datos sintéticos el cual contiene: i) secuencias de diferentes velocidades y texturas en un escenario urbano; ii) secuencias con movimientos complejos de la cámara dispuesta en el vehículo; y iii) secuencias con otros vehículos en movimiento en la escena. El flujo óptico correspondiente a cada secuencia es obtenido mediante una técnica de *ray-tracing*. Adicionalmente, se presentan algunas aplicaciones de flujo óptico en ADAS. Primeramente se propone una técnica robusta basada en RANSAC para estimar la línea de horizonte. Seguidamente se presenta una estimación del *egomotion* para comparar la representación de espacio-variante propuesta con los esquemas clásicos. Como contribución final, se propone una modificación en el término de regularización que mejora notablemente los resultados en las aplicaciones ADAS. Los resultados experimentales en una base de datos pública (KITTI) validan las ventajas de la utilización de la modificación propuesta.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The developments in human living during the last decades have led to increased movements that have made the world small with respect to reachability. It has resulted in continuous increase in mobility. This has raised the number of road accidents, questioning life and safety of human beings. Hence there is a concern and continuous effort to make mobility safe, easy and cheap. One way to increase safety is to build better and easier driving infrastructures (such as roads, signals and sign boards), and to improve driving vehicles (e.g., by airbags, breaking systems). Despite these, the main cause of accidents is human driver errors. Human drivers by nature are equipped with eyes which work as stereo cameras and processing and decision making unit brain. Although this human vision system is the best in performance in difficult situations, accidents can happen due to driver's distraction and un-attentiveness as humans can get tired.

In recent decades computer vision has emerged as a significant technology providing solutions to wide areas that affects / improves human life. Medical imaging, surveillance, industrial automation, entertainment, to name but a few broader areas where computer vision technology is being researched and applied. Many researchers and engineers from academia and industry are exploring the ways and trying to use computer vision technology for the safety of automobiles and human life. The basic idea is to use vision sensors (cameras) to acquire images / video frames and to process them in order to help the driver by means of warning or by taking automatic control of the vehicle. We call such systems advanced driver assistance systems (ADAS). Figure 1.1 shows a car fitted with a monocular camera. ADAS using computer vision include applications such as egomotion estimation, lane departure warning, collision detection and avoidance, blind spot detection, parking assistance, driver alertness checking, obstacle detection, and in a broader view, environment perception and autonomous driving. Many of these ADAS applications make use of motion information. Perception of motion is an important function of the human visual system. The estimation of motion information, which is the prominent source of temporal variations in image sequences, is one of the key problems in computer vision.

Motion in image sequences acquired by a video camera is induced both by movements of objects in a three-dimensional scene and by camera motion. A well-known motion estimation tool is optical flow estimation. Horn and Schunck [46] define motion field as a pure geometric concept; without any ambiguity, it is the projection of three-dimensional motion

Figure 1.1: On-board camera.

vectors onto a two-dimensional image. In other words, optical flow is the pattern of apparent motion in a visual scene caused by the relative motion between an observer and the scene. On the other hand, optical flow is defined as the velocity field in the image, which transforms one image into the next image in a sequence. The motion estimation, in particular the optical flow techniques are being investigated and used for many applications such as human action recognition, structure from motion, autonomous navigation, just to mention but a few. In addition to the extensions of optical flow towards different applications, a continuous effort to improve the accuracy can be observed in the literature. Even though the use of optical flow dates back to decades, considering its importance in different applications, it is still one of the hot research topics in computer vision. The literature in this topic suggests that variational techniques produce accurate motion estimates while being able to perform in realtime recently. Hence the research in this field is focused more on variational techniques which give dense flow fields. Figure 1.2 shows an image from a pair and the computed optical flow (the colormap used to depict flow field is shown in Fig. 2.4).

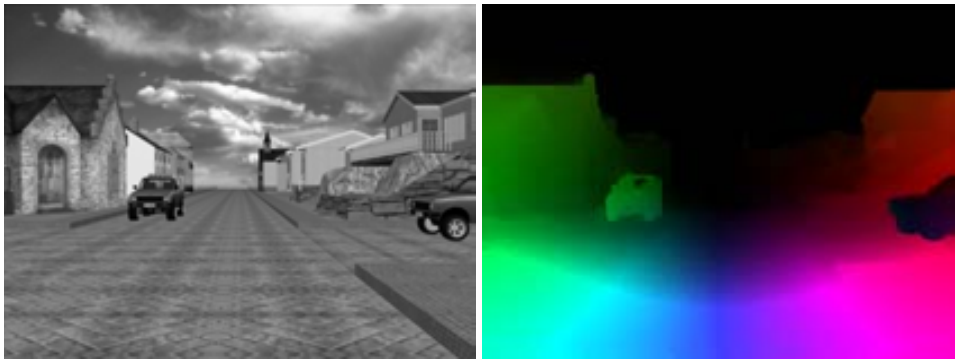


Figure 1.2: (*left*) First image from a pair. (*right*) Corresponding computed optical flow.

The importance of ADAS in improving safety of humans and vehicles, and the vitalness of motion information in making ADAS, have motivated us to consider the research in this direction. Our research concentrates on optical flow estimation specifically in driving scenarios towards making it more accurate and fast, and its affectedness by the characteristics of a driving scene. Also, in this dissertation we present some of the applications of optical flow in ADAS.

1.1 Motivation and contributions

Traditional computer vision represents an image by the Cartesian co-ordinate system typically with uniform sampling in both x and y axes. Unlike computer vision, biological vision (human eye) has higher sampling at the center where it is gazing and reduced resolution at the surround. This phenomenon can be well defined through log-polar sampling and is generally called a space-variant representation. One of the major advantages of space-variant representations is reduction in the data to be processed. Since optical flow estimation is a very time consuming task, space-variant representation motivates us to enable faster / real-time ADAS applications which involve optical flow. Motivated by the advantage of the space-variant representation, the need of faster computation in ADAS domain has resulted in the proposal of a novel space-variant representation that we refer as the Reverse Log-Polar representation. The proposed space-variant representation has the similar advantage of reduction in data and being more rich in information for ADAS.

In a driving scenario, a vehicle might be driven in a variety of roads, speeds, daylight conditions, seasons, environments (such as urban, highway, countryside). Thus, the situation while driving is unpredictable with added complexity, e.g., moving vehicles and pedestrians in the scene. Figure 1.3 shows a few images of real and synthetic sequences of different driving scenarios. Accuracy of the optical flow in any of such scenarios is expected to be good enough for reliability of ADAS. Having different algorithms for different scenarios is not a feasible option for real-time applications. Hence, there is a need for an optical flow algorithm that gives reasonable accuracy in all the scenarios. This motivates to tune the algorithm for different characteristics of a driving scene. There are no such previous studies in the literature. In this research, we study the influence of different speeds and different road textures on optical flow accuracy. For this analysis we choose an optical flow algorithm that uses polar representation of flow vectors. The formulation of this algorithm provides the advantage of having different regularization on different polar coordinates. In this research we also show that such a representation makes sense on flow fields in driving scenarios.

There exists no suitable dataset for the above mentioned analysis. Obtaining sequences of driving scenarios with exactly the same geometrical and textural scene but with different speeds and road textures is impossible unless done in a laboratory environment with limitations. This drives us to develop a 3D urban scenario and to generate required driving sequences with corresponding ground-truth flow fields for the analysis. Thus we have developed a synthetic dataset of an urban scenario containing several sequences of different speeds and road textures with ground-truth data.

Having made attempts to reduce the amount of information to be processed and to analyze the influence of the driving scene characteristics on optical flow accuracy, further we have made an effort to use it for some of the ADAS applications. First, we propose a RANSAC

Figure 1.3: (*top*) Real images with moving vehicles and shadows. (*bottom*) Synthetic images with different road textures and scenarios (forest and urban).

based robust vanishing point estimation technique using dense optical flow. In turn, the horizon line is also estimated. This horizon line can be used to limit the searching space for pedestrian detection or any moving object detection in a road. Furthermore, we present an egomotion estimation application using the optical flow computed on space-variant represented images. It is shown by this application that optical flow on the proposed space-variant representation is more accurate for ADAS applications than that on the log-polar one.

In addition to all of the above efforts on optical flow and its applications, we have made an attempt to adapt an existing state of the art optical flow approach to driving scenarios. In a driving scenario, a scene may contain complexities such as large displacements, specularities, reflections etc. A typical optical flow formulation involves the gradient of flow components in the regularization part. For large variations such as large displacements, the Laplacian of flow components is used which notably improves the results. The experiment on a real dataset demonstrates the improvement in results.

In summary, the work in this thesis contributes to the following topics:

- A novel space variant representation is proposed. Unlike the traditional log-polar representation, it is suitable for ADAS applications.

- A synthetic urban driving scenario dataset with optical flow ground-truth is developed. The dataset consists of sequences of different speeds, different road textures, and with

added complexities of camera motions and moving vehicles in the scene.

- It is demonstrated that in driving scenarios the polar represented optical flow formulation exhibits statistical independence of the flow components upon which regularization can be varied independently.
- Analysis of variations of optical flow accuracy for different speeds and road textures is performed and tuning weights of regularization to improve the results are suggested.
- A few applications of optical flow in ADAS such as horizon line estimation and egomotion estimation are presented.
- A modification to the regularization term, based on the use of the Laplacian derivatives of the flow components, is proposed; and it is shown that the results are notably better in ADAS scenarios.

1.2 Thesis outline

The rest of the thesis is organized as follows. Chapter 2 presents the related work on optical flow techniques, different datasets, space-variant image representations and ADAS applications. This chapter also presents the basic formulation, different error measures and the colormap used to depict optical flow in the whole thesis. Chapter 3 describes the proposed novel space variant image representation. In Chapter 4 the framework to generate synthetic dataset along with ground-truth flow fields is presented. Chapter 5 presents an empirical study on the influence of several driving scene characteristics on optical flow accuracy. A few optical flow applications in ADAS are given in Chapter 6 in addition to the proposed adaptation of an optical flow technique to ADAS. Finally, Chapter 7 concludes the thesis with directions for the future research.

Chapter 2

Related Work

2.1 Introduction

This chapter details the state of the art of the different topics tackled during this research. First it introduces optical flow, the formulation of basic variational optical flow and presents the related work in the different areas of optical flow. Later it describes the related work on space-variant representations. Also it presents several datasets available in the optical flow community. Finally, related work on some of the optical flow applications in ADAS are given.

2.2 Optical flow

Even though many attempts have been made to estimate motion since a long time, concrete formulations are proposed by Horn and Schunck [46], and Lucas and Kanade [56] in 1981. Generally, optical flow methods can be classified as feature based methods, area based methods, phase based methods and gradient based methods. Feature-based methods seek correspondences for sparse image features such as edges or corners (e.g., [28, 101]). Whereas area-based approaches rely on matching complete image patches by aggregating local information (e.g., [8, 81]). Phase-based approaches make use of velocity-tuned filters in the Fourier domain (e.g., [40, 36]). Gradient based methods can be classified as local methods and global methods. Local methods optimize local energy functions while global approaches, also referred to as variational approaches, attempt to minimize a global energy functional. Local differential methods that are based on the same constancy assumptions as variational techniques but minimize local energy-like expressions (e.g., [56, 47]). The main advantage of variational techniques is that they generate dense flow field and as manifested in recent literature on optical flow (e.g., [61, 26, 73, 107, 97, 110]), the variational techniques offer a high accuracy. In this thesis we focus on variational optic flow approaches that give currently the best results in the literature. Some of the key developments in the optical flow research are shown in the timeline in Fig. 2.1.

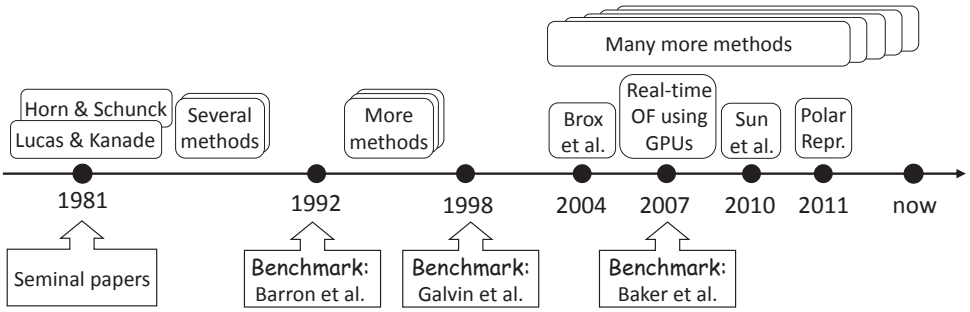


Figure 2.1: Optical flow research timeline.

Variational optical flow

The classical variational optical flow method has been first proposed by Horn and Schunck [46]. It is based on two assumptions: the brightness constancy assumption (BCA) and the homogeneous regularization. The first one, which is also called as optical flow constraint, assumes that the gray value of objects remains constant over time; the second one assumes that the resulting flow field varies smoothly all over the image. The BCA can be formulated as:

$$I_1(\boldsymbol{x}+\boldsymbol{u(x)}) - I_0(\boldsymbol{x}) = 0, \tag{2.1}$$

where I_0 and I_1 forms an image pair, $\boldsymbol{x} = (x_1, x_2)^T$ is the pixel location within a rectangular image domain $\Omega \subseteq \mathbf{R}^2$; $\boldsymbol{u} = (u_1(\boldsymbol{x}), u_2(\boldsymbol{x}))^T$ is the two-dimensional displacement field. Linearising the above equation using first-order Taylor expansion we get optical flow constraint as:

$$I_{x_1} u_1 + I_{x_2} u_2 + I_t = 0, \tag{2.2}$$

where subscripts denote the partial derivatives. Since optical flow is a highly ill-posed inverse problem, using only local intensity constraints does not provide enough information to infer meaningful flow fields. In particular, optical flow computation suffers from two problems: first, no information is available in un-textured regions. Second, one can only compute the normal flow (i.e., the motion perpendicular to the edges). This problem is generally known as the aperture problem. In order to solve this problem it is clear that some kind of regularization is needed. The Horn and Schunk [46] method overcomes this by assuming the resulting flow field is globally smooth all over the image. This can be formulated as penalizing large spatial flow gradients ∇u_1 and ∇u_2. Combining BCA and smoothness assumptions in a single variational framework and squaring both constraints in order to penalize negative and positive deviations in the same way, the following energy functional is obtained:

$$E(\boldsymbol{u}) = \int_\Omega \{ \underbrace{(I_{x_1} u_1 + I_{x_2} u_2 + I_t)^2}_{\textit{Data Term}} + \alpha \underbrace{(|\nabla u_1|^2 + |\nabla u_2|^2)}_{\textit{Regularization}} \} \, d\boldsymbol{x}, \tag{2.3}$$

where α is a regularization parameter.

Since the proposal of first variational method in 1981 [46], a lot of research has been carried out to improve the performance of such techniques. As the variational models consist of data term and regularization term (also referred to as smoothness term), the developments in corresponding concepts are discussed further separately. Also the approaches proposed for solving these formulations are discussed.

Data term:

Regarding the data term in the variational approaches, there are two fields of research in the literature, which are robust data terms that use non-quadratic penaliser functions to improve the performance in the presence of outliers in the image data (e.g., [17, 16, 61]) and modified constraints that allow for a more accurate estimation in the case of varying illumination, large displacements and noise (e.g., [65, 79, 7, 26]).

Robust data terms: Black and Anandan [17, 16] suggested the use of M-estimators from robust statistics. As compared to quadratic penalizer proposed in [46], these functions penalize outliers less severely and thus reduce the influence of corrupted data on the result. The usefulness of M-estimators is also explored by Memin and Perez [61], but solves the resulting non-convex optimization problem by an iteratively weighted least square method. Aaubert et al. [10] propose a robust L1 norm in the data fidelity term and Zack et al. [107] uses dual formulation to solve energy function with L1 data term. Recently, Govindu [43] presents a probabilistic formulation for brightness constraints and demonstrated its superiority over previous methods.

Modified constraints: Many ideas have been proposed in the literature regarding modifying the constraints in the data term. Schnorr [79] proposes suitable constraints with respect to a changing image brightness. Nagel [63] proposes the approximation via a second order Taylor expansion for more accurate estimation of small displacements. Nagel and Enkelmen [65] and Alvarez et al. [7] propose to use non-linear constancy assumption for estimating large displacements. Bruhn et al. [26] propose the integration of local least square fit to improve the performance with respect to noise. The use of blended texture component of the image have been proposed by Wedel et al. [97] to get rid of violations in the optical flow constraints due to illumination changes. Papenberg et al. [20, 73] propose higher-order derivatives based constancy assumption addressing additive or multiplicative illumination changes. The concept of separate robustification, which employs a robust penalizer function to each of the constancy assumptions, has been proposed [26]. The work in [110] uses HSV color space for multiplicative illumination changes and in particular under shadow and shading.

Modified and other data constrainsts such as higher order terms or by using several channels overcomes the violation in constancy assumptions due to illumination changes or different motion patterns (rotational, translational or divergent). Whereas robust penalizers penalyze the variations apropriately, but non-convexity of the energy functions makes it hard to minimize.

Smoothness term:

Regarding smoothness term the research in literature is focused on preserving the motion discontinuities (e.g., [65, 45, 67, 75, 99]) and the integration of temporal information (e.g., [64, 17, 100]). The approaches in these categories are presented below.

Discontinuity-Preserving smoothness terms: The first approach to adapt regularization has been proposed by Nagel [65]; it orients smoothness constraints inhibiting the filling-in-effect across image discontinuities. Heitz and Bouthemy [45], and Nesi [67] propose alternative approaches based on the use of robust statistics in the context of discrete variational approaches. These methods penalize the smoothness term less severely as compared to Horn and Schunck, thus allowing discontinuities in the flow field. Evolving flow-field driven non-quadratic regularisers have been proposed by Wickert and Schnorr [99] in the context of continuous formulation of variational techniques. Their work classifies continuous smoothness terms based on their induced diffusion process such as isotropy, anisotropy in combination with image features and flow features. Total variation based regularization is also presented in [107, 97], where they use a dual formulation to solve it. A structure and motion adaptive regularization for high accuracy optical flow is also proposed by Wedel et al. [96] that improves the results using the rigid motion information.

Spatiotemporal regularizers: Murray and Buxton [62] first propose a spatio-temporal smoothness term that use a discrete optical flow method based on spatio-temporal Markov random fields in order to estimate multiple flow fields simultaneously. Later Nagel extends his previous work toward spatio-temporal smoothness term [64]. Black and Anandan [17] propose to use computed flow estimates as prior knowledge to obtain temporally piecewise homogeneous motion field. Later, spatiotemporal concept is adopted to discontinuity-preserving regularisers by Weickert and Schnorr [100] and in Papenberg et al. [73]. These smoothness terms preserve both spatial and temporal discontinuities in the unknown flow field.

Regularization term is a necessary term to make the optical flow constraint solvable. Homogeneous regularization makes the flow field smooth. Therefore, similar to the robust penalization of data terms, robust penalizing functions of flow derivatives in regularization term preserves flow edges. The direction of penalization can be driven by flow and image features. Further, the same smoothness assumption on temporal axis improves the results.

Minimization / Solvers:

Variational optical flow energy functions can be minimized in several of ways. The most used way is to express and solve the set of Euler-Lagrange equations of the energy model. The thesis [23] presents various numerical linear and non-linear equation system solvers such as basic Gauss-Seidel method, its variants, advanced methods such as successive over-relaxation (SOR) techniques, unidirectional multigrid methods in the form of coarse-to-fine strategies and bidirectional multigrid methods. Brox et al. [20] propose to solve the energy function keeping the data term as nonlinear and solving the Euler-Lagrange equations using nested fixed point iterative methods and SOR. Bruhn et al. [24, 25] experiments various solvers on many variational techniques and demonstrates the real-time performance of multigrid methods. Recently Zach et al. [107] propose a dual formulation based on iterative alternating steps to solve TV-L1 optical flow energy model. Further they also proposed a numerical scheme to implement it on graphic processing units with real-time performance. In summary, the minimization approach also matters for the optical flow as the complex modelling might result into non-convexity of the terms. Hence, proper selection of scale in multi-resolution computation and number of iterations needs consideration for accurate optical flow. Further use of multigrid methods and GPUs improves the efficiency.

Recent advances:

Apart from the discussions in the previous sections about efforts to improve data terms and smoothness terms separately, here we discuss few very recent attempts, which involve either: both data and smoothness terms; or attempts to adapt to a particular application; or a different interpretation; or it makes use of a different concept into the variational framework. Wedel et al. [97] improve the TV-L1 optical flow proposed in [107] by using blended version of the image texture component and also used a median filter to reject outliers. Zimmer et al. [110] propose an optical flow technique which is the rank one in [1] as on that day with regard to angular error and it is based on the concept of complementarity between data term and smoothness term. Here the data term incorporates HSV color channels with higher-order constancy assumptions, separate robust penalization with constraint normalization, while the anisotropic smoothness term reduces smoothing in the data constraint direction instead of image edge direction, thus enforcing a filling-in effect. Structure and motion adaptive regularization [96] proposed by Wedel et al. was at rank two according to the average angular error (AAE), but rank one according to end-point error during that time when evaluated at [1]. A segmentation based variational model for accurate optical flow estimation is proposed by Xu [103]; each segment in the image is constrained by an affine motion and a confidence map is used for global minimization. Brox et al. [22] propose hierarchical region-mapping based variational optical flow techniques for large displacements. A variational method to segment the image motion in parallel to optical flow computation is proposed by Brox et al. [21]. This method uses level set framework following the idea of motion competition. Sun et al. [84] propose to learn statistics from both ground-truth optical flow and brightness constancy to formulate a fully learned probabilistic model for optical flow estimation. A method combining the advantages of both local and global approaches is proposed in [26].

Recently, Sun et al. [83] explore the concepts such as pre-processing, coarse-to-fine warping, graduated non-convexity, interpolation, derivatives, robustness of penalty functions and median filtering, and then their influence on optical flow accuracy is revealed. Using the best of the explored concepts and weighted non-local median filtering, an improved model is proposed in [83]. A multiframe optical flow estimation technique based on temporal coherence of flow vectors across image frames is proposed in [94]. It is based on the assumption that optical flow is smooth along motion trajectories. The authors proposed multiframe spatial regularization and also higher order trajectorial regularization and it is shown that their method performs better than many state of the art methods. Motivated by the natural representation of a vector and the statistical independence of the polar coordinates to the cartesian coordinates, recently, an optical flow estimation approach based on polar representation is proposed in [4]. The polar represented optical flow estimation [4] involves two regularization terms: one of orientation and the other of magnitude. This formulation gives the advantage of tuning each one of them independently unlike in cartesian represented optical flow estimation. One of the top performing method presently in [1] is proposed in [104]. The method intelligently preserves small and large motion in a coarse-to-fine approach that was a common problem with all previous coarse-to-fine approaches.

One can notice that in almost all optical flow methods, the weights for regularization is empirically chosen. There are very few attempts in this direction to automatically select such parameters. Krajsek et al. [49] presents a Bayesian model that automatically weights different data terms and a regularization term. This model, that estimates optical flow and

several parameters together, is very complex to minimize. Recently, Zimmer et al. [110] propose to automatically select the regularization weight based on the *optimal prediction principle*. In their work the optimal regularization weight is obtained as the one that can produce a flow field with which the next frame in a sequence is best predicted. Inherently, this approach involves a brute-force method to select the optimal weight based on average data constancy error and hence it is computationally expensive. On the other hand, there is an attempt [57] to use several different optical flow methods for a sequence, by selecting the best suitable method per pair of frames or per pixel.

During last three decades plenty of approaches on optical flow estimation have been proposed to improve the accuracy and efficiency, and there has been increased interest on optical flow approaches during last years, which can be appreciated on the number of publications and released codes [1]. Despite the volume of research on the topic of optical flow, few attempts have been made to empirically evaluate the performance of optical flow algorithms on complex image sequences. A major obstacle in performing any empirical study in computer vision is obtaining ground-truth data. The major previous works in the area of quantitative optical flow evaluation has been the work of Barron et al. [13] and Otte and Nagel [72]. Barron et al. conduct an empirical and qualitative analysis of nine optical flow algorithms. The algorithms are tested on five synthetic image sequences where ground-truth motion fields were available and four real image sequences for which no ground-truth motion field were available. Otte and Nagel [72] present a good evaluation of their optical flow algorithm by comparing it against others using one of the synthetic sequences from Barron et al. Their paper is significant because they actually measured the ground-truth motion field for their real sequence and have made the sequence and the motion field publicly available. Galvin et al. [41] evaluate eight different optical flow algorithms among those, two approaches are other than those evaluated by Barron et al. Presently the vision group at Middlebury [1] maintain evaluation results of almost all optical flow techniques including those to be published yet. The different approaches are rank ordered separately based on different error measures by experimenting on various sequences with ground-truth. Recently McCane et al. [59] propose benchmarking suite of image sequences and tools for the purpose of evaluating optical flow algorithms. They provide a comprehensive set of complex synthetic scenes, a simple interactive method for extracting ground-truth motion from real polyhedral scenes, and also three new ground-truth motion data sets from such scenes. Baker et al. [11] has recently proposed few sequences with ground-truth and evaluation methodology. The most commonly used error measures to compare optical flow techniques are average angular error (AAE) [13] and average end-point error (EPE) [59]. Other error measures in the literature are interpolation error and normalised interpolation error [1]. EPE is the absolute distance between the end-points of estimated and ground-truth flow vectors. According to [13], the angular error e between two vectors (u_1, v_1) and (u_2, v_2) is given by:

$$e((u_1,v_1),(u_2,v_2)) = \arccos\left(\frac{u_1u_2 + v_1v_2 + 1}{\sqrt{(u_1^2+v_1^2+1)(u_2^2+v_2^2+1)}}\right). \tag{2.4}$$

2.3 Space-variant vision

Current technology allows taking larger and larger images; at the same time computational power is also increasing, being able to process large amount of information in short period of time. Unfortunately, the trade-off between computation time and amount of information to process is still there and needs specific solutions. One way to tackle this challenging problem is by means of the use of space-variant representations, which have been motivated by the mammalian biological vision systems [18], [80], and allows to reduce the amount of information to be processed. The mammalian retina is a space-variant sensor with higher sensory neurons at fovea and gradually reducing towards periphery. This leads to high resolution at the center and wide field of view simultaneously without many sensing elements in the periphery. There have been many attempts to design space-variant imaging sensors [91] as well as approaches based on such representations for computer vision applications (e.g., [95, 38]). On the other hand, space-variant represented images can be obtained from the conventional rectangular sensors by space-variant sampling and can have the same advantage of space-variant sensors. A typical space-variant log-polar sampling is shown in Fig. 2.2. The distance between consecutive samples along radial axis increases along the space from the center to the periphery. The Log-Polar Representation (LPR) of images has been originated from such biological vision and has been extensively studied in the computer vision community. Additionally, it has been also exploited in robotics and active vision communities for pattern recognition [89] and navigation [32] tasks. The LPR has many advantages with respect to the conventional cartesian representation of images [88]; the most important are the reduction in the data and the invariance to scale and rotation. The data reduction due to the polar mapping and logarithmic sub-sampling leads to high resolution in the fovea and low resolution in the periphery, which is a desired feature for instance in the active vision community. The importance of loss of information in LPR due to reduced resolution in the periphery depends on the application of interest. Figure 2.3 shows the log-polar sampled image of a cartesian image, even though it look like a disk cuted from Fig. 2.3 (*left*), this polar representation contains higher sampling in the center and lower in the periphery.

A review of log-polar imaging is presented in [91] for robotic vision applications such as: visual attention, target tracking and 3D perception. All these applications benefit from the high resolution of the fovea region. Also, there have been attempts to use LPRs for motion analysis (e.g., [32, 90, 19]), mainly based on the estimation of optical flow (e.g., [33, 34, 12, 92]). For instance, [88] presents the advantages of polar and log-polar mapping to the cartesian representation and proposes a technique to estimate time-to-impact using optical flow. Log-polar mapping and its properties are described in [34]; and it proposes the computation of normal optical flow. In [105], a novel optical flow computation approach is proposed. It is based on the concept of variable window and generalized dynamic image model. The variable window adapts its size along the LP space. Also working in the LP space, [33] analyzes the polar deformation and proposes several local optical flow estimation techniques on log-polar plane. Based on LPR, an approach for active monocular fixation is proposed in [106] and an application of docking using rotational-component of log-polar optical flow is proposed in [12]. The authors [90] proposed a method to estimate motion by defining the projections in log-polar represented images. The comparision of this result to the cartesian concludes that the error is due to motion mapping model and not due to the distortions of space-variant representation. An approach to track motion that is independent

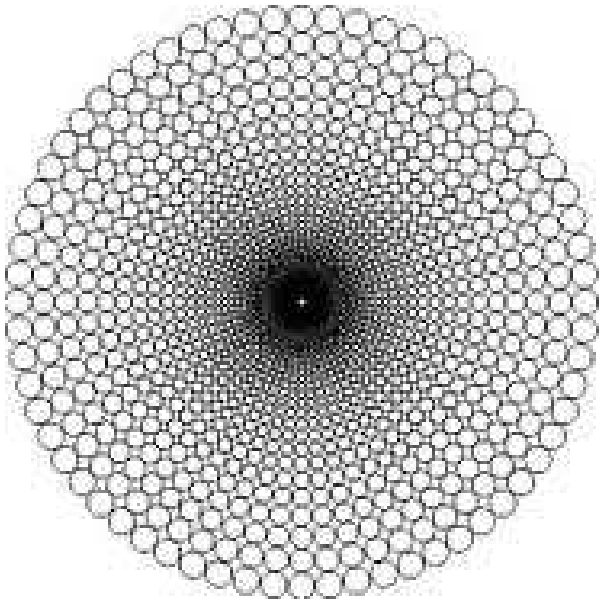


Figure 2.2: A typical log-polar sampling.

Figure 2.3: An example of log-polar sampling.

of camera motion is demonstrated on log-polar images in [19]. It has proved to be useful in the case of a camera with the translation in the optical axis and/or rotation around it. The work in [92] propose a Least-Square based method to estimate optical flow on Log-polar sampled images. The optical flow approaches presented in [33, 34, 12, 88] and [105] are all local approaches. However the dense flow fields would be more useful for driver assistance applications such as obstacle detection, egomotion estimation and collision avoidance.

2.4 Datasets

There are several datasets (e.g., [11, 27, 42, 55, 13, 93]) for optical flow available today. The most widely used and well-known dataset is Middlebury [11]. This dataset contains both real and synthetic sequences. The ground-truth for real sequences is generated with hidden texture in a controlled laboratory environment. Most of the contributions on optical flow are evaluated on this dataset [1], which contains limited scenarios and image pairs with small displacements. Few images and ground-truth flow fields of this dataset are shown in Fig. 2.5. All the flow fields in this thesis are depicted using the colormap shown in Fig. 2.4 where color corresponds to the vector orientation and intensity to the corresponding magnitude. One of the first performance evaluation work has been done in [13] while presenting a few synthetic sequences with an evaluation methodology. These sequences are very simplistic. Later, McCane et al. [59] introduce several complex synthetic sequences and also compared several optical flow methods on them. A real sequence is provided by Liu et al. [55]. In this work the authors also present dense ground-truth data. They annotate an image into different layered segments and compute optical flow using existing methods for layers separately. Obviously, this process adds several errors into the ground-truth. Few real sequences were proposed by Otte and Nagel [71]. These sequences are also simplistic with geometry, texture and small displacements. A big challenge when real dataset with realistic scenarios need to be obtained lies on the difficulty in obtaining ground-truth optical flow. Recently, Geiger et. al [42] have proposed a new real dataset of driving scenarios containing large displacements, specularity, shadows and different illuminations. They have also provided sparse ground-truth flow fields with a density of around 50%. This dataset is referred to as KITTI. Figure 2.6 shows an image from a pair with its corresponding ground-truth flow from KITTI dataset. Although this dataset contains data from the real cameras, it does not provide dense ground-truth and the ground-truth is not accurate due to errors on the registration of laser scanner point clouds to the image plane. The work in [93] introduces few synthetic sequences of driving scenarios with ground-truth optical flow. Sample images and ground-truth flow fields are shown in Fig. 2.7. The ground-truth flow fields in these sequences do not show the occlusion areas when there are moving vehicles in the scene. The authors point out that the algorithms which perform better on synthetic may not perform better on real sequences and vice-versa. Also a set of simple sequences are provided by [58]. In this work, the authors attempts to find the best suitable optical flow algorithm based on the flow confidence at every pixel.

Considering the drawbacks of the existing datasets, recently Butler et al. [27] present a large synthetic dataset from the animated short film Sintel. This dataset is referred to as MPI-Sintel. They have incorporated several complexities such as motion blur, defocus blur, shading and atmospheric effects. The dataset contains the same image sequences with three levels of complexities. This dataset is expected to be a new benchmark for optical flow re-

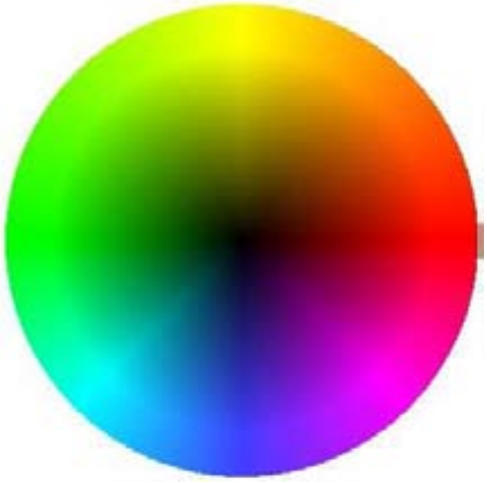


Figure 2.4: Colormap to display flow field.

search [3]. Figure 2.8 shows the few highest complexity images and the ground-truth flow fields. These datasets ([11, 27], [42]) have separate training and evaluation sets. The evaluation set does not provide ground-truth flow data to the public for the purpose of evaluation. One can think that the state of the art methods that give better results on Middlebury dataset can also perform similarly on KITTI dataset and MPI-Sintel. However, by analyzing the KITTI flow evaluation [2] and Sintel evaluation [3] one can appreciate that such a statement is wrong due to the difficulties of the particular datasets. This proves that a diverse collection of datasets will take forward research to new levels. Among the existing datasets, the sequences from [42] and [93] are for ADAS.

2.5 Applications

During the last decade on-board vision has gained popularity in the automotive applications due to the increase of traffic accidents in modern age. According to the World Health Organization, every year almost 1.2 million people are killed and 50 million are injured in traffic accidents worldwide [74]. A key solution to this is the use of on-board intelligent vision systems that are able to predict dangerous situations and anticipate accidents. As mentioned in Chapter 1 there are many uses of optical flow such as for egomotion estimation, obstacle detection and many more. Here, related work on some of the applications of optical flow in ADAS is presented briefly.

2.5.1 Horizon line estimation

In general, monocular based approaches tackle the camera pose problem by using the prior knowledge of the environment as an extra source of information. For instance, Coulombeau and Laurgeau [31] assume that the road observed on images has a constant known width; Liang et al. [54] assume that the vehicle is driven along two parallel lane markings, which are

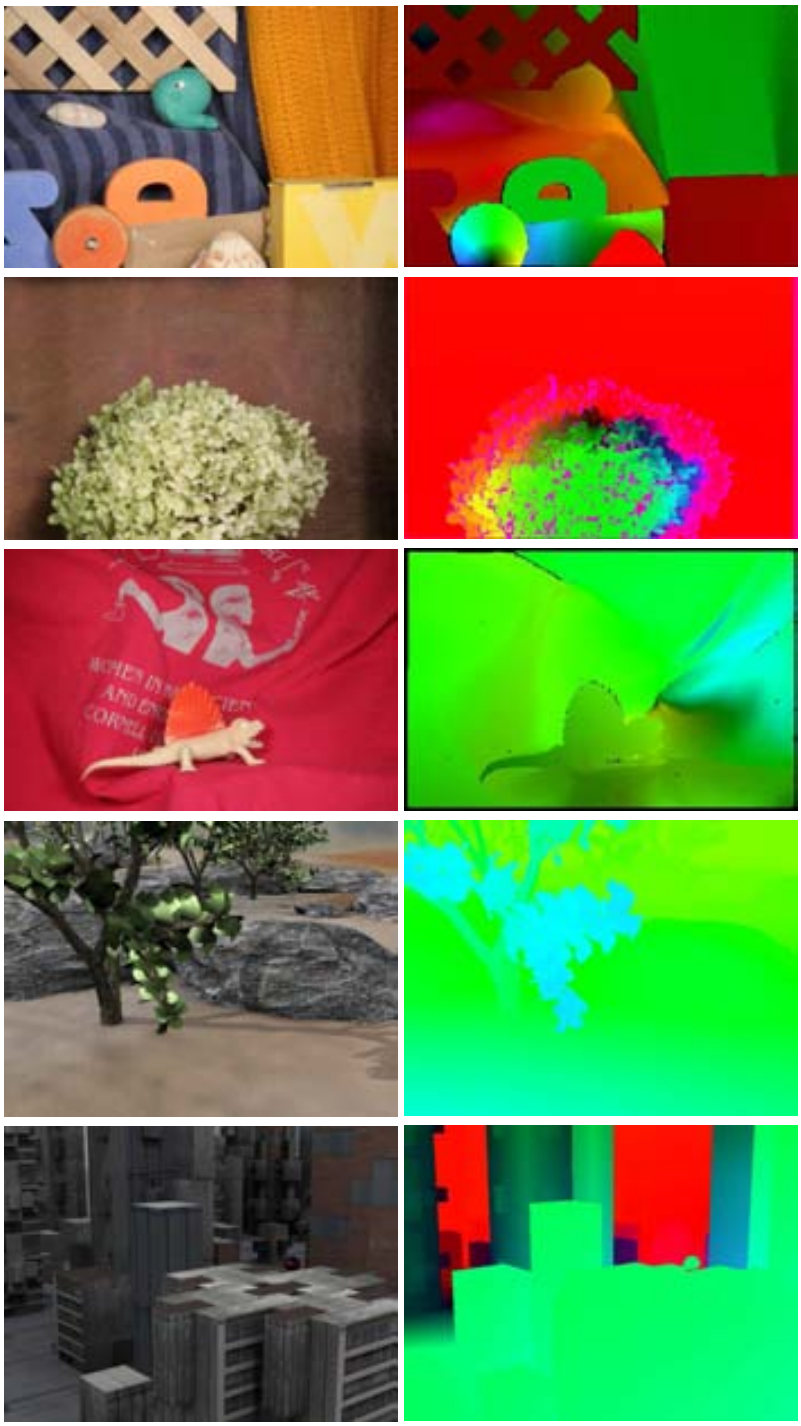


Figure 2.5: Some samples of Middlebury dataset [1]; (*left*) first frame of a pair; (*right*) corresponding ground-truth optical flow.

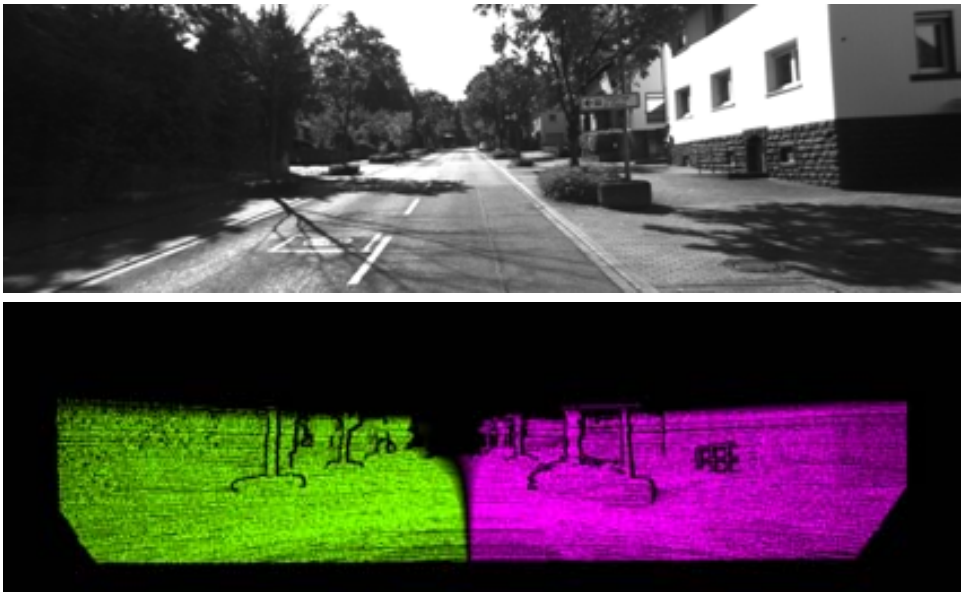


Figure 2.6: An example of KITTI dataset [42].

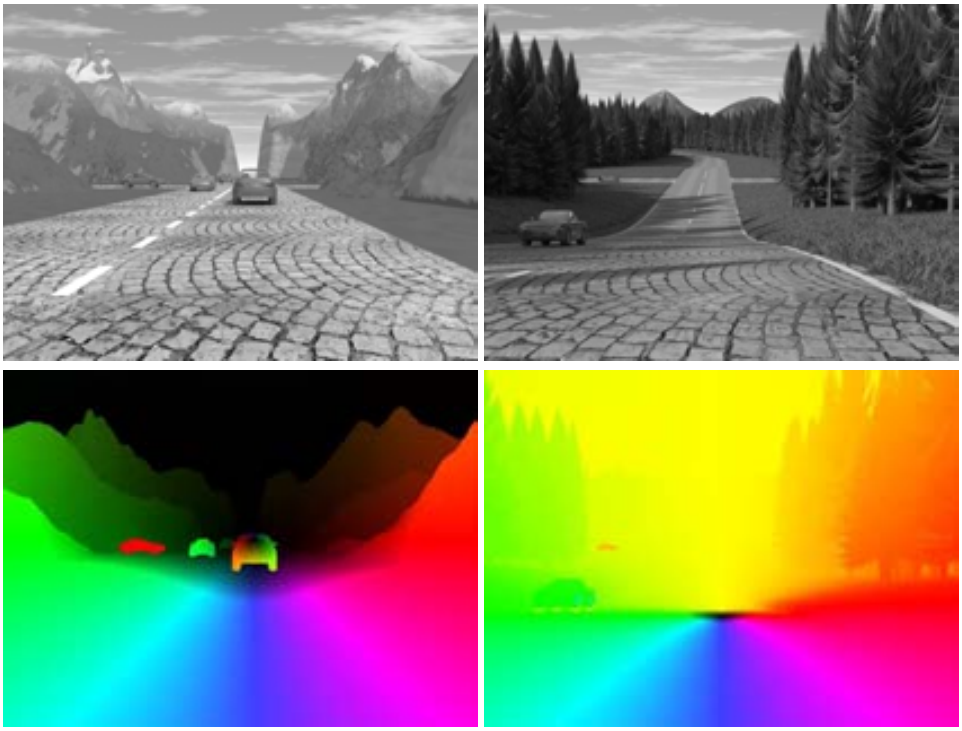


Figure 2.7: An example of EISATS dataset [93].

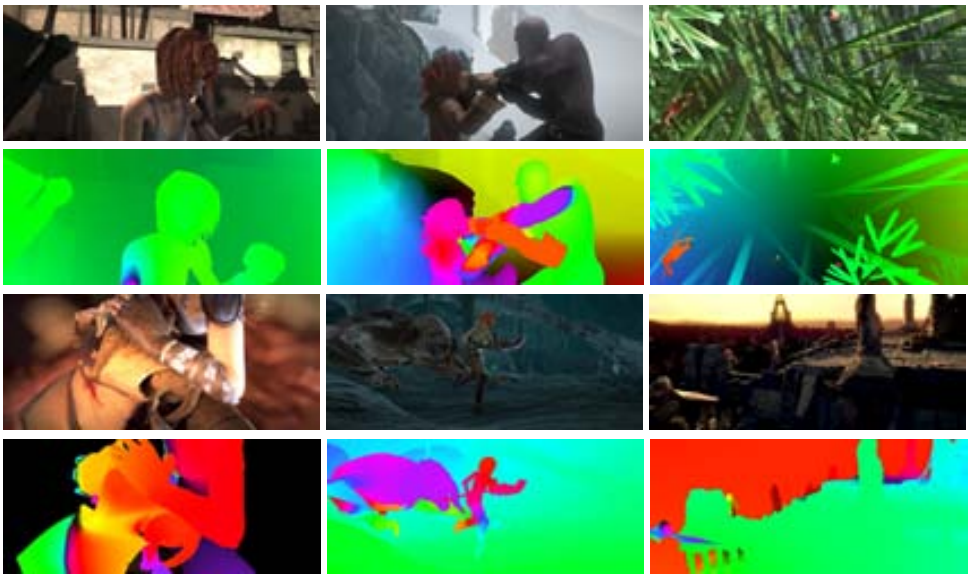


Figure 2.8: Sample images and corresponding ground-truth flow fields from MPI-Sintel dataset [27].

projected to the left and to the right of the image; Bertozzi et al. [15] assume that the camera's position and orientation remain constant through the time. Obviously, the performance of these methods depend on the fulfillment of these assumptions, which in general cannot be taken for granted.

On the other hand, stereo based approaches have also used prior knowledge of the scene to simplify the problem and to speed up the whole process by reducing the amount of information to be handled. For instance, [66] proposes to reduce the processing time by computing 3D information only on edge points (e.g., lane markings on the image). Similarly, the edge based *v*-disparity approach proposed in [52], for an automatic estimation of horizon lines and later used for applications such as obstacle or pedestrian detection (e.g., [14, 51]), only computes 3D information over local maxima of the image gradient. A different stereo vision based approach has been proposed in [78]. It uses dense depth maps and is based on the extraction of a dominant 3D plane that is assumed to be the road plane. Camera's position and orientation are directly computed, referred to that plane. A recent work [35] proposes a novel paradigm that is based on raw stereo images provided by a stereo head. This paradigm includes a stochastic technique to track vehicle pose parameters given stereo pairs arriving in a sequential fashion. In [35], the assumption is that the selected region only contains road points, as well as the road surface is assumed to be a plane. Some of the previous monocular vision systems rely on local formulations: a technique based on optical flow with template matching scheme was used in [85], while a maximum likelihood formulation over small patches was introduced in [82].

2.5.2 Egomotion estimation

The egomotion also called visual odometry is the estimation of the 3D rigid motion (rotation and translation) of a camera from the image sequence acquired by it. The camera motion involves six DOF (Degrees Of Freedom). Egomotion estimation involves estimation of three translation and three rotation (orientation) parameters. Figure 2.9 illustrates the orientations (*pitch*, *yaw* and *roll*) and translations with respect to world coordinate system (*X*,*Y* and *Z* axes). This is one of the fundamental tasks in many advanced driver assistance and robotic applications. Egomotion methods can be classified as feature [9] and appearance based [108] methods. A feature based five-point algorithm that is regarded as a classical in visual odometry is proposed by Nister [68]. There are several approaches proposed in literature; the work in [87] compares some of them. Estimation of motion parameters can be ambiguous (e.g.,translation on X axis can be incorrectly estimated as rotation on Y axis and vice-versa) under a reduced field of view or with insufficient depth variation (e.g., [6, 37]) because different motions can induce similar flow vectors. Therefore, another way to evaluate these three orientations is just by comparing the rotation matrices directly instead of comparing individual orientations [87]. An extensive review of egomotion estimation techniques, from the ADAS field, can be found in [30]. A polynomial eigenvalue solution based on five-point sets is proposed in [50]. These five-point sets can be randomly selected by a RANSAC [39] based scheme that allows to reject outliers and noisy data.

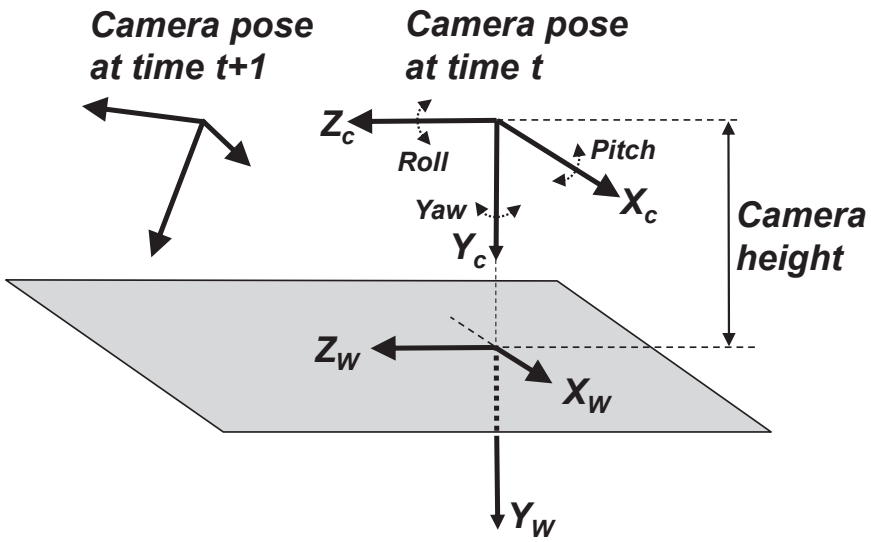


Figure 2.9: Illustration of coordinate systems and orientations.

Chapter 3

Space-Variant Representations

3.1 Introduction

Traditionally, in machine vision images are represented using cartesian coordinates with uniform sampling along the axes. On the contrary, biological vision systems represent images using polar coordinates with non-uniform sampling. For various advantages provided by space-variant representations many researchers are interested in space-variant computer vision. One of the significant advantage of a space-variant representation is the reduction in the amount of data. For instance, a cartesian image of size 640×480 when sampled according to typical polar representation of size 240×360 in radial and angular axes reduces the 1/5th of the the original data. With the same resolution traditional space-variant Log-Polar Representation (LPR) enables to have higher resolution at centre. In this direction the current work proposes a novel and simple space-variant representation of images. The proposed representation is compared with the classical log-polar mapping. The log-polar representation is motivated by biological vision having the characteristic of higher resolution at the fovea and reduced resolution at the periphery. On the contrary to the log-polar, the proposed new representation has higher resolution at the periphery and lower resolution at the fovea. Our proposal is proved to be a better representation in navigational scenarios such as driver assistance systems and robotics. The experimental results involve analysis of optical flow fields computed on both proposed and log-polar representations. The experimental analysis comprises results from synthetic as well as real sequences.

In the particular contexts of robotics and advanced driver assistance systems (ADAS), LPR has attracted the attention of many researchers. In general, in these fields LPRs are obtained using the *vanishing point* (VP) as a center of the log-polar reference system. $VP_{(x,y,z)}$ corresponds to a point at $z \rightarrow \infty$ where two parallel lines of a road appear to converge in the image plane. Since LPR results in a high sampling in the fovea region, points in the periphery are undersampled. It should be noted that the periphery of the 2D image (in particular the lateral and bottom parts) corresponds to regions in the 3D space near to the camera reference system, hence are the most important areas for robotics navigation tasks and ADAS applications. Furthermore, points near to the camera are not only useful for detection tasks but also for an accurate calibration; note that the accuracy of 3D data decreases with the depth.

In the current work classical LPR is studied and a new space-variant representation scheme

called Reverse Log-Polar Representation (RLPR) is proposed. It is intended to overcome the disadvantage of LPR with respect to the sampling in periphery. In addition to the benefit of reduction in information to be processed similar to LPR, RLPR produces higher accurate results in navigational tasks. Unlike LPR which has advantages in pattern recognition applications with rich fovea and sparse periphery, the newly proposed spatial representation has advantages in navigational tasks. These space-variant representations are used for optical flow estimation in forward facing motion problems. In this chapter, the performance of both representations is analyzed using dense optical flow. The chapter is organized as follows. Section 3.2 describes the proposed space-variant representation. Experimental results and comparative studies are given in Section 3.3. Finally, the work is concluded in Section 3.4.

3.2 Space-variant representations

As discussed in the previous section a polar representation offers advantages with respect to the cartesian one depending on the application. In the current section polar space-variant image representations are discussed.

A log-polar representation is a polar mapping with logarithmic distance along the radial axis. For a given pixel (x,y), the log-polar (ρ,θ) are defined as:

$$\rho = log(\sqrt{(x-x_0)^2+(y-y_0)^2}), \qquad \theta = \arctan((y-y_0)/(x-x_0)), \tag{3.1}$$

where (x_0,y_0) is the origin of mapping. Since the whole thesis is focussed on ADAS applications, which run on forward facing moving platforms, the origin of the reference system is assumed to be at the vanishing point position.

As mentioned in Section 3.1, LPR oversamples the fovea and undersamples the periphery. This leads to the non-preservation of vital information of the periphery useful for forward facing vision applications. The latter motivates us to propose in this chapter a more appropriate space-variant representation, where a (x,y) pixel is mapped to as:

$$\rho = log(r_{max} - \sqrt{(x-x_0)^2+(y-y_0)^2}), \quad \theta = \arctan((y-y_0)/(x-x_0)), \tag{3.2}$$

where r_{max} is the radius of the circle around VP; typically r_{max} is the largest inner circle within the cartesian image boundary around VP. The image contained inside the circular region is to be mapped. The proposed mapping is different from LPR in the sense that logarithmic subsampling is from the periphery towards the center and will be referred to as *Reverse Log-Polar Representation* (RLPR). Figure 3.1(*right*) shows LP (*top*) and RLP (*bottom*) representations of the same image Fig. 3.1(*left*). In both cases the images are sparsely sampled as depicted in Fig. 3.1(*middle*). Since the LP/RLP transformations involve both many-to-one and one-to-many mapping, the LP/RLP images cannot be straight forwardly dense. The dense images presented in Fig. 3.1(*right*) are obtained by querying for each (ρ,θ) to the cartesian and by bilinear interpolation—horizontal axis is angles ($\theta's$) and vertical axis is distances ($\rho's$). As can be seen in the grids in Fig. 3.1(*middle*), qualitatively, the RLPR sampling better preserves the periphery information, which covers most part of the road at the bottom in the scenario of a moving vehicle. Further experiments in this chapter are performed on these images (shown in Fig. 3.1(*right*)), represented as rectangular images, but sampled using LP and RLP sampling (Fig. 3.1(*middle*)) from the original image shown in Fig. 3.1(*left*).

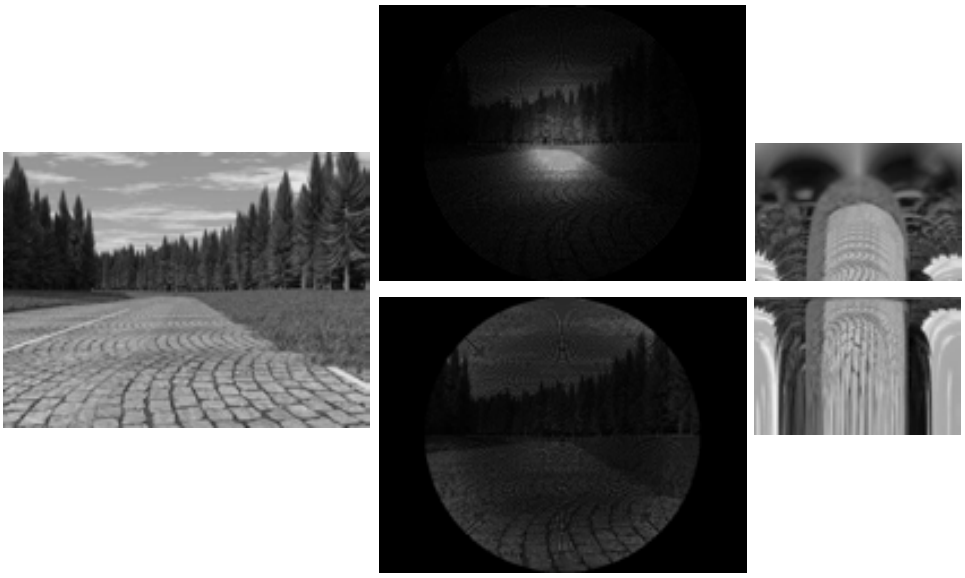


Figure 3.1: ($left$) A synthetic image. ($middle-top$) Log-Polar sampling. ($middle-bottom$) Reverse-Log-Polar sampling. ($right-top$) Cartesian representation of LP image. ($right-bottom$) Cartesian representation of RLP image.

3.3 Experimental results

As mentioned in Section 2.3, there have been many applications using LP represented images, some of them based on the optical flow estimation in that space. The current work aims to estimate the optical flow on RLP represented images and compare them with results from LPRs.

In LP/RLP representations of images the origin of mapping should be the vanishing point in the scenario of a forward facing moving vehicle, so that the mapped images better suit the applications. In the current work, vanishing points computed from a RANSAC based approach [69] are used. The images are mapped to LP/RLP as explained in Section 3.2 and then, the optical flow is computed on these LP and RLP rectangular represented images. The bottleneck to compare the flow fields from LP and RLP representations is that the flow field patches at a particular location in both representations correspond to different regions of the image in cartesian with varied resolution. Hence, the framework proposed to perform the comparisons consists in inverse mapping the flow fields back to cartesian and compare them in the cartesian space. Figure 3.2 shows an image pair in cartesian (*top-left*), their LPRs (*middle-left*) and RLPRs (*bottom-left*) and in the (*top-right*) the ground-truth flow field, the computed flow field of LPR (*middle-right*) and RLPR (*bottom-right*). The color map used to display optical flow is shown in Fig. 3.2 bottom right corner. Since the image pairs correspond to translation along the camera focal axis, the ground-truth flow field in cartesian looks diverging (see Fig. 3.2 (*top-right*)), vectors appear originating from a VP toward the boundary in all directions. Hence, the computed flow fields in both LP/RLP representations looks blue in color indicating all the vectors point downwards. In order to compare the flow

fields of LP and RLP, these flow fields are mapped back to cartesian. Figure 3.3 depicts the flow fields of both LP and RLP mapped back to cartesian. These flow fields in cartesian are sparse. Hereinafter, the LP and RLP representations of flow fields refer to these mapped back to cartesian.

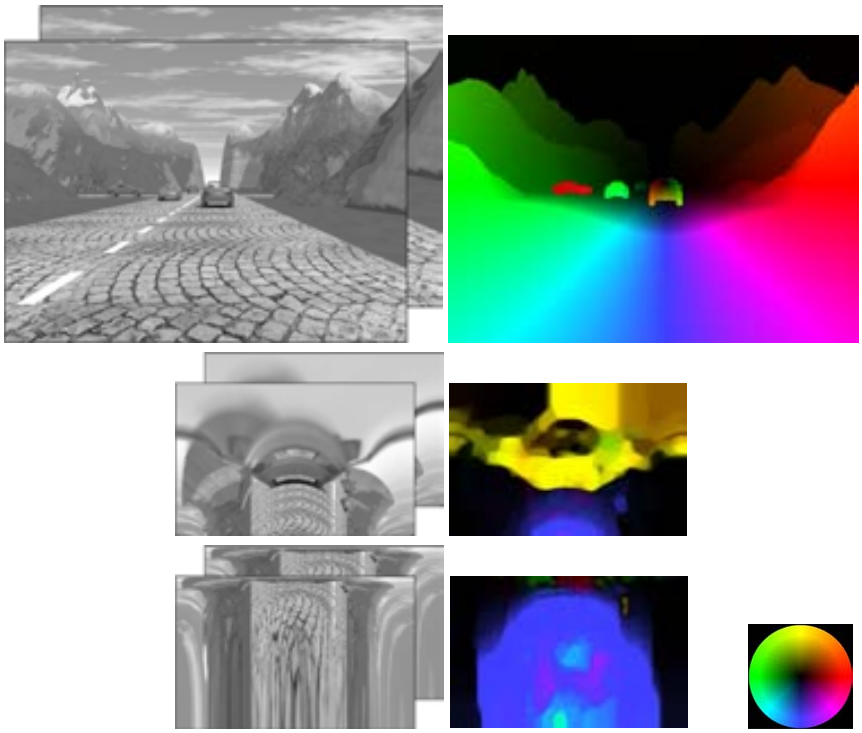


Figure 3.2: (*top*) Pair of images and flow fields in Cartesian. (*middle*) LP representations. (*bottom*) RLP representations and (*bottom* – *right* – *corner*) colormap used to display flow fields.

Since the flow fields from LP and RLP representations are sparse and of different resolutions along radial direction, in order to do a fair comparison a common set of pixels (intersection mask) is selected. Figure 3.4 shows the masks (i.e., locations where the flow vectors exist) of LPR (*left*) and RLPR (*middle*) flow fields and the intersection mask (*right*) that is the common set of positions those have flow values in both representations. Then, AAE (see Equation 2.4) between LP and ground-truth flow fields, and between RLP and ground-truth flow fields are computed using the obtained intersection mask. Table 3.1 shows AAEs of flow fields from ten selected different image pairs from sequence-1 of set-2 of [93]. The images are of resolution 480×640. They are mapped to LP and RLP representations of resolution 230×360, placing the vanishing point at $(230,340)$, computed from [69]. Then, optical flow is computed on these images using [83]. The flow fields are mapped back to cartesian and then using the intersection mask shown in Fig. 3.4 (*right*), the AAEs are computed. The AAEs in Table 3.1 show that flow fields estimated in RLP representations are more accurate than flow fields from LPRs.

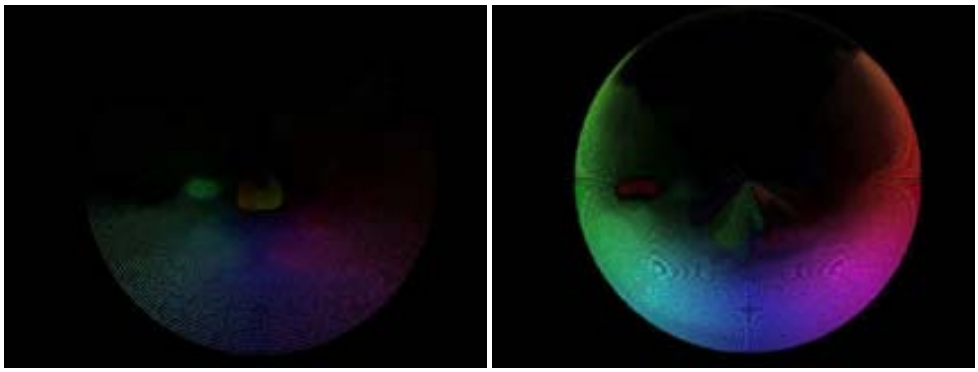


Figure 3.3: Inverse mapped flow fields from (*left*) LP and (*right*) RLP.

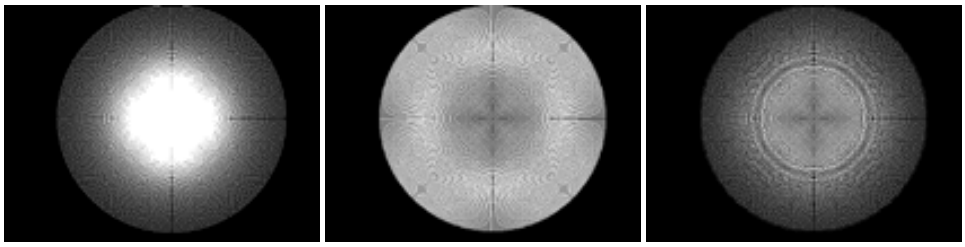


Figure 3.4: (*left*) LP mask. (*middle*) RLP mask. (*right*) Mask from their intersection.

Table 3.1
AAEs (DEG.) FOR SELECTED TEN FLOW FIELDS FROM SEQUENCES [93] IN LPR AND RLPR

		1	2	3	4	5	6	7	8	9	10
Seq. 1	LPR	24.3	24.3	23.9	23.9	23.9	23.8	23.6	23.7	23.5	23.4
	RLPR	20.9	19.1	19.0	19.4	18.3	18.4	18.4	18.0	17.9	18.9
Seq. 2	LPR	24.3	24.5	27.4	27.1	24.3	24.6	24.7	24.5	24.7	24.8
	RLPR	21.6	21.8	27.2	26.6	21.6	23.3	24.2	23.8	23.8	22.0

A similar experiment on sequence-2 of set-2 of [93] is performed; results are also presented in Table 3.1. Vanishing points in sequence-2 varies, but the most of the frames' VP lies in $(240, 320)$; hence the resolution of the mapped images is fixed at 240×360. Also in the results of sequence-2, the RLPR flow fields are more accurate than LPR flow fields. The difference in AAEs between LP and RLP representations of sequence-2 is smaller than the results of sequence-1 because the displacement between consecutive images in sequence-2 is very high. These large displacements lead to more stretching in RLP represented images and hence increase in errors in flow fields.

In order to analyze the behavior of space-variant representations, we have done experiments for different resolutions in the two sequences presented above (sequence-1 and sequence-2). The experiments consist of testing different radial and angular resolutions. Since it is intuitive to keep the angular resolution equal to or higher than 360, otherwise leads to higher loss of information, we have performed experiments for different angular resolutions such as 360, 720 and 1440. From this first test we conclude that increasing the angular resolution does not improve results, it only leads to higher processing. On the contrary, for radial resolution, in addition to the deep study that will be presented in Fig. 3.5, we tested two values for sequence-1. These values correspond to the maximum inner circle (230) and minimum bounding circle (378) covering the whole rectangular cartesian image respectively from the VP. Similarly, for sequence-2 we considered 240 and 400 corresponding to inner circle and outer circle. In sequence-1, 230 is the distance from the VP to the nearest boundary. Note that in the case of using the outer circle, a lot of empty space in LP and RLP representations will be included, which is not desired. Values smaller than 230, but covering the same area, would yield to more sparse sampling in both LPR and RLP that could introduce more errors in the estimated optical flow fields. Therefore, as mentioned above, we have considered 230×360 (radial$\times$angular) and 240×360 (radial$\times$angular) as appropriate resolutions for sequence-1 and sequence-2 respectively. These resolutions are the ones used in all our experiments, which correspond to the regions contained in the largest inner circle centered on the VP.

Further experiments are done to analyze how the error evolves along the space in these representations. Different concentric circular regions of the flow field around the vanishing point, with an increase in the radius of the circle within the flow field boundary, are considered. At each sized circle, the AAE is calculated for the region inside the circle and also for the region outside the circle. This experiment is done for both LPR and RLPR on sequence-1. Since the radial axis for the flow fields in sequence-1 is of length 230, nine circles with increasing radius from 23 till 207, in steps of 23, are considered. Figure 3.5(*top-left*) and (*middle-left*) show the AAEs in colormap for the region inside the circle at radius 115 for LPR and RLPR. Figure 3.5(*top-right*) and (*middle-right*) show the AAEs in colormap for the region outside the circle at radius 115 for LPR and RLPR respectively. In Fig. 3.5(*bottom-left*), solid line indicates AAEs (the average of 10 flow fields' region inside the circle) in LPR with the increase in radius. The AAE increases as the inner area increases. This proves that the flow field near to the fovea is more accurate than in the periphery in LPR. The dashed line correspond to AAEs in RLPRs with the increase in radius. In the plot Fig. 3.5(*bottom-left*) the AAE of RLPR decreases from radius 138 till the boundary. At radius 207, where most of the image area is covered, the AAE of RLPR is lower than the AAE of LPR. This shows RLPR is better at periphery than LPR.

Figure 3.5(*bottom-right*) shows the AAEs of LPR and RLPR, outside the circles, with the

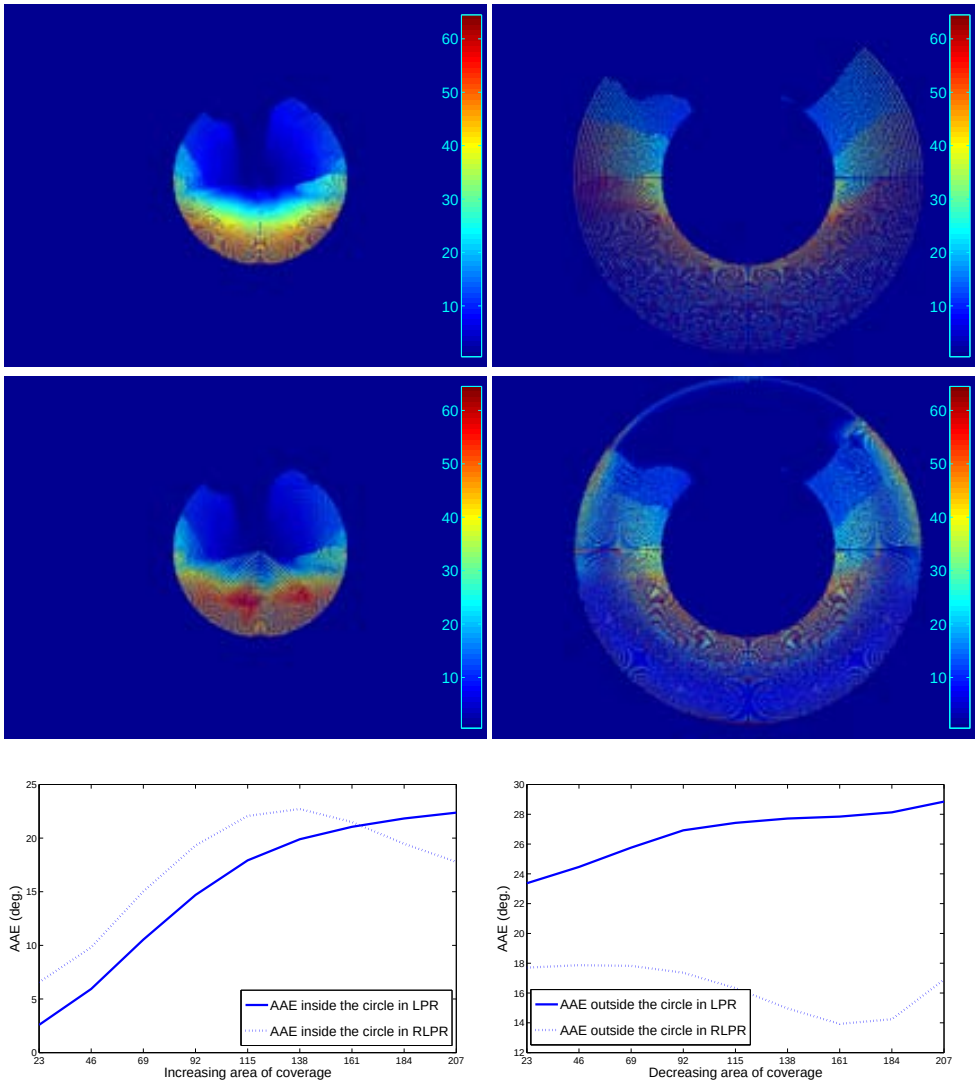


Figure 3.5: Analysis of AAEs over space in LPR and RLPR (values in colormap scale computed from eq. (2.4)). (*left*) Region inside circle. (*right*) Region outside circle.

increase in radii of the circles. The outer area is getting reduced with the increase in radius of the circle. The solid line indicating AAE of LPR increases as the outer area decreases, whereas the dashed line indicating AAE of RLPR decreases as the outer area decreases till the circle with radius 161. Then, it increases due to some artifacts in the extreme periphery of RLPR flow field. Figure 3.5(*middle-right*) shows the artifact, thin band of circular arc on the top, whereas this band is absent in the LPR (*top-right*) flow field. This plot (*bottom-right*) gives the same conclusion obtained from the plot in (*bottom-left*).

RLPR is better than LPR not only by AAE, Table 3.2 shows the mean AAE and mean EPE (see at the end of Section 2.2) of all frames in sequence-1 and sequence-2 of set-2 [93] named as Sequence 1 and Sequence 2 in the table. Sequence-1 contains 99 flow-fields and sequence-2 contains 394 flow fields. These two sequences resembles a countryside scenario with very good road texture. Table 3.2 also provides results of similar experiments on another synthetic sequence and a real sequence. Sequence 3 in the table is another synthetic sequence of ten image frames [70] of an urban scenario with asphalt road texture, which does not contain as much texture as previous two sequences. The error values for this sequence also show that still RLPR is better in this case. Finally, a real sequence is also used to validate the usefulness of the proposed space-variant representation. Figure 3.6 shows a pair of images from the real sequence used here with an annotated flow field (*top-right*), and the corresponding LP (*bottom-left*) and RLP (*bottom-right*) representations. Results from this real sequence [55] are depicted in the last row of Table 3.2. This sequence contains 37 image frames of an urban driving scenario. This dataset has annotated ground-truth optical flow. The results for this sequence also reveal the goodness of RLPR. From the experiments on these four sequences, it can be confirmed that RLPR is better than LPR in both AAE and EPE on both synthetic and real sequences.

Figure 3.6: ($top-left$) Pair of images of a real sequence [55]. ($top-right$) Annotated ground-truth flow field between the pair of consecutive frames in the real sequence. ($bottom-left$) LP representations and ($bottom-right$) RLP representations.

Table 3.2
MEAN AAE AND MEAN EPE OF SYNTHETIC SEQUENCES: [93] (SEQUENCE-1 AND SEQUENCE-2 IN SET-2) AND [70], AND A REAL SEQUENCE [55].

		AAE	EPE
Sequence 1	RLPR	18.3145	1.4043
	LPR	23.9895	1.7151
Sequence 2	RLPR	35.9757	6.1254
	LPR	47.41	6.4059
Sequence 3	RLPR	26.1972	3.2783
	LPR	36.8119	3.9058
Real Sequence	RLPR	36.5783	2.1985
	LPR	63.7911	2.6398

3.4 Conclusions

The current work proposes to change image sampling, resulting in a novel space-variant representation. This validates the initial intuition that more accurate results can be obtained if we change the philosophy of oversapling fovea region to the higher sampling in peripheral region. This improvement is useful when forward faced on-board vision systems are considered, where translation in the optical axis is the predominant motion (e.g., mobile robotics, driver assistance). The proposed RLP representation is evaluated and compared with classical LP by computing optical flow on them. The rectangularized representations of space-variant sampling may introduce systematic errors in flow fields when traditional optical flow approaches are used on them. Hence we can say that there is a need of research in variational optical flow methods to be applied directly on the space-variant sampled images. The experimental analysis is performed on both synthetic and real video sequences. Experimental results show that RLPR is better than LPR in navigational applications.

Chapter 4

Synthetic Sequence Generation Framework

4.1 Introduction

Computer Vision has got applications in different fields of life. The research in computer vision is always motivated, as well as supported by the benchmarking dataset with ground-truth information. The availability of ground-truth information makes the dataset very useful for the evaluation of different methods. Generally, ground-truth can be obtained by manual labelling and/or sophisticated equipments such as for detection, recognition and segmentation tasks. Optical flow technique is an important approach in motion estimation that is useful in many fields such as action recognition, surveillance, image compression, obstacle detection, to mention a few. In spite of the fact that the research on optical flow started more than three decades ago, the seminal methods were just proposed in 1981 [46] [56]. There are many advances in this field since then and it has got momentum since several years from now. This is mainly due to the availability of datasets with ground-truth flow information as well as due to the large increase in computational power. The availability of dataset challenges the existing state of the art methods and promotes research to propose newer methods. Also it allows the evaluation of existing methods. Another benefit is that the ground-truth data satisfies the need of the learning based approaches. Particular to the optical flow, there are issues in obtaining ground-truth optical flow of real scenarios. There is no such direct sensor to obtain ground-truth information with a good accuracy in real scenarios unless performed in a restricted laboratory environment. It can be possible in a laboratory environment for limited motion using hidden texture [11], but for natural outdoor scenes, there is no way at present to have ground-truth optical flow with good accuracy. In such a scenario, the alternate viable solution is to have synthetic datasets. Hence, the developments in the area of computer graphics have given the advantage of creating such synthetic datasets facilitating the validation of computer vision algorithms.

Advanced driver assistance systems (ADAS) is an upcoming area where many computer vision techniques have potential to tackle challenging situations. The safety is one of the top priority in the present days, and the optical flow estimation has an important role in ADAS. There are several variations in a driving scenario affected by environments such as

urban, highway, countryside where the vehicle is being driven, types of road (well structured, differently textured), weather conditions, and daylight conditions. There exist no sensor to directly acquire ground-truth flow field along with the image sequences and it is not possible to obtain accurate ground-truth by other ways such as by using depth sensors. The alternative way for this situation is to create virtual scenarios using 3D designing tools. The advantage to go with synthetic sequences is that one can create all the possible different environments and scenarios as mentioned before in the case of driving. Although there exist a question that how realistic are these synthetic ones compared to real ones, one can thrive to integrate realism into virtual scenes with the latest advances in computer graphics. Actually, there are some work in this direction, for instance recently in [60] the authors did an attempt to create realistic synthetic scenarios. They show that it is possible to create more realism by varied lighting conditions, varied material properties and by exploiting state of the art in computer graphics.

As discussed above, driving scenarios involve varied complexities due to motion of on-board camera, dynamic scene with independently moving vehicles and other additional factors. There are several of such characteristics that need to be analyzed to develop a good optical flow method in driving scenarios. For example, change in road texture influences the optical flow accuracy. To do such a study one needs to have image sequences of the same structural scene but with different road textures. In reality, it is very difficult to create such one and impossible to generate ground-truth optical flow of good accuracy. The other best solution is to have synthetic sequences generated and then obtaining ground-truth will also be easy. The existing datasets do not provide any of such sequences. In this work we are proposing a framework to generate such sequences along with the ground-truth flow fields. The proposed framework is presented in Section 4.2 followed by the technique of generating ground-truth optical flow in Section 4.3 and dataset generated in Section 4.4. The generated ground-truth optical flow is validated by back-projecting images to match previous images in a sequence. Also our dataset is compared with state of the art synthetic driving sequences. Further a simple analysis of effectiveness of different optical flow algorithms on our dataset is also performed. This analysis and validation is provided in Section 4.5. Then, the Chapter is concluded in Section 4.6.

4.2 Synthetic sequence generation framework

The objective of the work in the current Chapter is to generate synthetic sequences along with ground-truth flow fields. We present a framework to generate sequences in a driving scenario considering three particular cases: i) on-board vision system in a vehicle with different speeds; ii) roads with different textures; iii) scenarios with independently moving vehicles. For analyzing the influence of speed on optical flow accuracy, we need to have image sequences of the same scene, but the on-board vision system vehicle moving with different speeds. Similarly, for analyzing the impact of texture, we need image sequences of the same scene (i.e., surrounding scene structure) but with just different textures. In reality, it is impossible to have such scenarios and to generate ground-truth optical flow in real life. Hence, in the current work we propose a framework similar to the one presented in [58]. We use Maya[1] to develop a 3D scene. We have built a synthetic 3D urban scenario that consists of a straight

[1] www.autodesk.com/maya

road and buildings around it with appropriate textures. A camera assumed to be fixed in a vehicle referred to as on-board camera moves along the road in the model. The images are rendered using in-built Maya software with production quality. All the images are rendered with a resolution of 640×480 pixels. Figure 4.1 shows a snapshot of the 3D urban scenario designed in Maya.

Figure 4.1: Snapshot of our 3D design in Maya.

For case i) the on-board camera moves at different speeds along the same path straight on the road and the images are rendered. Some of the rendered images are shown in Fig. 4.2. The $top-left$ is the first image which is common to all the sequences. 2^{nd} $row-left$ is the second image in a sequence and $bottom-left$ is the second image of another sequence of higher speed. In ADAS scenarios, the road surface covers a major part in the images taken through vehicle's camera. The flow vectors computed from this surface are more reliable as there could be more inaccuracies in other areas of the image due to occlusions, specularities etc. For case ii) to analyze texture influence particularly, only road texture can be changed very easily without disturbing the 3D design. Hence, for a given speed, several sequences with different road textures are rendered. Figure 4.3 shows images with three different textures on the road surface.

In 3D design, the designer has full control over all the things such as motion of camera, lighting, textures and motion of different objects in the scene. With this capability, for case iii) we have added two moving vehicles in the scene. One vehicle coming towards the on-board camera vehicle and another coming from a cross road are considered. To add complexity, pitch and yaw variations to the on-board camera are also incorporated. The yaw is 0.25

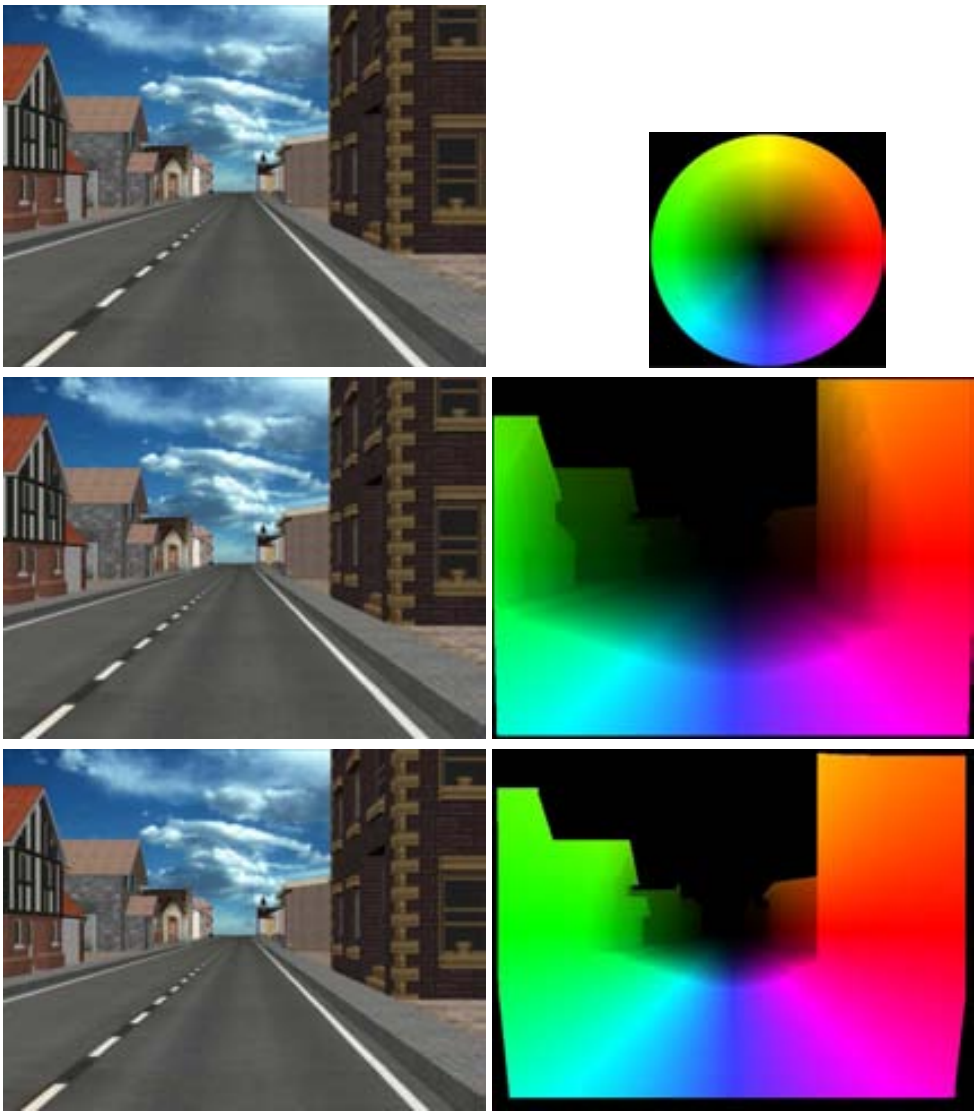


Figure 4.2: Images from sequences of different speeds: ($top - left$) first frame common for all sequences; ($top - right$) colormap used to show the flow fields; ($left - column$) second frames from the sequences of different speeds in the increasing order (2^{nd} *and* 3^{rd} *rows*); ($right - column$) the ground-truth flow fields between the respective first and second frames.

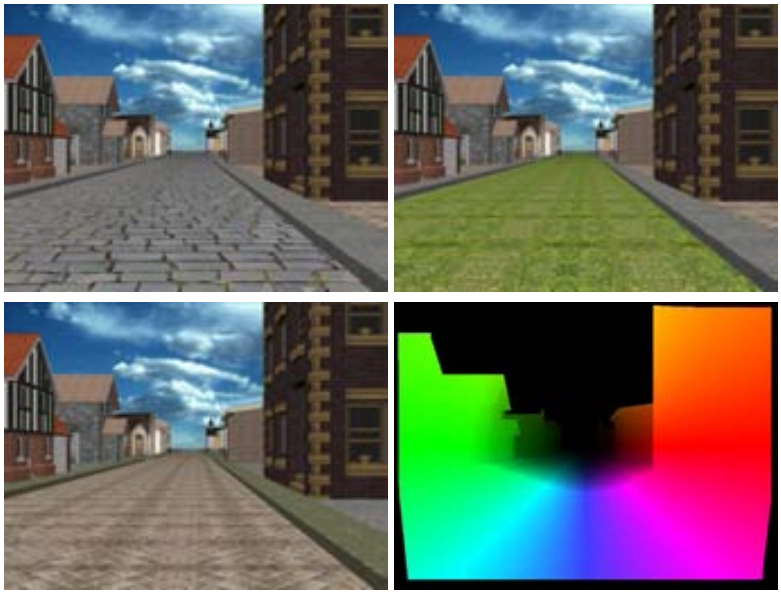


Figure 4.3: $(top-left)$, $(top-right)$ and $(bottom-left)$ frames with different texture from different sequences; $(bottom-right)$ Ground-truth flow field for all the pairs of images in $(top-left)$, $(top-right)$ and $(bottom-left)$; all of them have the same scene geometry and same speed but different textures.

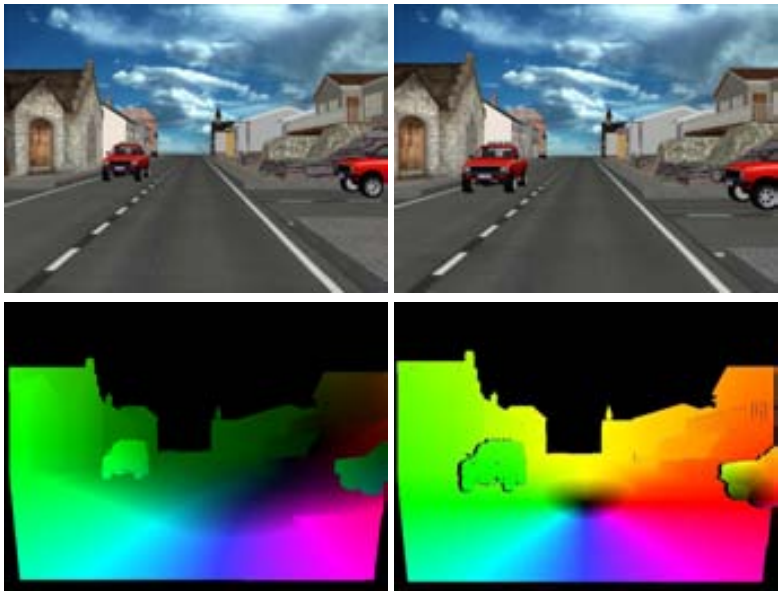


Figure 4.4: (top) Two different image frames from a sequence with independently moving vehicles and different egomotion. $(bottom)$ Ground-truth flow fields between the above frames and to their next ones in the sequence.

degrees to the left/right and the pitch is 0.25 degrees to the up/down. Rendered images from this sequence are shown in Fig. 4.4. In all these above cases, the camera focal length is 35mm and the frame-rate is 24fps.

4.3 Optical flow ground-truth

This section describes the generation of ground-truth flow fields. It is based on the well known ray-tracing approach. Ray-tracing is basically a technique of tracing the path of a light ray. It is being used in 3D computer graphics environments to render photorealistic images. The same idea has been used in the current work to estimate displacement vector of each pixel. The complete information of designed 3D model enables us to use such a technique to compute the distplacement vector. A pixel P in an image plane at time t is traced to its position in the 3D scene. Then this 3D point is projected back to the image plane at time $t+1$. Since we know the simulated camera motion in time, difference in pixel position at different times on the image plane gives the displacement vector. Figure 4.5 depicts the ray-tracing approach. A vector from P_t to P_{t+1} is the flow vector at P_t with respect to image at time t. The computed ground-truth flow fields for different speeds are shown in Fig. 4.2. Figure

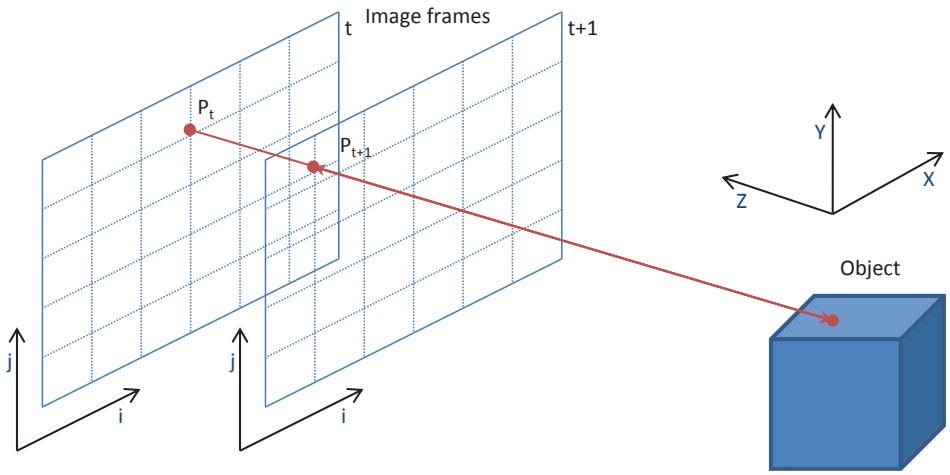


Figure 4.5: Ray-tracing to estimate displacement vector.

4.2 (*top-right*) is the colormap used to depict flow fields. The color indicates the direction and intensity indicates the magnitude. Figure 4.2 (*middle-right*) is the ground-truth flow field between the frames in (*top-left*) and (*middle-left*). Similarly, Fig. 4.2 (*bottom-right*) is the ground-truth flow field between (*top-left*) and (*bottom-left*). One can notice large blank space at the bottom of Fig. 4.2 (*bottom-right*) that indicates larger occluded areas at a higher speed than the one at lower speed in (*middle-right*). The maximum displacement in the lowest speed sequence is 8.31 pixels and that in the highest speed sequence is 33.67 pixels. The ground-truth flow fields for differently textured image pairs are shown in Fig. 4.3 (*bottom-right*). Since the scene geometry is the same, the ground-truth flow fields for all differently textured sequences are the same. Further ground-truth flow fields for two image pairs from

complex sequences with moving vehicles are shown in Fig. 4.4. Notice that the flow field in Fig. 4.4 (*bottom-left*) has flow vectors at all pixels except the sky and occluded pixels at the boundary, whereas the flow field in (*bottom-right*) does not have values at the edge of moving vehicles depicting the occluded regions. Hence in synthetic sequence generation one has full control of all the possible scenarios and it is very useful.

4.4 Dataset

Using the framework presented in the current work, we have generated four sequences of different speeds. The sequence with highest speed has a displacement of four times the displacement of the sequence with lowest speed. Then, we have also created three sequences of different road textures for each speed. Hence we have generated twelve sequences of different combinations of speeds and textures. In the case of analysis of optical flow accuracy for different speeds, if we generate equal number of frames for each speed, then the scene geometry covered by the distance varies and it might affect estimated flow accuracy. Thus we have generated frames for a constant distance and hence sequences of different speeds have different number of frames. The sequences generated with different textures have different textural properties, particularly in the increasing order of texture contrast. The third set consists of complex sequences with two independently moving vehicles; pitch and yaw motion of on-board camera. It gives an opportunity to study more dynamic scenes. These sequences are also generated for different combinations of road textures and on-board camera speeds without the constraint of constant covered distance. Here all the complex sequences have ten frames. The number of frames in all these sequences are depicted in Table 4.1.

Table 4.1
DIFFERENT SYNTHETIC SEQUENCES

Sequence	Number of frames (For each of the three textures T1, T2, T3 and T4)
Speed S1	40
Speed S2	20
Speed S3	13
Speed S4	10
Complex sequences (For each of the four speeds)	10

4.5 Analysis and validation

As described in sections 4.3 and 4.4, we have generated several sequences with different characteristics of a driving scenario. In this section, we evaluate how good is the generated ground-truth optical flow. In order to evaluate the accuracy, we consider several pairs of images from the generated sequences. Hence, for a given pair, we back-project the second image

using the ground-truth optical flow to match the first frame. Figures 4.6 and 4.7 show two of such examples. Further, we calculate normalized-root-mean-square-error (NRMSE) between the first image and the corresponding back-projected one, considering only the region where the back-projected pixels exist. NRMSE for several pairs are shown in Table 4.2.

Table 4.2
NRMSE FOR SEVERAL PAIRS FROM EISATS AND OUR SEQUENCES

Image pairs from our sequences	NRMSE	Image pairs from EISATS	NRMSE
S1T1	0.0079	Pair1	0.0174
S4T4	0.0089	Pair2	0.0346
Complex S1T1	0.0100	Pair3	0.0139
Complex S4T4	0.0115	Pair4	0.0193

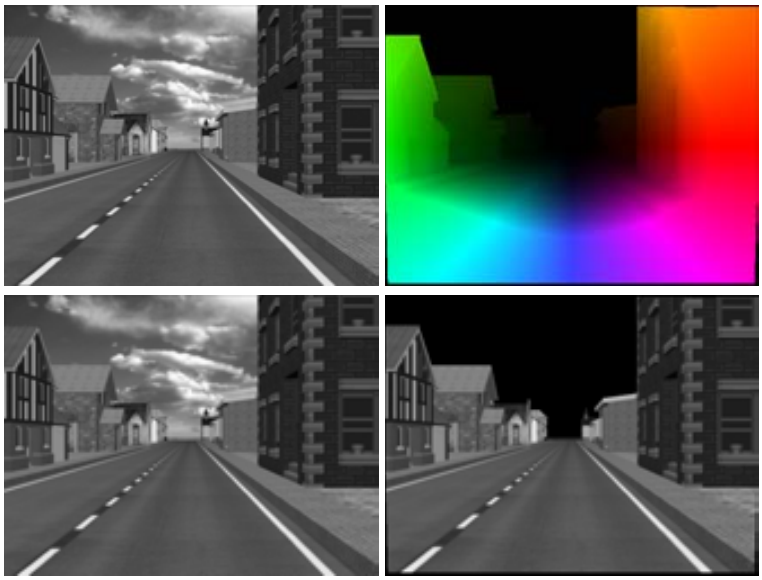


Figure 4.6: Backward registration of a pair from S1T1 sequence in our sequnces; $(left)$ An image pair. $(top-right)$ Ground-truth flow field. $(bottom-right)$ Synthesized image obtained by back-projecting frame two $(bottom-left)$ using the flow field $(top-right)$.

As presented in Section 2.4, the EISATS dataset is the most appropriated one to perform comparisons with the proposed approach. Hence, we took several pairs of images from sequence-2 of set-2 in EISATS. The similar back-registration procedure has been performed and NRMSE computed. Figs. 4.8 and 4.9 show two examples of back-registration from the EISATS sequence. Quantitative results are presented in Table 4.2. Similarly to our dataset the NRMSE is not zero, as could be expected; actually in the EISATS dataset errors are higher than in our case. One can atrribute the higher errors in EISATS to the complex scene con-

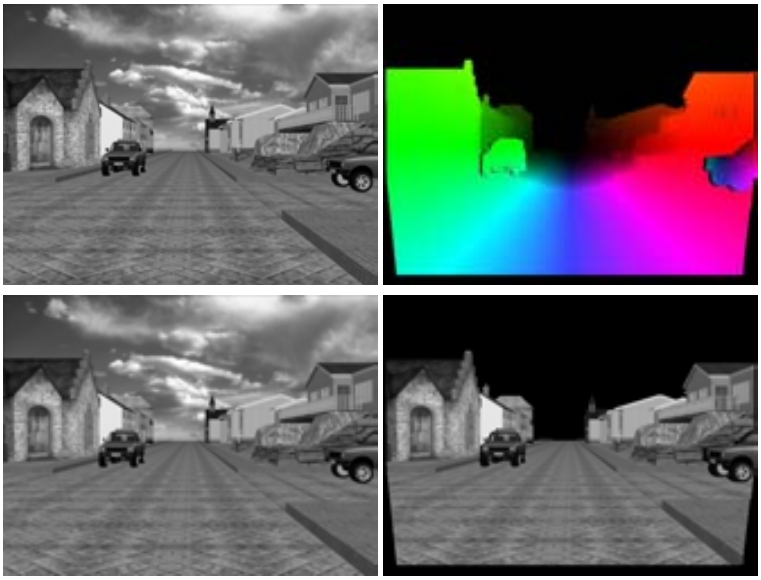


Figure 4.7: Backward registration of a pair from complex-S4T4 sequence in our sequences; $(left)$ An image pair. $(top-right)$ Ground-truth flow field. $(bottom-right)$ Synthesized image obtained by back-projecting frame two $(bottom-left)$ using the flow field $(top-right)$.

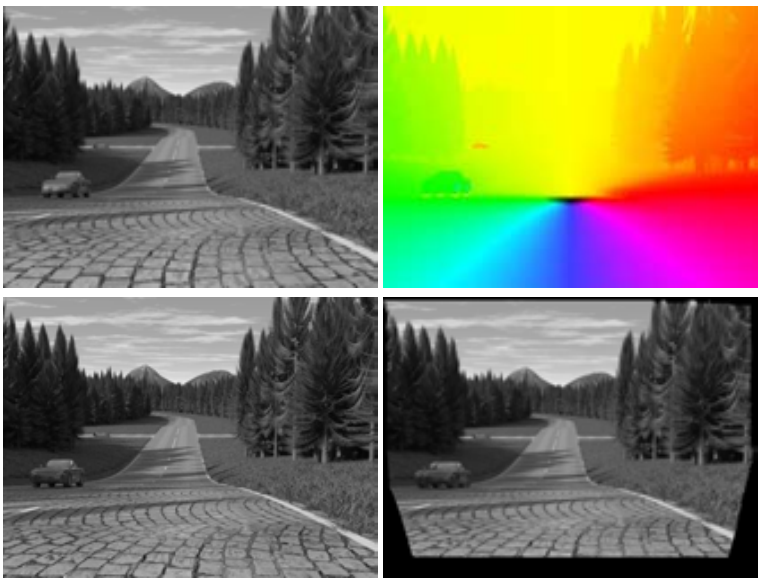


Figure 4.8: Backward registration of Pair2 in EISATS sequence; $(left)$ An image pair. $(top-right)$ Ground-truth flow field provided by EISATS. $(bottom-right)$ Synthesized image obtained by back-projecting frame two $(bottom-left)$ using the flow field $(top-right)$.

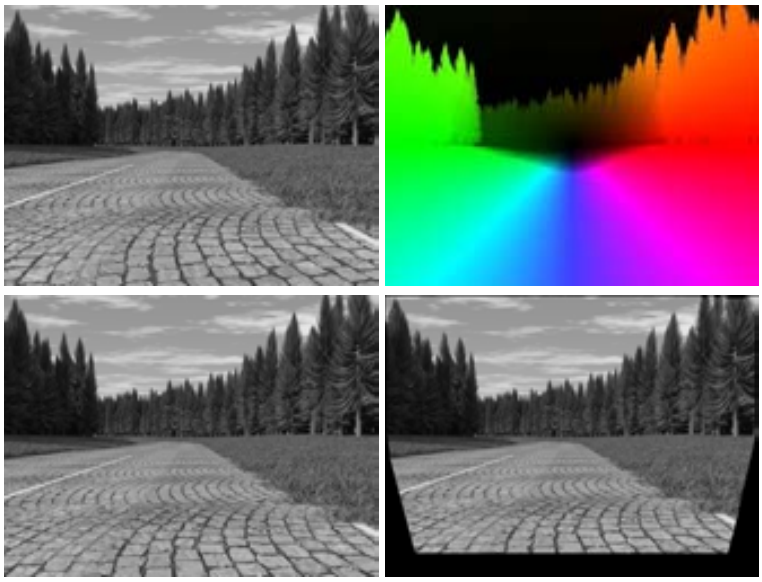


Figure 4.9: Backward registration of Pair4 in EISATS sequence; $(left)$ An image pair. $(top-right)$ Ground-truth flow field provided by EISATS. $(bottom-right)$ Synthesized image obtained by back-projecting frame two $(bottom-left)$ using the flow field $(top-right)$.

taining trees compared to building in our case. The important characteristics of our dataset is that occluded areas (black region at the boundary of moving vehicles in Fig. 4.7$(top-right)$ flow field) are identified while generating ground-truth flow vectors in those regions, whereas EISATS dataset does not give any information about occluded areas. The drawback of this can be seen in Fig. 4.8$(bottom-right)$ back-projected image. The moving vehicle boundary looks double at that position.

On further curiosity that how the state of the art optical flow methods perform on our dataset, we have computed optical flow on selected pairs of images in our sequence. We have considered four optical flow algorithms. Both average-angular-error (AAE) and average-end-point-error (EPE) are calculated and are shown in Table 4.3. These results reveal that the method MDP-Flow2 [104], which is top-rank in Middlebury evaluation, has a similar performance to Classic+NL-Fast approach [83] on our dataset. Computed flow fields for the pair from Complex-S4T4 sequence is shown in Fig. 4.10.

4.6 Conclusions

This chapter presents a framework to generate synthetic sequences using Maya. The computation of ground-truth flow fields corresponding to the generated sequences is also detailed in the current work. This framework is used to generate sequences of driving scenarios, which are used throughout this thesis to validate and evaluate our contributions. The scenarios include different speeds, different road textures, independently moving objects and complex motion of the on-board camera. The generated ground-truth data are validated by NRMSE;

Table 4.3
AVERAGE ANGULAR ERROR (AAE) AND AVERAGE END POINT ERROR (EPE), COMPUTED AS IN [13] AND [59], FOR TWO PAIRS BY DIFFERENT OPTICAL FLOW ALGORITHMS.[59]

An image pair from sequence	TV-L1[98]		Classic+NL-Fast[83]		PolarOF[5]		MDP-Flow2[104]	
	AAE	EPE	AAE	EPE	AAE	EPE	AAE	EPE
S1T1	3.3133	0.2442	1.9884	0.0841	2.4817	0.1014	1.9697	0.0888
S4T4	1.4535	0.2746	0.9879	0.1261	1.1579	0.1402	0.7870	0.1292
Complex S1T1	3.8310	0.1625	2.6277	0.1185	3.6925	0.1544	2.5182	0.1206
ComplexS4T4	2.0844	0.5386	1.5853	0.1352	1.7304	0.1374	1.3722	0.1403

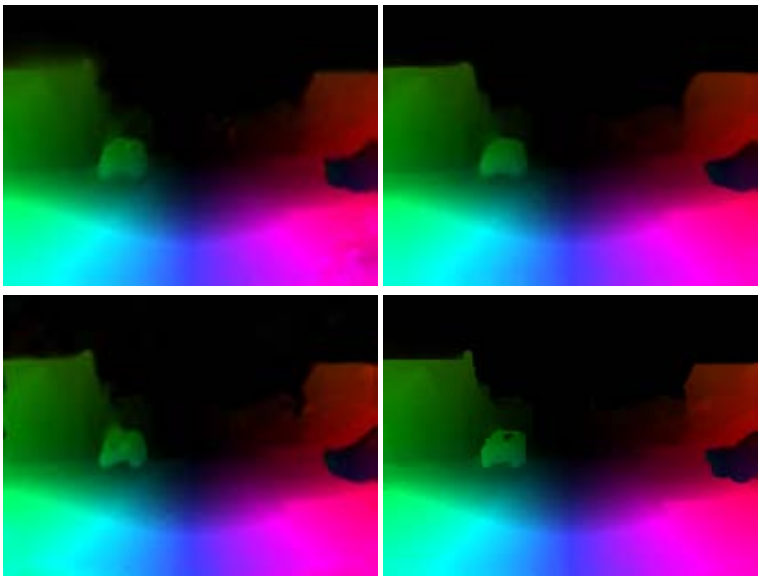


Figure 4.10: Estimated optical flow fields by different methods on a pair from Complex S4T4 sequence; $(top-left)$ TV-L1 [98]. $(top-right)$ Classic+NL-Fast [83]. $(bottom-left)$ Polar optical flow [5]. $(bottom-right)$ MDP-Flow2 [104].

its accuracy is compared to state of the art approaches those used to provide datasets of synthetic driving scenarios. We anticipate that the proposed framework and dataset will create interest in the driving assistance systems community to explore and improve current optical flow approaches. The obvious future goal tends towards incorporating more realism by motion blur, material characteristics and atmospheric effects.

Chapter 5

Speed and Texture on Optical Flow in ADAS

5.1 Introduction

In a driving scenario, optical flow is estimated between successive video frames captured by the camera mounted on a vehicle as mentioned in Chapter 1. The seminal methods to estimate optical flow were proposed in 1981 by [46] and [56]. The literature shows that there have been plenty of attempts to improve optical flow accuracy with increased interest in recent years, specifically on variational approaches those typically involves a data term and a regularization term. The balancing between the regularization and the data term has to be tuned to get better flow fields. Almost all the state of the art approaches select this weight empirically for a fixed set of images used for evaluation.

In the ADAS domain, it can happen that the vehicle is driven in different environments (e.g., urban, highway, countryside, etc.) [86] with different speeds and with different road textures, making it difficult to achieve the same optical flow accuracy all over the vehicle's trajectory, in turn it reduces the confidence and effectiveness of ADAS applications. It is very important to adjust the regularization weight based on the environment where the vehicle is being driven. This motivates us in this work to analyze the effect of some specific properties of the driving environment on the optical flow accuracy. There are many factors affecting the flow accuracy such as illumination, occlusion, specularity, texture, structure, large displacement etc. Specifically, in this work, we study the influence of on-board vision system speed as well as the road texture affect the accuracy of optical flow estimations.

As motivated by the natural way of representing a vector in terms of polar coordinates, it is also demonstrated that this representation exhibits statistical independence on image sequences of ADAS scenarios. The polar represented optical flow estimation [4] involves two regularization terms: one of orientation and the other of magnitude. This formulation gives the advantage of tuning each one of them independently unlike in cartesian represented optical flow estimation. The importance of having the right tuning of regularization terms can be appreciated in the following case study. Figure 5.1 shows image frames of different speeds and textures and the estimated flow fields using [4]. The error values (AAE: Average Angular Error; EPE: Average End Point Error) for the same flow fields are given in Table 5.1,

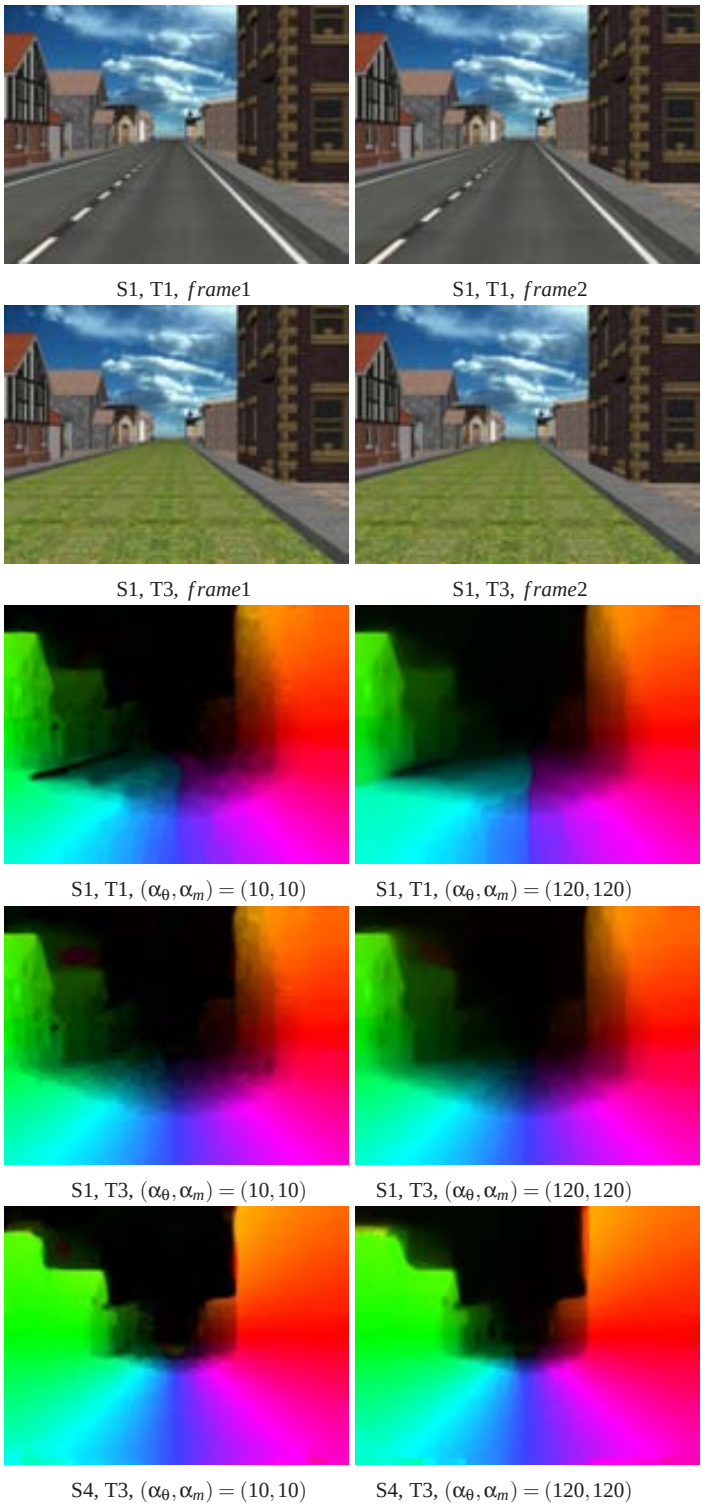


Figure 5.1: Image frames of different texture and speed, and computed optical flow for different regularization weights.

S1 corresponds to the sequence with lowest speed while S4 to the one with highest speed. On the other hand T1 corresponds to the lowest texture contrast and T3 to the highest texture contrast. Analysis of errors for a fixed set of regularization weights and different speeds and textures, in both Fig. 5.1 and Table 5.1 reveals the importance of regularization weights for an accurate flow field estimation. In the current Chapter we analyze the variation in accuracy of optical flow by varying the weights of regularization on several sequences of different speeds and road textures. Firstly, the analysis of the influence of just speed is performed. Secondly, different textures are analyzed. Finally, the analysis combining both speed and textural properties is done.

Table 5.1
AAEs AND EPEs FOR FIXED REGULARIZATION WEIGHTS FOR SEQUENCES OF DIFFERENT TEXTURES AND SPEEDS (FLOW FIELDS ARE SHOWN IN FIG. 5.1)

Sequence	(AAE, EPE) for $(\alpha_\theta, \alpha_m) = (10, 10)$	(AAE, EPE) for $(\alpha_\theta, \alpha_m) = (120, 120)$
S1,T1	(2.3974, 0.1006)	(3.4395, 0.1537)
S1,T3	(1.4916, 0.0657)	(1.5065, 0.0628)
S4,T3	(1.0999, 0.2210)	(1.1156, 0.1736)

This empirical analysis requires image sequences: i) for analyzing the influence of speed, it is needed to have sequences of different speeds with the same geometrical structure and texture; ii) for analyzing the influence of texture, it is needed to have sequences with the same geometrical scene structure but with different texture. It is impossible to have such real-life scenarios and also the corresponding ground-truth optical flow. In the current work, as presented in the previous chapter several synthetic sequences of an urban scenario for the required cases are rendered using 3D models generated with the graphic editor Maya[1]; the corresponding ground-truth flow fields are also generated using a ray-tracing technique as presented in Chapter 4.

In summary, in this chapter the statistical independence of polar representation is exploited when ADAS scenarios are considered. Additionally, the dependency of regularization weights (both for magnitude and orientation) are analyzed for different speed of the on-board vehicle camera, for different road textures, and for different combinations of both speed and texture together.

The chapter is organized as follows. Next section presents a brief comparative study of the use of polar representations with respect to cartesian in the context of ADAS applications. Section 5.3 presents the polar optical flow formulation used in the current study. The texture measures needed to evaluate the different scenarios are presented in Section 5.4 Experimental results and conclusions are given in Sections 5.5 and 5.7 respectively.

[1] www.autodesk.com/maya

5.2 Polar versus Cartesian representation of flow vectors

The most commonly used representation in optical flow estimation is through the cartesian coordinate system. However, representing a vector in terms of its magnitude an orientation is a natural way that is referred to as polar representation. As presented in [4], the analysis of spatial derivatives distribution of a flow field represented in polar shows significant statistical difference among its components as compared to that of among components of a cartesian representation. Further, the polar components show higher statistical independence compared to the cartesian components when the mutual information between derivatives of flow components in the respective representations are analyzed as shown in [4] and [77].

A similar analysis is shown in Fig. 5.2. This analysis is performed on the estimated optical flow field from a pair of images in urban driving scenario (shown in Fig.4.2 ($left-column$). Figure 5.2 shows joint histograms of flow derivatives in both cartesian and polar coordinate systems. The mutual information (MI) between the coordinate components computed using these joint histograms are depicted on top of each plot in Fig. 5.2. Lower the values of MI, higher the statistical independence. As can be observed from the illustrations in Fig. 5.2, the representation of flow field in polar is more independent than the cartesian system. Similar analysis on the ground-truth flow field between the same pair of images has shown zero MI (in the cases of ($bottom-left$) and ($bottom-right$) plots in Fig. 5.2) for polar coordinates. For cartesian coordinates (in the cases of Fig. 5.2 ($top-left$) and ($top-right$)) the MI values are 0.27082 and 0.50335 respectively, when ground-truth flow field is considered. This shows that in the ideal case of translational motion, polar coordinates are mutually exclusive (totally independent).

Polar representation of flow vectors for optical flow estimation is proposed in [4] and its implications are studied. It is shown that polar represented optical flow performs almost similar as with the state of the art cartesian coordinates represented optical flow estimation on traditional image data sets. Further, it is shown that for specular and fluid flow image sets, polar representation adds the advantage by allowing regularization in either of the coordinate components independently. In the vehicle driving scenario, the majority of the motion is translation. The expected flow field in such a scenario is diverging and the variation in magnitude is higher compared to the variation in orientation. In such a motion scenario, polar optical flow becomes convenient. The current work exploits the possibility of independent tuning of regularization terms.

5.3 Overview of polar optical flow

As presented in Chapter 2 a typical variational formulation of optical flow energy function using cartesian representation looks like:

$$E(u,v) = \int\int_{\Omega} \{\underbrace{(I(x+u,y+v,t+1)-I(x,y,t))}_{Data\ Term} + \alpha\,(\underbrace{|\nabla u_1|^2+|\nabla u_2|^2}_{Regularization})\}\,dx\,dy\,, \tag{5.1}$$

that contains a data term and a regularization term. Where $I(x,y,t)$ is the pixel intensity value

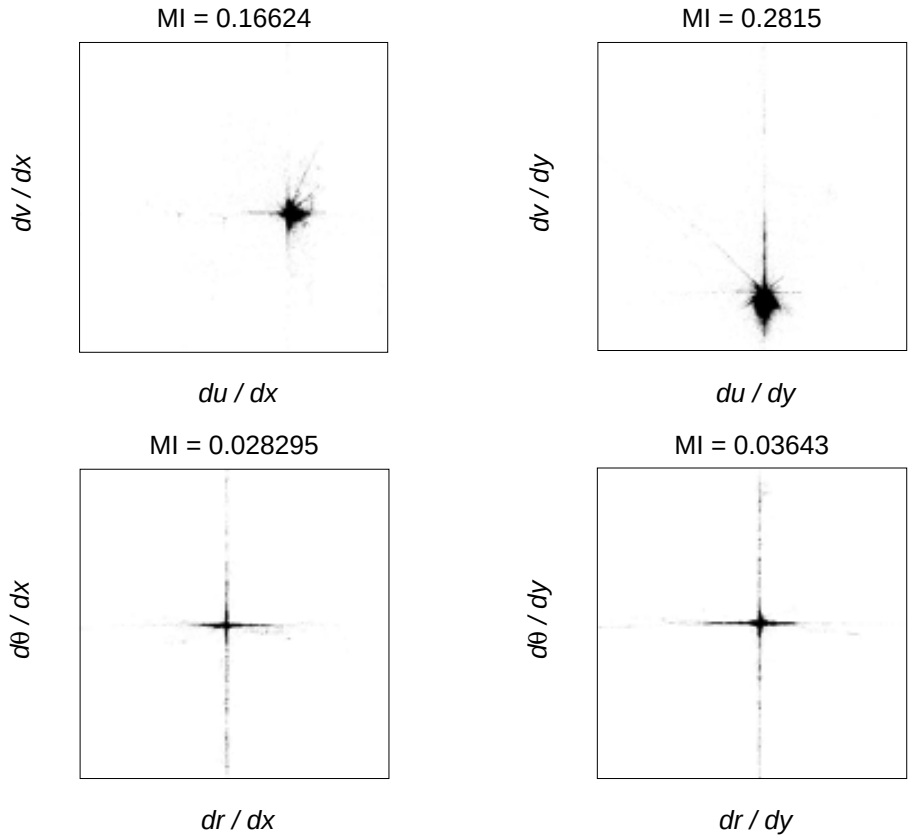


Figure 5.2: Joint histograms of flow derivatives in cartesian and polar coordinates of an estimated flow field in a synthetic sequence of an urban road scenario. On top of each plot MI value is depicted.

at (x,y) at time t; α is the regularization weight; (u,v) is the flow field vector to be estimated using Euler-Lagrange equations [20] or alternative methods [98].

This section presents a brief description of the polar optical flow formulation proposed in [4]. According to that work, the flow vector at a pixel (x,y) can be represented in terms of polar coordinates as:

$$flow(x,y) = (m(x,y), \theta(x,y)) \,, \tag{5.2}$$

where m is the magnitude and θ is the orientation at (x,y). The energy formulation using the polar representation allows us to separate the regularization terms as defined below:

$$\begin{aligned} E(\theta(x,y), m(x,y)) = \\ \int\int_{\Omega} \{\psi(I(x+m\cos\theta, y+m\sin\theta, t+1) - I(x,y,t)) \\ + \alpha_\theta \psi_\theta(\rho_\theta(\theta)) + \alpha_m \psi_m(\rho_m(m)) \} \, dx \, dy \,, \end{aligned} \tag{5.3}$$

where ψ is a robust penalty function for the data term; ψ_θ and ψ_m are robust penalty func-

tions respectively for orientation and magnitude components' regularization (see [4] for more details); similarly, α_θ and α_m are regularization weights; ρ_θ and ρ_m are differential operators (in a simpler case first derivative). All these ρ_*, ψ_* and α_* can be varied depending on image sequences or the application of interest.

To avoid the difficulty of m being negative, the following equivalence relation is defined over values of m and θ:

$$(m,\theta) \sim \begin{cases} (m,\theta) & \text{if } m > 0 \\ (-m,\theta+\pi) & \text{if } m < 0\,. \end{cases} \tag{5.4}$$

Due to the periodic nature of θ, the orientation is expressed in terms of two parameters as:

$$\begin{aligned} s(x,y) &= \sin\theta(x,y)\,, \\ c(x,y) &= \cos\theta(x,y)\,, \end{aligned} \tag{5.5}$$

where the constraint $s^2+c^2=1$ is called as *coherence constraint* which ensures proper representation of orientation.

Using Lagrange multiplier λ and assuming it as a pre-determined parameter, the energy function can be formulated to minimize three parameters (c,s,m) as:

$$\begin{aligned} E(c,s,m) = \int\int_\Omega \{\, & \lambda(s^2+c^2-1)^2 \\ & +\psi(I(x+mc,y+ms,t+1)-I(x,y,t)) \\ & +\alpha_\theta\psi_\theta(\rho_\theta(c),\rho_\theta(s))+\alpha_m\psi_m(\rho_m(m))\,\}\,dx\,dy\,, \end{aligned} \tag{5.6}$$

where λ is pixel-wise pre-determined parameter updated every iteration as $\lambda = e^{(s^2+c^2-1)^2}$ using previous iteration values of c and s. Equation 5.6 can be minimized using Euler-Lagrange equations.

5.4 Texture measures

In order to study the effect of texture on the optical flow accuracy, it is necessary to quantify the texture property. There are several ways of measuring the texture content of a given sequence [102]; in the current work three of the most widely used statistical texture metrics: *contrast*, *correlation* and *homogeneity*, are considered. These metric values are computed over co-occurrence matrix of gray values of images [44] and are correlated with the optical flow error measures. The texture metrics computed over co-occurance matrix that is also called normalized Gray-Level Co-occurrence Matrix (GLCM) of an image are defined as:

$$Contrast = \sum_{n=0}^{N_g-1} n^2\{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p(i,j)\};\ |i-j|=n\,, \tag{5.7}$$

$$Correlation = \frac{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(ij)p(i,j)-\mu_x\mu_y}{\sigma_x\sigma_y}\,, \tag{5.8}$$

$$Homogeneity = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p(i,j)}{1+|i-j|} , \tag{5.9}$$

where $p(i,j)$ is the $(i,j)^{th}$ entry in normalized GLCM; N_g is the number of distinct gray levels in the quantized image; μ_x, μ_y, σ_x, and σ_y are the means and standard deviations of p_x and p_y: $p_x(i) = \sum_{j=1}^{N_g} p(i,j)$ and $p_y(j) = \sum_{i=1}^{N_g} p(i,j)$.

5.5 Experimental analysis

This section presents the empirical study of the optical flow accuracy of scenes where i) the camera moves at different speeds; ii) the texture of the scene changes; and iii) both speed and texture changes are considered together. First, we perform the study for all these three cases on a set of simple sequences, where there is no complexity, the vehicle's camera moves straight on a road with different speeds and with different road textures. Such a simple sequence enables us to analyze the influence of speed and textures easily. Later on we also present the study of influence of speed and texture together with another set of sequences that has complex egomotion (for details on sequence generation see Chapter 4).

5.5.1 Analysis for speed

From the framework presented in Chapter 4, four sequences of different speeds are considered. They have an incremental translation of 0.25cm, 0.5cm, 0.75cm and 1cm along the optical axis of the vehicle camera in the Maya model. Let us call these sequences as S1, S2, S3 and S4 in the increasing order of speed. The ground-truth optical flows for these sequences are also considered. The scene and texture of all these sequences are as shown in Fig. 4.2. The first aim is to study the optical flow accuracy for the change in speed and to find its relationship with respect to the regularization parameters in the optical flow formulation. We are using the polar optical flow presented in section 5.3 since this formulation provides the possibility of tuning different regularization parameters separately which is an attractive feature in the ADAS domain. Furthermore, it involves two regularization terms that allow an independent study of their influence. Initially, an experimentation is performed to find the optimal range for the regularization weights. It consists of computing optical flow on a pair of images from one of the sequences for a wide range of weights of both regularization terms. From this experiment it is determined that the following range of values for experimentation: 1, 2.5, 5, 10, 20, 30, ..., 120 is sufficient.

For further analysis of influence of speed, it is not good to have equal number of frames in all of the sequences of different speeds. Since each sequence has different displacement per frame, having equal number of frames in all of them will result in that the vehicle camera moved different distances and end-up processing different scenes. Since there is different scene geometry with different buildings in the 3D model, n^{th} frame in S1 will have different scene geometry from the n^{th} frame in S2 and S3 and S4. Since the scene geometry also affects optical flow accuracy, in this experiment, we have generated sequences of different speeds but the vehicle camera travelling a constant distance in all of them along the camera axis of the 3D model, hence generating varying number of frames in different sequences. In this way, all the sequences cover exactly the same geometric scene, but with different number of frames.

So we have 40 frames in S1, 20 frames in S2, 13 frames in S3 and 10 frames in S4 (see Table 4.1). The average of error measures of all the frames in a sequence are considered for analysis. We have considered both AAE and EPE for analysis. All the errors in this analysis are computed over a RoI (Region of Interest) of size 320×480 in the center of the flow field. The considered RoI is shown in Fig. 5.3 $(left)$.

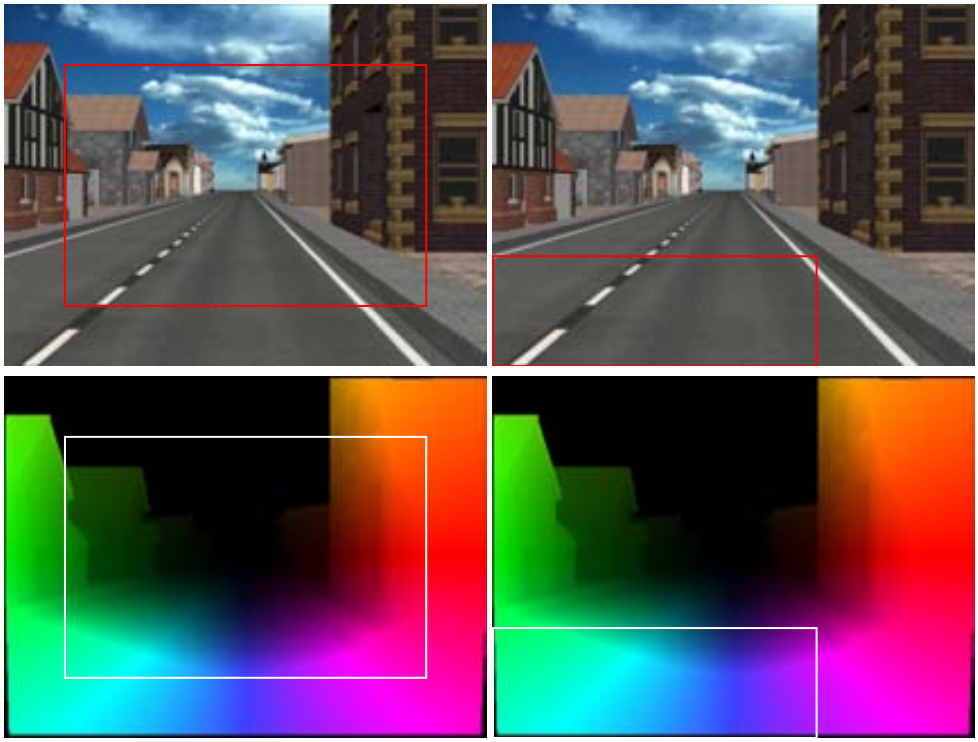


Figure 5.3: RoIs used to calculate error measures. $(left)$ Speed analysis; $(right)$ Texture and Texture together with Speed analysis.

Figure 5.4 shows a 3D representation of AAE for sequence S1 for varying values of the two regularization weights α_θ and α_m. The 3D error representations of AAEs from all the four sequences are shown in Fig. 5.5. Minimum AAEs and the corresponding regularization weights for all the sequences are given in Table 5.2. Observing the meshes in Fig. 5.5 and by analyzing the minimum AAE values in Table 5.2, we can conclude that the error in the sequence of lower speed is always higher than the error in the sequence of higher speed, at almost all combinations of regularization weights. The values of α_θ and α_m in Table 5.2 reveals that α_θ is constant around 2.5 and 5, where α_m values decrease as the speed increases. It can be inferred that, overall AAE decreases with the increase in speed of the vehicle, and α_θ has to be slightly increased and α_m should be tuned with the change in the speed of the vehicle.

A similar analysis is also done using EPE. Figure 5.6 shows the 3D representation of EPE for S1 for all combinations of two regularization weights. The 3D representations of EPEs

of all four sequences are depicted in Fig. 5.7. The minimum EPEs for all four sequences with corresponding regularization weights are shown in Table 5.3. It is observed by the error maps in Fig. 5.7 and the Table 5.3 that the EPE in a lower speed sequence is lower than in a sequence of higher speed for any combination of both regularization weights. In Table 5.3, α_θ increases from a smaller value as the speed increases, whereas α_m keeps constant at around value 60. From the point of view of EPE, α_m has to be kept constant at a higher value and α_θ should be tuned according to the change in speed of the vehicle. One interesting conclusion from this first study is that depending on the required accuracy (AAE or EPE, i.e., angular or magnitudinal), needed for a given application, different tuning of regularization parameters have to be applied. Furthermore it is clear that there is a relationship between this parameter tuning and the current speed of the vehicle.

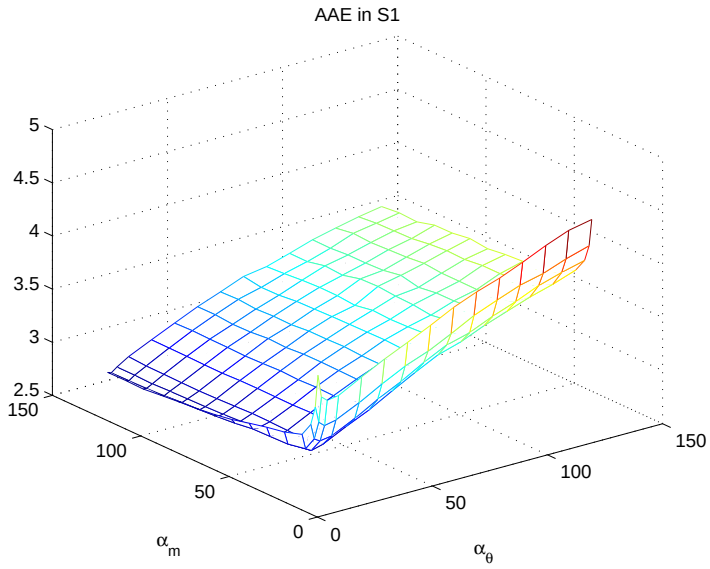


Figure 5.4: 3D plot of AAEs from S1 for varying α_θ and α_m values.

Table 5.2
REGULARIZATION PARAMETER VALUES PRODUCING LOWEST AAES IN EACH OF THE SEQUENCES

Sequence	α_θ	α_m	AAE
S1	2.5	120	2.9261
S2	2.5	90	2.2481
S3	5	60	1.7618
S4	5	50	1.5695

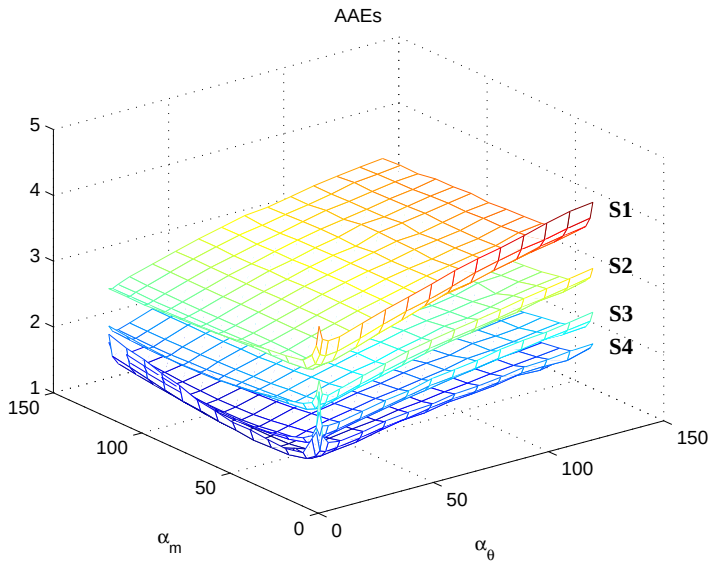


Figure 5.5: 3D plot of AAEs of all the sequences for varying α_θ and α_m values.

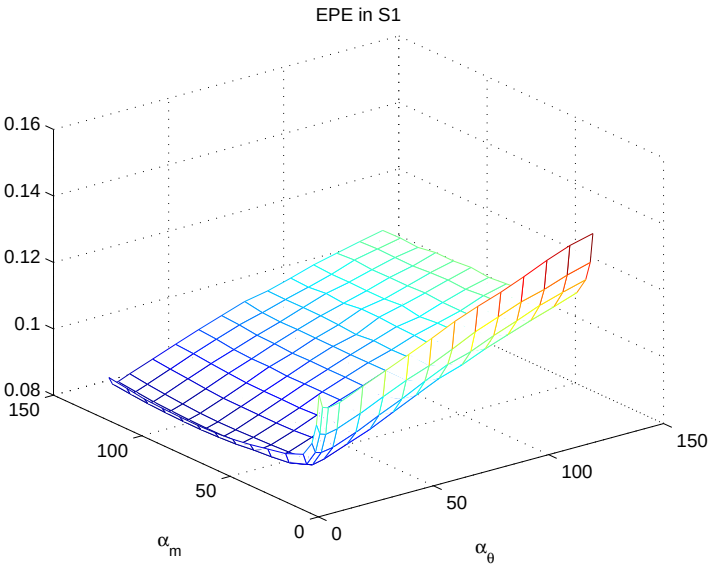


Figure 5.6: 3D plot of EPEs from S1 for varying α_θ and α_m values.

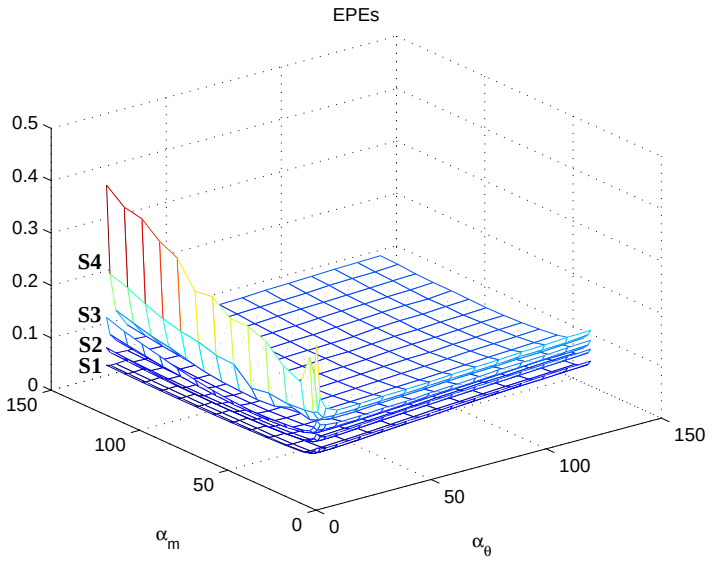


Figure 5.7: 3D plot of EPEs of all the sequences for varying α_θ and α_m values.

Table 5.3
REGULARIZATION PARAMETER VALUES PRODUCING LOWEST EPES IN EACH OF THE SEQUENCES

Sequence	α_θ	α_m	EPE
S1	2.5	70	0.0887
S2	5	60	0.1063
S3	10	60	0.1163
S4	20	60	0.1338

5.5.2 Analysis for texture

The aim of the work in this section is to analyze the influence of road texture on the optical flow accuracy and to identify the way of adjusting the regularization weights for better results. We have generated several sequences with different road textures, some of the images of these sequences are shown in Fig. 4.3. The study in this section is performed considering three sequences with the increasing value of texture *contrast*. Hereinafter they are referred to as T1, T2 and T3. These sequences are of the same speed but with different road textures. The texture metrics are computed over a small RoI of size (146×430) on the road surface. This RoI is shown in Fig. 5.3 $(right)$. Again in this section the polar represented optical flow described in section 5.3 is used. The optical flow is computed on all the image pairs from these sequences which were obtained assuming the on-board vision system is travelling at the same speed. The average error values of all the flow fields in the same small RoI (where texture metrics were calculated) in a sequence are computed. We are considering both AAE and EPE for analysis. Table 5.4 gives the texture metrics for the sequences. Figs. 5.8 and 5.9 shows 3D representations of AAEs and EPEs respectively for three sequences of different textures T1, T2 and T3. The minimum AAEs and EPEs with the corresponding regularization weights are shown in Tables 5.5 and 5.6 respectively. By observing Figs. 5.8 and 5.9, and Tables 5.5 and 5.6, it can be easily confirmed that both AAE and EPE measures decrease with the increase in texture contrast. The regularization weights in the Tables 5.5 and 5.6 reveal that both values should increase with the increase in texture *contrast* for better results. Similarly, these results can be correlated with other textural properties such as *correlation* and *homogeneity* in Table 5.4.

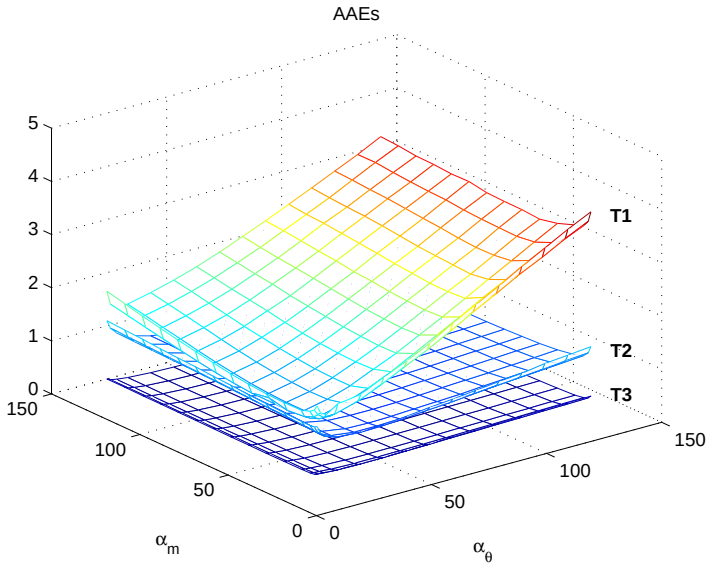


Figure 5.8: 3D plot of AAEs from three different textured sequences for varying α_θ and α_m values.

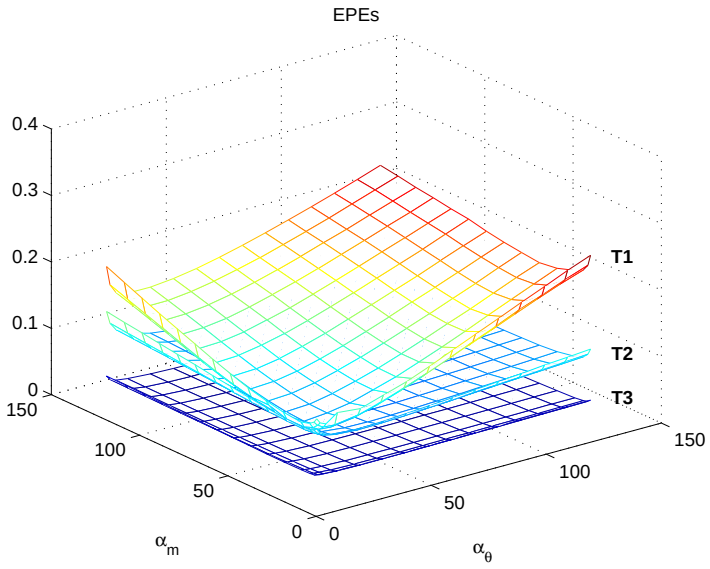


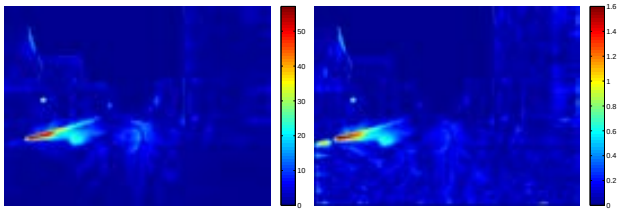
Figure 5.9: 3D plot of EPEs from three different textured sequences for varying α_θ and α_m values.

Table 5.4
TEXTURE METRICS FOR DIFFERENT SEQUENCES

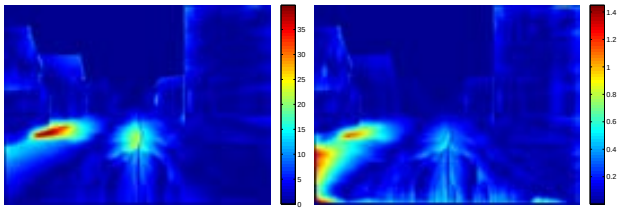
Sequence	*Contrast*	*Correlation*	*Homogeneity*
T1	0.0488	0.9386	0.9765
T2	0.0850	0.8611	0.9575
T3	0.1255	0.7168	0.9372

Table 5.5
REGULARIZATION PARAMETER VALUES WITH LOWEST AAES

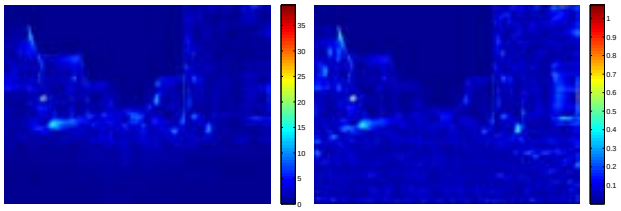
Sequence	α_θ	α_m	AAE
T1	10	20	1.5564
T2	40	90	0.7237
T3	60	120	0.6057



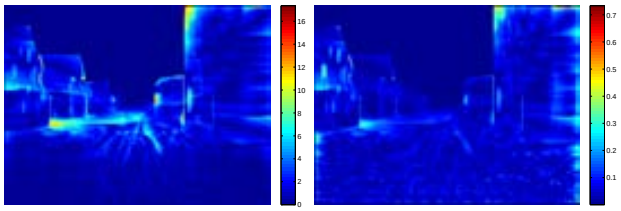
S1, T1, $(\alpha_\theta, \alpha_m) = (10, 10)$



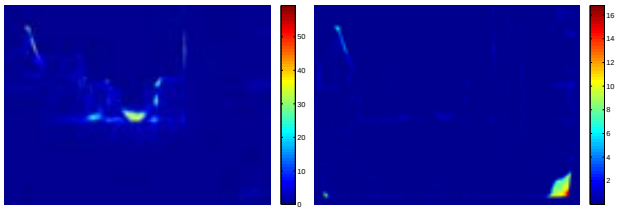
S1, T1, $(\alpha_\theta, \alpha_m) = (120, 120)$



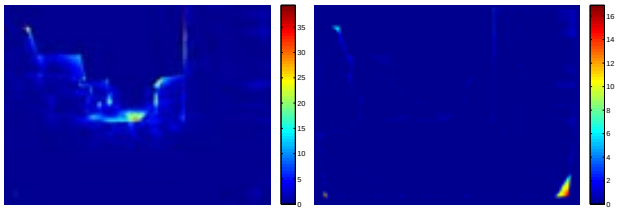
S1, T3, $(\alpha_\theta, \alpha_m) = (10, 10)$



S1, T3, $(\alpha_\theta, \alpha_m) = (120, 120)$



S4, T3, $(\alpha_\theta, \alpha_m) = (10, 10)$



S4, T3, $(\alpha_\theta, \alpha_m) = (120, 120)$

Figure 5.10: Error images for the same image pairs shown in Fig. 5.1 and Table 5.1. $(left)$ AAEs; $(right)$ EPEs

Table 5.6
REGULARIZATION PARAMETER VALUES WITH LOWEST EPES

Sequence	α_θ	α_m	EPE
T1	10	20	0.1095
T2	60	80	0.0548
T3	80	120	0.0466

5.5.3 Analysis for both speed and texture

Further we have performed experiments to analyze the influences of both speed and texture together. We use twelve different sequences of four different speeds and three different textures. Optical flow is estimated on all the frames in these sequences and errors are computed. The error for a particular sequence is the average of errors in all the flow fields in that sequence. Error heatmaps are shown in Fig. 5.10 for the same flow fields shown in Fig. 5.1. The errors are calculated on a small region of interest of size (146×430 on the road surface), which is the same as in the previous subsection. Table 5.7 shows minimum AAEs for twelve different sequences of different speeds and textures. The regularization weights corresponding to the minimum errors are also mentioned in brackets in Table 5.7. 3D plots of minimum AAEs, and corresponding α_θs and α_ms are shown in Fig. 5.11, 5.12 and 5.13 respectively. From Table 5.7 and Fig. 5.11, we can notice that AAE reduces with the increase in texture *contrast* also with the increase in speed. With respect to AAE, Fig. 5.12 indicates that α_θ has to be kept small and increase slightly when speed increases for a sequence of lower texture, but it has to be higher and has to increase when speed increases for a sequence of higher texture. In conclusion, α_θ has to increase with the increase in speed and texture. The 3D representation in Fig. 5.13 indicates that α_m has to decrease with the increase in speed and has to increase with the increase in texture.

Table 5.7
MINIMUM AAES AND REGULARIZATION WEIGHTS (α_θ, α_m)

Sequence	T1	T2	T3
S1	1.5564 (10,20)	0.7237 (40,90)	0.6057 (60,120)
S2	0.9472 (10,10)	0.5307 (50,60)	0.4126 (80,70)
S3	0.7402 (20,10)	0.4382 (70,50)	0.3305 (100,70)
S4	0.6694 (30,5)	0.3984 (80,20)	0.2906 (110,20)

A similar experiment on all the twelve sequences is performed considering EPEs. Table

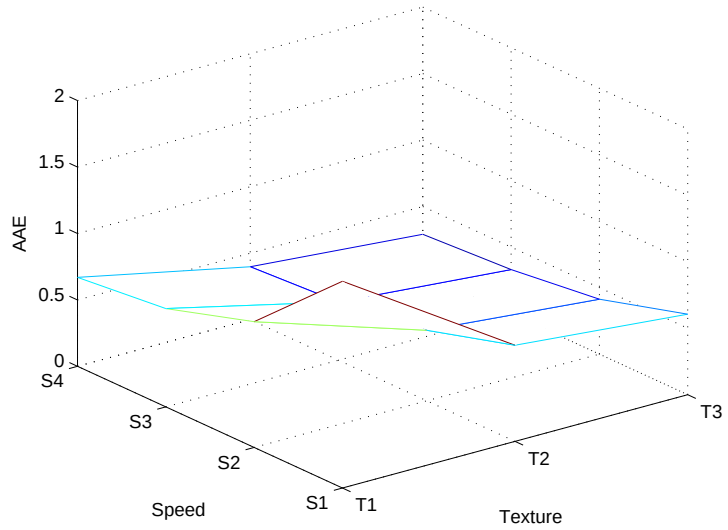


Figure 5.11: 3D plot of minimum AAEs for all sequences with different speeds and textures for varying α_θ and α_m values.

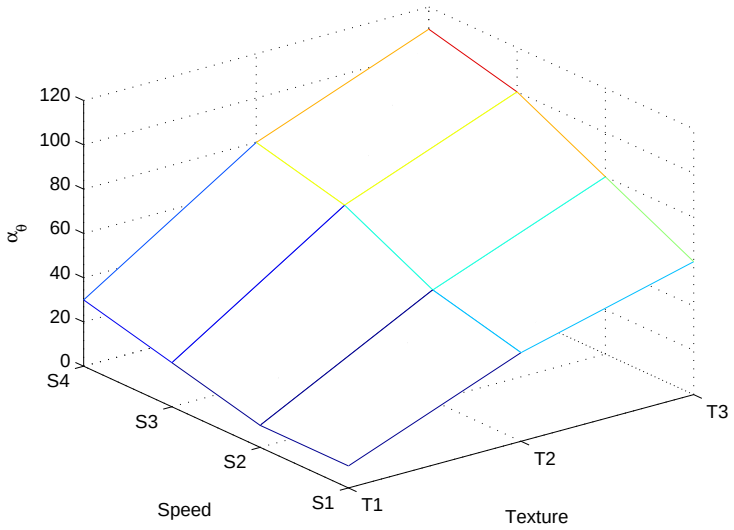


Figure 5.12: 3D plot of α_θ corresponding to minimum AAEs for all sequences with different speeds and textures.

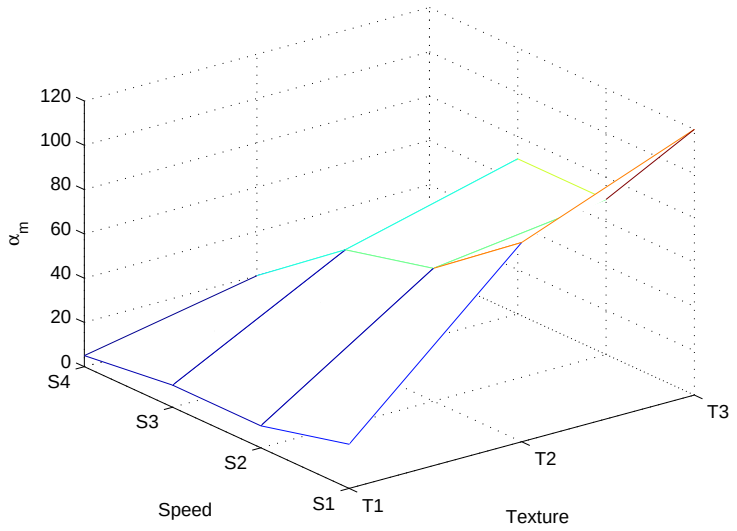


Figure 5.13: 3D plot of α_m corresponding to minimum AAEs for all sequences with different speeds and textures.

5.8 shows minimum EPEs and the corresponding regularization weights in brackets. Figure 5.14, 5.15 and 5.16 are the 3D representations of minimum EPEs, α_θs and α_ms respectively. In Table 5.8 and Fig. 5.14 we can observe that EPE reduces with increase in texture contrast but increases with the increase in speed. Figure 5.15 shows that α_θ has to increase with the increase in speed as well as with the increase in texture contrast. Whereas the 3D representation in Fig. 5.16 indicates that α_m has to increase with the increase in texture, but has to decrease with the increase in speed except for lower textured sequences (e.g., sequence with texture T1 and with speeds S1, S2, S3 and S4).

Further, to assure the conclusions made out from Table 5.2 and Table 5.7 about decrease in AAE with the increase in speed, and conclusion that AAE will decrease with the increase in texture contrast, we analyzed the AAEs of all twelve sequences keeping α_θ and α_m constant at different values. Table 5.9 shows AAEs in all sequences for fixed α_θ and α_m values at 40. The AAEs in this Table confirms our conclusion that AAE decreases with the increase in speed as well as texture. Comparing the error values in Table 5.9 and Table 5.7, it is clear that tuning of regularization weights is very important in getting accurate optical flow. Similarly Table 5.10 shows EPEs for fixed α_θ and α_m value of 40 for the sequences. This also re-affirms that EPE increases with the increase in speed but decreases with the increase in texture, and comparing values from Table 5.10 and Table 5.8 reveals that tuning of regularization weights is needed.

Finally, a similar analysis than the previous one has been performed but by adding complexity to the motion. The new sequences involve large changes in (*yaw* and *pitch*) angles. All these sequences are of ten frames long. The optical flow is computed for varying regularization weights and the errors are computed on a small RoI (the one shown in Fig. 5.3

Table 5.8
MINIMUM EPES AND CORRESPONDING REGULARIZATION WEIGHTS (α_θ, α_m)

Sequence	T1	T2	T3
S1	0.1095 (10,20)	0.0548 (60,80)	0.0466 (80,120)
S2	0.1434 (20,10)	0.0809 (90,70)	0.0664 (120,120)
S3	0.1735 (20,10)	0.1061 (110,60)	0.0854 (120,80)
S4	0.2223 (60,10)	0.1435 (120,10)	0.1141 (120,50)

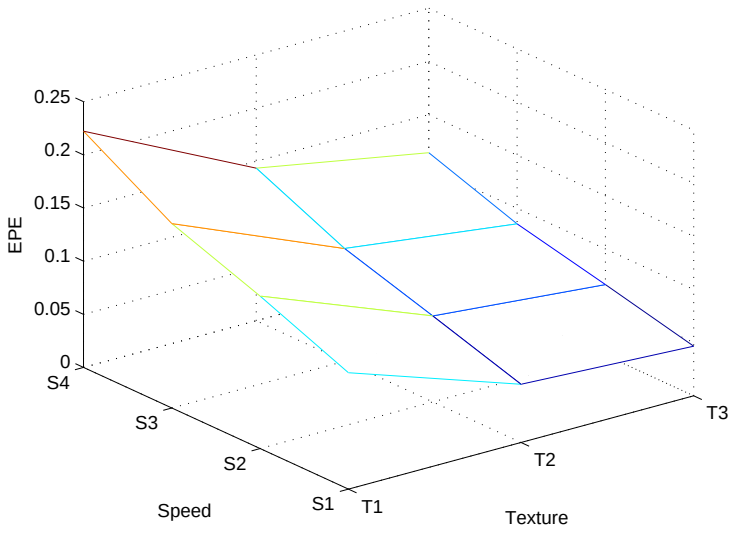


Figure 5.14: 3D plot of minimum EPEs for all sequences with different speeds and textures for varying α_θ and α_m values.

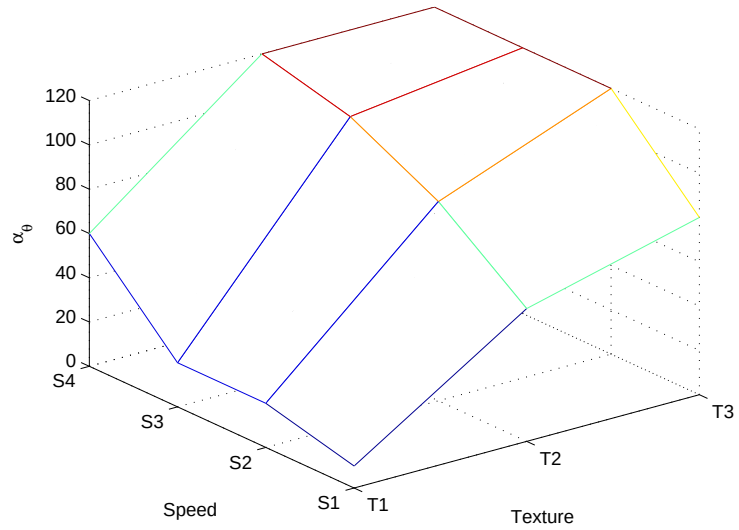


Figure 5.15: 3D plot of α_θ corresponding to minimum EPEs for all sequences with different speeds and textures.

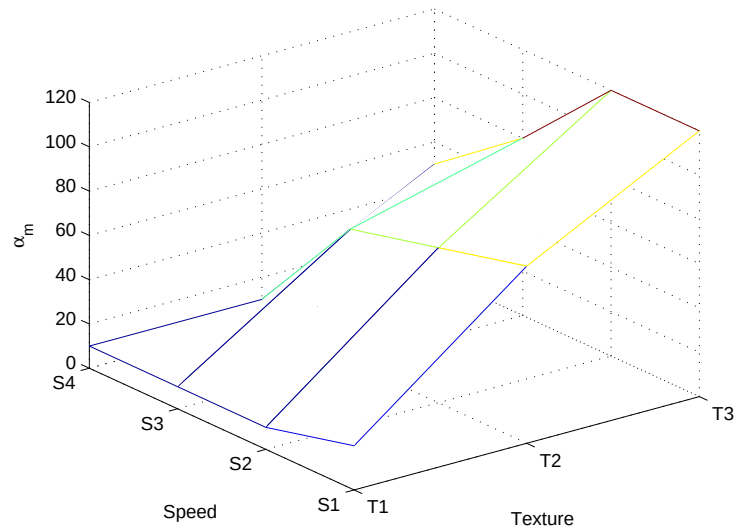


Figure 5.16: 3D plot of α_m corresponding to minimum EPEs for all sequences with different speeds and textures.

Table 5.9
AAEs FOR FIXED REGULARIZATION WEIGHTS: $\alpha_\theta = 40$ AND $\alpha_m = 40$

Sequence	T1	T2	T3
S1	1.9815	0.7376	0.6282
S2	1.0912	0.5336	0.4154
S3	0.8105	0.4447	0.3371
S4	0.7250	0.4116	0.2979

Table 5.10
EPEs FOR FIXED REGULARIZATION WEIGHTS: $\alpha_\theta = 40$ AND $\alpha_m = 40$

Sequence	T1	T2	T3
S1	0.1333	0.0559	0.0494
S2	0.1649	0.0828	0.0696
S3	0.1988	0.1112	0.0915
S4	0.3049	0.1672	0.1255

$(right)$). The minimum AAE and EPE are shown in Tables 5.11 and 5.12 respectively for all the twelve different sequences in this complex set. All these sequences have the same degree of egomotion, but the on-board camera moves at different speeds and on different textures. Here we can observe almost the same trends in error values and regularization weights as in the previous study. α_θ has to be increased when speed and texture contrast increase, for both AAE and EPE, whereas α_m has to be increased when texture contrast increases for both AAE and EPE. Since sequences have egomotion, changes in α_m does not affect much in AAE, it is almost constant with increase in speed for AAE and it has to slightly decrease with increase in speed for EPE.

For the completeness of our study we have added few independently moving vehicles in the scene and performed similar analysis. In this particular case, the RoI corresponds to the one shown in Fig. 5.3($left$). As expected, independently moving objects are another source of errors that cannot be tackled merely by tuning regularization weights although it could improve. The relative motion of on-board camera and moving vehicles causes different groups of flow vectors. Errors due to occlusions and due to moving objects present in the scene are not under our control by tuning regularization weights. Overall we can tune on the things which are related with static and under our control (such as speed and texture), but not on the behavior of dynamic moving vehicles present in the given scenario.

5.6 Discussion

Although out of the scope of current work the question on how we can tune the regularization parameters could arise. In general, from the study presented above we can say that having

Table 5.11
MINIMUM AAES AND CORRESPONDING REGULARIZATION WEIGHTS (α_θ, α_m)

Sequence	T1	T2	T3
S1	1.6947 (5,10)	0.7859 (30,60)	0.7030 (40,60)
S2	1.2964 (10,10)	0.6670 (40,50)	0.5370 (40,60)
S3	1.2213 (30,20)	0.5392 (60,50)	0.4328 (70,60)
S4	0.9361 (40,20)	0.4582 (80,20)	0.3469 (110,60)

Table 5.12
MINIMUM EPES AND CORRESPONDING REGULARIZATION WEIGHTS (α_θ, α_m)

Sequence	T1	T2	T3
S1	0.1388 (10,30)	0.0589 (60,80)	0.0520 (70,110)
S2	0.1644 (20,20)	0.0830 (80,70)	0.0683 (100,100)
S3	0.2240 (40,20)	0.1035 (90,50)	0.0824 (120,90)
S4	0.2529 (60,20)	0.1271 (110,30)	0.1000 (120,50)

the best set of parameters would depend on current scenario, however for a given set of regularization parameters (independently whether it is the best set or not) we can adapt its values according with the speed and texture using the information presented in previous section. We perceive that this analysis is just a tip on how to proceed, a rigorous study and validation should be performed to define a rule to adapt regularization parameters to a particular characteristics of a sequence. A much more deep study is required to conclude adaptation rules in the case of combinations of several characteristics of any sequence. The study in this work is a starting point in that direction.

Even though an increase in speed of vehicle camera can be compensated by increasing the camera cycle time, that is by higher number of frames per second (FPS), the increase in number of frames leads to higher computation burden. A related work [48] in that direction proposes to change resolution of images for varied FPS in a variational hierarchical framework. Moreover the maximum number of FPS of a camera is limited by its hardware. In the current work, the influence of vehicle speed for a fixed cycle time of a camera is considered.

5.7 Conclusions

This chapter shows that the polar representation of flow vectors is very convenient in ADAS scenarios due to its freedom of weighting regularization terms differently and further uses polar represented optical flow estimation for the analysis. The analysis of optical flow accuracy to specific characteristics of a driving scenario such as vehicle speed and road texture is performed. It is concluded that there is a need to tune the regularization parameters depending on the needed accuracy (angular or magnitudinal) and for varying speeds and textural properties of the road.

Chapter 6

Applications of Optical Flow in ADAS

6.1 Introduction

This chapter presents some of the usages of optical flow in ADAS for preprocessing tasks, such as camera pose and horizon line estimation and application tasks such as egomotion estimation. Additionally it presents the adaptation of an existing state of the art optical flow method to driving scenarios through the change in regularization of flow components from gradient to Laplacian.

6.2 Vanishing point and horizon line estimations

This section presents a robust technique for estimating on-board monocular vision system pose. The proposed approach is based on dense optical flow that is robust against shadows, reflections and illumination changes. A RANSAC based scheme is used to cope with the outliers in the optical flow. The proposed technique is intended to be used in driver assistance systems for applications such as obstacle or pedestrian detection. Experimental results on different scenarios, both from synthetic and real sequences, show the usefulness of the proposed approach.

On-board vision systems can be classified into two different categories: *monocular* or *stereo*. Although each one of them has its own advantages and disadvantages, both approaches have a common problem: real-time estimation of on-board vision system pose—position and orientation—, which is a difficult task since: (a) the sensor undergoes motion due to the vehicle dynamics, and (b) the scene is unknown and continuously changing.

The current work proposes a novel approach for estimating camera's position and orientation for monocular vision system, which are finally represented as a single value. It is based on dense optical flow estimated by means of the TV-L^1 formulation. The main advantage of the proposed approach with respect to other monocular based approaches is that it does not require feature extraction neither imposes restrictive assumptions. The advantage with respect to the previous optical flow based approaches is that the current one is based on an accurate variational dense optical flow formulation. Finally, since it is based on a monocular vision, a system cheaper than stereo based solutions can be reached.

The remainder of this work is organized as follows. Section 6.2.1 briefly introduces the TV-L^1 formulation used to compute dense optical flow, together with the proposed adaptation to reduce the processing time and to increase the accuracy of the flow estimation. The model formulation and proposed approach are presented in Sections 6.2.2 and 6.2.3 respectively. Experimental results on different sequences/scenarios are presented in Section 6.2.4. Finally, conclusions are given in Section 6.2.5.

6.2.1 TV-L^1 optical flow

State of the art in optical flow techniques unveil that variational approaches give dense estimation with more accuracy as compared to other approaches. TV-L^1 technique is a variational optical flow technique proposed in [107] that gives dense flow field. In the current work, an improved version [97] is used, which is briefly presented in this section. As presented in Chapter 2 the formulation of the variational method proposed by Horn and Schunck [46] involves a data term and a regularization term. The formulation in [97] also involves an optical flow constraint and a regularization term but both of them with L^1 norm. The TV-L^1 optical flow is obtained by minimizing the following energy function:

$$E = \int_{\Omega} \{ \alpha \underbrace{|I_1(\boldsymbol{x}+\boldsymbol{u}(\boldsymbol{x})) - I_0(\boldsymbol{x})|}_{Data\ Term} + \underbrace{|\nabla \boldsymbol{u}|}_{Regularization} \} \, d\boldsymbol{x}, \tag{6.1}$$

where I_0 and I_1 are two images; $\boldsymbol{x} = (x_1, x_2)$ is the pixel location within a rectangular image domain $\Omega \subseteq \mathbf{R}^2$; and $\boldsymbol{u} = (u_1(\boldsymbol{x}), u_2(\boldsymbol{x}))$ is the two dimensional displacement field. The α parameter weighs between data term and regularization term. The objective is to find the displacement field $\boldsymbol{u}$ that minimizes the energy function in (6.1). The regularization term $|\nabla \boldsymbol{u}|$ with L^1 norm is called total variation regularization. Replacing these data and regularization terms with L^2 norm lead us to the original Horn and Schunck formulation [46]. Since the terms in (6.1) are not continuously differentiable, the energy function can be minimized using dual formulation for minimizing total variation as proposed in [29] and adapted to optical flow in [107]. Linearizing I_1 near to $(\boldsymbol{x}+\boldsymbol{u}_0)$, where $\boldsymbol{u}_0$ is a given flow field, the whole data term is denoted as an image residual $\rho(\boldsymbol{u}) = I_1(\boldsymbol{x}+\boldsymbol{u}_0) + \langle \nabla I_1, \boldsymbol{u}-\boldsymbol{u}_0 \rangle - I_0(\boldsymbol{x})$. Then, by introducing an auxiliary variable $\boldsymbol{v}$, the data term and regularization term in (6.1) can be rewritten as indicated in (6.2), making easier the minimization process. Without loss of generality, in the two-dimensional case, the resulting energy can be expressed as:

$$E = \int_{\Omega} \left\{ \alpha |\rho(\boldsymbol{v})| + \sum_{d=1,2} (1/2\theta)(u_d - v_d)^2 + \sum_{d=1,2} |\nabla u_d| \right\} d\boldsymbol{x}, \tag{6.2}$$

where θ is a small constant, such that $\boldsymbol{v}$ is a close approximation of $\boldsymbol{u}$; and d indicating the dimension takes value as 1 and 2. This convex energy function is optimized by alternative updating steps 1 and 2 for $\boldsymbol{u}$ and $\boldsymbol{v}$:

Step 1. By keeping $\boldsymbol{u}$ fixed, $\boldsymbol{v}$ is computed as:

$$\min_{\boldsymbol{v}} \{ \alpha |\rho(\boldsymbol{v})| + \sum_{d=1,2} (1/2\theta) \, (u_d - v_d)^2 \}, \tag{6.3}$$

Step 2. Then, by keeping v_d fixed for every d, u_d is computed as:

$$\min_{u_d} \int_{\Omega} \{ 1/2\theta (u_d - v_d)^2 + |\nabla u_d| \} d\boldsymbol{x}. \tag{6.4}$$

Equation (6.4) can be solved for each dimension using the dual formulation. The solution is given by:

$$u_d = v_d - \theta \, \mathbf{div} \boldsymbol{p}_d, \tag{6.5}$$

where the dual variable $\boldsymbol{p} = [p_1, p_2]$ for a dimension d is iteratively defined by

$$\tilde{\boldsymbol{p}}^{n+1} = \boldsymbol{p} + \tau/\theta(\nabla(v_d + \theta \, \mathbf{div} \boldsymbol{p}^n)), \tag{6.6}$$

$$\boldsymbol{p}^{n+1} = \tilde{\boldsymbol{p}}^{n+1} / \max(1, |\tilde{\boldsymbol{p}}^{n+1}|), \tag{6.7}$$

where $\boldsymbol{p}^0 = \mathbf{0}$ and the time step $\tau \leq 1/4$.

The solution of equation (6.3) is a simple thresholding step since it does not involve derivative of $\boldsymbol{v}$, and is given by:

$$\boldsymbol{v} = \boldsymbol{u} + \begin{cases} \alpha\theta\nabla I_1 & \text{if} \quad \rho(\boldsymbol{u}) < -\alpha\theta|\nabla I_1|^2 \\ -\alpha\theta\nabla I_1 & \text{if} \quad \rho(\boldsymbol{u}) > \alpha\theta|\nabla I_1|^2 \\ -\rho(\boldsymbol{u})\nabla I_1/|\nabla I_1|^2 & \text{if} \quad |\rho(\boldsymbol{u})| \leq \alpha\theta|\nabla I_1|^2 \end{cases} \tag{6.8}$$

In this optical flow method, the structure-texture blended image that is robust against sensor noise, illumination changes, reflections and shadows as explained in [97] is used. Additionally, in the current implementation, an initialization step is proposed for reducing the CPU time and increasing the accuracy. This step consists in using the optical flow computed between the previous couple of frames as initial values for the current couple instead of initializing by zero.

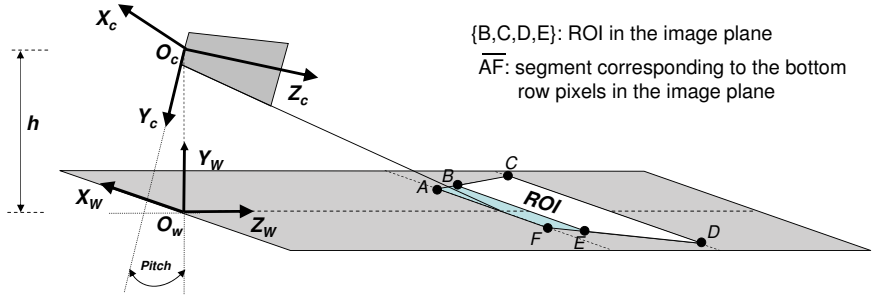


Figure 6.1: Camera coordinate system (X_C, Y_C, Z_C) and world coordinate system (X_W, Y_W, Z_W).

6.2.2 Model formulation

Before detailing the approach proposed to estimate the monocular vision system pose, the relationships between the coordinate systems (world and camera) and the camera parameters, assuming a flat road are presented. Camera pose parameters are computed relative to a world coordinate system (X_W, Y_W, Z_W), defined for every frame, in such a way that: the $X_W Z_W$ plane is co-planar with the current road plane. Figure 6.1 depicts the camera coordinate system (X_C, Y_C, Z_C) referred to the road plane. The origin of the camera coordinate system O_C is contained in the Y_W axis—it implies a $(0, t_y, 0)$ translation of the camera w.r.t. world coordinate system. Hence, since *yaw* angle is not considered in the current work (i.e., it is

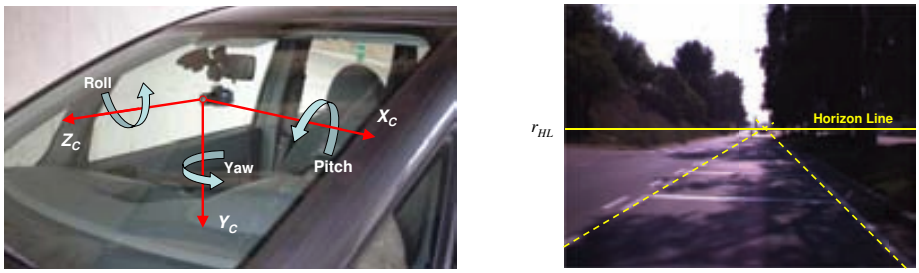


Figure 6.2: (*left*) On-board camera with its corresponding coordinate system. (*right*) Horizon line (r_{HL}) estimated by the intersection of projected lane markings.

assumed to be zero), the six camera pose parameters[1] $(t_x, t_y, t_z, yaw, roll, pitch)$ reduce to just three $(0, t_y, 0, 0, roll, pitch)$, denoted in the following as (h, Φ, Θ) (i.e., camera height, roll and pitch). Figure 6.2(*left*) shows the onboard camera used for testing the proposed approach.

Among the parameters (h, Φ, Θ), the value of the roll angle (Φ) will be very close to zero in most situations, since when the camera is rigidly mounted on the car, a specific procedure is followed to ensure an angle at rest within a given range, ideally zero, and in regular driving conditions this value scarcely varies (more details can be found in [78]). Finally, the variables (h, Θ) that represents the camera pose parameters are encoded as a single value, which is the *horizon line* position in the image plane (e.g., [109],[76]). The horizon line corresponds to the back-projection of a point, lying over the road at an infinite depth. Assuming the road can be modeled as a plane, let $ax + by + cz + h = 0$ be the road plane equation and h the camera height, see Fig. 6.1 (since $(h \neq 0)$ the plane equation can be simplified dividing by $(-h)$). Let $P_i(0, y, z)$ be a point lying over the road plane at an infinite depth z from the camera reference frame with $x = 0$; from the plane equation the y_i coordinates of P_i corresponds to $y_i = \frac{1-cz_i}{b}$. The backprojection of y_i into the image plane when $z_i \to \infty$ defines the row coordinate of the horizon line r_{HL} in the image. It results into:

$$r_{HL} \quad = \quad r_0 + f\frac{y_i}{z_i} = r_0 + \frac{f}{z_i b} - f\frac{c}{b} \quad , \tag{6.9}$$

where f denotes the focal length in pixels, r_0 represents the vertical coordinate of the camera principal point, and z_i is the depth value of P_i. Since $(z_i \to \infty)$, the row coordinate of the horizon line in the image is finally computed as $r_{HL} = r_0 - f\frac{c}{b}$. Additionally, when lane markings are present in the scene, the horizon line position in the image plane can be easily obtained by finding the intersection of these two parallel lines, see Fig. 6.2(*right*).

6.2.3 Horizon line estimation

In the current work, a RANSAC based approach is proposed to estimate the horizon line position. It works directly in the image plane by using the optical flow vectors computed between two consecutive frames. The TV-L^1 optical flow [97] with a minor modification as explained in the previous section is used. The flow vectors within a rectangular region

[1] A 3D translation and a 3D rotation that relates O_C with O_W.

centered in the bottom part of the image are used instead of considering the flow vectors through the whole image. The specified region is a rough estimation of the minimum free space needed for a vehicle moving at 30km/h to avoid collisions—rectangle defined by the points $\{A,C,D,F\}$, in Fig. 6.3. Note that, at a higher speed this region should be enlarged. Actually from this rectangular free space only the top part is used (rectangular ROI defined by the points $\{B,C,D,E\}$ in Fig. 6.3), since the flow vectors at the bottom part (image boundary) may not be as accurate as required. Figure 6.3 presents a couple of synthetic frames with the optical flow computed over that ROI; an enlarged and sub-sampled illustration of these flow vectors is given in the top-right part.

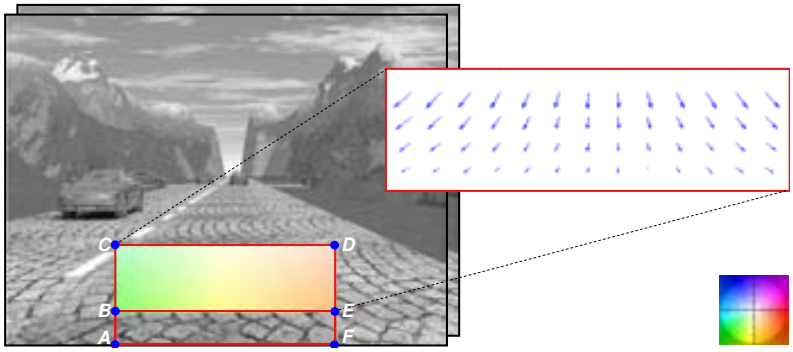


Figure 6.3: A couple of consecutive synthetic frames illustrating the rectangular free space {A,C,D,F}, containing the ROI {B,C,D,E} from which computed flow vectors are used for estimating horizon line position. ($top-right$) Enlarged and sub-sampled vector field from the ROI. ($bottom-right$) Color map used for depicting the vector field in the ROI.

Let $\boldsymbol{u}$ be the computed flow field corresponding to a given ROI $\{B,C,D,E\}$. This vector field can be used for recovering the camera motion parameters through a closed form formulation (e.g., [85] and [82]). However, since it could be noisy and contains outliers, a robust RANSAC based technique [39] is proposed for computing the horizon line position. It works as follow:

Random sampling: *Repeat the following three steps K times*

1. Draw a couple of vectors, $(\boldsymbol{u}^1,\boldsymbol{u}^2)$ from the given ROI where $\boldsymbol{u}^1=(u_1^1,u_2^1)$ and $\boldsymbol{u}^2=(u_1^2,u_2^2)$.
2. Compute the point (S_x,S_y) where these two vectors intersect.
3. Vote into the cell $\boldsymbol{C}_{(i,j)}$, where $i=\lfloor S_y \rfloor$ and $j=\lfloor S_x \rfloor$ and (i,j) lie within the image boundary.

Solution:

1. Choose the cell that has the highest number of votes in the voting matrix $\boldsymbol{C}$. Let $\boldsymbol{C}_{(i,j)}$ be this solution.
2. Set the sought horizon line position r_{HL} as the row i.

Figure 6.4: Horizon line computed by the proposed approach on a synthetic sequence.

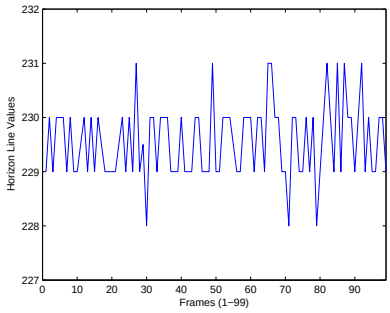


Figure 6.5: Plot of variations in horizon line in a sequence of 100 frames.

6.2.4 Experimental results

The proposed technique has been tested on several synthetic and real sequences. Firstly, a synthetic sequence (gray scale sequence-1 in set 2 of EISATS [93]) was used for validating the proposed approach. Figure 6.4 shows some frames with the horizon line computed by the proposed technique. Note, that in this case, since a perfect flat road without any vehicle dynamics, camera pose almost remains constant (horizon line variation through this synthetic sequence is presented in Fig. 6.5). On the contrary, horizon line undergoes large variations in Fig. 6.6. This synthetic sequence (gray scale sequence-2 in Set 2 of EISATS [93]) contains uphill, downhill and flat road scenarios. Figure 6.7(*left*) presents the variations of horizon line for the whole sequence. Figure 6.7(*right*) depicts the pitch angle variation from the ground-truth data. The similarity between these two plots confirms the effectiveness of the presented approach. The sequences in Fig.6.4, and Fig.6.6 are of resolution 480×640 pixels, and the ROI contains 96×320 pixels placed above 48 pixels from the bottom of the image.

Figure 6.8 shows a frame from a real sequence (Intern-On-Bike-left sequence in set 1 of EISATS sequences [93]) with the horizon line estimated by the proposed approach. The variation of the horizon line over a set of 25 frames of that sequence is presented in Fig. 6.8(*right*). Additionally, few different real frames, with horizon line estimated by the proposed approach, are shown in Fig.6.2(*right*) and Fig. 6.9. Notice that the horizon lines estimated by intersecting the projected lane markings (dotted lines) also coincide with those obtained by the proposed approach, in spite of the fact that some frames contain outliers (see lane barriers in the top-left frame in Fig. 6.9). The video frames in Fig. 6.9 are captured at a

resolution of 480×752 pixels at about 30fps. The value of K is empirically determined and the better value is about half of the total number of flow vectors in the specified ROI. The specified ROI contains 96×376 pixels and is placed above 48 pixels from the bottom of the image.

Figure 6.6: Horizon lines computed by the proposed approach on a synthetic video sequence illustrating different situations: uphill, downhill and flat roads.

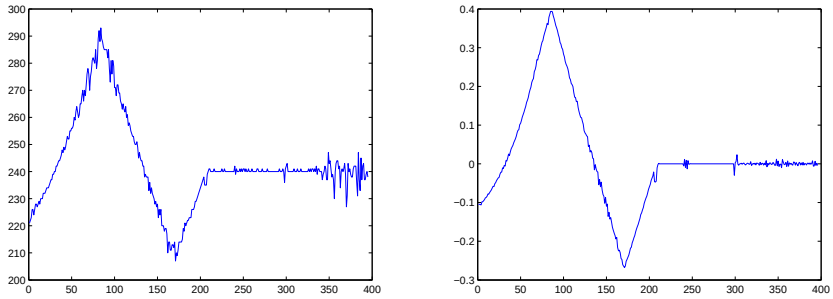


Figure 6.7: (*left*) Variations in horizon line position over a sequence of 396 frames. (*right*) Pitch angle variations from the ground-truth.

6.2.5 Conclusions

A robust technique for horizon line estimation of an on-board monocular vision system has been presented. It uses dense flow field from a state of the art variational optical flow technique that is robust against common obstructions in real traffic such as shadows, reflections and illumination changes. The proposed modified initialization step to the optical flow estimation has the advantage to be more accurate and less computation time. The camera pose parameters estimation is modeled as a horizon line estimation problem and has been solved using a RANSAC based approach that is robust against outliers in the flow field. The proposed approach is validated on both synthetic and real sequences. With the advancement in real-time implementation of optical flow algorithms and particularly, for our problem of estimating the flow vectors only in the specified region instead of the whole image, the proposed approach can be implemented on real applications with real-time performance.

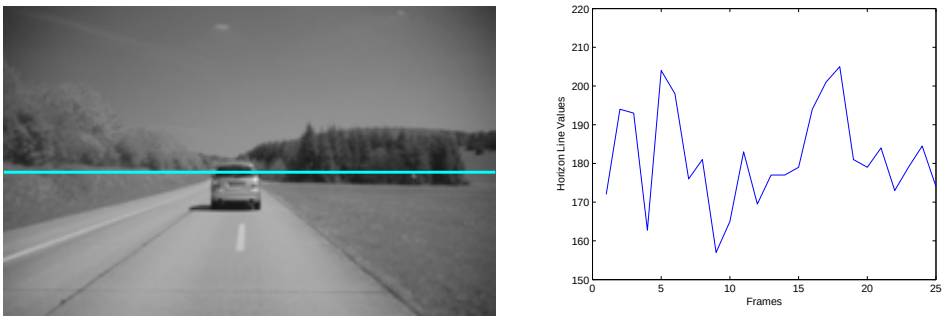


Figure 6.8: Horizon line for a real sequence and its variations for 25 frames.

Figure 6.9: Real video frames with the horizon lines estimated by the proposed approach (note that they correspond with the intersections of the projected lane markings).

6.3 Egomotion estimation

From the experiments in Chapter 3 in Section 3.3, it is evident that RLPR gives more accurate optical flow than LPR. In order to show the advantages of the better accuracy in RLPR, both representations are further evaluated by computing camera egomotion [87] on them. Particularly, here we compare only three orientation parameters (*pitch*, *yaw* and *roll*) of the camera egomotion using the optical flow of the space-variant representations.

6.3.1 Five-point algorithm

One technique to estimate orientation parameters is by using five-point algorithm. Five-point algorithm is first proposed by Nister in [68]. He proposes an algorithm to find the relative camera pose between two calibrated views using five corresponding points. Later Li et al. [53] propose much easier five-point algorithm based on hidden variable resultant technique. Figure 6.10 shows a typical five point correspondences from two views of a camera. Below describes the algorithm to find the camera pose parameters.

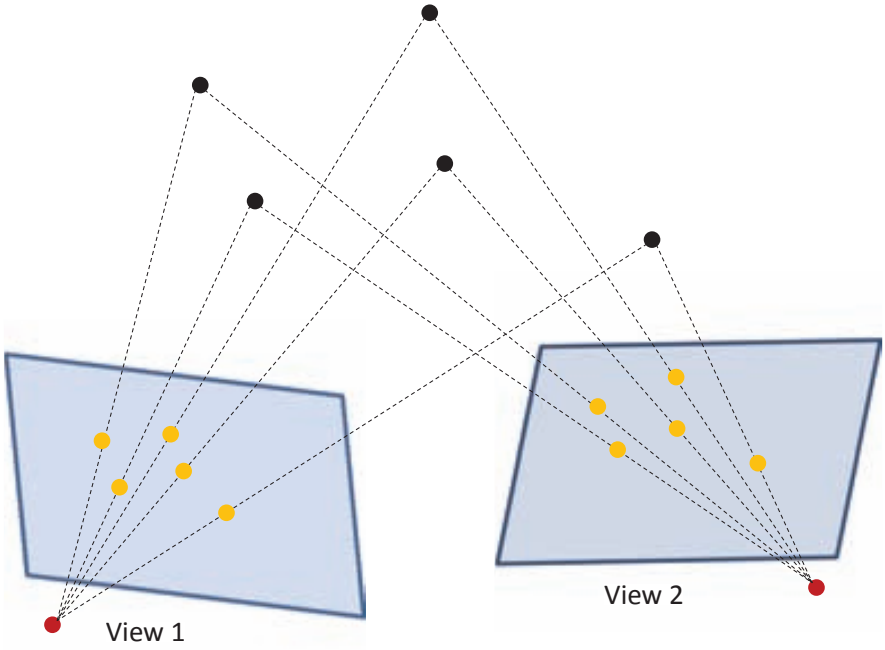


Figure 6.10: Two views of a camera with five point correspondences.

Steps in Five-point algorithm:

- Selection of five corresponding point sets from two consecutive images in a sequence
- Normalization using the calibration matrix
- Extracion of the null space of a 5×9 matrix

- Gauss-Jordan elimination with pivoting on the 10×20 matrix
- Obtain the tenth degree polynomial
- Extraction of roots of the tenth degree polynomial
- Find essential matrices for each of the roots
- Find the best essential matrix based on the Sampson error
- Obtain rotation and translation from the essential matrix

In our case, since we estimate dense optical flow between consecutive image frames, dense correspondences can be established using optical flow information.

6.3.2 Experimental results

This experiment is performed on sequence-2 of set-2 of EISATS [93] that has different motions of camera. We use the fast and simple five-point algorithm presented in [50] to estimate these orientation parameters. For inputting the point sets to five-point algorithm, we use a RANSAC [39] based approach that randomly selects *k* sets of ten points and compute the rotation angles. In the current work *k* has been set to 25 samples. Both space-variant representations (LP and RLP) use only a quarter of the information used in the original Cartesian representation. Hence, in order to have a fair comparison, we subsampled the original Cartesian images to the resolution of the space-variant representations (LPR and RLPR), i.e., to 240×360; this is referred hereinafter to as Subsampled-Cartesian representation. The orientation parameters are estimated using the optical flow computed on these representations. Table 6.1 shows the mean (in radians) and standard-deviation of the errors in *pitch*, *yaw* and *roll* angles obtained using ground-truth, Subsampled-Cartesian, LPR and RLPR flow fields of all frames in the sequence. The mean and standard-deviation of errors in *pitch*, *yaw* and *roll* angles by ground-truth optical flow are smaller than those computed by using optical flow estimated in Subsampled-Cartesian, LPR and RLPR. The results in Table 6.1 shows that LPR and RLPR are more accurate than Subsampled-Cartesian results. The mean and standard-deviation of *pitch* and *yaw* angles' errors are lower in RLPR than those in LPR. Note that *roll* angle do not change so much through the whole sequence (Fig 6.11 depicts ground-truth of *pitch*, *yaw* and *roll* angles in radians). Hence, the mean error of *roll* estimation, though lower in LPR than those in RLPR, is not as much significant as in the case of *pitch* and *yaw* angles. In Table 6.1, it can be observed that roll angle from GTOF is in the same order (E-04) as those from LPR and RLPR, whereas *pitch* and *yaw* angles from GTOF are far more accurate as compared to those from LPR and RLPR. Regarding *roll* angle variations, up to our understanding, visually there is no variation all over the original sequence. Hence, it seems to be an error introduced in the process of generating ground-truth values provided by [93]. Estimation of motion parameters can be ambiguous (e.g.,translation on X axis can be incorrectly estimated as rotation on Y axis and vice-versa) under a reduced field of view or with insufficient depth variation (e.g., [6],[37]) because different motions can induce similar flow vectors. Therefore, another way to evaluate these three orientations is just by comparing the rotation matrices directly instead of comparing individual orientations [87]. Table 6.2 provides the error values with rotation matrices. These values also reproduce

the same conclusion that RLPR is better than LPR and Subsampled-Caretesian. Overall it can be concluded that orientations estimated using optical flow from RLPR is more accurate than those estimated from LPR. Additionally, results are better than those obtained by the Subsampled-Cartesian representation that contains the same number of data points.

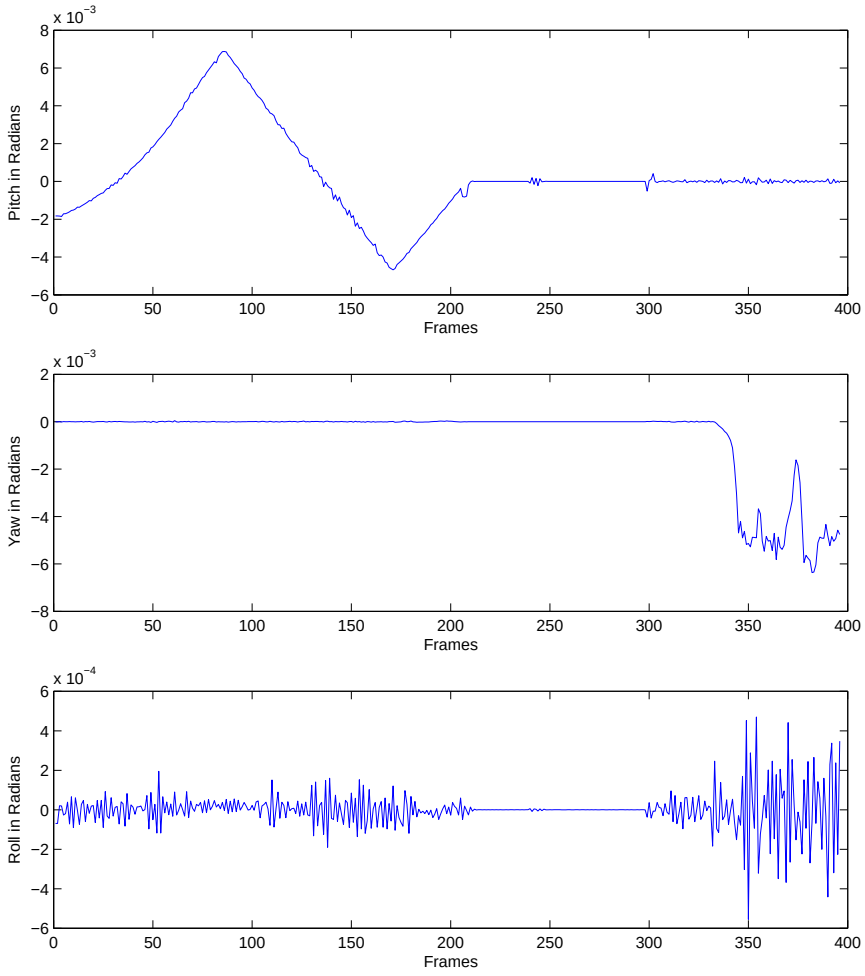


Figure 6.11: Plots of ground-truth *pitch*, *yaw* and *roll* angles through the whole sequence.

6.3.3 Conclusions

An egomotion estimation application is shown. Additionally the accuracy of both space-variant representations to estimate the egomotion parameters of a moving camera is evaluated. The results of RLPR are more accuarate than those of LPR, and the results from both space-variant representations are better those of Subsampled-cartesian.

Table 6.1
MEAN AND STANDARD-DEVIATION OF ERRORS IN *pitch*, *yaw*, *roll* IN RADIANS OF SEQUENCE-2 IN SET-2 [93].

	Pitch		Yaw		Roll	
	Mean	Std	Mean	Std	Mean	Std
GTOF	4.08E-07	4.34E-07	5.06E-07	5.90E-07	1.15E-04	1.66E-04
Sub. Cart.	2.39E-03	5.44E-03	1.43E-03	4.59E-03	2.63E-03	3.71E-03
LPR	1.67E-03	2.62E-03	9.16E-04	3.10E-03	5.89E-04	8.83E-04
RLPR	1.33E-03	1.63E-03	7.08E-04	1.49E-03	6.77E-04	8.54E-04

Table 6.2
MEAN AND STANDARD-DEVIATION OF ERRORS IN *rotation matrices* [87] IN RADIANS OF SEQUENCE-2 IN SET-2 [93].

	Rotation Matrix	
	Mean	Std
GTOF	2.46E-04	1.41E-04
Sub. Cart.	4.77E-03	7.60E-03
LPR	2.62E-03	4.04E-03
RLPR	2.15E-03	1.93E-03

6.4 Adaptation

Existing state of the art optical flow approaches, which are evaluated on standard datasets such as Middlebury, not necessarily have a similar performance when evaluated on driving scenarios. This drop on performance is due to several challenges arising on real scenarios during driving. Towards this direction, in this paper, we propose a modification to the regularization term in a variational optical flow formulation, that notably improves the results, specially in driving scenarios. The proposed modification consists on using the Laplacian derivatives of flow components in the regularization term instead of gradients of flow components. We show the improvements in results on a standard real image sequences dataset (KITTI).

6.4.1 Introduction

There have been a lot of contributions in optical flow techniques (as detailed in Chapter 2 Section 2.2) which are targeted and evaluated on few standard datasets (Chapter 2 Section 2.4) those were meant for general scenes. There is not much work on optical flow techniques particularly for driving scenarios. Driving scenarios vary very largely by environment, weather conditions and day-light conditions. The driving environment itself involves the situations such as urban, highway, countryside with different geometry of scenes and textures. Apart from these, the vehicle speed [70] and the turnings in the road also matters causing very large displacement. Thus, developing an optical flow technique that withstands all such difficult scenarios is a challenging research topic. Actually there is a lack of specialized methods for driving scenarios where occurs a variety of difficulties. In the current work, we propose an improvement over an existing state of the art method [83]. In this work, we specifically deal with the importance of regularization. We propose a modification to the derivative operator in the regularization that deals with large variations in speed and rotations that exist in KITTI dataset. The performance analysis done on the KITTI dataset shows that the proposed modification improves the results.

The work is organized as follows. Section 6.4.2 gives an overview of the basic optical flow formulation and the proposed modification. Experimental results are provided in section 6.4.3 followed by the conclusions in section 6.4.4.

6.4.2 Laplacian derivative based regularization

As presented in Chapter 2 Section 2.2 a basic variational optical flow technique can be formulated as:

$$E(\boldsymbol{u}) = \int_{\Omega} \{ \underbrace{(I_{x_1}u_1 + I_{x_2}u_2 + I_t)^2}_{Data\ Term} + \alpha \underbrace{(|\nabla u_1|^2 + |\nabla u_2|^2)}_{Regularization} \} \, d\boldsymbol{x}, \tag{6.10}$$

Based on the above basic formulation, the authors in [83] propose a formulation using median filtering in addition to the other improvements proposed in previous literature and

explored by them. It is known that median filtering at every iteration of flow computation improves the results. The work in [83] incorporates this filtering heuristics into the objective function. This improved non-local median filtering based method is called C+NL. In most of the methods in literature authors try to penalize the gradient of the estimated flow vectors using different and combinations of robust penalizing functions. In a driving sequence, there occurs large variations in magnitude and orientations due to change in speed of the vehicle, turn of the vehicle, specularity, and scene dynamics. In general, driving scenarios are very dynamic with large variations. Hence, in the current Chapter we propose to penalize the Laplacian of flow components instead of their gradients. With the basic formulation notation, the equation (6.10) becomes:

$$E(\boldsymbol{u}) = \int_{\Omega} \{ \underbrace{(I_{x_1}u_1 + I_{x_2}u_2 + I_t)^2}_{Data\ Term} + \alpha \underbrace{(|\triangle u_1|^2 + |\triangle u_2|^2)}_{Regularization} \} \, d\boldsymbol{x}. \tag{6.11}$$

In summary, we propose to modify the derivative of flow components in the regularization to second derivative as shown in equation (6.11) in the approach presented in [83]. We will refer to this method as C+NL-M. With second derivative regularization, it allows more variations in flow components. Hence, as shown in the next section the proposed modification results in more accurate optical flow estimations.

6.4.3 Experimental results

The proposed modification has been evaluated with respect to the state of the art method C+NL, which is one of the best approach on Middlebury dataset. The analysis of performance is carried out on the standard dataset KITTI [42]. This dataset contains image pairs of real driving scenarios with varied real characteristics that make optical flow computation a real challenge in such scenarios. This dataset consists of 194 training image pairs and 195 testing image pairs. The results on few of the testing pairs from KITTI are shown in Figures 6.12, 6.13 and 6.14. In these figures, the (top) is the 1st image of individual pairs, $(middle)$ is the error map, and the $(bottom)$ is the computed flow field. The red area in the $(middle)$ indicates the occluded pixels falling outside image boundary.

The evaluation performed by the KITTI server computes the average number of bad pixels for non-occluded or all pixels for available ground-truth. This evaluation is performed over the optical flow computed on testing set with our modified approach that has been uploaded to the KITTI server. Table 6.3 shows the errors for the image pair shown in Fig. 6.12 for the approach C+NL, whereas Table 6.4 shows the errors for the same pair for the proposed approach C+NL-M. It can be appreciated that C+NL-M gives better results and as presented below C+NL-M is ranked higher than the C+NL by the KITTI evaluation procedure. It should be noted that both C+NL and C+NL-M in this work use fast version of their implementations.

The evaluation table ranks all methods according to the number of non-occluded erroneous pixels at the specified end-point error threshold. At the time of submission (on 5th April 2013), our proposed method ranks 8^{th}, whereas C+NL ranks 16^{th} for 2 pixel threshold. The ranking table from the KITTI web service is shown in Fig. 6.15. For 3 pixel threshold our method ranks at 9^{th} as shown in Fig. 6.16. This shows that changing the regularization

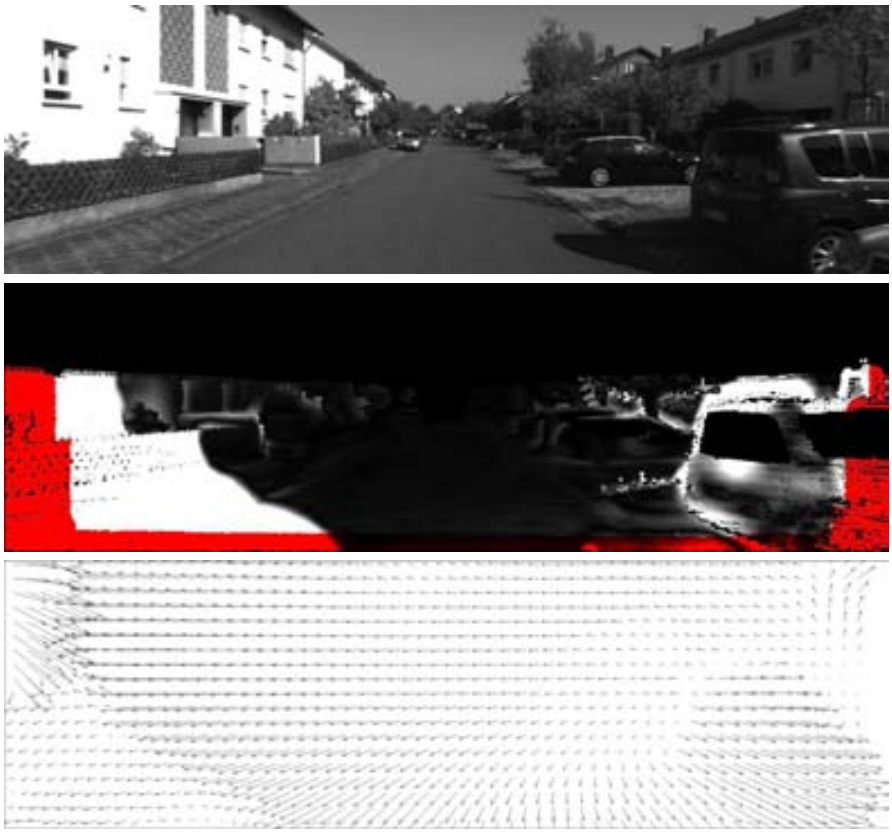


Figure 6.12: Results for a pair of images; (top) 1^{st} image of the pair; $(middle)$ error map; and $(bottom)$ computed flow field.

to Laplacian notably improves the results, specifically in the sequences of driving scenarios. At this time of writing the thesis, the previous entry of C+NL in [2] has been replaced by an improved modified approach by the original authors. Note that our proposed modified method better performs compared to the original approach in [83].

6.4.4 Conclusions

We explored and realized that the state of the art optical flow methods does not necessarily perform well for driving scenarios. Towards this, in this work we propose a modification of the regularization term in a state of the art method. The derivative of flow components are changed to Laplacian from gradient. The experimentation is performed on a standard benchmark data set (KITTI) that contains real image pairs of driving scenarios with challenging characteristics. The evaluation shows that the proposed modification performs better. We envisage that the KITTI dataset will lead research to the development of new approaches that can perform better in very complex scenarios and our future work concentrates on this line.

Figure 6.13: Results for a pair of images; (*top*) 1^{st} image of the pair; (*middle*) error map; and (*bottom*) computed flow field.

Table 6.3
ERROR VALUES FOR THE IMAGE PAIR SHOWN IN FIG. 6.12 BY C+NL

Error	Out-Noc	Out-All	Avg-Noc	Avg-All
2 pixels	32.68 %	42.11 %	11.1 px	17.5 px
3 pixels	30.74 %	40.00 %	11.1 px	17.5 px
4 pixels	29.56 %	38.48 %	11.1 px	17.5 px
5 pixels	28.56 %	37.13 %	11.1 px	17.5 px

Figure 6.14: Results for a pair of images; (*top*) 1^{st} image of the pair; (*middle*) error map; and (*bottom*) computed flow field.

Table 6.4
ERROR VALUES FOR THE IMAGE PAIR SHOWN IN FIG. 6.12 BY C+NL-M

Error	Out-Noc	Out-All	Avg-Noc	Avg-All
2 pixels	25.27 %	32.90 %	9.1 px	16.1 px
3 pixels	22.43 %	30.04 %	9.1 px	16.1 px
4 pixels	21.14 %	28.73 %	9.1 px	16.1 px
5 pixels	20.19 %	27.75 %	9.1 px	16.1 px

Rank	Method	Setting	Out-Noc	Out-All	Avg-Noc	Avg-All	Density	Runtime	Environment	Compare
1	PCBP-Flow	ms	**6.33 %**	**11.59 %**	**0.9 px**	**2.2 px**	100.00 %	3 min	4 cores @ 2.5 Ghz (Matlab + C/C++)	
2	MotionSLIC	ms	6.72 %	14.06 %	1.0 px	2.7 px	100.00 %	11 s	1 core @ 3.0 Ghz (C/C++)	
3	PR-Sceneflow		6.91 %	12.39 %	1.3 px	3.3 px	100.00 %	150 sec	4 core @ 3.0 Ghz (Matlab + C/C++)	
4	TGV2ADCSJPT		8.86 %	18.47 %	1.6 px	4.5 px	100.00 %	8s	GPU @ 1.5 Ghz (C/C++)	
5	Data-Flow		10.86 %	19.00 %	2.3 px	5.7 px	100.00 %	3 min	2 cores @ 2.5 Ghz (Matlab + C/C++)	
6	TGV2CENSUS		13.33 %	21.11 %	2.9 px	6.6 px	100.00 %	4 s	GPU+CPU @ 3.0 Ghz (Matlab + C/C++)	
7	PGM		14.56 %	25.94 %	3.2 px	12.2 px	100.00 %	60 s	1 core @ 2.4 Ghz (C/C++)	
8	C+NL-fit		21.13 %	28.37 %	7.4 px	14.5 px	100.00 %	5 min	2 cores @ 2.5 Ghz (Matlab)	
9	ePGM		22.01 %	31.50 %	5.2 px	10.8 px	100.00 %	0.026 s	GPU @ 700 Mhz (C/C++)	
10	HS		22.02 %	31.18 %	5.8 px	11.7 px	100.00 %	3 min	1 core @ 2.5 Ghz (Matlab + C/C++)	
11	RSRS-Flow		22.68 %	31.81 %	6.2 px	12.1 px	100.00 %	4 min	1 core @ 2.5 Ghz (Matlab)	
12	GC-BM-Bino	ms	23.07 %	33.10 %	5.0 px	12.0 px	83.71 %	1.3 s	2 cores @ 2.5 Ghz (C/C++)	
13	ALD		24.28 %	33.63 %	10.9 px	16.0 px	100.00 %	110 s	1 core @ 2.5 Ghz (C/C++)	
14	LDOF		24.43 %	33.87 %	5.5 px	12.4 px	100.00 %	1 min	1 core @ 2.5 Ghz (C/C++)	
15	GC-BM-Mono	ms	24.79 %	34.59 %	5.0 px	12.1 px	84.33 %	1.3 s	2 cores @ 2.5 Ghz (C/C++)	
16	C+NL		26.42 %	35.28 %	9.0 px	16.4 px	100.00 %	3 min	1 core @ 2.5 Ghz (C/C++)	

Figure 6.15: Evaluation table for 2 pixel error threshold (data from [2]).

Rank	Method	Setting	Out-Noc	Out-All	Avg-Noc	Avg-All	Density	Runtime	Environment	Compare
9	C+NL-fit		19.17 %	26.35 %	7.4 px	14.5 px	100.00 %	5 min	2 cores @ 2.5 Ghz (Matlab)	
16	C+NL		24.64 %	33.35 %	9.0 px	16.4 px	100.00 %	3 min	1 core @ 2.5 Ghz (C/C++)	

Figure 6.16: Evaluation table for 3 pixel error threshold (data from [2]).

Chapter 7

Conclusions

This Chapter presents a summary of the whole research work done in this thesis. It is organized as follows. Section 7.1 briefly summarizes our work and contributions highlighting the contents of the different chapters as well as their usage in the ADAS context. Finally, future work and possible research lines identified during the development of this dissertation are described in Section 7.2.

7.1 Summary

The contributions of this research are on optical flow in the context of ADAS. Broadly, we aimed to adapt optical flow estimation to driving scenarios in order to to make it faster and accurate, and to use it for some of the ADAS applications. With this objective in mind, we propose, in Chapter 3, a novel space-variant representation. We refer it to as Reverse Log Polar Representation (RLPR). It is shown that the results from this representation are more accurate compared to the traditional space-variant representation (LPR) when optical flow is computed on them. This fact is then confirmed in Chapter 6 through the egomotion estimation application. These space-variant representations drastically reduce the data to be processed and hence would make ADAS faster. Although space-variant representations lose some information, one can choose the representation based on the application to be developed.

Apart from making optical flow faster, this thesis also attempted to make optical flow accurate. Almost all existing optical flow approaches have evolved from their evaluations on general datasets. In a driving scenario, there are many different complexities. Thus, in Chapter 6 Section 6.4 we propose a modification in the regularization that notably improves the results. The advantage of the proposed modification is shown using a state of the art approach. The experiment is performed on a standard dataset (KITTI) with real images, which contains several image pairs of complex driving scenarios.

Thus, having a fast and best performing optical flow algorithm is not sufficient because in a driving scenario the environment varies significantly (e.g., highway, country side, urban) and situations are unpredictable (e.g., daylight conditions, roads, moving vehicles, shadows, illuminations). Although we have a good optical flow approach with robust data and regularization terms, it cannot cope-up with all these complexities. In this context there is a need

to adapt regularization weights depending on some characteristics of the scene to improve results. In this direction Chapter 5 presents an analysis of optical flow accuracy for the two characteristics of a scene such as vehicle speed and road texture. It is inferred that the regularization weight has to be tuned depending on the needed accuracy (either AAE or EPE) as well as on the speed and road texture. This analysis is performed considering a polar represented optical flow algorithm. It is also shown in Chapter 5 that the polar representation of flow vectors suits driving sequences (where predominant motion is translation). The polar representation based optical flow formulation gives freedom to use different regularizers and regularization weights.

For the above mentioned analysis, there was a need of sequences with different speeds and road textures. Finding such scenarios and obtaining dense ground-truth optical flow with good accuracy is nearly impossible. Thus we have created an urban scenario using the 3D modeling and visualization software (Maya) that gives full control of different attributes of the scene and camera movements. The sequences for the required scenarios are rendered and the ground-truth flow fields are also obtained using the ray-tracing technique. Thus we have generated a synthetic dataset for different driving scenarios, which is valid for evaluating the performance of optical flow algorithms in different situations.

Furthermore, Chapter 6 also presents several uses of optical flow such as for horizon line estimation and egomotion estimation. A RANSAC based algorithm to estimate vanishing point using dense optical flow is proposed. This vanishing point in-turn gives the position of the horizon line. The egomotion estimation is presented as a part of the experiments to compare the two space-variant representations evaluated in this dissertation.

7.2 Future work

This section details some of the ideas emanating from the research work carried out in this thesis during these years. They are listed as follows.

- The proposed space-variant representation needs to be represented in a rectangular grid to be able to apply the existing optical flow approaches. This mapping and remapping back to the space-variant representation induces some systematic errors in images as well as in flow fields. Hence there is a big scope to propose global optical flow approaches which can be applied directly on space-variant representations. Also, there is a need to study systematic errors. Apart from the optical flow computation, the proposed space-variant representation needs to be analyzed in terms of the reduced size of data and information loss.

- The influence on optical flow accuracy can be analyzed for many more characteristics of driving scenarios. In the current work just speed and texture were considered as an initial study. Other characteristics of driving scenarios could be considered, for instance, the relative speed, direction of moving vehicles present in the given scenario, textural and structural properties of the whole scene. Obviously, there is a need of new datasets for all such scenarios.

- From the knowledge obtained above, an interesting future work would be the development of rules to automatically adapt regularization parameters to a particular charac-

teristic of a given scene. A much deeper study is required when all the characteristics are combined.

- The synthetic datasets can be made more realistic by adding atmospheric effects, materials, complex scenes with trees and other road-side structures, motion blur, shades etc. It is clear that this kind of research is more oriented to computer graphics than computer vision, which somehow is a trend nowadays where computer vision takes advantage of the realism in computer graphics.
- Finally, it would be interesting to study the usage of optical flow for other applications such as moving object / obstacle detection or at a higher level towards 3D environment perception, automatic cruise control, to mention but a few.

Appendix A

Publications

Journals

- Naveen Onkarappa and Angel D. Sappa, "Speed and texture: an empirical study on optical flow accuracy in ADAS scenarios", *IEEE Transactions on Intelligent Transportation Systems*. (Published online as early access)
- Naveen Onkarappa and Angel D. Sappa, "A Novel Space Variant Image Representation", *Journal of Mathematical Imaging and Vision*. Vol. 47, No. 1-2, 2013, pp. 48-59.
- Naveen Onkarappa and Angel D. Sappa, "Synthetic sequences and ground-truth flow field generation for algorithm validation", *Multimedia Tools and Applications*. (Accept with minor revision)

Conferences

- Naveen Onkarappa and Angel D. Sappa, "Laplacian Derivative based Regularization for Optical Flow Estimation in Driving Scenario", *International Conference on Computer Analysis of Images and Patterns (CAIP'13)*, York, UK, August 27-29, 2013. LNCS 8048, pp. 483–490.
- Naveen Onkarappa, Sujay M. Veerabhadrappa and Angel D. Sappa, "Optical Flow in Onboard Applications: A Study on the Relationship Between Accuracy and Scene Texture", *International Conference on Signal and Image Processing (ICSIP'12)*, Coimbatore, India, December 13-15, 2012. LNEE 221, pp.257-267.
- Naveen Onkarappa and Angel D. Sappa, "An Empirical Study on Optical Flow Accuracy Depending on Vehicle Speed", *IEEE Intelligent Vehicles Symposium (IV'12)*, Alcala de Henares, Spain, June 3-7, 2012. pp. 1138-1143.
- Naveen Onkarappa and Angel D. Sappa, "Space Variant Representations for Mobile Platform Vision Applications", *International Conference on Computer Analysis of Images and Patterns (CAIP'11)*, Seville, Spain, August 29-31, 2011. LNCS 6855, pp.146-154.

- Naveen Onkarappa and Angel D. Sappa, "On-Board Monocular Vision System Pose Estimation through a Dense Optical Flow", *International Conference on Image Analysis and Recognition (ICIAR'10)*, Povoa de Varzim, Portugal, June 21-23, 2010. LNCS 6111, pp. 230-239.

References

[1] http://vision.middlebury.edu/flow/. [Pages **5, 19, 20, 23** and **25**]

[2] http://www.cvlibs.net/datasets/kitti/. [Pages **7, 24, 87** and **90**]

[3] http://sintel.is.tue.mpg.de/results/. [Page **24**]

[4] Y. Adato, T. Zickler, and O. Ben-Shahar. A polar representation of motion and implications for optical flow. In *IEEE Int. Conference on Computer Vision and Pattern Recognition*, pages 1145–1152, Colorado Springs, USA, June 2011. [Pages **19, 51, 54, 55** and **56**]

[5] Y. Adato, T. Zickler, and O. Ben-Shahar. A polar representation of motion and implications for optical flow. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1145–1152, Colorado Springs, CO, USA, June 2011. [Page **49**]

[6] G. Adiv. Inherent ambiguities in recovering 3-d motion and structure from a noisy flow field. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(5):477–489, 1989. [Pages **28** and **82**]

[7] L. Álvarez, J. Weickert, and J. Sánchez. Reliable estimation of dense optical flow fields with large displacements. *International Journal of Computer Vision*, 39(1):41–56, 2000. [Page **17**]

[8] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310, 1989. [Page **15**]

[9] X. Armangué, H. Araújo, and J. Salvi. A review on egomotion by means of differential epipolar geometry applied to the movement of a mobile robot. *Pattern Recognition*, 36(12):2927–2944, 2003. [Page **28**]

[10] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. *SIAM Journal on Applied Mathematics*, 60:156–182, 1999. [Page **17**]

[11] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, 2011. [Pages **20, 23, 24** and **39**]

[12] N. Barnes and G. Sandini. Direction control for an active docking behaviour based on the rotational component of log-polar optic flow. In *European Conference on Computer Vision*, pages 167–181, Dublin, Ireland, June-July 2000. [Pages **21** and **23**]

[13] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994. [Pages **3, 20, 23** and **49**]

[14] M. Bertozzi, E. Binelli, A. Broggi, and M. Del Rose. Stereo vision-based approaches for pedestrian detection. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, San Diego, USA, June 2005. [Page **27**]

[15] M. Bertozzi, A. Broggi, M. Carletti, A. Fascioli, T. Graf, P. Grisleri, and M. Meinecke. IR pedestrian detection for advaned driver assistance systems. In *Proc. 25th. Pattern Recognition Symposium*, pages 582–590, Magdeburg, Germany, September 2003. [Page **27**]

[16] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996. [Page **17**]

[17] M.J. Black and P. Anandan. Robust dynamic motion estimation over time. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 296 –302, Maui, HI, USA, June 1991. [Pages **17** and **18**]

[18] M. Bolduc and M. D. Levine. A review of biologically motivated space-variant data reduction models for robotic vision. *Computer Vision and Image Understanding*, 69(2):170–184, 1998. [Page **21**]

[19] J. A. Boluda, J. Domingo, F. Pardo, and J. Pelechano. Detecting motion independent of the camera movement through a log-polar differential approach. In *Computer Analysis of Images and Patterns*, pages 702–709, Kiel, Germany, September 1997. [Pages **21** and **23**]

[20] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision*, volume 3024 of *LNCS*, pages 25–36. Springer, May 2004. [Pages **17, 18** and **55**]

[21] T. Brox, A. Bruhn, and J. Weickert. Variational motion segmentation with level sets. In *Eurpean Conference on Computer Vision (1)*, volume 3951 of *Lecture Notes in Computer Science*, pages 471–483, May 2006. [Page **19**]

[22] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(3):500–513, 2011. [Page **19**]

[23] A. Bruhn. *Variational Optic Flow Computation: Accurate Modelling and Efficient Numerics*. PhD thesis, Department of Mathematics and Computer Science, Saarland University, Saarbrucken, 2006. [Page **18**]

[24] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnörr. Variational optical flow computation in real time. *IEEE Transactions on Image Processing*, 14(5):608–615, 2005. [Page **18**]

[25] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. *International Journal of Computer Vision*, 70(3):257–277, 2006. [Page **18**]

[26] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005. [Pages **15, 17** and **19**]

[27] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision (6)*, volume 7577, pages 611–625, Florence, Italy, October 2012. [Pages **5, 23, 24** and **27**]

[28] B. F. Buxton and H. Buxton. Computation of optic flow from the motion of edge features in image sequences. *Image Vision Comput.*, 2(2):59–75, 1984. [Page **15**]

[29] A. Chambolle. An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.*, 20(1-2):89–97, 2004. [Page **74**]

[30] D. Cheda, D. Ponsa, and A.M. Lopez. Camera egomotion estimation in the adas context. In *IEEE Int. Conference on Intelligent Transportation Systems*, pages 1415–1420, Madeira Island, Portugal, September 2010. [Page **28**]

[31] P. Coulombeau and C. Laurgeau. Vehicle yaw, pitch, roll and 3D lane shape recovery by vision. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 619–625, Versailles, France, June 2002. [Page **24**]

[32] K. Daniilidis. Computation of 3-d-motion parameters using the log-polar transform. In *Computer Analysis of Images and Patterns*, pages 82–89, Prague, Czech Republic, September 1995. [Page **21**]

[33] K. Daniilidis and V. Krüger. Optical flow computation in the log-polar-plane. In *Computer Analysis of Images and Patterns*, pages 65–72, Prague, Czech Republic, September 1995. [Pages **21** and **23**]

[34] J. Dias, H. Araújo, C. Paredes, and J. Batista. Optical normal flow estimation on log-polar images. a solution for real-time binocular vision. *Real-Time Imaging*, 3(3):213–228, 1997. [Pages **21** and **23**]

[35] F. Dornaika and A. Sappa. A featureless and stochastic approach to on-board stereo vision system pose. *Image and Vision Computing*, 27(9):1382–1393, August 2009. [Page **27**]

[36] M. Felsberg. Optical flow estimation from monogenic phase. In *First International Workshop on Complex Motion*, volume 3417, pages 1–13, Günzburg, Germany, October 2004. [Page **15**]

[37] C. Fermüller and Y. Aloimonos. Ambiguity in structure from motion: Sphere versus plane. *International Journal of Computer Vision*, 28(2):137–154, 1998. [Pages **28** and **82**]

[38] B. Fischl, M. A. Cohen, and E. L. Schwartz. Rapid anisotropic diffusion using space-variant vision. *International Journal of Computer Vision*, 28:199–212, 1998. [Page **21**]

[39] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381–395, June 1981. [Pages **28, 77** and **82**]

[40] D. J. Fleet and A. D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990. [Page **15**]

[41] B. Galvin, B. Mccane, K. Novins, D. Mason, and S. Mills. Recovering motion fields: An evaluation of eight optical flow algorithms. In *British Machine Vision Conference*, pages 195–204, 1998. [Page **20**]

[42] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR)*, Providence, USA, June 2012. [Pages **5, 23, 24, 26** and **86**]

[43] V. M. Govindu. Revisiting the brightness constraint: Probabilistic formulation and algorithms. In *European Conference on Computer Vision (3)*, volume 3953 of *Lecture Notes in Computer Science*, pages 177–188, May 2006. [Page **17**]

[44] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621, 1973. [Page **56**]

[45] F. Heitz and P. Bouthemy. Multimodal estimation of discontinuous optical flow using markov random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(12):1217–1232, 1993. [Pages **17** and **18**]

[46] B. K. P. Horn and B. G. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981. [Pages **9, 15, 16, 17, 39, 51** and **74**]

[47] J.Bigün, G. H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(8):775–790, 1991. [Page **15**]

[48] Y. Kameda, A. Imiya, and T. Sakai. Hierarchical properties of multi-resolution optical flow computation. In *ECCV 2012 Workshops and Demonstrations*, volume 7584 of *LNCS*, pages 576–585. Springer Berlin Heidelberg, 2012. [Page **72**]

[49] K. Krajsek and R. Mester. Bayesian model selection for optical flow estimation. In *DAGM-Symposium*, Lecture Notes in Computer Science, pages 142–151, Heidelberg, Germany, September 2007. [Page **19**]

[50] Z. Kukelova, M. Bujnak, and T. Pajdla. Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. In *British Machine Vision Conference*, Leeds, UK, 2008. [Pages **28** and **82**]

[51] R. Labayrade and D. Aubert. A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 31–36, Columbus, OH, USA, June 2003. [Page **27**]

[52] R. Labayrade, D. Aubert, and J. Tarel. Real time obstacle detection in stereovision on non flat road geometry through 'V-disparity' representation. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 646–651, Versailles, France, June 2002. [Page **27**]

[53] Hongdong Li and Richard I. Hartley. Five-point motion estimation made easy. In *18th International Conference on Pattern Recognition (ICPR)*, pages 630–633, Hong Kong, China, August 2006. [Page **81**]

[54] Y. Liang, H. Tyan, H. Liao, and S. Chen. Stabilizing image sequences taken by the camcorder mounted on a moving vehicle. In *Proc. IEEE Int. Conf. on Intelligent Transportation Systems*, pages 90–95, Shangai, China, October 2003. [Page **24**]

[55] C. Liu, W. T. Freeman, E. H. Adelson, and Y. Weiss. Human-assisted motion annotation. In *IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, Alaska, USA, June 2008. [Pages **3, 5, 23, 36** and **37**]

[56] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision (darpa). In *DARPA Image Understanding Workshop*, pages 121–130, April 1981. [Pages **15, 39** and **51**]

[57] O. Mac Aodha, G. J. Brostow, and M. Pollefeys. Segmenting video into classes of algorithm-suitability. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1054–1061, San Francisco, CA, USA, June 2010. [Page **20**]

[58] O. Mac Aodha, A. Humayun, M. Pollefeys, and G. J. Brostow. Learning a confidence measure for optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(5):1107–1120, 2013. [Pages **23** and **40**]

[59] B. McCane, K. Novins, D. Crannitch, and B. Galvin. On benchmarking optical flow. *Computer Vision and Image Understanding*, 84(1):126–143, October 2001. [Pages **3, 20, 23** and **49**]

[60] S. Meister and D. Kondermann. Real versus realistically rendered scenes for optical flow evaluation. In *14th ITG Conference on Electronic Media Technology (CEMT)*, pages 1–6, Dortmund, 2011. [Page **40**]

[61] É. Mémin and P. Pérez. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Transactions on Image Processing*, 7(5):703–719, 1998. [Pages **15** and **17**]

[62] D. W. Murray and B. F. Buxton. Scene segmentation from visual motion using global optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(2):220–228, 1987. [Page **18**]

[63] H-H Nagel. Displacement vectors derived from second-order intensity variations in image sequences. *Computer Vision, Graphics, and Image Processing*, 21(1):85 – 117, 1983. [Page **17**]

[64] H-H Nagel. Extending the 'oriented smoothness constraint' into the temporal domain and the estimation of derivatives of optical flow. In *ECCV*, Lecture Notes in Computer Science, pages 139–148, Antibes, France, April 1990. [Pages **17** and **18**]

[65] H-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593, 1986. [Pages **17** and **18**]

[66] S. Nedevschi, C. Vancea, T. Marita, and T. Graf. Online extrinsic parameters calibration for stereovision systems used in far-range detection vehicle applications. *IEEE Trans. on Intelligent Transportation Systems*, 8(4):651–660, December 2007. [Page **27**]

[67] P. Nesi. Variational approach to optical flow estimation managing discontinuities. *Image Vision Comput.*, 11(7):419–439, 1993. [Pages **17** and **18**]

[68] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(6):756–777, 2004. [Pages **28** and **81**]

[69] N. Onkarappa and A. D. Sappa. On-board monocular vision system pose estimation through a dense optical flow. In *Int. Conference on Image Analysis and Recognition*, pages 230–239, Póvoa de Varzim, Portugal, June 2010. [Pages **31** and **32**]

[70] N. Onkarappa and A.D. Sappa. An empirical study on optical flow accuracy depending on vehicle speed. In *IEEE Intelligent Vehicles Symposium*, pages 1138 –1143, June 2012. [Pages **3, 36, 37** and **85**]

[71] M. Otte and H-H. Nagel. Optical flow estimation: Advances and comparisons. In *European Conference on Computer Vision (1)*, volume 800 of *Lecture Notes in Computer Science*, pages 51–60, Stockholm, Sweden, 1994. [Page **23**]

[72] M. Otte and H-H Nagel. Estimation of optical flow based on higher-order spatiotemporal derivatives in interlaced and non-interlaced image sequences. *Artificial Intelligence*, 78(1-2):5 – 43, 1995. [Page **20**]

[73] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision*, 67(2):141–158, 2006. [Pages **15, 17** and **18**]

[74] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A. Hyder, E. Jarawan, and C. Mathers. *World Report on road traffic injury prevention*. World Health Organization, Geneva, Switzerland, 2004. [Page **24**]

[75] M. Proesmans, L. J. Van Gool, E. J. Pauwels, and A. Oosterlinck. Determination of optical flow and its discontinuities using non-linear diffusion. In *European Conference on Computer Vision (2)*, Lecture Notes in Computer Science, pages 295–304, Stockholm, Sweden, May 1994. [Page **17**]

[76] C. Rasmussen. Grouping dominant orientations for ill-structured road following. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 470–477, Washington, USA, June 2004. [Page **76**]

[77] S. Roth and M. J. Black. On the spatial statistics of optical flow. In *IEEE International Conference of Computer Vision*, pages 42–49, Beijing, China, October 2005. [Page **54**]

[78] A. Sappa, F. Dornaika, D. Ponsa, D. Gerónimo, and A. López. An efficient approach to on-board stereo vision system pose estimation. *IEEE Trans. on Intelligent Transportation Systems*, 9(3):476–490, September 2008. [Pages **27** and **76**]

[79] C. Schnörr. On functionals with greyvalue-controlled smoothness terms for determining optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(10):1074–1079, 1993. [Page **17**]

[80] E. L. Schwartz, D. N. Greve, and G. Bonmassar. Space-variant active vision: Definition, overview and examples. *Neural Networks*, 8(7-8):1297–1308, 1995. [Page **21**]

[81] A. Singh. An estimation-theoretic framework for image-flow computation. In *ICCV*, pages 168–177, Osaka, Japan, December 1990. [Page **15**]

[82] G. Stein, O. Mano, and A. Shashua. A robust method for computing vehicle egomotion. In *IEEE Intelligent Vehicles Symposium*, pages 362–368, Dearborn Michigan, USA, October 2000. [Pages **27** and **77**]

[83] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *IEEE Int. Conference on Computer Vision and Pattern Recognition*, pages 2432–2439, San Francisco, CA, USA, June 2010. [Pages **19, 32, 48, 49, 85, 86** and **87**]

[84] D. Sun, S. Roth, J. P. Lewis, and M. J. Black. Learning optical flow. In *European Conference on Computer Vision (3)*, volume 5304 of *Lecture Notes in Computer Science*, pages 83–97, 2008. [Page **19**]

[85] T. Suzuki and T. Kanade. Measurement of vehicle motion and orientation using optical flow. In *Proc. IEEE Int. Conf. on Intelligent Transportation Systems*, pages 25–30, Tokyo, Japan, October 1999. [Pages **27** and **77**]

[86] I. Tang and T. P. Breckon. Automatic road environment classification. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):476–484, 2011. [Page **51**]

[87] T. Y. Tian, C. Tomasi, and D. J. Heeger. Comparison of approaches to egomotion computation. In *IEEE Int. Conference on Computer Vision and Pattern Recognition*, pages 315–320, San Francisco, CA, USA, 1996. [Pages **3, 28, 81, 82** and **84**]

[88] M. Tistarelli and G. Sandini. On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(4):401–410, 1993. [Pages **21** and **23**]

[89] V. J. Traver and F. Pla. The log-polar image representation in pattern recognition tasks. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 1032–1040, 2003. [Page **21**]

[90] V. J. Traver and F. Pla. Motion analysis with the radon transform on log-polar images. *Journal of Mathematical Imaging and Vision*, 30(2):147–165, 2008. [Page **21**]

[91] V. Javier Traver and A. Bernardino. A review of log-polar imaging for visual perception in robotics. *Robotics and Autonomous Systems*, 58(4):378–398, 2010. [Page **21**]

[92] H. Tunley and D. Young. First order optic flow from log-polar sampled images. In *European Conference on Computer Vision (vol. 1)*, pages 132–137, Stocklholm, Sweden, 1994. [Pages **21** and **23**]

[93] T. Vaudrey, C. Rabe, R. Klette, and J. Milburn. Differences between stereo and motion behaviour on synthetic and real-world stereo sequences. In *Image and Vision Computing New Zealand*, pages 1–6, Christchurch, New Zealand, November 2008. [Pages **3, 5, 23, 24, 26, 32, 33, 34, 36, 37, 78, 82** and **84**]

[94] S. Volz, A. Bruhn, L. Valgaerts, and H. Zimmer. Modeling temporal coherence for optical flow. In *IEEE International Conference of Computer Vision*, pages 1116–1123, Barcelona, Spain, November 2011. [Page **19**]

[95] R. S. Wallace, P-W. Ong, B. B. Bederson, and E. L. Schwartz. Space variant image processing. *International Journal of Computer Vision*, 13:71–90, 1994. [Page **21**]

[96] A. Wedel, D. Cremers, T. Pock, and H. Bischof. Structure- and motion-adaptive regularization for high accuracy optic flow. In *IEEE International Conference of Computer Vision*, pages 1663–1668, Kyoto, Japan, 2009. [Pages **18** and **19**]

[97] A. Wedel, T. Pock, C. Zach, D. Cremers, and H. Bischof. An improved algorithm for TV-L1 optical flow. In *Proc. of the Dagstuhl Motion Workshop*, pages 23–45, Dagstuhl Castle, Germany, September 2008. [Pages **15, 17, 18, 19, 74, 75** and **76**]

[98] A. Wedel, T. Pock, C. Zach, D. Cremers, and H. Bischof. An improved algorithm for TV-L1 optical flow. In *Dagstuhl Motion Workshop*, pages 23–45, Dagstuhl Castle, Germany, September 2008. [Pages **49** and **55**]

[99] J. Weickert and C. Schnörr. A theoretical framework for convex regularizers in pde-based computation of image motion. *International Journal of Computer Vision*, 45(3):245–264, December 2001. [Pages **17** and **18**]

[100] J. Weickert and C. Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3):245–255, 2001. [Pages **17** and **18**]

[101] J. Wills, S. Agarwal, and S. Belongie. A feature-based approach for dense segmentation and estimation of large disparity motion. *International Journal of Computer Vision*, 68(2):125–143, 2006. [Page **15**]

[102] P. Wu, Y. Man Ro, C. S. Won, and Y. Choi. Texture descriptors in mpeg-7. In *International Conference on Computer Analysis of Images and Patterns*, pages 21–28, Warsaw, Poland, September 2001. [Page **56**]

[103] L. Xu, J. Chen, and J. Jia. A segmentation based variational model for accurate optical flow estimation. In *European Conference on Computer Vision (1)*, Lecture Notes in Computer Science, pages 671–684, Marseille, France, October 2008. [Page **19**]

[104] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(9):1744–1757, 2012. [Pages **19, 48** and **49**]

[105] M. Yeasin. Optical flow in log-mapped image plane-a new approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(1):125–131, 2002. [Pages **21** and **23**]

[106] A. Yeung and N. Barnes. Efficient active monocular fixation using the log-polar sensor. *International Journal of Intelligent Systems Technologies and Applications*, 1:157–173, July 2005. [Page **21**]

[107] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L^1 optical flow. In *Proc. 29th Annual Symposium of the German Association for Pattern Recognition*, pages 214–223, Heidelberg, Germany, September 2007. [Pages **15, 17, 18, 19** and **74**]

[108] L. Zelnik-Manor and M. Irani. Multi-frame estimation of planar motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(10):1105–1116, 2000. [Page **28**]

[109] C. Zhaoxue and S. Pengfei. Efficient method for camera calibration in traffic scenes. *Electronics Letters*, 40(6):368–369, March 2004. [Page **76**]

[110] H. Zimmer, A. Bruhn, and J. Weickert. Optic flow in harmony. *International Journal of Computer Vision*, 93(3):368–388, July 2011. [Pages **15, 17, 19** and **20**]