UAB
Universitat Autònoma
de Barcelona

Monocular Depth Cues in Computer Vision Applications

A dissertation submitted by **Diego Cheda** at Universitat Autònoma de Barcelona to fulfil the degree of **Doctor of Philosophy**.

Bellaterra, September 26, 2012

Director: **Dr. Daniel Ponsa Mussarra**
Centre de Visió per Computador
Universitat Autònoma de Barcelona

Co-director: **Dr. Antonio M. López Peña**
Centre de Visió per Computador
Universitat Autònoma de Barcelona

Thesis commite: **Dr. Juan Andrade Cetto**
Institut de Robòtica i Informàtica Industrial
Consejo Superior de Investigaciones Científicas

Dr. Ángel Domingo Sappa
Centre de Visió per Computador

Dr. David Masip Rodó
Dept. d'Estudis d'Informàtica Multimèdia i Telecomunicació
Universitat Oberta de Catalunya

Dra. Aura Hernandez Sabaté
Dept. de Ciències de la Computació
Universitat Autònoma de Barcelona

Dra. Carme Julià Ferré
Dept. d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili

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 978-84-940530-0-9

Printed by Ediciones Gráficas Rey, S.L.

A Martina y a Pau

Acknowledgements

Foremost, I would like to thank my advisors, Dr. Daniel Ponsa and Dr. Antonio M. López, for their continuous support, advice and patience during these years. They introduced me to the Computer Vision field and shared with me a lot of his expertise and research insight. I would like to express my gratitude towards all the members of ADAS group.

I would like to acknowledge the funding, academic and technical support of the Computer Vision Center, the Universitat Autònoma de Barcelona, and the Ministry of Education and Science (projects TRA2011-29454-C03-01 and TIN2011-29494-C03-02, and Consolider Ingenio 2010: MIPRCV (CSD200700018)).

I would also like to thank my committee for their time and comments: Dr. Juan Andrade Cetto, Dr. Ángel Domingo Sappa, Dr. David Masip Rodó, Dra. Aura Hernandez Sabaté, and Dra. Carme Julià Ferré. I also want to thank the anonymous conference and journal reviewers for their invaluable comments along these years.

I also owe a great deal to the many friends and colleagues that I have met at Computer Vision Center. Thank you all for share your time in one way or another along some endless days at CVC. I am indebted to Fernando Barrera and David Rojas for the innumerable talks about research, life, rare things and, almost as important, the morning coffee ritual.

I thank to my closest friends Juan José Tamagnini, Salvador Cavadini, and Daniel Romero for always being there.

Most importantly, I thank my wonderful family. First, to all my big family back in Argentina, I thank you for your continued support and encouragement from abroad. Second, to my small family in Catalonia: to Fernanda for those nice days in Tarragona, and to Mariana and Aarón for those incredible shared moments at Barcelona. Last, to my dear wife Martina for embarked with me in this adventure, for your continued love and support, and for keeping at my side –dealing with the good and bad times along these years. During this time, you had the generous experience of raising a family, teaching me how deep is your love. To my gorgeous son, Pau, I thank you for your smiles, your 3am wake-ups asking for milk, for make me playing again and again with the animals, and for showing me the impressive way of the life.

Abstract

Depth perception is a key aspect of human vision. It is a routine and essential visual task that the human do effortlessly in many daily activities. This has often been associated with stereo vision, but humans have an amazing ability to perceive depth relations even from a single image by using several monocular cues.

In the computer vision field, if image depth information were available, many tasks could be posed from a different perspective for the sake of higher performance and robustness. Nevertheless, given a single image, this possibility is usually discarded, since obtaining depth information has frequently been performed by three-dimensional reconstruction techniques, requiring two or more images of the same scene taken from different viewpoints. Recently, some proposals have shown the feasibility of computing depth information from single images. In essence, the idea is to take advantage of a priori knowledge of the acquisition conditions and the observed scene to estimate depth from monocular pictorial cues. These approaches try to precisely estimate the scene depth maps by employing computationally demanding techniques. However, to assist many computer vision algorithms, it is not really necessary computing a costly and detailed depth map of the image. Indeed, just a rough depth description can be very valuable in many problems.

In this thesis, we have demonstrated how coarse depth information can be integrated in different tasks following alternative strategies to obtain more precise and robust results. In that sense, we have proposed a simple, but reliable enough technique, whereby image scene regions are categorized into discrete depth ranges to build a coarse depth map. Based on this representation, we have explored the potential usefulness of our method in three application domains from novel viewpoints: camera rotation parameters estimation, background estimation and pedestrian candidate generation. In the first case, we have computed camera rotation mounted in a moving vehicle applying two novels methods based on distant elements in the image, where the translation component of the image flow vectors is negligible. In background estimation, we have proposed a novel method to reconstruct the background by penalizing close regions in a cost function, which integrates color, motion, and depth terms. Finally, we have benefited of geometric and depth information available on single images for pedestrian candidate generation to significantly reduce the number of generated windows to be further processed by a pedestrian classifier. In all cases, results have shown that our approaches contribute to better performances.

Resumen

La percepción de la profundidad es un aspecto clave en la visión humana. El ser humano realiza esta tarea sin esfuerzo alguno con el objetivo de efectuar diversas actividades cotidianas. A menudo, la percepción de la profundidad se ha asociado con la visión binocular. Pese a esto, los seres humanos tienen una capacidad asombrosa de percibir las relaciones de profundidad, incluso a partir de una sola imagen, mediante el uso de varias pistas monoculares.

En el campo de la visión por ordenador, si la información de la profundidad de una imagen estuviera disponible, muchas tareas podrían ser planteadas desde una perspectiva diferente en aras de un mayor rendimiento y robustez. Sin embargo, dada una única imagen, esta posibilidad es generalmente descartada, ya que la obtención de la información de profundidad es frecuentemente obtenida por las técnicas de reconstrucción tridimensional, que requieren dos o más imágenes de la misma escena tomadas desde diferentes puntos de vista. Recientemente, algunas propuestas han demostrado que es posible obtener información de profundidad a partir de imágenes individuales. En esencia, la idea es aprovechar el conocimiento a priori de las condiciones de adquisición de la imagen y de la escena observada para estimar la profundidad empleando pistas pictóricas monoculares. Estos enfoques tratan de estimar con precisión los mapas de profundidad de la escena empleando técnicas computacionalmente costosas. Sin embargo, muchos algoritmos de visión por ordenador no necesitan un mapa de profundidad detallado de la imagen. De hecho, sólo una descripción en profundidad aproximada puede ser muy valiosa en muchos problemas.

En nuestro trabajo, hemos demostrado que incluso la información aproximada de profundidad puede integrarse en diferentes tareas siguiendo una estrategia holística con el fin de obtener resultados más precisos y robustos. En ese sentido, hemos propuesto una técnica simple, pero fiable, por medio de la cual regiones de la imagen de una escena se clasifican en rangos de profundidad discretos para construir un mapa tosco de la profundidad. Sobre la base de esta representación, hemos explorado la utilidad de nuestro método en tres dominios de aplicación desde puntos de vista novedosos: la estimación de la rotación de la cámara, la estimación del fondo de una escena y la generación de ventanas de interés para la detección de peatones. En el primer caso, calculamos la rotación de la cámara montada en un vehículo en movimiento mediante dos nuevos métodos que emplean elementos distantes en la imagen a través de nuestros mapas de profundidad, aprovechando el hecho de que el componente traslacional en los vectores de flujo de la imagen pueden considerarse como insignificantes. En la reconstrucción del fondo de una imagen, propusimos un método novedoso que penaliza las regiones cercanas en una función de coste que integra, además, información del color y del movimiento. Por último, empleamos la información geométrica y de la profundidad de una escena para la generación de peatones candidatos. Este método reduce significativamente el número de ventanas generadas, las cuales serán posteriormente procesadas por un clasificador de peatones. En todos los casos, los resultados muestran que los enfoques basados en la profundidad contribuyen a un mejor rendimiento de las aplicaciones estudidadas.

Contents

Utopia
The Lost Thing, Shaun Tan, 2010.

The Lost Thing, written and illustrated by Shaun Tan, is a picture book and a short film that relates the story about a boy who discovers a strange creature. He guessed that it is lost, and begins the search for its owner or where it really belongs. However, he is faced with indifference by everyone else. After searching around the city, the boy found an Utopian land of lost things, where there are a lot of stranger beings. He returns the creature and continues on with his life.

Despite the extraordinary cyclopean creature shown in the image, we believe that monocular vision is not so stranger in the nature. Many animals have lack of binocular vision because they have their eyes on the sides of the head, providing a panoramic view of the environment to notice the approach of predators from any direction. Even people who have lost the sight in one eye can spatially get oriented by using non-stereoscopic clues, which have experiential nature. In this chapter, we introduce the motivation of our work, which is focus on how a simple set of image features can be used to obtain rough depth information and how this information can be useful for different purposes.

Chapter 1

Introduction

An understanding of how we categorize is central to any understanding of how we think and how we function, and therefore central to an understanding of what makes us human. **Women, Fire, and Dangerous Things, G. Lakoff, 1987.**

Perception is the ability to interpret and organize stimuli received from the environment in order to understand and behave effectively within it. One of the most important sources of stimuli for the human beings is the visual system. The visual system is a very complex system. It is composed by over one million axons from each eye, whose function is capture light reflected by objects. The processing of such inputs involves a huge number of neurons in the brain, where mental representations of reality are build. This process is called vision. It is an extraordinarily powerful sense that goes beyond simply capturing images, as a camera does. Additionally, it implies a variety of mechanisms by which the form, color, size, movements, and distance of objects are perceived.

The recognition of location in space is essential for almost all daily activities like navigating through a place, avoiding obstacles, jumping, catching and throwing objects, reaching or grasping something, and making size judgments. All these activities can be done thanks to our ability to extract three-dimensional (3D) representations of physical reality from our 2D retinal images. Humans do this effortlessly.

Depth perception has traditionally linked to stereopsis (i.e., perception of a scene using binocular vision). However, there are more information contained in bidimensional images that make us perceiving depth. This is the so-called cue theory, which is focused on identifying the information in the 2D images that is related to depth of the scene. According to this theory, we learn the connections between these cues and the actual depth by our accumulated experience about the spatial relations in the world. For instance, in Fig. 1.1, we can see an apocalyptic scenario where a family desperately struggle to cross what remains of a street. Of course, this is an optical illusion conceived by the German artist Edgar Mueller. Despite the fact that it is just

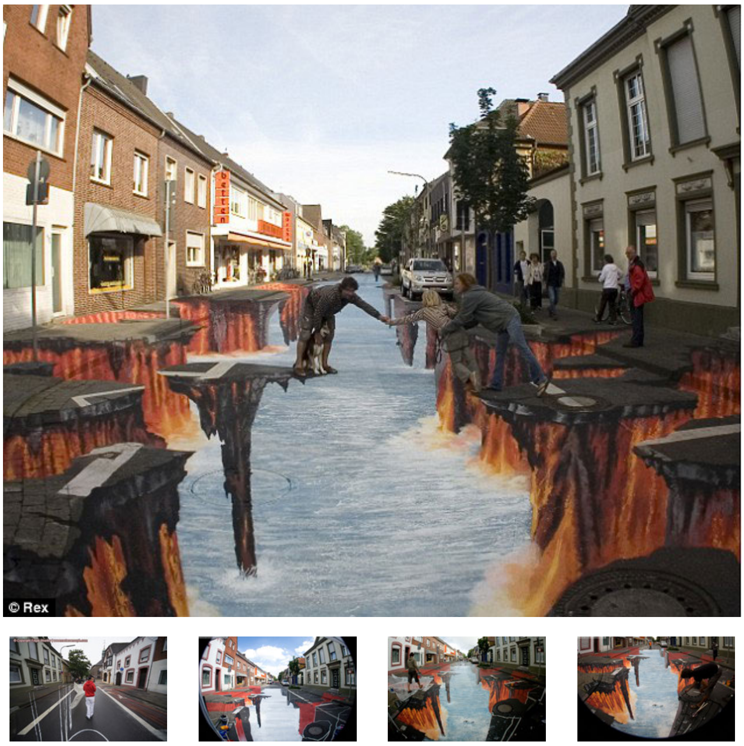


Figure 1.1: An incredible 3D illusion created by the street-painter Edgar Mueller and its painting process. Even from a drawing on a flat surface we are able to perceive depth. This is due to the fact that our visual system is supported by many monocular cues.

a drawing, we see a perfect 3D scene. The artist takes advantage of our knowledge of the world and applies many monocular cues (e.g. perspective, occlusion, texture) to create landscapes with incredible details, giving the illusion of depth on a flat surface that challenge the reality perception. For centuries, these techniques have been widely used by artists, and captured the interest of researchers in different science fields.

From the computer vision field, the reconstruction of 3D world from images has been commonly addressed as the process to create 3D models of real objects using two or more of images. In general, given multiple images taken from a calibrated camera, scene depth can be estimated by finding correspondences between, at least, two images and triangulating matched elements to determine their position in 3D space. Figure 1.2 shows impressive examples of state of the art approaches to reconstruct a scene using hundreds of images taken from different point of views by structure-from-motion.

Recently, the problem of estimating depths from a single image has received a growing interest. Current approaches are mainly focused on the design of sophisticated features and levels of reasoning for accurate depth estimation. Their final goal

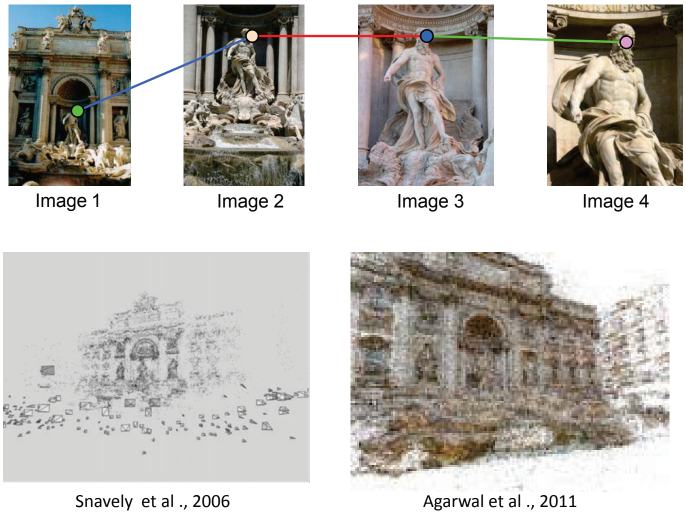


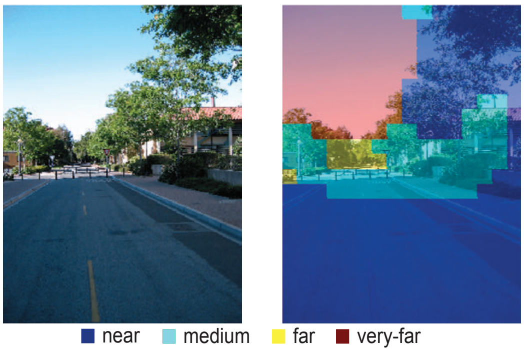
Figure 1.2: Reconstruction of a scene using multiple images from different viewpoints. The images are matched by finding interesting points. Then, a structure-from-motion algorithm is used to process the matched points for reconstructing the scene [103, 1].

is inferring the 3D structure of the scene as good as possible. However, with the aim of solving a given computer vision problem, the computation of 3D information could be just the starting step to subsequently analyze an image more reliably instead of being a goal in itself. Furthermore, even just a rough depth description could help to achieve better performances.

In this thesis, we first focus on how, by means of simple image features based on monocular cues, is possible obtaining a representation of a scene depth from a single image taken from a camera. This representation is computed by applying the learned relation between a set of visual features extracted from a scene image and scene depth. Figure 1.3 depicts an example of what kind of information we want to predict. With this objective in mind, we develop a low-cost method to estimate coarse depth information, but informative enough to tackle many computer vision problems from a different perspective, leading to alternative and more reliable solutions.

In particular, here we explore the use of coarse depth information in three different computer vision problems: egomotion estimation, background estimation, and pedestrian candidate generation. We pose each one of these problems by proposing novel ways of addressing them.

In the first case, we estimate the changes in position and orientation of a vehicle (i.e., the egomotion). This is a key component in most advanced driver assistance systems (ADAS) (e.g., adaptive cruise control, collision avoidance, lane-departure warning, autonomous driving, etc.), where the knowledge of the previous vehicle po-



(a) (b)

Figure 1.3: Example from our approach showing: (a) Original image and (b) Ideal depth segmentation considering four depth thresholds.

sition is required to properly act in the next time instant. In this work, we focus on the close relationship between motion and depth to improve monocular egomotion estimation results. Regarding to that, when an camera moves in a 3D world, this relative motion induces a corresponding image flow field. Image flow vectors are composed by two components: a translational component that depends on depth, and a rotational one which is independent of depth. Commonly, egomotion algorithms use image flow to update the camera pose. However, most of them ignore that the translation contribution to the image flow vectors is negligible at image zones corresponding to very distant elements in the scene (see Fig. 1.4). Unlike these *general* methods, we take that fact into consideration to estimate rotation uncoupledly from translation by incorporating depth information in the egomotion computation process. Then, since image flow vectors at enough distance from the camera are mainly dominated by rotational velocities, we propose two novel approaches to rotation estimation: one is based on tracking points located at distant regions between consecutive frames; and other one avoids point matching by directly tracking distant regions. Figure 1.5 shows an example of both distant points and regions, which are used by our methods to compute rotation estimation.

A second application where depth is useful is background estimation. It is usually the first step in background subtraction algorithms, where moving objects are detected by subtracting the observed image from an estimated reference background image. Segmentation of moving objects provides useful information from video processing applications such as image stitching, background substitution, compression, tracking, surveillance, etc. Here, we rely on figure-ground perception experience to estimate background, which involves depth perception. That is, in an image, foreground appears nearer than the ground part, and the ground appears to be occluded by the figure (see Fig. 1.6). Despite this strong relationship between figure-ground and depth, any previous related work in monocular background estimation has taken into account that depth information can be extracted from single images. Hence, we propose to integrate in the background estimation process the information about

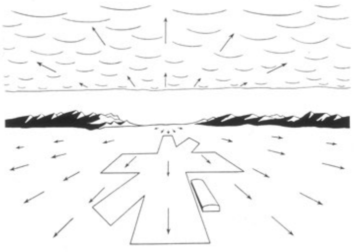
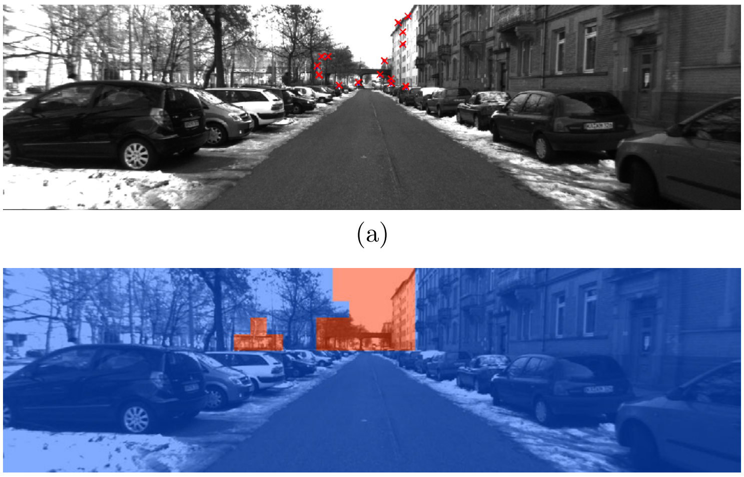


Figure 1.4: Image of optical flow depicting an flyby over the landing field (original image from [41]). In this case, the vectors are due to just translation displacement. We can see how the magnitude of image flow vectors decreases when they are located at far distances from the observer.

what is close and distant in an image. Our novel method reconstructs the background by selecting the appropriate pixels from a set of input images, showing a partially occluded scene background from the same or different view. To do that, we minimize a cost function that penalizes the deviations from the following assumptions: background represents objects whose distance to the camera is maximal, and background objects are stationary. Distance information is roughly obtained by our supervised learning approach that allows us to distinguish between close and distant image regions. Moving foreground objects are filtered out by using stationariness and motion boundary constancy measurements. Figure 1.7 depicts an example of a sequence with transient and moving objects and the occlusion-free result of our method.

In the last application, we focus on the use of our coarse depth information in a pedestrian detection system. This kind of ADAS systems are devoted to detect pedestrian in the surroundings of a vehicle to posteriorly warn the driver or perform some evasive action. In general, they are based on analyzing interesting regions extracted by exhaustively searching over the image (e.g., sliding window approaches), which are classified as pedestrian or non-pedestrian. This implies a huge number of generated windows to be further processed in the classification stage. However, for generating windows where the likelihood of containing a pedestrian is high, we can benefit of the physical regularity of our world, where relations between an object and its setting are arranged into a well-formed scene (see Fig. 1.8). According to this, we propose a novel method that fuses geometric and depth information available on single images to generate pedestrian candidates based on an underlying model which is focused only on objects standing vertically on the ground plane and having certain height, according with their depths on the scene. An example of our approach is illustrated in Fig. 1.9, where we can see how geometric and depth are fused to generate candidate windows.

Our final intention is following one of the primary goals in computer vision, which is characterized by holistic scene understanding. In that sense, instead of explaining certain aspects of a scene in isolation, we focus on how to exploit the existent dependencies between depth and other problems to achieve better performance in different computer vision tasks.



(a)

(b)

Figure 1.5: Our proposals use (a) points located at very-far regions (beyond than 70 m) or (b) directly distant regions to rotation estimation, since they contain mainly rotational information. Distant regions are computed from our coarse depth map.

1.1 Contributions and outline

Given the relevance of depth for many computer vision tasks, in this thesis we have developed a method to estimate coarse depth information from a single image. Based on their results, we have proposed novel algorithms in three different computer vision problems, leading to robust and accurate performance. Summarizing, the outline of the main contributions of this thesis are discussed as following:

- **Monocular coarse depth map estimation**:

  As a basis of our work we define a simple algorithm to classify the pixels of an image in just four categories: near, medium-distance, far and very-far, conforming with that a coarse depth map. To this aim, in Chapter 2 we analyze the cues involved in monocular human depth perception and, by constraining our domain to outdoor images taken from a camera parallel to the ground, we propose a reduced set of low-level features to infer depth information. Our approach, detailed in Chapter 3, uses these features to describe small regions of an image, which are processed by classifiers whose output is taken to segment the image into depth categories.

- **Egomotion estimation methods based on distant regions**:

  In Chapter 4, we contribute with two novel methods that take advantage of the fact that points distant enough from the camera just provide information about camera rotation, since they are negligibly affected by translation. These methods proceeding by first classifying distant regions of the image, and then

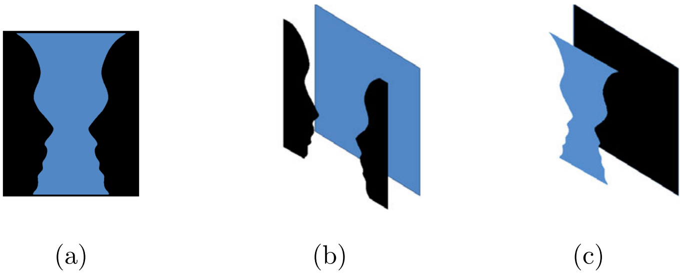


Figure 1.6: Our proposal for estimating the scene background relies on figure-ground perception experience: (a) Reversible figure-ground image, (b) When the faces are seen as figure, they appear in front of a blue background, (c) When the vase is seen as figure, it perceives in front of a dark background.

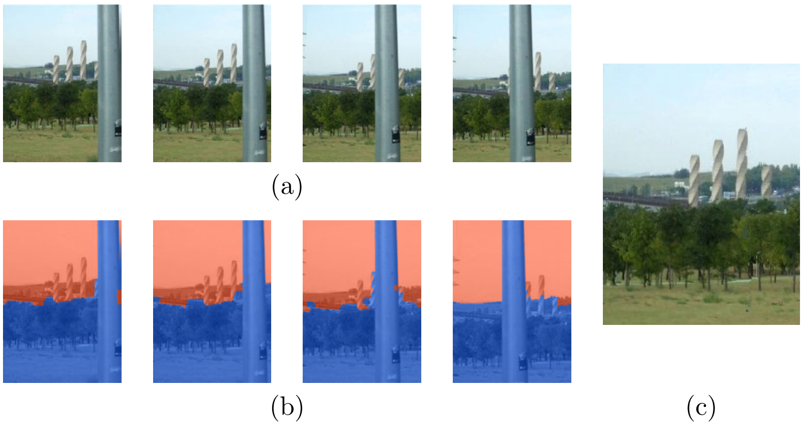


Figure 1.7: Example of sequence with occluding objects and the estimated background: (a) Some frames of a sequence with occluding objects, (b) Close and distant regions, and (c) Reconstructed background without occluding objects.

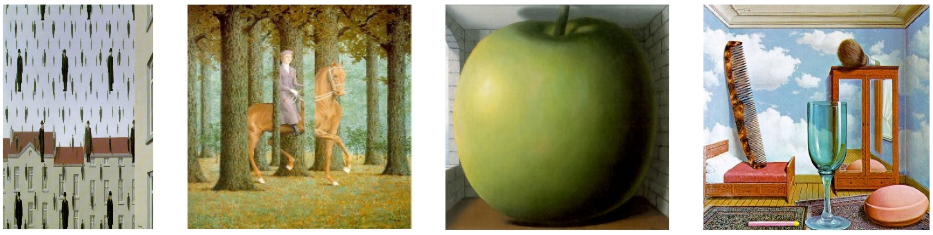
robustly estimate the camera rotation by tracking points or regions placed on those areas.

- **Background estimation method exploiting close/distant regions**:

  In Chapter 5, we propose a novel approach for background estimation where the background model takes into account the distinction between close/distant regions from the camera. The background is composed by selecting the appropriate pixels from the input images such that a cost function penalizing the deviations from our background model is minimized by a graph cuts method.

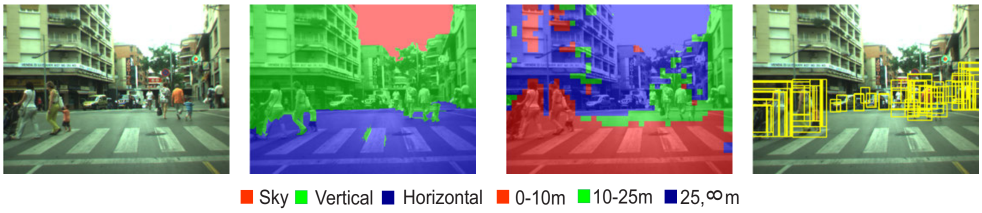
- **Pedestrian candidates generation exploiting coarse depth maps**:

  In Chapter 6, we propose a novel method for pedestrian candidate generation that significantly reduces the number of windows to be considered by a classifier. Our method exploits geometric and depth information available on single



(a) Support (b) Interposition (c) Size (d) Probability and position

Figure 1.8: In our world, scenes establish strong relations between objects than compose them. Four paints of the Belgian surrealist artist René Magritte(Gloconde, Blank check, The listening room, and Personal values) where well-formed rules of a scene are violated [13]: (a) Support: in our world things appears resting on a surface; (b) Interposition: an opaque object occludes the object behind it; (c) Familiar size of objects: an object appears too small or too large relative to other objects in the scene; (d) Probability and position: likelihood of a given object being in a given scene, and occupy specific positions.

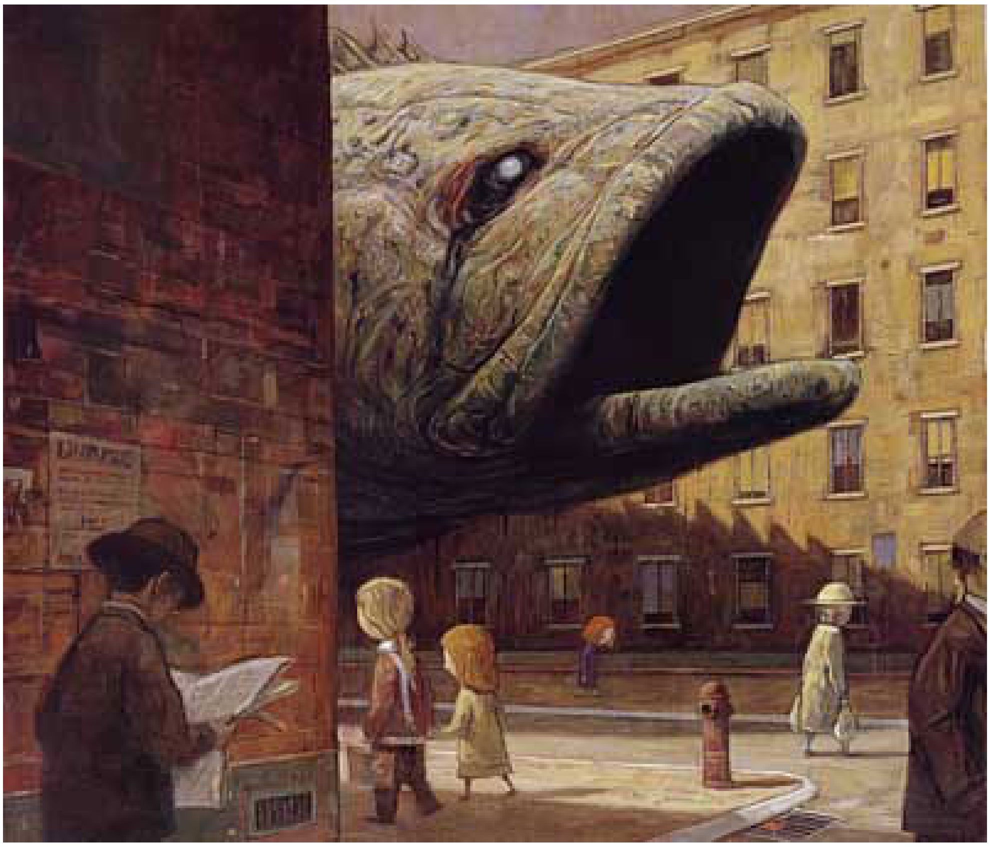


(a) (b) (c) (d)

Figure 1.9: Example of our pedestrian candidate generation: (a) Original image, (b) Geometric information, (c) Depth information, and (d) Candidate windows.

> images, which are fused together to generate candidates based on an underlying model which is focused only on objects standing vertically on the ground plane and having certain height, according with their depths on the scene.

Each chapter of this thesis is organized following a similar scheme. First, the addressed problem and the contributions are introduced. Then, the proposed method is detailed in the central part of the chapter, followed by experiments and discussion on the obtained results. Each chapter finishes with the corresponding conclusions.



Darkness overcomes you
The Red Tree, Shaun Tan, 2001.

The Red Tree, written and illustrated by Shaun Tan, is a picture book that relates the story of a little girl, which appears in every picture, passing through many dark moments, and by surreal and imaginary worlds. At the end of her journey, she found the hope, which is represented by a red-leafed tree growing in her bedroom.

In the selected image, we can see the little girl trudging through a city street in the shadow of a huge fish which floats above her. Beyond the symbolism of the scene, our knowledge and beliefs about relationships in our well-organized world allow us to understand that there is an inconsistency in that environment, which is incompatible with our set of memory descriptions. This evidences that relational constraints play an important role in scene understanding. We use these constraints to our purpose, as we will describe in the next chapters.

Chapter 2

Depth perception

We cannot clearly be aware of what we possess till we have the means of knowing what others possessed before us. We cannot really and honestly rejoice in the advantages of our own time if we know not how to appreciate the advantages of former periods. **J. von Goethe (1749-1832).**

In this chapter, we identify monocular cues that the human being uses during depth perception process. These cues has been used as basis of many previous works to extract depth information. Then, we review relevant works in computer vision about 3D estimation and scene understanding from single images, and place our work in relation to them.

2.1 Introduction

Studies on how human perceive suggest that the visual system make use of a variety of information sources to understand and derive the depth structure of scenes. In this chapter, we first introduce these cues, focusing our interest on monocular cues[1]. Then, we overview the state-of-the-art in depth estimation from a single view, and put our proposal in context.

2.2 Depth perception

Several stimuli present in 2D retinal images provide information that contributes to the spatial-depth perception of the human visual system. These sources of depth

[1] In Appendix A, we a historical overview from the perspective of vision research is addressed, focusing primarily on the mechanism by which humans can perceive depth is addressed.

information are typically classified into monocular and binocular cues[2] [43].

On the one hand, monocular cues depend on a single view, and can be separated onto pictorial and motion-based cues. The former are based on visual features observed in a static view of a scene. The latter exploit the observer motion, taking advantage of motion parallax, which relies on the fact that nearby objects apparently move “faster” in the retinal image than distant ones [51]. On the other hand, the perception of depth in binocular cues is mainly founded on the existent disparity between two different view-points of the same scene that allows triangulating the distance to an object with a high accuracy [52].

All these cues are complementary and integrated during depth perception. However, Cutting and Vishton [24] suggested that different sources of information are used at three distinguishable spatial regions surrounding the observer. For the human vision system, in regions delimited to be within 30 m, motion and binocular cues are most important, while beyond 30 m depth perception is supported only by pictorial cues. In this work, we put our interest in estimating depth of outdoor scenes, where there are larger distance ranges. Then, we consider monocular pictorial criteria as candidates for implementing our approach aimed at identifying the depth ranges of regions in outdoor images.

According to psychophysical studies described in [43], there are seven pictorial cues supporting depth perception. Figure 2.1 depicts these cues. In the following, we briefly describe each one:

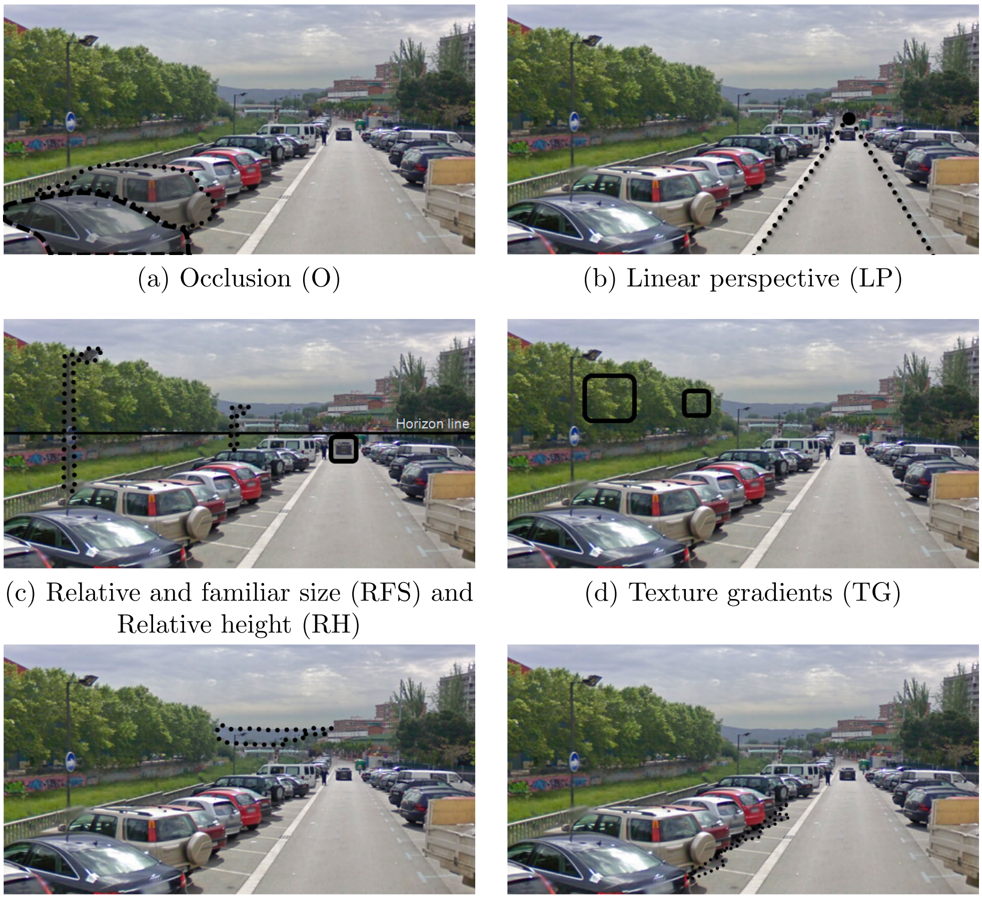
- **Occlusion (O)**: An occlusion occurs when one object partially hides another one in a view. The occluded object appears farther to the observer, which provides information about depth order rather than distance. In Fig. 2.1(a), a car marked by a dotted line is perceived as being father away than the other one car marked by a dashed line.

- **Linear perspective (LP)**: As a direct consequence of the perspective projection, the linear perspective refers to the fact that parallel lines that recede into the distance apparently converge toward a vanishing point in the projected image, which is illustrated in Fig. 2.1(b).

- **Relative and familiar size (RFS)**: Given two objects of equal size, the one farther to the observer appears to be smaller. This fact is also an effect of the perspective projection. For instance, the two lamp post in Fig. 2.1(c) have the same size, but the farther one appears smaller than the other. Note that prior knowledge about the objects size is required to infer their exact distance.

- **Relative height (RH)**: This cue is related to the position of objects with respect to the horizon line in the visual field. When an object is near the horizon, this implies that it is distant. In Fig. 2.1(c) also we can see how a

[2] Additionally, humans also take advantage of oculomotor cues, which are related to the ability to sense the position and muscle-tension of the eyes. These are inherent to human nature, and cannot be emulated by a camera.



(a) Occlusion (O)

(b) Linear perspective (LP)

(c) Relative and familiar size (RFS) and Relative height (RH)

(d) Texture gradients (TG)

(e) Aerial perspective (AP)

(f) Shadows (S)

Figure 2.1: Monocular pictorial cues used by the human being during depth perception.

farther car (marked by a solid line) which is near to the horizon line appears smaller.

- **Texture gradients (TG)**: The patterns of a textured surface vary as a function of the distance from the observer. At a greater distance, they get finer and appear smoother, as with the textured trees in Fig. 2.1(d).
- **Aerial perspective (AP)**: Details of distant object are degraded by atmospheric conditions like haze, fog, and smoke. Under these situations, the further away objects are unclearer and less detailed with respect to those which are closer. Fig. 2.1(e) depicts the aerial perspective effect over mountains, which are unclear.
- **Shadows (S)**: The cast shadow of objects can provide information about the 3D object shape (as the shadows projected by cars in Fig. 2.1(f)), and its relative location in the scene [81], as it has been shown by shape from shading approaches [123].

From a computational point of view, to obtain depth information from a single image, it is necessary to extract information correlated with these cues.

2.3 Depth estimation from a single image

In this section, we analyze how the previously introduced cues have been used in different approaches to estimate depths from single images. This is a topic that has gained considerable interest in recent years, mainly due to work of [111, 56, 95].

Table 2.1 summarizes recent works on estimating depth from a single image. Methods based on using occlusions as the principal cue [27] have been demonstrated in experiments done with very simple scenes, or under laboratory conditions. These approaches require objects in the image to have well defined contours, which is commonly not satisfied, and hence, limits their usage for outdoor images.

Other proposals rely on the linear perspective cue [23, 61, 10, 118]. In general, these proposals make strong assumptions about the scene geometry, and only work for images with clear parallel lines and orthogonal planes such as the case of man-made/indoor environments.

Judging the distance from the camera based on the relative and familiar size cue has been used in [17, 46, 77]. They incorporate contextual information to guide depth perception by recognizing what is the object class that a region in the image belongs to [46, 77]. The main problem of these approaches is that require a preclassification step to determine region classes over a limited set of object labels. Once semantic information is gathered, it is used to reinforce depth and geometric constraints of the learning model.

Based on the texture gradients' cue, the approach of Torralba and Oliva [111] relies on the idea that the global image structure (encoded using a Fourier transform)

Table 2.1: Depth estimation approaches that use different monocular cues. With "X" we mark cues used in each paper. Application to indoor/outdoor scenes is denoted by I/O, respectively.

Paper	Cues							Scene	Results
	O	LP	RFS	RH	TG	AP	S		
Dimicolli et al. [27]	X							I/O	Object order
Criminisi [23]		X						I/O	3D model
Battiato et al. [10]		X						I/O	Depth map
Wang et al. [118]		X						I/O	Measurement information
Huang et al. [61]		X						I	3D model
Chen et al. [17]			X					I/O	Object depth
Torralba et al. [111]					X			I/O	Scene mean depth
He et al. [54]						X		O	Depth map
Hoiem et al. [56]		X		X	X			O	3D model
Saxena et al. [96]					X	X		O	3D model + depth map
Nedovic et al. [83]		X			X	X		I/O	3D model
Liu et al. [77]			X	X	X			O	Depth map
Our approach				X	X			O	Coarse depth map

strongly differs in close-ups and panoramic views, urban and natural environments, etc. Given an image, their proposal estimates the magnitude order of its absolute mean depth. Note that in this case depth is inferred at the level of the whole image.

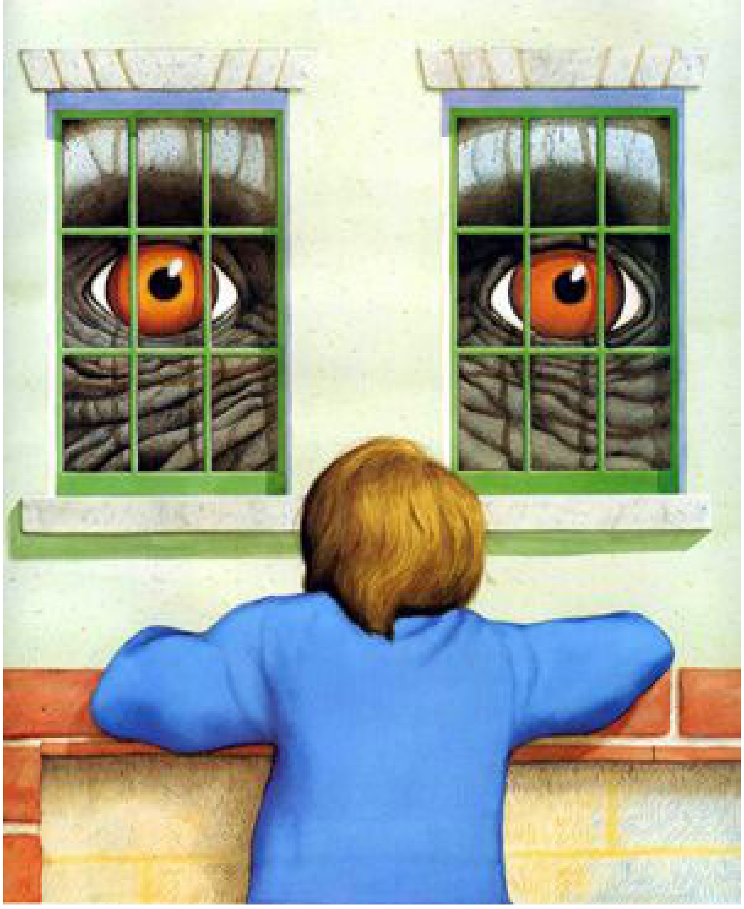
An example of the use of aerial perspective cue is found in [54]. In this work, He et al. propose an approach for removing the haze of an image, and as a consequence of this process a scene depth map is generated. The drawback of this method is that the haze presence is required to make the depth map estimable, and obviously that does not always happens.

Other proposals combine several monocular cues in order to reconstruct a 3D model or a depth map. For instance, a rough geometric model of the global scene is obtained in [83]. To this end, scenes are categorized into different geometric types (called stages). Based on visual features for depth estimation, a support vector machine (SVM) is trained to classify images into their corresponding stage category, which is a first approximation of scene depth. Another approach proposed by Hoiem et al. [56] computes impressive 3D "popup" models for outdoor scenes from a single image. Their approach is based on the statistical modeling of geometric classes that consists of ground, sky, and vertical objects. For this purpose, an image is oversegmented into uniform regions, each of which belongs to a particular geometric class, and it is described by depth cues. Then, from a logistic regression form of AdaBoost previously trained, the geometric class of each region is inferred. Finally, the 3D model is constructed by estimating the position of vertical objects with respect to ground. However, since they assume a "ground-vertical" structure of the environment, the method fails on environments that do not satisfy this assumption. Saxena et al. [96] introduce an approach to recover a detailed 3D reconstruction and a depth map of arbitrary outdoor scenes. A Markov Random Field (MRF), trained by supervised learning, is used to infer 3D information of homogeneous regions in an image. MRF allows modeling a set of constraints that capture image depth as well as relations between neighboring regions.

In the next chapter, we describe our proposal to coarse depth estimation. Our approach has resemblances to the one proposed by Hoiem et al. [57], in the sense that both works address an image labeling problem. However, while their goal is labeling image regions according to geometric classes, our method aims to do that in terms of depth categories, which for a given camera correspond to absolute depth ranges. Technically our approach is also different. In our design we have prioritized the use of low-level features without employing too much computation. We impose that requirement because the results of our proposal are not a final objective, but just the starting point to tackle different computer vision tasks from a different and novel perspective.

2.4 Summary

In this chapter, we have studied different cues that human visual system use to support depth perception. We have also focused on how each of these cues has been applied in computer vision to develop systems that extract depth information from single images. Finally, taking into account all these works, we have putted our work in relation to others.



Changes, Anthony Browne, 2008.

According with Anthony Browne, *Changes* is a picture book that deal with the concept of changing one thing into something very opposite. This book tells the story of how the world of a child suffers physical changes, when in reality they are mental changes and changes on daily routine.

The image is surrealist in the sense that it is playing an ambiguous game. It contains a disturbing mixture between a photographic realism and mystery. On the one hand, the child seems to be prisoner of its self room. In the other hand, the gorilla's eyes reveal what are its feeling about the scene. In relation with our work, this express that an image is very informative. An image can expose many things like, for instance, what is thinking or feeling someone, but also information for understanding the scene that we are seeing like, for example, depths, relation between objects, etc. This chapter is about what simple visual cues are useful to depth estimation.

Chapter 3

Recovering depth information from a single image

En el ovillo, al final de la madeja, siempre aparece lo más sencillo y básico para andar por casa [. . .]. **El viaje a la felicidad, E. Punset, 2010.**

In this chapter, we propose a simple algorithm to classify the pixels of outdoor images in just four categories: near, medium-distance, far and very-far. Based on the analysis of monocular cues done in Chapter 2 and by constraining our domain to outdoor images taken from a camera parallel to the ground, we propose a reduced set of low-level features to infer depth information. These features are used to describe the small regions of a grid defined in the image, which are processed by four Adaboost classifiers whose output is used to segment the image into depth categories. The quantitative evaluation of this simple approach shows a reliable performance, overcoming state-of-the-art proposals requiring higher computation.

3.1 Introduction

Recently, some proposals on depth estimation and 3D reconstruction from a single outdoor image have been done [56, 57, 96, 77]. These works focus on the design of sophisticated features and levels of reasoning for the accurate depth estimation. Their final goal is inferring the 3D structure of the scene as good as possible. However, with the aim of solving a given computer vision problem, the computation of 3D information could be just the starting step to analyze posteriorly an image more reliably instead of being a goal in itself. Taking that into consideration, our work has focused on developing a low-cost method to estimate from single images coarse depth information, bu informative enough to tackle many computer vision problems from a different perspective, leading to alternative and more reliable solutions. In this chapter, we describe our proposal to build such system.

To develop such system, we fix the application domain for which it will be developed. Extracting depth information from a single image is possible thanks to the prior knowledge that we can apply, so developing a generic method is not be feasible (no assumptions could be done). In our case, we have developed the system for outdoor urban images, taken with a camera whose X axis is parallel to the ground.

We have decided to describe depth in terms of 4 discrete categories: near, medium-distance, far and very far. For a given camera, each of these relative concepts is in fact translated to a depth range, that we have established taking into account properties of the elements observed inside that ranges. Although the chosen categorization is somewhat arbitrary and subjective, we think that can be useful for many tasks. In fact, for some applications, just using the image information inside one of the categories may be enough.

Once established these core decisions, we have designed our system following a common pipeline in multiclass image segmentation. First the image is divided in small regions, and a descriptor is computed for each of them. Then several classifiers are applied on these regions, and their output is used to infer the desired discrete depth map using a conditioned random field model. Results obtained with this standard approach have been satisfactory, overcoming the performance of previous approaches.

In the following sections we detail the work carried on. Section 3.2 describes and justifies our technical approach to build the desired depth segmenter: the image region descriptor selected, the definition of the depth categories for a given camera, the classifier learning approach, and finally the depth map inference algorithm used. In Section 3.3 experiments conducted to quantify the performance of our proposal are detailed. Several tests have been done to analyze the effect that different implementation possibilities have on the performance. The suitability of the proposed region descriptor is shown, as well as the validity of the processing pipeline chosen. Finally, Section 3.4 provides some conclusions.

3.2 Depth-based region segmentation

Our proposal for estimating the coarse depth maps of an image is depicted in Fig. 3.1. Our method first defines small regions in the image, and describe them by means of a vector of local image features. These descriptors are then processed by previously trained classifiers, which are used to predict the likelihood of each region of belonging to each depth category. The final depth map is inferred by processing the classifier outputs in a conditional random field model. This is in fact a quite standard multilayer segmentation pipeline. In the following we detail the different alternatives that we have considered to implement this pipeline. The final setting of our system will be fixed according to results achieved in experimental work.

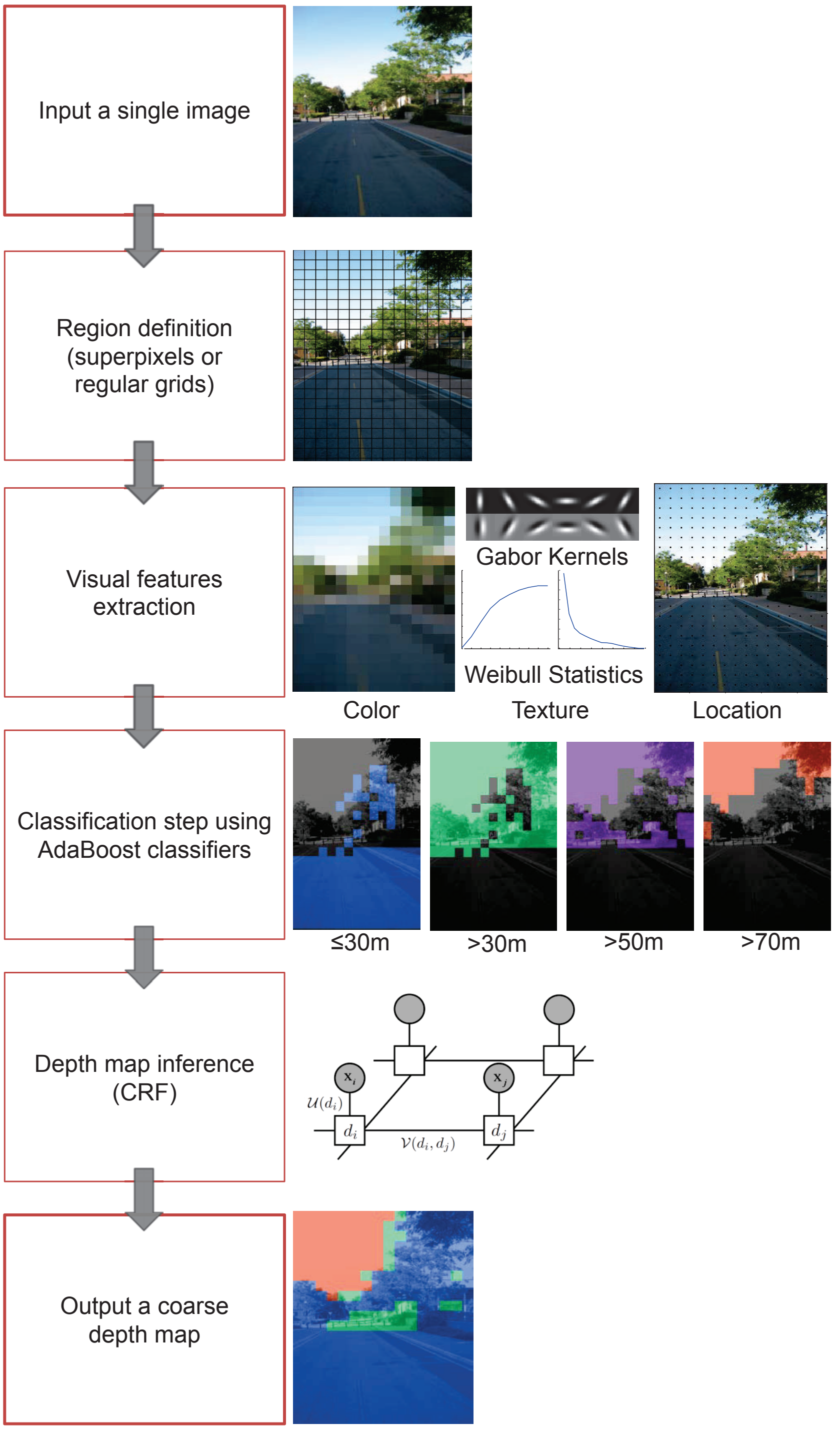

Input a single image
Region definition (superpixels or regular grids)
Visual features extraction
Gabor Kernels
Weibull Statistics
Color
Texture
Location
Classification step using AdaBoost classifiers
≤30m
>30m
>50m
>70m
Depth map inference (CRF)
x_i
x_j
$\mathcal{U}(d_i)$
d_i
d_j
$\mathcal{V}(d_i, d_j)$
Output a coarse depth map

Figure 3.1: A schematic overview of our approach.

3.2.1 Small regions definition

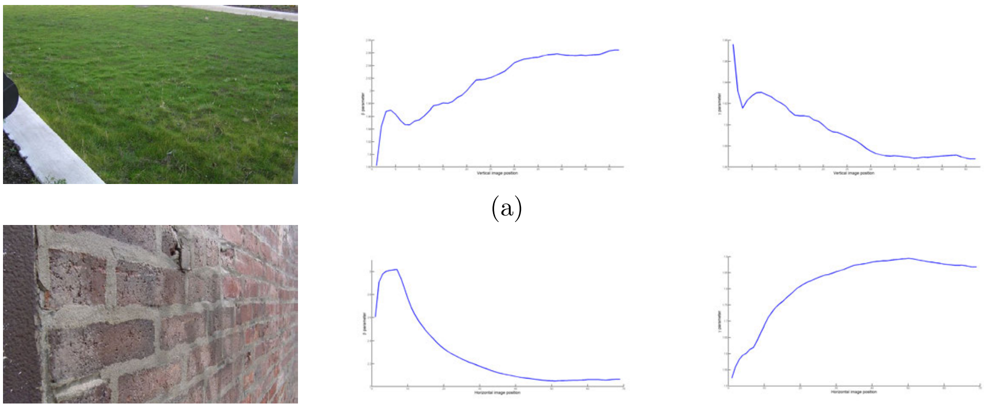
Our process start by first defining a lattice of small regions on the image. We decide to infer depth at a region level rather than at pixel level because in general depth is spatially correlated, and since our depth inference is coarse, there is a high likelihood that neighboring pixels correspond to the same depth category. Moreover, most visual cues involved in depth perception take the pixels surroundings into consideration, hence working with regions increases computational efficiency because a same descriptor is shared for all grouped pixels.

So, given an image, first it is oversegmented into small regions. To do that, we consider applying a superpixel approach [92]; that is, oversegment the image taking into account intra-region similarities (e.g., with uniform color and texture). This increases the likelihood that all pixels in a region indeed belong to the same depth category. However, for the sake of reducing computational needs, we also consider defining regions just by means of a regular grid. Although this will provoke misclassification errors in the borders between categories, the inaccuracy of the resulting depth map may be tolerable for posterior depth map uses.

3.2.2 Visual features

From the studies presented in Chapter 2, we observe that cues used by the different approaches imply distinct levels of reasoning. For the sake of simplicity we have decided to build our system using just low-level reasoning (i.e., discarding the explicit use of perspective, object recognition, etc.), and check the performance that can be achieved with that. Indeed, we are aware that for the selected application domain, these high-level cues can be informative. For instance, in man-made outdoor environments, a strong indicator of depth are the parallel lines receding into the distance, which apparently meet at a vanishing point. Then, the vanishing point location can be useful to depth estimation because gives important information on the distance of scene objects and the 3D structure. Closely linked to this, if information about size of familiar objects is available, we can use that to determine their distance from the camera. This is due to the perceived distance increases as the image size of an object decreases. Additionally, the relative height of an object and its location with respect to the horizon is informative of depth. The occlusions between objects can be useful to define depth discontinuities and object boundaries in depth, providing ordinal depth information. Cast shadows originating from one object and falling on another are valuable to determine relative depth. Attached shadows are also helpful in judging 3D shape.

In order to define a descriptor for the image regions, we have considered visual features remarked by different depth perception studies, as well as the ones used in previous effective depth estimation approaches. Obviously we have also taken into account the application domain for which our system is developed: outdoor urban images taken with a camera whose X axis is parallel to the ground. As result of that, we have defined our region descriptor based just on color, texture and location



(a)

(b)

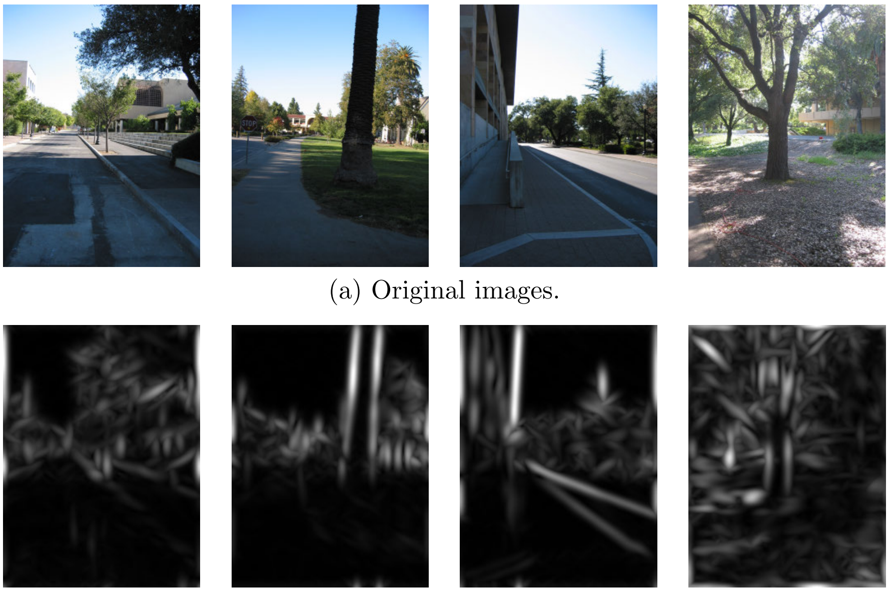
Figure 3.2: Example of Weibull parameter fitting as a function of depth for two surfaces: (a) grass and (b) wall bricks.

features due to the following reasons:

- Color is useful to both distinguish between different objects in an image and, in itself, as a strong depth cue [113]. Experiments in color perception have shown that objects colored in red or yellow are judged to appear closer than objects in blue or green [49, 6]. Additionally, the colors of distant objects appear much less saturated as a cause of atmospheric scattering. These properties can be coded by using RGB and HSV color spaces [112, 57]. We account for them by computing the mean and the histogram of the region's values in the three channels of both RGB and HSV color spaces. The histogram is constructed by concatenating three-bin histograms computed independently at each channel.

- Texture gradients change in objects according to distances from the camera.For each region, a compact representation of them can be obtained by fitting a Weibull distribution to a histogram of Gaussian derivatives in x and y directions, resulting in β and γ parameters for each direction. These parameters are indicative of local depth order and the direction of depth [39]. Figure 3.2 demonstrates the change in Weibull parameters over depth. The Weibull parameters are estimated for each row/column of the image respectively, showing that the β parameter decreases when the depth in the scene increases, and γ has the opposite behavior.

  Additionally, as shown in [111], in urban scenarios, the scene structure is dominated by smooth surfaces on the bottom (e.g., roads) and also on the top due to the sky, whereas the center contains buildings with high frequency textures (see Fig. 3.3). In order to capture this information, we propose to compute the energy response of a Gabor filter bank [73] on the whole image, which is appropriate for texture representation and discrimination. We use Gabor filters with six orientations and three scales to emphasize regions in the image of high



(a) Original images.

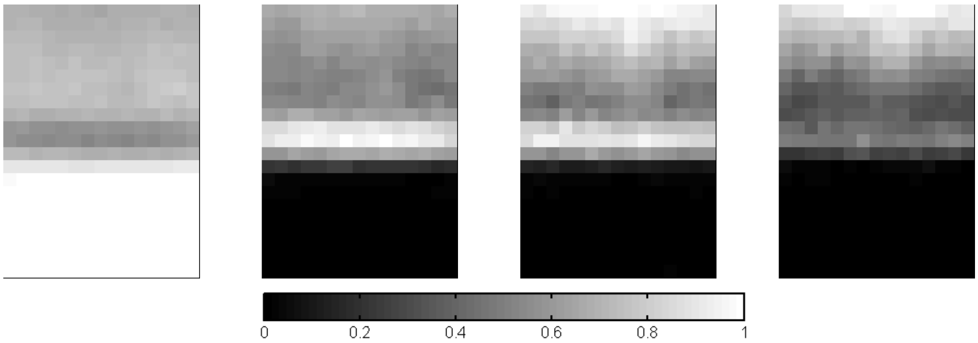
(b) Gabor filter responses.

Figure 3.3: Examples of energy response of a Gabor filter bank.

texture and capture the unknown scale variations that may occur. By computing them on the whole image and then collecting their energy at each region we take both texture and spatial information into account, which have been proved useful as depth perception cues [111].

- Location provides useful information to distinguish some image regions that commonly tend to stay in specific image areas in the application domain considered. For example, sky regions are usually placed at the top of the image, while the ground tends to be at the bottom. The distance of these regions with respect to the camera can be (roughly) guessed. Hence, for images taken with a camera with the X axis parallel to the ground, the y position of a region in the image is very informative. Figure 3.4 shows for a dataset of urban images, the likelihood of different depth ranges given the x and y position in the image. We can observe that regions in the range [0,30] m are mainly located at the bottom half of the image. Regions in the range (30,50] m tend to be positioned on the center, near an hypothetical horizon line, and some at top. As the depth range increases, regions tend to be located closer to the top of the image.

These are the visual features that our system will use. Except for the Weibull parameters and color histograms, we compute the 25th, 50th, and 75th percentiles of the feature value inside the region. With that, we want to capture minimally how the feature value is distributed. The complete region descriptor is built by concatenating all features, resulting in a 97-dimensional vector.



(a) near [0,30] m (b) medium (30,50] m (c) far (50,70] m (d) very-far (70,∞) m

Figure 3.4: Likelihood of different depth categories given their location in the image.

3.2.3 Depth categories

Our system classifies the image pixels in four depth categories. However, for many applications even a coarser depth map could be enough. Hence, depending on the posterior use of the depth map, the categorization that we propose could be varied. The key is defining categories that can be discriminated with pictorial cues. In our case, we have decided to distinguish between the regions near, medium, far and very far because that can be of help in many posterior applications, and also demonstrates the validity of the proposed system.

We have defined the four categories in our system taking into account how pictorial information is reflected on images according to the characteristics of the acquisition camera. Considering that, our depth categories correspond to the following depth ranges:

- Near [0,30] m. In the human visual system, at distances greater than 30 m depth perception is based just on monocular pictorial cues [24]. For the camera used in our work (effective focal length of 348 pixels) a similar statement can be done. If the considered camera were used in a typical stereo rig configuration (i.e. baseline of 12 cm) the disparity values at 30 m would be around 1.5 pixels. Hence, further than that, the depth estimation from stereoscopy would be quite inaccurate. Taking that into consideration, we define the near category as the depth range [0,30] m, where depth usually would be estimated through stereoscopy.
- Very far: (70,∞) m. We have established this category using as criterion how the effect of camera translation is observed in two acquired frames (i.e., the parallax). For the camera used, it turns out that for camera translations of 1 meter magnitude, elements beyond 70 meters generate optical flow vectors of subpixel magnitude. That is to say, if images were taken at 25 frames per second from a car moving forward at 90 Km/h, no pixel motion would be appreciated at this region.

- Medium: (30,50] m and Far: (50,70] m. To define these categories, we have arbitrarily set a threshold at the middle point of the range [30,70].

3.2.4 Depth classifiers

Our problem is clearly a multiclass learning problem since we have four depth categories to be identified in images. In machine learning, there are classification algorithms that directly deal with this kind of problems (e.g., [22, 21, 124]). In contrast to these algorithms, other schemes cope with multiclass problems by different combinations of binary classification algorithms. Experimental evidence shown that binary-based multiclass schemes perform as well as those that naturally support multiclass [94]. In this work, we prefer the former due to their computational and conceptual simplicity.

Regarding binary-based multiclass schemes, a set of binary classifiers can be combined using two simplest and commonly used strategies. Given M different classes, one of the these schemes is training M different binary classifiers (i.e., one for each class) to distinguish the examples in a single class from the rest of the classes. When it is desired to classify a new example, the M classifiers are run, and the example is labeled according to the highest response provided by one of the classifiers. This scheme is referred to as the "one-vs-all". Other scheme is the so-called "all-pairs" or "all-vs-all" scheme. In this approach, $\binom{M}{2}$ binary classifiers are trained (i.e., all possible pair combinations of a set of M classes). In this case, each classifier separates a pair of classes.

We have tested these two popular methods to perform multiclass classification, considering the four classes defined in the previous section (i.e., $[0, 30]$, $(30, 50]$, $(50, 70]$, and $(70, \infty)m$). In the one-vs-all approach, we have trained one classifier per class to distinguish between depth ranges (i.e., a classifier to distinguish regions belonging to $[0, 30]$ m from the rest, another classifier for distinguish regions in $(30, 50]$ m from the rest, and so on). In the all-vs-all scheme, we build a classifier to discriminate between a pair of classes (i.e., a classifier to distinguish between regions belonging to $[0, 30]$ m from those in $(30, 50]$ m, a classifier to distinguish regions in $[0, 30]$ m from $(50, 70]$ m regions , a classifier to distinguish regions in $[0, 30]$ m from $(70, \infty)$ m regions, etc). Results using both approaches showed a discouraging low performance. Due to this, we have devised a different way to solve this problem.

The proposed pipeline requires having three classifiers, whose output will be used to predict the depth category of each region. Instead of using the depth ranges previously stated, we have used the following ones: $(30, \infty)$ m, $(50, \infty)$ m, and $(70, \infty)$ m.

A binary classifier has been trained for each range by using the following procedure. From a training set of images containing regions described by our selected visual features, we have labeled a set of positive/negative examples to train a classifier able to distinguish between regions below/over one of our previously defined distance ranges. To do that, ground truth information about the depth of the observed scene is required. This can be provided, for instance, by a laser sensor, whose output is mapped on acquired images. Then, each example is labeled as positive if at least 75% of pixels is within the current threshold.

We use boosted decision trees for the classifier trained by using Real Adaptive Boosting (AdaBoost) [98]. Boosting is a method for building a strong classifier by an ensemble of weak classifiers, each of which is moderately accurate. Formally, following the notation of [98], AdaBoost takes as input a sequence of training examples $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_K, y_K)\}$ in the domain X, where each $\mathbf{x}_k$ is a vector of our visual features describing a region i and y_k belongs to $\{-1, +1\}$ depending if that region is labeled as negative or positive, with $k = 1, \ldots, K$. In each t of $1 \ldots T$ rounds, a weak classifier with hypothesis $h_t : X \to \mathbb{R}$ is build. The sign of h_t is interpreted as the predicted label of $\mathbf{x}_k$, and the confidence in this prediction is given by its magnitude $|h_t|$.

A strong classifier is computed based on a weighted linear combination of T weak classifiers as

$$H(x) = sign\left(\sum_{t=1}^{T} h_t(\mathbf{x}_k)\right) \text{with } k = 1, \ldots, K \ . \tag{3.1}$$

AdaBoost is adaptive in the sense that each new weak classifier is forced to focus on those examples which were wrongly classified by previous classifiers. For this purpose, the algorithm maintains a set of weights D over S, where each $\mathbf{x}_k$ on round t its weighted by $D_t(k)$. Initially, all weights are set equally, but on each round, the weights of incorrectly classified examples are increased, which causes that the weak learner focuses on the hard examples [31].

Although each training region is described by a set of descriptors, only the ones which are more discriminants for the depth segmentation task are used. These are selected automatically during the training process.

Once we have trained our classifiers, we can apply them to a new image. First, we oversegment it, and then compute for each image region i the likelihood of belonging to each depth category $d_i \in \{1, \ldots, M\}$ applying the previously trained classifiers. To do that, the response of each classifier is transformed into the conditional probability based on the analogies between AdaBoost and logistic regression [32, 99] using the sigmoid fuction as:

$$P(i) = \frac{1}{(1 + e^{(-\beta F(i))})} \ ,$$

where $F(i) = \sum_{t=1}^{T} h_t(\mathbf{x}_k)$ is the classifier score function from Eq. 3.1, and β is a parameter controlling the slope of the curve. Intuitively, the shape of the curve is more threshold-like when β increases. We have selected $\beta = 3$ since it perform better for our purposes. An example of this sigmoid function is depicted in Fig. 3.5.

Finally, the probability of a region i belonging to certain depth category is computed as follows by combining the output of the 3 learned classifiers:

- $P(\text{near}|i)$ is obtained from the first classifier as $1 - P((30, \infty)|i)$.

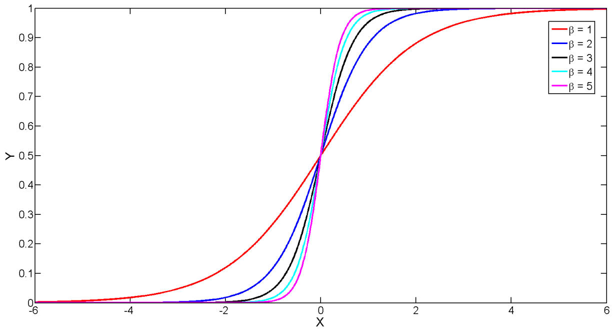


Figure 3.5: Plot of the sigmoid function, varying the values of β from 1 to 5.

- $P(\text{very far}|i)$ is directly obtained from the last classifier (i.e., the one discriminating the range $(70, \infty)$ m).

To derive the probabilities of medium and far regions, we have transformed the probabilities of $(30, \infty)$ and $(50, \infty)$ using the addition law of probability for mutually exclusive events, stating that:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) \ .$$

Thus, applying this rule we have the following:

$$P((30, \infty)|i) \quad = \quad P((30, 50]|i) + P((50, \infty)|i) - P((30, 50] \cap (50, \infty)|i) \quad .$$

Arranging terms, we obtain:

$$P((30, 50]|i) \quad = \quad P((30, \infty)|i) - P((50, \infty)|i) \quad ,$$

since $P((30, 50] \cap (50, \infty)|i) = 0$ because both ranges do not intersect at all. Then,

- $P(\text{medium}|i) = P((30, 50]|i) = P((30, \infty)|i) - P((50, \infty)|i)$

In a similar form, the same derivation is applied for far distance category:

- $P(\text{far}|i) = P((50, 70]|i) = P((50, \infty)|i) - P((70, \infty)|i)$

3.2.5 Depth map inference

A depth map is computed by assigning a depth category to each region (i.e., infer the desired depth map). We pose that as an energy minimization problem using the following conditional random field expression:

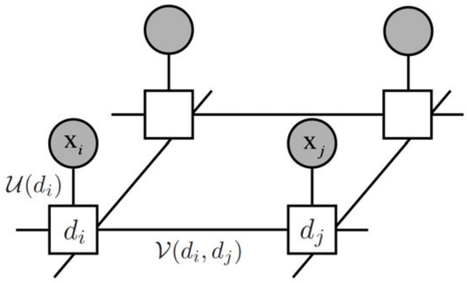


Figure 3.6: Illustrative example of the graphical model in order to estimate the depth map segmentation. Square nodes represent discrete variables and circular nodes continuous variables. Shaded nodes denote observed variables; non–shaded nodes are hidden variables. The conditional dependencies between variables are represented by solid lines.

$$E(\mathbf{d}) = \sum_{i=1}^{N} \mathcal{U}(d_i) + \lambda \sum_{(i,j)\in\mathcal{E}} \mathcal{V}_{i,j}(d_i, d_j). \tag{3.2}$$

$E(\mathbf{d})$ is the energy of a coarse depth segmentation $\mathbf{d} = [d_1, \ldots, d_N]$, being N the number of image regions. $\mathcal{U}(d_i)$ is the likelihood to belong to a depth range, and $\mathcal{V}_{i,j}(d_i, d_j)$ is a regularization term that models the compatibility between depth ranges of pairwise neighboring images regions defined in $\mathcal{E}$. The parameter λ weights the influence of the regularization term in the energy minimization.

With this scheme we want to encourage the association of neighboring image regions to the same category. This is managed by the term $\mathcal{V}_{i,j}(d_i, d_j)$, which applies a contrast sensitive Potts model [68] to the likelihood of the depth categories of the regions in a neighborhood. When depth categories differ significantly, the penalty of this term is relaxed, in order to keep depth discontinuities. However, in the cases where the disparity between regions is *weak*, this term promotes a unification of their final labeling.

Finally, we infer the global optimal coarse depth segmentation by applying graph cuts [15], since it guarantees to find a global maximum likelihood result. Results of this process are shown in Fig. 3.8.

3.3 Experimental results

In our experimental work, we have performed three kinds of experiments. First, we have analyzed the performance achieved by different configurations of our depth segmentation approach, in order to identify the best performing one. Then, we have made an analysis of the significance of the proposed visual cues for the sake of depth segmentation. Finally, we have compared the performance of our proposal with respect to other proposals.

For all these evaluations, we have used the dataset provided by Saxena et al. [96][1]. This dataset is composed by 534 images taken at diverse urban and natural areas. Each image has a ground truth depth map associated, acquired by a laser scanner with a maximum range of 81 m. Images are taken with the camera's horizontal (X) axis parallel to the ground, and Z axis corresponds to the camera's principal axis, so they are in accordance with the acquisition conditions we assume. In our tests, we have resized images to 240×320 pixels, using a focal length of 348 pixels.

To perform these experiments, we have trained our classifiers using 400 images from the dataset, and we have used 134 images for testing. We learn strong classifiers composed of 100 weak classifiers, being each one of them a decision stump. We trained classifiers using the GML AdaBoost Matlab Toolbox[2]

To quantify the performance of our proposal, we take into account the performance achieved by segmenting each one of the considered regions, as well as the classification ratio of the overall scheme. In the first case, we use the Jaccard index [72] as criterion, while in the latter, we use the average of that index. This is a typical approach for evaluating multiclass segmentation problems [28].

Jaccard index is defined as $\frac{TP}{(TP+FP+FN)}$. Intuitively, it measures the level of agreement with respect to an ideal classification result. If a prediction has many true positives, and relatively few false positives and false negatives, the level of agreement is high.

Additionally, to analyze the performance we use a confusion matrix, where the diagonal terms correspond to correctly classified instances, whereas off-diagonal terms represent incorrectly classified ones.

3.3.1 Analysis of segmentation performance

In the following experiments, we quantify the performance achieved by of our depth segmentation approach for different configurations of the image oversegmentation process. Specifically, we evaluate the effect on the performance of the number of regions for both regular grid and superpixels. Regular grids are evaluated using different levels of resolutions with windows of 10×10, 15×15 and 20×20 pixels. The use of superpixels is evaluated using approximately 200, 400 and 800 regions, which is the amount of regions equivalent to the considered regular grid configurations. We plot the results of our experiments in Tab. 3.1.

Note that, the advantage of using superpixels is appreciated just qualitatively when regions are quite large (i.e., considering approximately 200 superpixels regions versus a 20×20 grid). In this case, the use of superpixels is the best configuration. However, it is only 1% better with respect to a grid with the same number of regions.

Figure 3.7 shows a qualitative example of our results. This example illustrates the performance of our approach using different resolutions and varying the algorithm to

[1] http://www.cs.cornell.edu/~asaxena/learningdepth

[2] http://graphics.cs.msu.ru.

Table 3.1: Comparative of average Jaccard index with respect to the number of regions. We use regular grids and TurboPixels of different resolutions.

Oversegmentation	Number of regions		
algorithm	20x20	15x15	10x10
TurboPixels	0.3623	0.3567	0.3561
Grid	0.3586	0.3602	0.3570

conform regions. Misclassification using larger regions is due to the fact that regions are quite inconsistent, mixing features of regions located at different depths, specially in the case of regular grids. Increasing the number of considered regions improves the segmentation, and decreases the existing artifacts at boundaries (e.g., between sky and trees). But when regions are too small, the accuracy of segmentation decreases and the result contains misclassified regions because they include less contextual information, making the classifier more sensitive to noise and variations.

Experiments conducted in the rest of the paper have been made using a regular grid of 15×15. We chose this option due to the following criteria:

- Segmentation: A regular grid only must be computed once for a certain image configuration, while superpixels must be computed for each image.
- Performance: The chosen configuration has the best grid performance. The difference in average Jaccard index performance with respect to the best configuration (i.e., superpixels of 200 regions) is less than 1%.
- Time: Using a regular grid of 15×15, our approach is approximately 4 times more faster than using the best configuration of grid or superpixels.
- Applications: This level of resolution is enough for the applications described in Chapters 4, 5, and 6, as the obtained results will show.

3.3.2 Analysis of visual features

Once chosen a framework configuration, we have focused our experiments to analyze the significance of the visual features selected for the proposed coarse depth segmentation. In human perception, there are complex interactions between the various depth cues. Different cues cooperate or compete between them to support the depth perception. In that sense, the presence of one or more depth cues may either enhance or suppress the effect of other cues [66]. Considering that the discriminability of each feature can be different, we also analyze the behavior of each one for our goal. Table 3.2 shows the features actually used by the classifiers. Although all features are available in our configuration, a process to select which features cooperate in disambiguating the information provided by them is performed by AdaBoost. Those features with little relative weight are ignored.

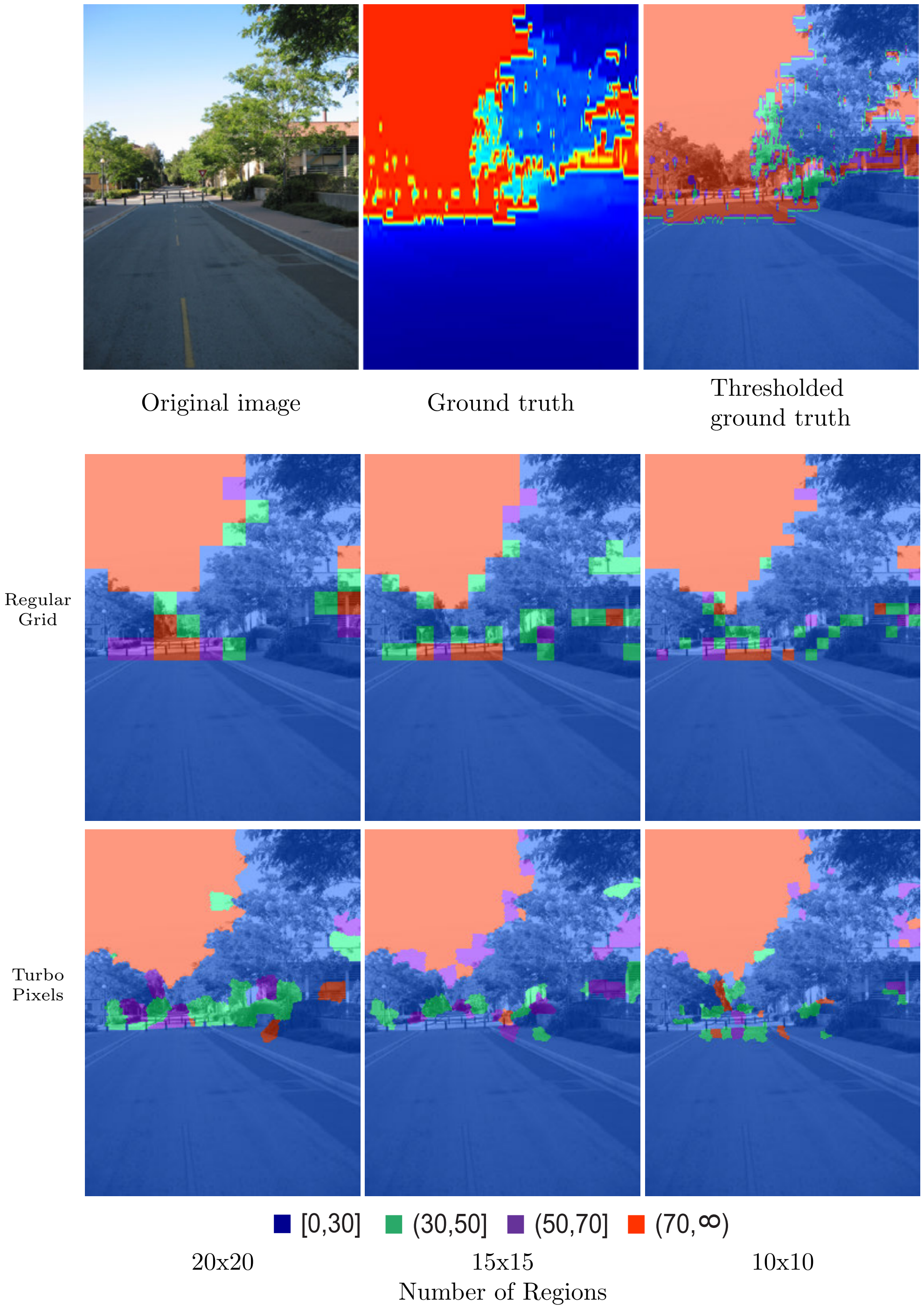


Figure 3.7: Qualitative example of our results varying the resolution and the algorithm to over-segment the image.

Table 3.2: Features computed on regions. The "Dim" column gives the number of considered features. Since the classifier selects the more discriminative features, the "Relevant" column shows the number of features actually used by it.

Features	**Dim.**	**Relevant**
Color	**33**	**30**
- RGB median	3	2
- 25th and 75th percentiles of RGB	6	6
- HSV median	3	3
- 25th and 75th percentiles of HSV	6	6
- RGB histogram (3 bins)	9	7
- HS histogram (3 bins)	6	5
Texture	**58**	**32**
- β and γ Weibull in x & y-directions	4	4
- Gabor responses median (6 orientations, 3 scales)	18	9
- Gabor responses 25th and 75th percentiles	36	19
Location	**6**	**2**
- Region centroid coordinates	2	2
- 25th & 75th percentiles of x & y	4	0

We note that classifiers' decisions first substantially rely on color features. This is because in our target domain, a big part of the image is dominated by ground and sky regions. Since these regions are mainly untextured color regions and they are highly correlated with a given depth category (near and very far, respectively), this captures the attention of Adaboost in the first iterations. Next considered features are the Weibull texture description and the position coordinate of regions, being the Gabor filter responses the last features used.

In order to assess the complementarity of the three visual cues selected (color, texture and location), we have also analyzed the performance achieved by different combinations of them.

Table 3.3 summarizes the results achieved. The first three rows correspond to results obtained when considering cues in isolation. Color and texture have strong correlation with respect to depth. For instance, sky is categorized as very-far since it has approximately light-blue in all images and also it is practically untextured. Its worth to remark that image location is not too strong as we expected. This is due to the fact that location in isolation is not discriminative enough. Although each depth range tends to reside in a given image area (e.g., near are more likely located at bottom, while far and very-far categories are placed on top of the image), when the classifiers responses are combined, the probability that a region being classified as close is maximal with respect the rest of classifier results. This is due to the fact that the near-region classifier is more confident because it has been trained using a great number of positive examples compared to the rest of depth categories classifiers. These results show that location needs to be supplemented with other cues to obtain better performance.

Table 3.3: Combination of several features, and the Jaccard index obtained for each depth category. The best results obtained for each distance appear in bold.

	Feature	*Jaccard index*				
		Near	Medium	Far	Very-far	Avg
1.	Color	0.8634	0.0155	0.0040	0.4451	0.3320
2.	Texture	0.8484	0.0072	0.0022	0.3429	0.3002
3.	Location	0.6792	0.0119	0.0317	0.2098	0.2331
4.	Color + Texture	0.8619	0.0228	0.0142	0.4478	0.3367
5.	Color + Location	0.8696	0.0552	0.0286	0.4622	0.3539
6.	Texture + Location	0.8626	0.0332	0.0128	0.4179	0.3316
7.	All (w/o priors)	0.8706	0.0753	0.0265	0.4684	0.3567
8.	All + CRF (w/ priors)	0.8649	**0.1172**	0.0403	**0.4389**	**0.3653**
9.	Saxena et al.	**0.8798**	0.0983	**0.0412**	0.2838	0.3258

Classifiers trained with a combination of features have a superior performance - (rows 4 to 6), being color and location the best performing combination. As expected, the best configuration uses all cues (row 7). Performance is significantly increased by taking into account contextual information modeled by relations between neighboring image regions, as we show in row 8.

In general, the performance reached by our approach in medium and far regions is quite poor; while in very-far region, it is significantly better, and in near region, it is remarkable. These results are discussed in detail in the following section.

3.3.3 Comparison with other frameworks

Finally, we have compared the performance of our proposal against the one achieved using a more ambitious depth map estimation method. Specifically, we compare against the result of the Saxena et al. method [96], thresholding their estimated depth maps according to the values used to define our depth categories. The overall performance of our proposed method outperforms the Saxena et al. algorithm (see row 8 and 9 in Tab. 3.3), using a remarkable inferior number of low-level features (64 vs 646 respectively).

Table 3.4 shows the confusion matrices for both Saxena et al. and our method. Both methods are very precise in classifying near regions. However, mid-distance categories are difficult to be correctly classified: they are mainly confused as near regions. This ambiguity is due to the lack of enough positive examples in the training set. In the case of very-far regions, our method remarkably exceeds the accuracy of Saxena et al. Although not all very-far regions are correctly identified, in this class there are very few false positives, so the very-far areas can be reliably used for applications as the one described in Chapter 4.

A qualitative view of the performance achieved is shown in Fig. 3.8. Our coarse

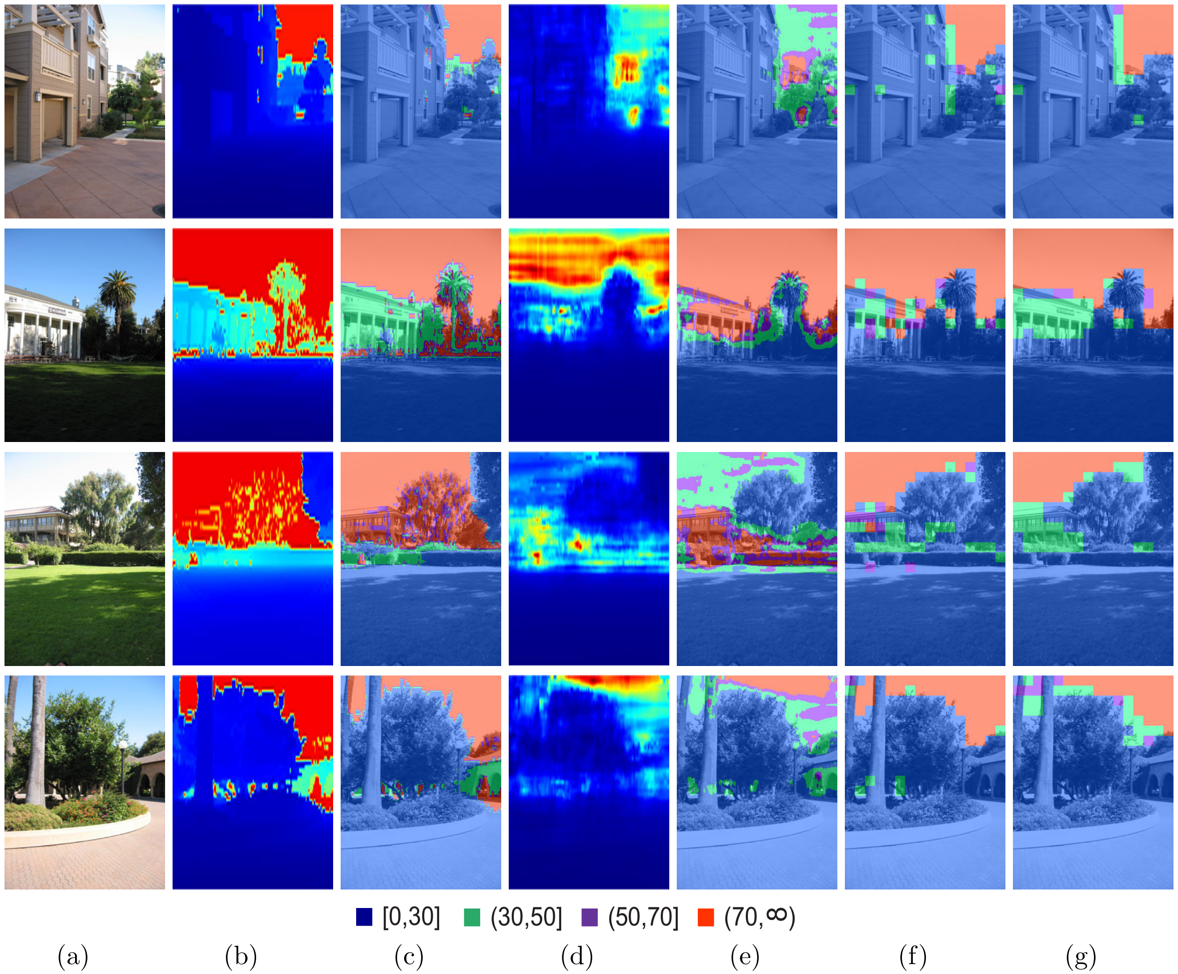


Figure 3.8: Depth segmentation results: (a) Original image, (b) Ground truth, (c) Thresholded ground truth, (d) Saxena's depth map, (e) Saxena's thresholded depth map, (f) Results of the proposed method without prior (only local information), and (g) Results of the proposed CRF method (within prior).

Table 3.4: Confusion matrices for both Saxena et al. and our approach.

		Predicted classes			
		Near	Medium	Far	Very-far
True classes	Near	**0.9447**	0.0411	0.0093	0.0049
	Medium	0.5354	**0.2665**	0.1005	0.0976
	Far	0.4911	0.2594	**0.1054**	0.1441
	Very-far	0.2296	0.2782	0.1816	**0.3106**

(a) Confusion matrix summarizing Saxena et al. results.

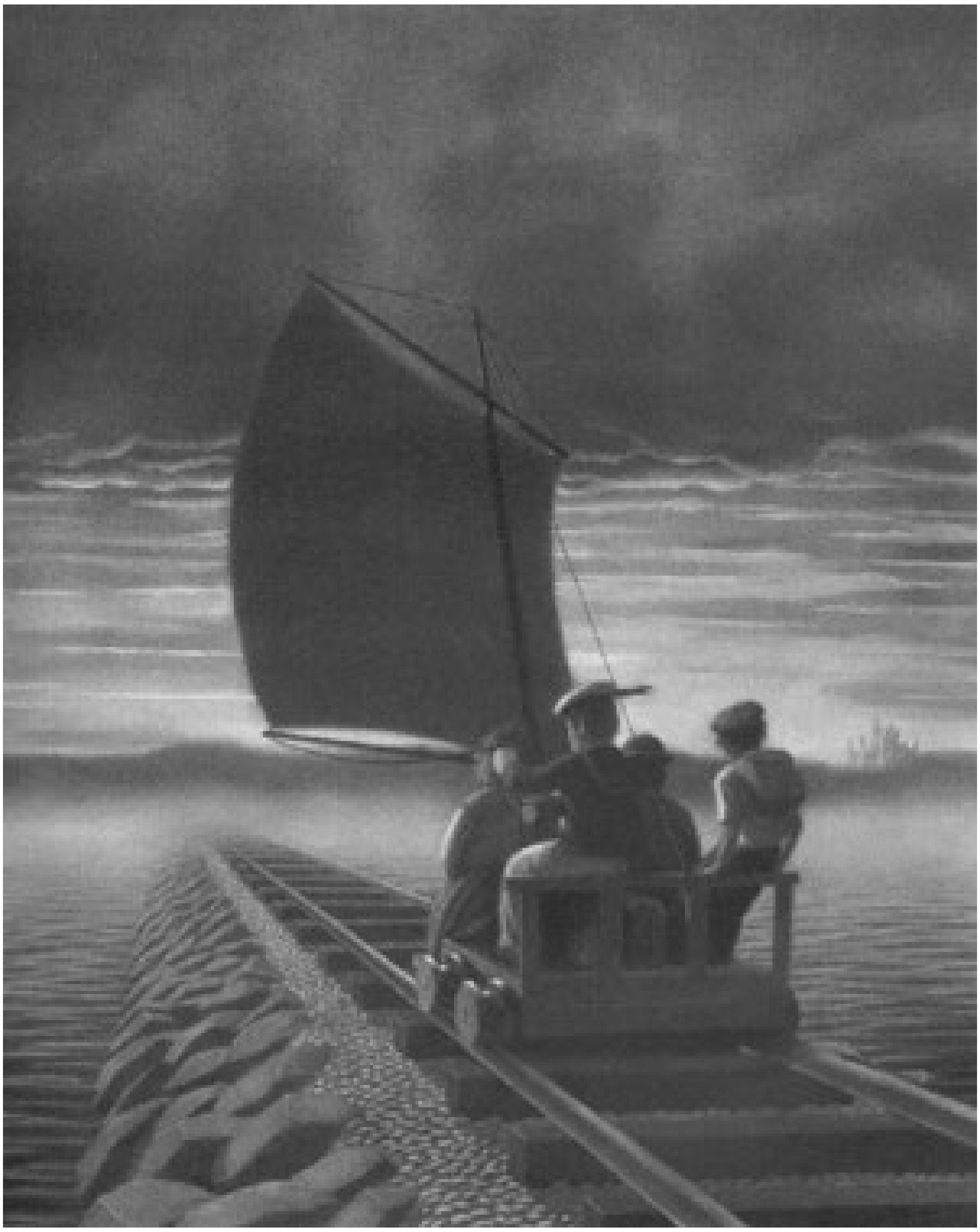
		Predicted classes			
		Near	Medium	Far	Very-far
True classes	Near	**0.9377**	0.0468	0.0109	0.0046
	Medium	0.5916	**0.3036**	0.0656	0.0392
	Far	0.5559	0.3109	**0.0801**	0.0531
	Very-far	0.2692	0.1875	0.0801	**0.4633**

(b) Confusion matrix summarizing our results.

depth maps have a good appearance regarding thresholded ground truth (Fig. 3.8, column (c) vs. (g)). Mid-distance regions (in the range between 50 and 70 m) are occasionally misclassified, which we attribute to the lack of examples in the training set. Our coarse depth maps achieve a more homogeneous and coherent aspect by including spatial coherence using the CRF model (as we can see in Fig. 3.8, column (f) vs. (g)). Regarding Saxena's approach, we can see that it performs poorly on sky pixels, whose depths are underestimated (e.g., see Fig. 3.8(e), row 1, 2, 4 and 5). Furthermore, reflective surfaces of buildings are confused as sky, and incorrectly positioned as far-away (e.g., see Fig. 3.8(e), row 3, and confronting it with respect to our estimation at column (g)).

3.4 Conclusions

In this chapter, we have presented a supervised learning approach to segment an image according to certain depth categories. Based on how humans perceive depth at far distances, several monocular visual features have been studied to estimate information about region depth. Several classifiers have been trained leading to good segmentation results with respect to state of the art approaches. From our experiments, we have observed that region descriptions including color, texture and location lead to a high-performance. Also, we have identified that the usage of all features is the best performing configuration.



Another Place, Another Time
The Mysteries of Harris Burdick, Chris Van Allsburg, 1984.

The Mysteries of Harris Burdick is a picture book by the American author Chris Van Allsburg. Each image is just accompanied by a title that suggests magical or fantastical stories, and inspires the reader to apply one's own experience in exploring every aspect of the drawing.

The picture shows children as sailors navigating in a ship on rails across the sea. The ship's heading is guided by the rails. However, when early sailors ventured out into the sea, they looked to the heaven for a more reliable means of navigation. They oriented and guided its ships using the stars as natural landmark. Despite stars are at different light years away one of each other, the celestial vault behaves like a plane at infinity. Due to the Earth's rotation, all the stars seem rotating with respect to the Pole Stars, which remains virtually fixed in the same position. In this chapter, we develop methods that rely on distant image regions that behave like a plane at infinity to estimate the egomotion.

Chapter 4

Egomotion estimation based on distant regions

> *La geometría de Tlön comprende dos disciplinas algo distintas: la visual y la táctil. La última corresponde a la nuestra y la subordinan a la primera. La base de la geometría visual es la superficie, no el punto. Esta geometría desconoce las paralelas y declara que el hombre que se desplaza modifica las formas que lo circundan.* **Tlön, Uqbar, Orbis Tertius en Ficciones, J. L. Borges, 1944.**

In this chapter we propose a novel approach to estimate the 3D motion of a vehicle from the images provides by a monocular camera. The goal is recovering the vehicle rotation and translation from the information provided by the image flow field. Unlike previous approaches, our proposal exploits the fact that the contribution of the vehicle translation on the image flow field is negligible in image zones corresponding to distant elements in the scene. Hence, if we identify distant elements in the image, then we can estimate vehicle rotation uncoupledly from vehicle translation. This allows to tackle the visual odometry problem from a different perspective, leading to more precise results. To this aim, we first segment distant elements in a frame by a classifier based approach. Then, we present two different methods to process the segmented regions in order to accurately estimate the vehicle rotation. Finally, we use the rotation information to determine the vehicle translation up to a scale factor. The performance of our proposals is evaluated on real sequences with available ground truth, achieving better results than previous approaches.

4.1 Introduction

The estimation of the vehicle egomotion consists in determining the changes in the 3D rigid vehicle position and orientation. In general, it concerns the estimation of six degrees of freedom (DOF) corresponding to the rotational and translational velocities. Egomotion information is valuable for most ADAS applications (e.g., adaptive cruise control, collision avoidance, lane-departure warning, autonomous driving, etc.), where the future position of the vehicle hosting is predicted based on the precise knowledge of previous vehicle's state. The process of estimating the vehicle egomotion is also referred as odometry [97].

Different kinds of devices are available for obtaining egomotion information, for example: rotary encoders to measure wheel rotations, inertial sensors (including accelerometers and gyroscopes), and laser systems. The use of cameras has also been demonstrated useful for this task, which is commonly known as visual odometry [84]. Cameras are insensitive to soil mechanics, and have lower drift rates than most expensive sensors [101]. Additionally, cameras are easily integrable, low-cost, low-power consumption, and can be used in conjunction with information coming from other sources such as global positioning system (GPS).

Visual odometry systems acquire input images using either single or multiple-cameras. In the case of stereo-based odometry, the egomotion is accurately determined in all 6 DOF because a calibrated stereo-rig allows to recovering the 3D world coordinates for objects in the scene [85]. Monocular systems suffer from ambiguities arising from the lack of knowledge of scene depth instead, leading to less precise results in general. Despite the fact that stereo-based systems are better posed for solving this task in terms of accuracy, there is a great interest in the ADAS context in developing systems based on a single camera. We indeed believe that monocular systems will be present in mid/low-priced vehicles solving other applications (signal recognition, lane departure warning, etc.), and this same sensor could be used for the visual odometry problem. This would extend the functionalities of existing systems without adding extra cost. Moreover, monocular systems avoid the need for recalibration required by stereo or multiple-camera systems, which is of special interest for the car industry since it makes the system more robust and reduces its maintenance needs and cost. Here, we focus on egomotion estimation using a monocular camera mounted rigidly in a vehicle, near the rear-view mirror.

Basically, visual odometry methods recover the motion parameters from the observed motion on an image sequence. They can be classified as feature-based and appearance-based approaches. Feature-based methods use interest points detected in images that are tracked over frames, while appearance-based methods use the intensity information of all pixels in the image or subregions of it. In general, both types of methods treat all image points/regions in the same way with respect to their distance from the camera. However, the image flow of points belonging to distant scene objects from the camera encodes mainly information about camera rotation[1].

Taking advantage of that, in the current chapter we propose to compute the vehicle egomotion in two phases. First the vehicle rotation is estimated, processing just information corresponding to distant elements in the scene. Then, this rotation estimation is used to determine the vehicle translation. We propose two different approaches to

[1] Specifically, in the following, distant points/regions are the ones whose optical flow due to camera translation between two consecutive frames is smaller than one pixel.

estimate rotation: one feature-based method that tracks points belonging to distant scene objects, and an appearance-based method that tracks the image projection of distant scene elements. Our two-phase strategy has several advantages with respect to most *general* egomotion methods. First, processing the image flow of selected distant points/regions provides a most reliable estimate of camera rotation because translation contribution is negligible at that points, and translation estimation errors do not affect the rotation computation. Second, our methods are not affected by ambiguities produced by camera motion [30], since distant regions are mainly affected just by rotation. Obviously, the proposed approach will not be applicable when acquired images do not show distant regions due to obstructions in the field of view (e.g., a truck or a wall in front of the vehicle). These situations can be properly detected, applying then a *regular* monocular egomotion estimation method.

The rest of the chapter is organized as follows. Section 4.2 introduces related work. In Section 4.3 we formally describes the problem of egomotion. The proposed techniques are presented in Section 4.4. Section 4.5 gives experimental results and comparisons with state-of-the-art approaches. Finally, conclusions are presented in Section 4.6.

4.2 Related work

The egomotion problem concerns the estimation of the 3D rigid motion of the camera along a sequence, involving six DOF. The goal is estimating a translation vector $\dot{\mathbf{t}}$ and the angles $\boldsymbol{\omega}$ of a rotation matrix $\mathbf{R}$ from the image motion observed in subsequent frames. This problem has been faced in the computer vision field as a part of more general problems. One of these problems is determining the optical characteristics (the intrinsic parameters) and the orientation and position (the extrinsic parameters) of the camera. The determination of such parameters is called camera calibration. This is done by using prior information (e.g., a calibration pattern) in a controlled scene [91] to determine the correspondences between 3D world coordinates and 2D image coordinates in several views.

Another classical problem addressing the egomotion estimation is the so-called structure from motion (SFM) [87], which simultaneously recovers the egomotion and the scene structure by analyzing jointly the N-views of a sequence in an off-line process, which involves non-linear techniques (such as bundle adjustment) to refine both. However, SFM does not fit to on-line applications [1].

Simultaneous localization and mapping (SLAM) techniques [7, 8] cope with on-line scene reconstruction, trying to solve an insurmountable chicken-and-egg problem: a precise map is needed for localization whereas, at the same time, an accurate camera pose estimation is needed to generate that map. SLAM requires tracking a feature along certain time to obtain an estimation. When tracking frequently fails, SLAM methods are ineffective for estimating egomotion [108]. Additionally, SLAM tries to reach a global map consistency by coping with loop-closure, but this implies a more complex and computationally expensive system [97].

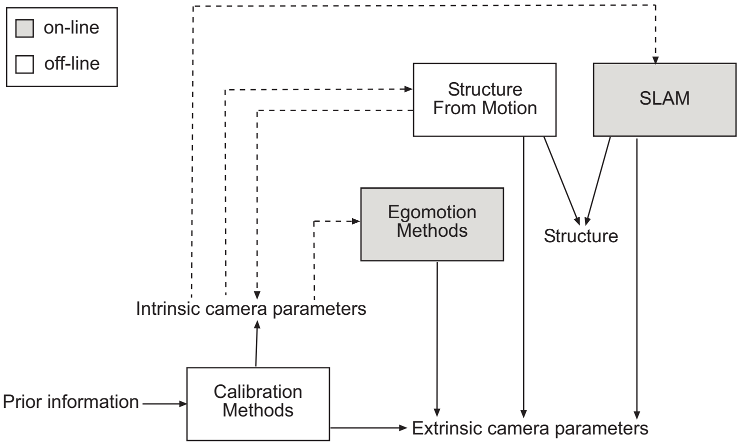


Figure 4.1: Diagram of the relationships between different fields that involve recovering camera parameters. Dotted lines depict an optional relation between calibration and other approaches, since some methods do not require as a prerequisite calibrated cameras.

In contrast to SLAM, visual odometry recovers the camera pose without estimating the 3D scene structure. Visual odometry methods compute the camera path incrementally by accumulating each new egomotion estimation frame by frame. This generates a drift of the estimated trajectory with respect to the real one, which can be reduced by applying a bundle adjustment technique or by combining with other information sources like IMU or GPS [70]. Particularly in our case, using a single camera, the DOF are reduced to five since translation can be only recovered up to a scale factor (i.e., only translation direction is estimated, while its magnitude cannot be recovered due to lack of depth information). So monocular systems require information from an additional sensor to ascertain the translation scale factor.

Summarizing, in Fig. 4.1, we illustrate the relations between different fields of the computer vision that involve recovering camera parameters, as we describe above.

As we have stated in the Section 4.1, egomotion methods can be classified as feature- [109, 3] and appearance-based methods [122, 20]. A historical review of both kind of methods can be found in [97].

Feature-based methods select interest points or corners, which are matched between views to compute egomotion. These methods use corners because they can be putted into correspondence with less ambiguity than other features. In general, feature-based egomotion methods consider all points in the similar way with respect to their depth in the scene. However, some recent works, try to take advantage of the relationship between image motion and scene depth. For instance, using a monocular camera, the authors in [16] proposed an iterative process relying on RANSAC to select optical flow vectors that can be explained only by a particular camera rotation. Implicitly, this process tries to select image points located at far-away distance from the camera. Then, using the selected set of optical flow vectors, camera rotation is estimated by solving a linear equation system. Using a stereo camera, in [86] a voting strategy is used to egomotion estimation. Rotation is estimated through a voting schema where a vote weighted with the distance of each point to the camera is assigned for each motion vector. Once rotation is computed, the rest of motion is due to translation, which is also estimated by a voting strategy. In [108], rotation and translation are estimated using far/near features, respectively. The point classification is done by checking their disparity over seven views of the same scene provided by a special omnidirectional sensor.

As the previous approaches, the proposals described in this chapter also exploit the information of distant point to estimate the vehicle egomotion. However, in our case we propose to identify distant regions from depth information inherently available in monocular cues, that allows to discriminate between far/close scene region.

Our first approach consists on estimating the camera rotation parameters through a feature-based method that uses corresponding points between consecutive images located at far distances in the image. Although a nice performance is achieved, this approach does not exploit all the available information in images to estimate rotation. For instance, some distant regions like sky or mountains are discarded by feature-based approaches because no interest points are detected on nearly uniform and low-textured regions.

In contrast to feature-based methods, appearance-based or direct methods avoid feature matching, and use only measurable information from images [64, 62, 20, 74]. Most of these approaches require a dominant physical plane existing in the scene to obtain an accurate estimation of the camera motion. However, if no such planar surface exists, then this kind of methods cannot be applicable [63]. Instead of requiring a real scene plane, we propose an appearance-based method that extracts camera rotation information from distant regions since it can be considered that they behave like a plane at infinity.

4.3 Problem formalization

In this section, we define some concepts needed for understanding the rest of the chapter. First, we establish the assumed camera model. Then, we focus in the relation between two camera viewpoints described through a rigid body motion. Assuming that the camera moves according with that model, we formalize how corresponding points between two camera views describe the motion or image flow field. Our methods rely on this relation.

4.3.1 Camera model

In this work, we assume a pinhole camera model, which forms the image by intersecting the light rays from the objects through the projection center (i.e., the lens) with the image plane $\mathcal{I}$ [52]. Formally, given a 3D point $\mathbf{p} = [p_x, p_y, p_z]$, its 2D projection $\mathbf{q} = [q_x, q_y, q_z]$ in the image plane $\mathcal{I}$ is

$$\mathbf{q} = \mathbf{K}\mathbf{p} \ ,$$

where $\mathbf{K}$ is the camera calibration matrix defined as

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \ ,$$

where f_x and f_y are the focal length of horizontal and vertical directions, and $[o_x, o_y]$ are the coordinates of the principal point. Without loss of generality, we assume that $f = f_x = f_y$ and $[o_x, o_y]$ at origin $[0, 0]$.

4.3.2 Rigid body motion relating two views

From the classical mechanic's point of view, the camera/scene can be assumed as a rigid body (i.e., the 3D distance between two points on a rigid body remains the same after motion). Rigidity implies a static scene where there is only one rigid motion between the camera and the observed scene. In this way, the motion of the camera in a stationary scene can be interpreted equivalently as the motion of this scene supposing a static camera. Henceforth, we consider the last one condition for our formulation of motion. The importance of conceptualizing the camera being static is that allow us to express all points and vectors with respect to the camera coordinate system.

The rigid motion of a scene with respect to a camera is described by a 3×3 rotation matrix $\mathbf{R}$ and a 3×1 translation vector $\mathbf{t}$. Since the distances between all pairs of point in a rigid body remain constant throughout the motion [44], the position and orientation of the scene can be represented by the motion of a single point, for example, the center of mass $\mathbf{c} = [c_x, c_y, c_z]^T$. Now, considering an arbitrary point $\mathbf{p}_0 = [p_x, p_y, p_z]^T$ at time 0 on the scene, as shown in Fig. 4.2, the location of that point at time t is the result of first rotating $\mathbf{p}_0$ with respect to $\mathbf{c}$, and then translating it

$$\mathbf{p}_t = \mathbf{R}_t \mathbf{p}_0 + \mathbf{t}_t \ . \tag{4.1}$$

Instead of expressing the rotation with respect to $\mathbf{c}$, now we express it relative to a point $\mathbf{x}$. This implies translating $\mathbf{p}_0$ to $\mathbf{x}$, then applying the rotation of $\mathbf{p}_0$ with respect to $\mathbf{x}$, and finally, returning $\mathbf{p}_0$ to its initial position

$$\mathbf{p}_t = \mathbf{R}_t(\mathbf{p}_0 + \mathbf{x}) - \mathbf{x} + \mathbf{t}_t \ . \tag{4.2}$$

In general, the motion of each point is described by its velocity vector $\dot{\mathbf{p}}$ which is obtained by deriving Eq. (4.2) with respect to time

$$\dot{\mathbf{p}}_t = \dot{\mathbf{R}}_t(\mathbf{p}_0 + \mathbf{x}) + \dot{\mathbf{t}}_t \ , \tag{4.3}$$

where $\dot{\mathbf{R}}_t$ describes the angular velocity with which the orientation of the rigid body is rotating, and $\dot{\mathbf{t}}_t$ is equal to the rate of change of linear position.

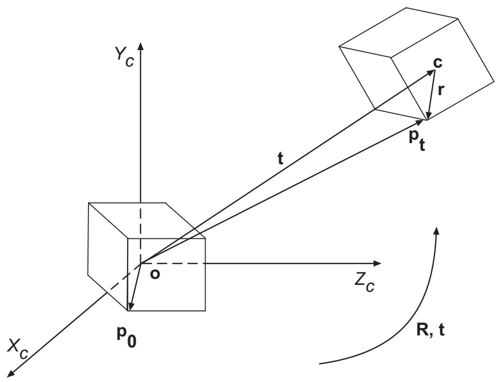


Figure 4.2: Body coordinate system with origin at the reference point $\mathbf{c}$ is fixed in the rigid body. The position and orientation of the point $\mathbf{p}_t$ is given by $\mathbf{p}_t = \mathbf{R_t}\mathbf{p}_0 + \mathbf{t}_t$.

We can reexpress Eq. (4.3) in order to put it in terms of $\mathbf{p}_t$ rather than $\mathbf{p}_0$. This is done substituting $\mathbf{p}_0$ by $\mathbf{R}_t^T(\mathbf{p}_t + \mathbf{x} - \mathbf{t}_t) - \mathbf{x}$ from Eq. (4.2), leading to

$$\dot{\mathbf{p}}_t = \dot{\mathbf{R}}_t\mathbf{R}_t^T(\mathbf{p}_t + \mathbf{x} - \mathbf{t}_t) + \dot{\mathbf{t}}_t \ .$$

If the rotation is performed with respect to the point $\mathbf{x} = \mathbf{t}$ (from now on, we avoid the use of subscript t since all variables are expressed in the same terms), then

$$\dot{\mathbf{p}} = \dot{\mathbf{R}}\mathbf{R}^T\mathbf{p} + \dot{\mathbf{t}} \ .$$

Equivalently, the previous equation can be written as

$$\dot{\mathbf{p}} = [\boldsymbol{\omega}]_\times \mathbf{p} + \dot{\mathbf{t}} \ , \tag{4.4}$$

where $[\boldsymbol{\omega}]_\times = \dot{\mathbf{R}}\mathbf{R}^T$ is a skew-symmetric or antisymmetric matrix. The vector $\boldsymbol{\omega}$ is the so-called angular velocity vector which specifies the axis about which the body is rotating (i.e., the direction of $\boldsymbol{\omega}$), and the angular speed of the body (i.e., the magnitude of rotation $|\boldsymbol{\omega}|$ per unit time).

Regarding Eq. (4.4), we can see that this equation has two components: a linear component given by $\dot{\mathbf{t}}$ and an angular component $[\boldsymbol{\omega}]_\times \mathbf{p}$, called angular velocity tensor [106].

Taking into account the cross product property that states that $\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b}$, Eq. (4.4) becomes

$$\dot{\mathbf{p}} = \boldsymbol{\omega} \times \mathbf{p} + \dot{\mathbf{t}} \ , \tag{4.5}$$

where $\dot{\mathbf{p}}$ is the velocity vector that describes the 3D motion of each point in a rigid body, with an angular velocity vector $\boldsymbol{\omega}$ and a translational velocity vector $\dot{\mathbf{t}}$.

Eq. (4.5) in components is

$$\begin{bmatrix} \dot{p_x} \\ \dot{p_y} \\ \dot{p_z} \end{bmatrix} = \begin{bmatrix} \omega_y p_z - \omega_z p_y + \dot{t_x} \\ \omega_z p_x - \omega_x p_z + \dot{t_y} \\ \omega_x p_y - \omega_y p_x + \dot{t_z} \end{bmatrix} \ . \tag{4.6}$$

4.3.3 Image flow field

Assuming that there is only one rigid motion between the camera and the observed scene, the image flow field is the projection of the 3D velocity on the image plane [59]. Here, we establish the relation between the velocity of a point $\dot{\mathbf{p}}$ in the space and its corresponding velocity of $\dot{\mathbf{q}}$ on the image plane. Each projected velocity vector $\dot{\mathbf{q}}$ is called image flow vector.

The perspective projection of a point $\mathbf{p} = [p_x, p_y, p_z]^T$ in the space corresponds to the point $\mathbf{q}_H = [q_x, q_y, f]^T$ in the image plane given by

$$\mathbf{q}_H = f\frac{\mathbf{p}}{p_z} \ , \tag{4.7}$$

where f is the camera focal length, and $\mathbf{q}_H$ is the projection of $\mathbf{p}$ expressed in homogeneous coordinates, symbolized by the subscript "H".

The equation of the image flow vector of $\mathbf{q}_H$ is computed by differentiating Eq. (4.7) with respect to time, thus obtaining

$$\dot{\mathbf{q}}_H = \frac{f}{p_z^2}(p_z\dot{\mathbf{p}} - \dot{p_z}\mathbf{p}) \ . \tag{4.8}$$

The components of $\dot{\mathbf{q}}_H$ corresponds to

$$\begin{bmatrix} \dot{q_x} \\ \dot{q_y} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{f}{p_z^2}(p_z\dot{p_x} - \dot{p_z}p_x) \\ \frac{f}{p_z^2}(p_z\dot{p_y} - \dot{p_y}p_y) \\ 0 \end{bmatrix} \ . \tag{4.9}$$

By replacing $\dot{\mathbf{p}}$ with the results obtained in Eq. (4.6) it follows that

$$\begin{bmatrix} \dot{q_x} \\ \dot{q_y} \end{bmatrix} = \begin{bmatrix} \frac{f}{p_z}(\omega_y p_z - \omega_z p_y + \dot{t_x}) - \frac{f}{p_z^2}(\omega_x p_y - \omega_y p_x + \dot{t_z})p_x \\ \frac{f}{p_z}(\omega_z p_x - \omega_x p_z + \dot{t_y}) - \frac{f}{p_z^2}(\omega_x p_y - \omega_y p_x + \dot{t_z})p_y \end{bmatrix}. \tag{4.10}$$

Rearranging terms and, according to Eq. (4.7), substituting $(\frac{fp_x}{p_z}, \frac{fp_y}{p_z})$ by (q_x, q_y) respectively, we obtain the following expression of the image flow vector components

$$\dot{\mathbf{q}} = \frac{1}{p_z}\mathbf{A}\dot{\mathbf{t}} + \mathbf{B}\boldsymbol{\omega} \ , \tag{4.11}$$

where

$$\mathbf{A} = \begin{bmatrix} f & 0 & -q_x \\ 0 & f & -q_y \end{bmatrix} \text{ and}$$

$$\mathbf{B} = \begin{bmatrix} -\frac{q_x q_y}{f} & f + \frac{q_x^2}{f} & -q_y \\ -f - \frac{q_y^2}{f} & \frac{q_x q_y}{f} & q_x \end{bmatrix}$$

Notice that the image flow in Eq.(4.11) is the sum of two component vectors: a translational component that only depends on velocity $\dot{\mathbf{t}}$, and a rotational component that only depends on angular velocity $\boldsymbol{\omega}$. Analyzing the translational component, it is observed that its magnitude is inversely proportional to the relative depth between the camera and the scene. This means that the image flow of points with a big p_z value are negligibly affected by $\dot{\mathbf{t}}$, and provide mainly just information about the camera rotation. That is, if p_z tends to ∞ then the image flow corresponds to

$$\dot{\mathbf{q}} = \mathbf{B}\boldsymbol{\omega} \tag{4.12}$$

In the next section we take advantage of that to propose a novel approach to estimate the vehicle egomotion.

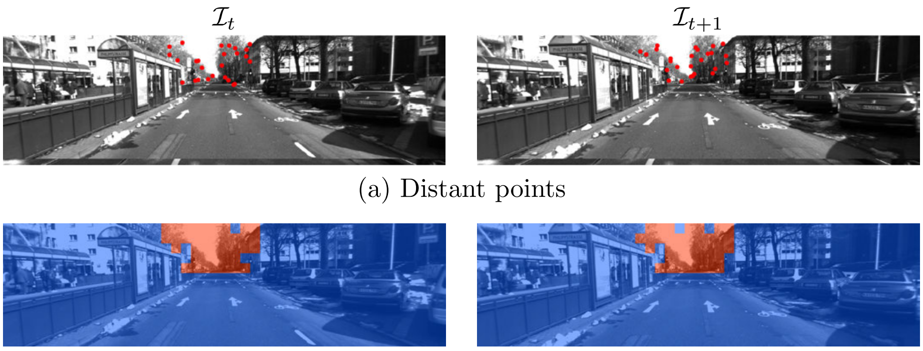
4.4 Proposed egomotion estimation approaches

Our camera motion estimation method is a two-stage approach: first, the camera rotation is computed, and then translation is estimated once rotation effect is canceled. To compute the camera rotational velocity, we propose two approaches which are based on the relation between motion and depth. The first one uses distant points (DP), and the second one is based on distant regions (DR).

Basically, given two consecutive frames $\mathcal{I}$ at instant t and $t+1$, our algorithms proceeds as follows:

1. Distant regions are identified in $\mathcal{I}_{t+1}$, and a template $\mathcal{T}$ is defined containing them.
2. Rotation $\mathbf{R}$ is computed by one of the following steps:
   (a) Feature-based approach (DP): Interesting points are detected in $\mathcal{I}_{\sqcup+\infty}$ and matched in $\mathcal{I}_t$. Those falling on $\mathcal{T}$ and their correspondences in $\mathcal{I}_{\sqcup}$ are selected as distant points. Then, $\mathbf{R}$ is estimated using the selected distant points.
   (b) Appearance-based approach (DR): The template $\mathcal{T}$ is aligned with respect to $\mathcal{I}_t$. Since the content of $\mathcal{T}$ can be considered a plane at infinity, its alignment determines the camera rotation between t and $t+1$.
3. The rotation effect between $\mathcal{I}_t$ and $\mathcal{I}_{t+1}$ is canceled, leading to just a translation $\mathbf{t}$ explaining the observed motion between both frames. Finally, translation $\mathbf{t}$ is computed using matching points between $\mathcal{I}_t$ and $\mathcal{I}_{t+1}$.

In the following sections, the main steps of the proposed algorithm are detailed.



(a) Distant points

(b) Distant regions

Figure 4.3: Distant points and distant regions identified in $\mathcal{I}_t$ and $\mathcal{I}_{t+1}$ from which our algorithms compute camera rotation parameters.

4.4.1 Distant regions segmentation

For the purposes of the application attained, it is not needed to perform a multiclass depth segmentation of images. Instead, we need just to identify the distant regions, and we can do that by applying a single properly trained classifier. Since available test sequences are in gray scale, the classifier used is based on location and texture features, obtaining good enough results for our purpose. Note that we use a regular grid of 15×15 pixels and discard the use of spatial coherence to devote the minimum of computational resources in this task.

4.4.2 Rotation estimation based on distant points

As described in Section 4.3.2, the relation between a 3D point $\mathbf{p}_i$ and its position in the next time instant is given by

$$\mathbf{p}'_i = \mathbf{R}\mathbf{p}_i + \mathbf{t} \ \ ,$$

where $i = 1 \ldots N$ and $N \geqslant 2$.

By reexpressing the point $\mathbf{p}_i$ as $\lambda_i \mathbf{n}_i$, where λ_i is the distance of $\mathbf{p}_i$ to the origin, and $\mathbf{n}_i$ its unit direction vector, previous expression can be formulated as

$$\lambda'_i \mathbf{n}'_i = \mathbf{R}\lambda_i \mathbf{n}_i + \mathbf{t} \ \ , \tag{4.13}$$

When $\lambda_i \to \infty$, and $|\mathbf{t}| \ll \lambda_i$, the effect produced by $\mathbf{t}$ is negligible, then Eq. (4.13) can be approximated as

$$\lambda'_i \mathbf{n}'_i = \mathbf{R}\lambda_i \mathbf{n}_i \ \ .$$

Note that the rotation $\mathbf{R}$ only affects the direction $\mathbf{n}_i$, and therefore λ'_i remains equal to λ_i. Thus, it is equivalent to

$$\mathbf{n}'_i = \mathbf{R}\mathbf{n}_i \ \ .$$

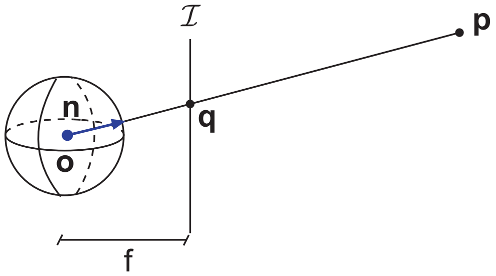


Figure 4.4: Projection of a point into a sphere of unit radio.

As we depict in Fig. 4.4, the unit vector $\mathbf{n}_i$ can be determined as the intersection between the projection line with the image plane, that is the 2D point $\mathbf{q}_{Hi} = \left[\frac{fp_{xi}}{p_{zi}}, \frac{fp_{yi}}{p_{zi}}, f\right]^T$. By normalizing $\mathbf{q}_{Hi}$, we obtain the unit vector direction of the projection line.

As has been shown in [107], this expression allows to estimate $\mathbf{R}$ from just two point correspondences between views. The two matched points determine the direction vectors $\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}'_1$, and $\mathbf{n}'_2$. Since the cross-product $\mathbf{n}_1 \times \mathbf{n}_2$ is related with $\mathbf{n}'_1 \times \mathbf{n}'_2$ by the same rotation, then $\mathbf{R}$ can be computed as

$$\mathbf{R} = [\mathbf{n}'_1, \mathbf{n}'_2, \mathbf{n}'_1 \times \mathbf{n}'_2]\,[\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_1 \times \mathbf{n}_2]^{-1} \quad .$$

The rotation estimated with this expression is very sensible to noise, since very few information is used. To elude that problem, we can take advantage of redundant information, using all available distant points to formulate an overdetermined system of equations. The only inconvenience is that the estimated $\mathbf{R}$ may not be a rotation matrix. To impose that constraint, we adapt the method proposed in [4] to our problem. This algorithm builds the following 3×3 matrix

$$\mathbf{H} = \sum_{i=1}^{N} \mathbf{n}_i \mathbf{n'}_i^T \quad . \tag{4.14}$$

The matrix $\mathbf{H}$ is decomposed by SVD as

$$\mathbf{H} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \quad .$$

Now, we calculate

$$\mathbf{R} = \mathbf{V}\mathbf{U}^T \quad , \tag{4.15}$$

which is an approximation matrix solution to a problem known as orthogonal Procustres problem [100] that minimize the following expression

$$min_{\mathbf{R}} \|\mathbf{n}'_i - \mathbf{R}\mathbf{n}_i\|^2 \quad \text{subject to} \quad \mathbf{R}^T\mathbf{R} = \mathbf{I} \quad .$$

One problem that may arise in using the above solution is that the obtained matrix $\mathbf{R}$ can have a determinant of -1. In other words, a reflection matrix of the desired rotation matrix is obtained. This can be easily solved by substituting Eq. (4.15) by the following expression

$$\mathbf{R} = \mathbf{V} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & det(\mathbf{V}\mathbf{U}^T) \end{bmatrix} \mathbf{U}^T \ .$$

To apply this approach, we select and match interest points by the method described in [65, 35]. This method provides a good distribution of feature points over the image, guaranteeing that the features not lie on independently moving objects, which yields an accurate egomotion estimation. The image coordinates of the interest poins at t and $t+1$ determine the vectors $\mathbf{n}_i$ in Eq. (4.14), from that the rotation is finally estimated.

4.4.3 Rotation estimation based on distant regions

Our direct method is based on the fact that image points belonging to scene objects which are placed at enough distant from the camera behave as lying on an infinity plane. The image projection of this plane remains in the same image coordinates when the camera just translates, and is only affected by camera rotation. Then, we can use the information provided by this plane to compute camera orientation robustly.

Taking advantage of that, we propose to track the image projection of distant regions in the scene to estimate the camera rotation. As a consequence of tracking this (infinity) plane over two consecutive frames, we obtain a pseudo-projective transformation relating both frames [11], which represents the planar image flow. This transformation is actually capturing the camera rotation.

From Eq. (4.12) we know that distant regions in consecutive frames are put in correspondence by means of the image flow field given by $\mathbf{B}\omega$. Given the template $\mathcal{T}$ of the distant regions in $\mathcal{I}_{t+1}$, we pose its matching to the distant regions of $\mathcal{I}_t$ as the problem of finding the value ω that minimizes the following objective function

$$\sum_{\mathbf{q}} \left(\mathcal{I}_t(\mathcal{W}(\mathbf{q}, \boldsymbol{\omega})) - \mathcal{T}(\mathbf{q})\right)^2$$

where $\mathcal{W}(\mathbf{q}, \boldsymbol{\omega}) = \mathbf{q} + \mathbf{B}\boldsymbol{\omega}$. To solve that, we apply a modified version of the Lucas-Kanade algorithm [80] based on the public available code described in [9]. Basically, a multi-resolution pyramidal scheme has been added, together with adaptations to consider templates of any arbitrary shape. When the algorithm converges we obtain the parameters ω aligning $\mathcal{T}$ to $\mathcal{I}_t$, and from ω we estimate $\mathbf{R}$ as

$$\mathbf{R} = \begin{bmatrix} 1 & -\omega_z & \omega_y \\ \omega_z & 1 & -\omega_x \\ -\omega_y & \omega_x & 1 \end{bmatrix} \ ,$$

assuming that the angular velocity in consecutive frames is small.

4.4.4 Translation estimation

Once rotation angles $\boldsymbol{\omega}$ are estimated, we can cancel the rotation effects on the images flow between consecutive images. Without rotation motion, the difference between both frames $\mathcal{I}_t$ and $\mathcal{I}_{t+1}$ is due to the camera translation.

We can compute translation direction by solving the following equation system from Eq. (4.11) by using matching points between $\mathcal{I}_t$ and $\mathcal{I}_{t+1}$

$$\dot{\mathbf{q}} = \frac{1}{p_z} \left[\begin{array}{ccc} f & 0 & -q_x \\ 0 & f & -q_y \end{array} \right] \dot{\mathbf{t}} \ .$$

Note that, when using only one camera $\dot{\mathbf{t}}$ can only be recovered up to an unknown scale factor $\frac{1}{p_z}$. Then, to solve the previous equation system, we arbitrarily set $\frac{1}{p_z}$ at 1, leading to translation direction as result. This scale ambiguity due to lack of depth information can be solved by using an additional sensor providing the vehicle speed.

4.5 Experimental results

In this section, we report the experiments done to quantify the performance of our methods described in Section 4.4 using real sequences.

We have used sequences from the Karlsruhe dataset[2] [34, 65], which provides videos taken from a stereo camera mounted on a vehicle in a driving environment. In total, there are more than 8000 frames captured in an urban scenario, driving along approximately 3 km. Translation vector and rotation angles ground truth (GT) are provided by measurements of an INS sensor. We have selected this dataset because, since sequences have been acquired with a stereo image, we can obtain from them the depth information needed to train our distant region classifier. Our algorithms are tested processing the left image of the sequences.

4.5.1 Evaluation of distant regions segmentation

Our classifier is trained with positive and negative examples taken from the computed disparity map of 700 randomly selected frames of different sequences. Then, this classifier is only applied to the image taken from the left camera of the stereo system. Figure 4.5 shows the results of applying our approach in some frames of the considered sequences, where a region is assumed as distant if it is composed by pixels whose stereo depth map value is over 70 meters (m). We have chosen this threshold value, since at this distance the effect of the vehicle translation on the optical flow is subpixel for the considered camera settings.

In Tab. 4.1 we depict the performance of our approach in terms of TP, TN, FP, FN rates plotted in a confusion matrix. Confusion matrix is computed at pixel level

[2] `http://www.cvlibs.net/datasets/karlsruhe_sequences.html`

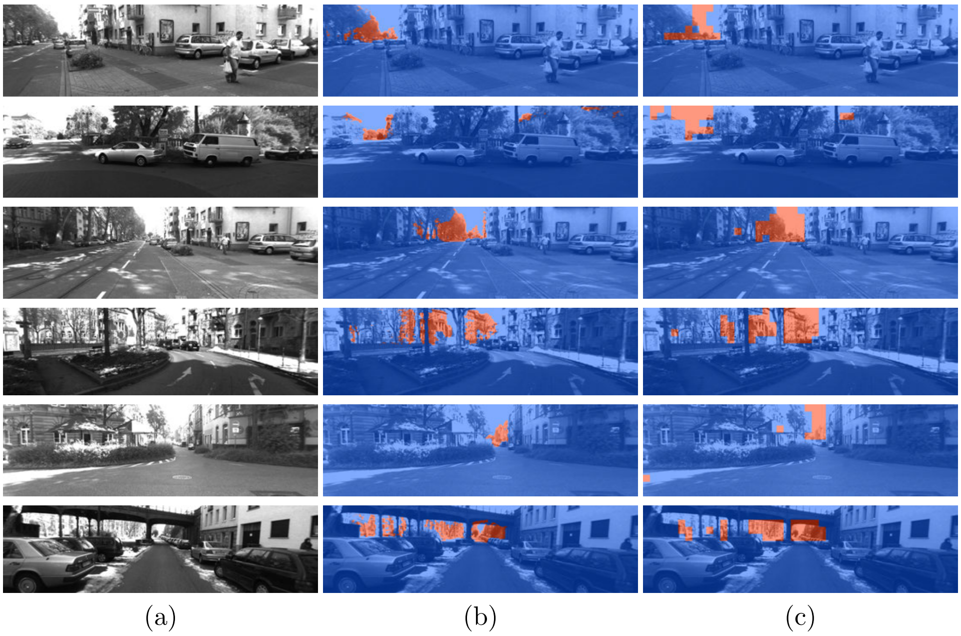


Figure 4.5: Depth segmentation results: (a) Original image, (b) Stereo depth map thresholded at 70 m (ground truth), (c) Results of the proposed method.

Table 4.1: Confusion matrix showing the performance of our segmentation approach.

		Predicted classes	
		Close	Distant
True classes	Close	**0.98**	0.02
	Distant	0.26	**0.74**

over all frames of Karlsruhe's sequences. We can observe that 74 % of pixels located at distant regions are correctly classified. This means that our classifier is good for distinguish between distant and close regions. Note that the number of FP is very low (2 % of close regions are incorrectly classified as distance), then distant regions used by our method have a low amount of noise.

The receiver operating characteristic curve (ROC) shown in Fig. 4.6 plots the true positive rate versus false positive rate for our classifier as its discrimination threshold is varied. We can observe that, admitting less than 20% of false positives, our classifier correctly classifies more than 90% of distant regions. Furthermore, if we do not accept false positives, more than 50% of the distant regions are correctly detected. Additionally, the segmentation performance of our segmentation is measured by using the area under the ROC curve (AUC). This measure reduces the ROC information into a single scalar value between 0 and 1, representing the classifier performance

[60]. For all the dataset, our classifier has an AUC of 0.83, which is a very good performance as our results shown.

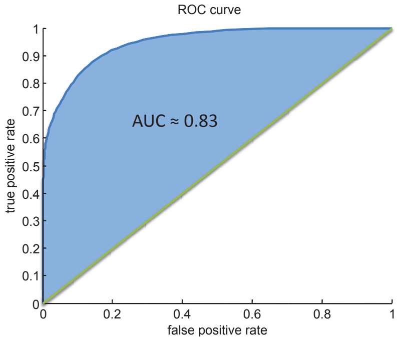


Figure 4.6: ROC curve showing the performance of our segmentation approach.

4.5.2 Evaluation methodology for egomotion methods

To evaluate how accurately our two proposals estimate the camera egomotion, we use two criteria. First, we compute the mean rotation error [109] to measure how well the camera rotation is recovered. Then, we evaluate the accuracy of the whole egomotion parameters estimated by computing the mean Euclidean distance between estimated and ground truth trajectories. We provide also plots of both trajectories for a qualitative analysis.

The rotation error is calculated as the difference angle between the true rotation $\boldsymbol{\omega}_f$ and the estimated rotation $\bar{\boldsymbol{\omega}}_f$, at frame $f = 1, \ldots, F$. For this purpose, rotation matrices $\mathbf{R}_f$ and $\bar{\mathbf{R}}_f$ for both $\boldsymbol{\omega}_f$ and $\bar{\boldsymbol{\omega}}_f$ are built. The product between $\mathbf{R}_f^T$ and $\bar{\mathbf{R}}_f$ is an identity matrix when both are equal. Thus, the difference between both matrices is defined as $\triangle\mathbf{R}_f = \mathbf{R}_f^T\bar{\mathbf{R}}_f$. In Euler terms, $\triangle\mathbf{R}_f$ can be characterized by a rotation axis defined by a unit vector, and a rotation angle. This angle is used as the mean rotation error. Since $trace(\mathbf{R}_f) = 1 + 2cos(\alpha_f)$, then the angle is equal to

$$\mu(\triangle\mathbf{R}_f) = cos^{-1}\left(\frac{1}{2}\ (trace(\triangle\mathbf{R}_f) - 1)\right) \ .$$

Then, we compute the mean of rotation error for the whole sequence as

$$MRE = \frac{1}{F}\sum_{f=1}^{F}\mu(\triangle\mathbf{R}_f) \ . \tag{4.16}$$

Trajectory errors are quantified by computing the Euclidean distance between both the ground truth and estimated trajectories for all algorithms and sequences as follows

$$e(\mathbf{c}_f, \mathbf{c}'_f) = \sqrt{(c_x - c'_x)^2 + (c_z - c'_z)^2} \ ,$$

where $\mathbf{c}_f = [c_x, c_y, c_z]^T$ are the 3D coordinates of the ground truth trajectory and $\mathbf{c'}_f = [c'_x, c'_y, c'_z]^T$ are the coordinates of the estimated trajectory at frame $f = 1, \ldots, F$. Notice that we exclude the Y coordinates in the computation of e. This is because ground truth data are very unreliable in this coordinate (recognized by their authors), and distort the comparison. Euclidean distance is measured frame-wise, and then averaged over all frames

$$MED = \frac{1}{F}\sum_{f=1}^{F}(e(\mathbf{c}_f, \mathbf{c'}_f)) \quad . \tag{4.17}$$

In the next sections, we show the results obtained by our approaches versus the following algorithms:

- The five-point algorithm (5pts) by Nister [84], which is considered a classical in visual odometry.
- The Burschka et al. method [16], which is a monocular method that tries to distinguish distant points by using RANSAC.
- The stereo-based algorithm by Kitt et al. [65] as a baseline to measure the performance of our monocular methods.

Table 4.2: Mean rotation error (in degrees) for each evaluated sequence.

Sequence	Algorithms				
	DR	DP	Burschka	5pts	Stereo
1	0.0725	0.1473	0.1077	0.0942	**0.0667**
2	**0.0579**	0.1007	0.1362	0.0744	0.0602
3	0.1222	0.1413	0.1647	0.1458	**0.1203**
4	0.0834	0.1237	0.1804	0.1021	**0.0757**
5	1.8305	1.9379	1.9358	1.8611	**1.8262**
6	0.2284	0.2353	0.3647	0.2119	**0.2143**
7	0.7623	0.7809	0.8412	0.7892	**0.7267**
8	0.0539	0.0645	0.0963	0.0607	**0.0551**

4.5.3 Rotation error

Table 4.2 shows the mean rotation error for all sequences. First, we can notice that our approaches overcome Burschka et al. algorithm. This is due to Burschka tends to select mid-regions, while our approaches are effectively based on distant points/regions. DR is more accurate than 5pts algorithm because distant regions provide very reliable information about rotation. We also note that in many cases the estimations of DR are very close to those performed by Stereo algorithm. Our results are supported

Table 4.3: Mean Euclidean distance error (in meters) for each evaluated sequences.

Sequence	Traveled distance	Algorithms DR	DP	Burschka	5pts	Stereo
1	456 m	4.21 m	4.38 m	7.32 m	10.70 m	4.18 m
2	631 m	15.83 m	25.96 m	28.29 m	68.00 m	9.21 m
3	386 m	4.15 m	4.31 m	7.67 m	9.91 m	4.10 m
4	361 m	4.38 m	10.42 m	14.44 m	4.51 m	2.61 m
5	287 m	4.34 m	4.51 m	11.18 m	10.44 m	4.04 m
6	103 m	0.65 m	0.74 m	0.95 m	0.84 m	0.67 m
7	515 m	4.31 m	4.50 m	8.87 m	8.42 m	4.44 m
8	76 m	1.10 m	1.15 m	2.11 m	1.64 m	0.72 m
Average		4.87 m	7.02 m	9.80 m	14.40 m	3.77 m

by some psychophysical studies that have been demonstrated that depth help to rotation estimation, and they are most reliable when based on distant points/regions [116, 115].

Figure 4.7 depicts a comparison between the yaw angle estimated by considered algorithms, and ground truth for the first 100 frames of Sequence 1. We can see that our results are very close to ground truth data and they look very similar. However, notice that the DR is smoother than DP estimates because we do a maximal use of the information available in frames to estimate the camera rotation, which lead to estimated parameters reflecting the behavior of the real camera motion in the sequence, characterized by slow and smooth movements.

A comparison between the yaw angle and the ground truth for Sequence 7 is depicted in Fig. 4.8. By exploiting depth information along the sequences, we can see that both methods give accurate rotation estimations, which is close to GT data. However, DP makes more mistakes in sudden orientation changes, as shown in the details of Fig. 4.8. But both behave similarly in approximately straight paths.

4.5.4 Visual odometry results

For this experiment, we compute the motion between two consecutive frames and then incrementally accumulate each estimation along the sequence to build the vehicle trajectory. Note that we do not refine the current estimate by bundle adjustment techniques, what would improve the performance achieved.

Since our methods are monocular ones, translation is estimated only up to a scale factor. To elude that problem, the absolute scale between consecutive poses is obtained from the ground truth vector norm, and then we apply it to the estimated translations at each frame. With that we emulate the reading of this information from the speedomenter of the car.

Table 4.3 shows the results for each sequence. The mean Euclidean distance error of DR and DP for all sequences correspond to approximately 1.38% and 1.98% of

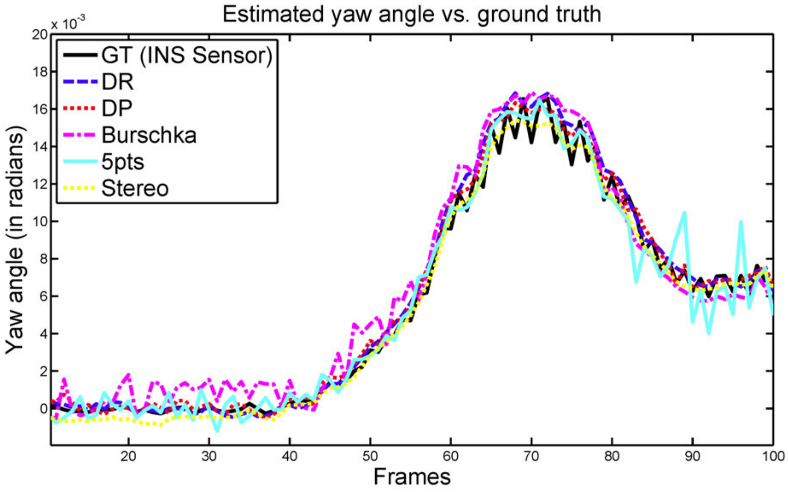


Figure 4.7: Comparison between the yaw angle computed with DR, DP, Burschka, 5pts and Stereo-based algorithms for the first 100 frames of sequence 1.

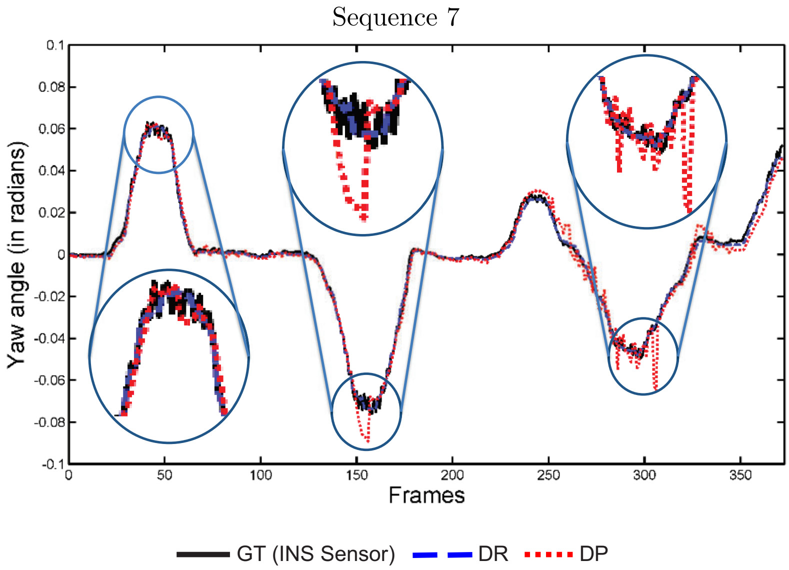


Figure 4.8: Comparison between the yaw angle and ground truth. In this case, the yaw angle is computed with DR and DP methods for sequence 7. We can observe that the results obtained by DR are less affected by sudden orientation changes, as shown in the details, while both behave similarly in approximately straight paths.

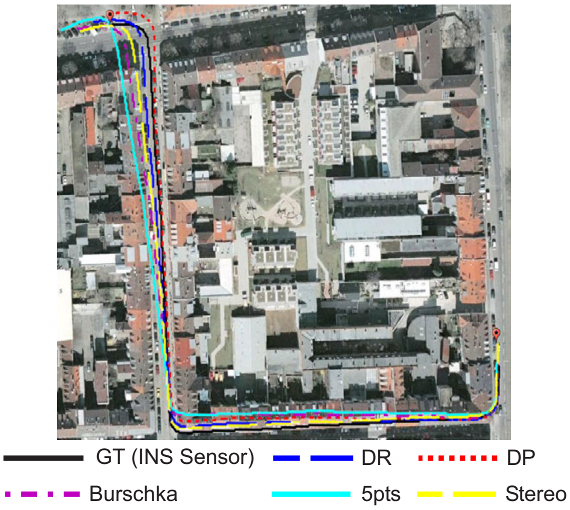


Figure 4.9: Comparison between visual odometry trajectories for sequence 1 computed with DR, DP, Burschka, 5pts and Stereo algorithms.

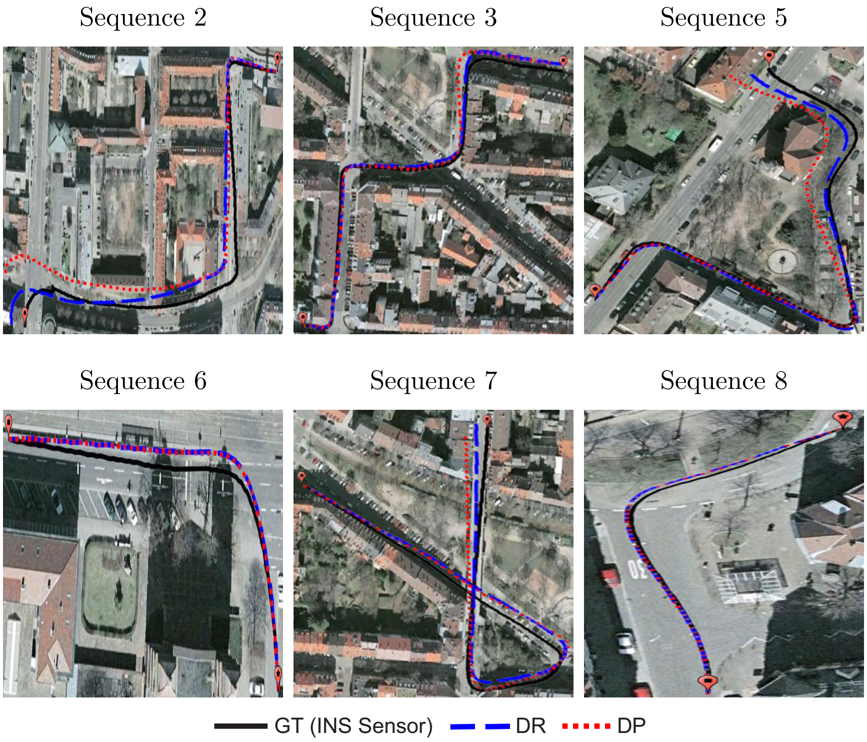


Figure 4.10: Comparison between visual odometry trajectories computed with DR and DP algorithms.

the traveled distance. Our error is lower than the obtained by the other monocular methods, and it is closer to both stereo' performance and the trajectories given by the INS sensor. Figure 4.9 depicts a comparison among the visual odometry paths computed for sequence 1 with DR, DP, Burschka, 5pts and Stereo algorithms. Our proposals remain close to the ground truth along the sequence in a comparable performance to Stereo algorithm, while Burschka and 5pts are more biased. This is because our methods accurately capture the dominant components of the rotation motion by selecting distant points/regions, which affect the trajectory estimation. Unlike our methods that effectively select distant points/regions, Burschka et al. algorithm, in general, selects points which are located at mid-distances, leading to less accurate rotation estimation. The 5pts algorithm use all available points in the image, without distinction between close and distant ones. Additionally, both rotation and translation are recovered by decomposing the essential matrix [52], which is a step where small amounts of noise significantly affect the final result.

Figure 4.10 shows the comparison between different ground truth trajectories having strong orientation changes and the estimations performed by DR and DP. We can see that both methods are very precise. However, the accuracy of DR outperforms DP because DR takes advantage of distant regions having valuable information for motion estimation, which is unexploited by our feature-based method. Most part of distant regions belong to low textured zones, so few interest points are detected and consequently matched.

4.5.5 Robustness to noisy segmentation

In this section, we evaluate the performance of our algorithms under noisy distant regions segmentation. In Tab. 4.4, we compare the egomotion results obtained in the previous sections using our segmentation approach with respect to those computed using stereo depth maps to segment the image in close/far regions. In this case, we can observe that, although our segmentation algorithm is not perfect, the egomotion results obtained are accurate, and very close to those obtained from an "ideal" segmentation from stereo. The mean difference error between using a "perfect" segmentation from stereo or our estimated depth map is 0.0243 and 0.01724 for DR and DP, respectively. Under a " perfect" segmentation of distant regions, DP increases its performance, but it does not overcome DR. Therefore, DP is most sensitive to a noisy segmentation of distant regions.

In another experiment, we randomly add and remove different amounts of near and distant regions for simulating misclassification error to stereo depth map, and use that to apply our egomotion methods. We introduce FP by using close regions randomly chosen as if they were distant ones. Additionally, we remove distant regions to produce an amount of FN. Table 4.5 shows the percentage of generated error and its relation with FN and FP. Figure 4.11 depicts an example of such nosy segmentations under different amounts of noise.

Table 4.6 shows the egomotion results under noisy segmentations. Notice that the performance of our algorithms does not degrade significantly due to classification

Table 4.4: The results of our egomotion approach in terms of mean rotation error (in degrees) is shown. We compare the egomotion accuracy of DR and DP using distant regions from our segmentation approach versus those obtained by thresholding the stereo depth maps.

Sequences	Algorithm	Depth segmentation approach	
		Our approach	Stereo depth map thresholding
Seq. 1	DR	0.0725	0.0521
	DP	0.1473	0.1164
Seq. 2	DR	0.0579	0.0502
	DP	0.1007	0.0820
Seq. 3	DR	0.1222	0.0979
	DP	0.1413	0.1175
Seq. 4	DR	0.0834	0.0644
	DP	0.1237	0.1033
Seq. 5	DR	1.8305	1.8142
	DP	1.9379	1.9234
Seq. 6	DR	0.2284	0.2028
	DP	0.2353	0.2303
Seq. 7	DR	0.7623	0.6839
	DP	0.7809	0.7603
Seq. 8	DR	0.0539	0.0507
	DP	0.0645	0.0605

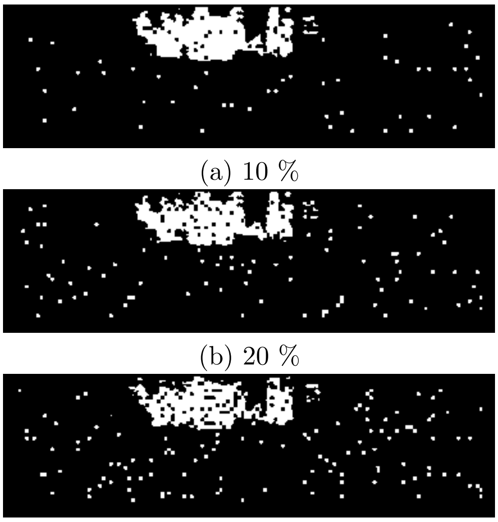
Table 4.5: Misclassification errors introduced in stereo depth maps to evaluate the robustness under noisy segmentations of our egomotion algorithms.

% Error	FP	FN
10 %	1 %	10 %
20 %	2 %	20 %
30 %	3 %	30 %

errors, even under a large number of outliers. For noisy segmentation of 10%, 20% and 30% compared to the case without noise, the mean of difference error is 0.0412, 0.0674 and 0.1066 for DR and 0.0465, 0.0759 and 0.1199 for DP, respectively.

4.6 Conclusions

In this chapter, we have shown that by exploiting depth information obtained from a single camera it is possible to obtain accurate camera motion estimations. For such purpose, we have proposed a feature-based and an appearance-based algorithm, relying on the fact that at distant scene regions, image flow is dominated by mainly rotational motion component. Distant regions are selected by applying a segmentation



(a) 10 %

(b) 20 %

(c) 30 %

Figure 4.11: Example of noisy segmentations used in our experiments to show the robustness of our egomotion algorithms.

approach which allow us to distinguish between close/distant regions in a image. The first algorithm uses corresponding distant points (DP) to estimate the camera rotation by solving an overdetermined set of equations. The second one tracks distant regions between two frames (DR). These regions can be assumed as located at the plane at infinity, inducing an infinity homography relation between two consecutive frames. By tracking that plane, we are able to estimate the camera rotation. Once rotation is computed, we cancel its effect on the images, leaving the resulting motion due to camera translation.

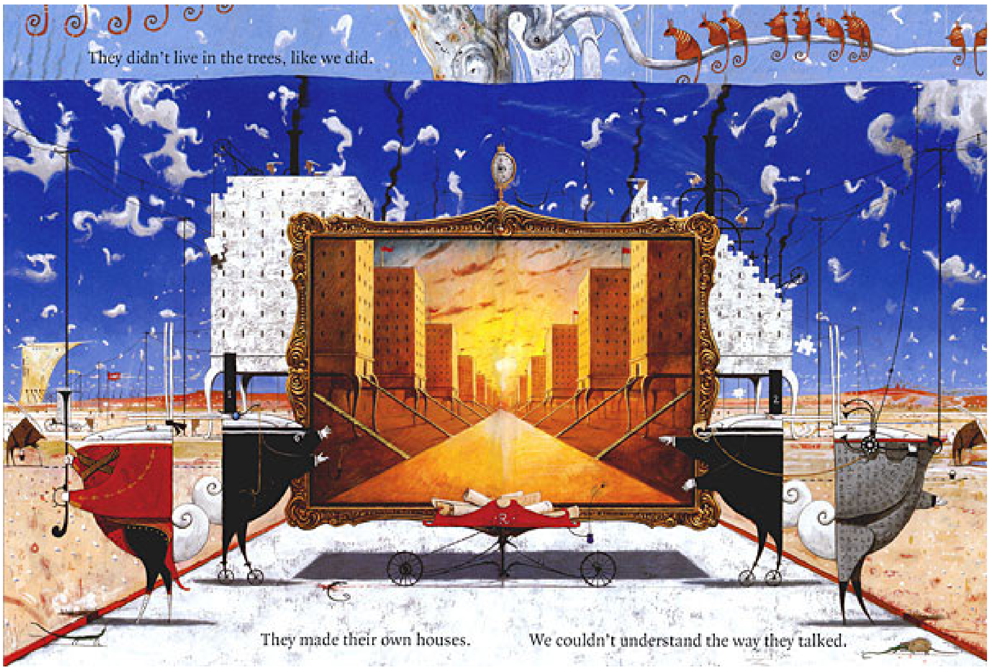
Real experiments in different sequences have shown that rotations are accurately estimated, since distant points/regions provide strong indicators of camera rotation. Our algorithms outperforms other monocular state-of-the-art methods, and have a comparable performance with respect to the stereo algorithm detailed in [65].

Despite our two algorithms are very precise, the accuracy of DR outperforms DP because DR exploits valuable information for motion estimation, which is available on these distant regions. DP requires distant points that commonly are located at the sky or other less textured image regions. These kind of points are hard to be detected and tracked in successive frames. Additionally, DR avoids the feature extraction and matching, which is valuable since small errors in the estimated image flow usually bring to large perturbations in the motion estimation.

Moreover, from several experiments performed, we can conclude that our methods are also robust to segmentation errors, leading to accurate results even under a significant number of outliers.

Table 4.6: Mean rotation error (in degrees) is shown for our approach computed using stereo depth maps under different amount of outliers.

Sequences	Algorithm	Outliers			
		0%	10%	20%	30%
Seq. 1	DR	0.0521	0.0570	0.0613	0.0667
	DP	0.1164	0.1292	0.1372	0.1493
Seq. 2	DR	0.0502	0.0557	0.0592	0.0644
	DP	0.082	0.0910	0.0967	0.1052
Seq. 3	DR	0.0979	0.1086	0.1154	0.1256
	DP	0.1175	0.1304	0.1385	0.1507
Seq. 4	DR	0.0644	0.0715	0.0759	0.0826
	DP	0.1033	0.1147	0.1218	0.1326
Seq. 5	DR	1.8142	2.0131	2.1387	2.3271
	DP	1.9234	2.1343	2.2674	2.4671
Seq. 6	DR	0.2028	0.2250	0.2391	0.2601
	DP	0.2303	0.2555	0.2715	0.2954
Seq. 7	DR	0.6839	0.7589	0.8062	0.8772
	DP	0.7603	0.8436	0.8963	0.9752
Seq. 8	DR	0.0507	0.0563	0.0598	0.0650
	DP	0.0605	0.0671	0.0713	0.0776


They didn't live in the trees, like we did.
They made their own houses.
We couldn't understand the way they talked.

The building of the houses, Shaun Tan, 1998.

The Rabbits is a picture book, written by John Marsden and illustrated by Shaun Tan, which is an allegorical fable about Australian colonization, told from the view-point of the colonized. It describes the coming of *rabbits* in an amazing detail, an encounter that is at first friendly and curious, but later darkens as it becomes clear that the visitors are in reality invaders.

In the image that we are showing, the underling idea of our background estimation approach is depicted. Our goal is to reconstruct the background of a scene without disturbing objects, composing an image like the one carried by the *rabbits*.

Chapter 5

Background estimation exploiting monocular depth information

> *Landscape is not merely the world we see, it is a construction, a composition of that world.***Social Formation and Symbolic Landscape, D. E. Cosgrove, 1984.**

In this chapter, we address the problem of reconstructing the background of a scene from a video sequence with occluding objects. Background reconstruction is useful for many computer vision task like tracking, surveillance, motion analysis, etc. Our method composes the background by selecting the appropriate pixels from previously aligned input images. To do that, we minimize a cost function that penalizes the deviations from the following assumptions: background represents objects whose distance to the camera is maximal, and background objects are stationary. Distance information is roughly obtained by our supervised learning approach that allows us to distinguish between close and distant image regions. Moving foreground objects are filtered out by using stationariness and motion boundary constancy measurements. The cost function is minimized by a graph cuts method. We demonstrate the applicability of our approach to recover an occlusion-free background in a set of sequences.

5.1 Introduction

During the last decade, the number of cameras in our environment has increased dramatically. This growth has been experienced in all areas, including traditionally ones such as video surveillance, video and photography for professionals and enthusiasts, systems for driving assistance, but also in newest ones like in smart-phones, and video gaming. This growing presence of cameras has been mainly motivated by reductions in cost and improvement in the quality of digital cameras. Additionally, the widespread use of computers has provided user-friendly ways to process images. Even applications for domestic use allow to any user manipulating an image to enhance it in many forms. Image editing software includes basic tools like adjusting colors and cropping images, but also more complex ones like removing disturbing elements and merging images to compose collages or panoramas.

In this chapter, we focus on how to apply our monocular depth estimation approach to automatically remove transient and moving objects from a set of images or a sequence with the aim of obtaining a background image of the scene. Besides the obvious uses of this for image enhancement (e.g., removing objects that spoil a beautiful landscape photograph, or creating images without cluttered foreground objects), it has many other applications in computer vision and graphics fields. For example, background estimation is usually the first step in background subtraction algorithms [90], where moving objects are detected by subtracting the observed image from an estimated reference background image. Segmentation of moving objects provides useful information from video processing applications such as image stitching, background substitution, compression, and tracking [121].

In this work, we rely on the fact that background is inherently behind foreground, as demonstrate the figure-ground perception. As a consequence of this, we define a background model under two main assumptions: First, the background represents objects whose distance to the camera is maximal; and second, background objects are stationary. On the one hand, this definition implies the knowledge of depth information that has been traditionally associated to stereoscopic vision. However, human beings easily identify which objects are in the foreground as well as those belonging to the background. This is because people can infer depth information, which is used by the visual system to understand their surroundings [43], as we discuss in Chapters 2 and 3. On the other hand, moving objects are considered as foreground objects since they tends to occlude the background.

For the purposes of background estimation, we propose a novel approach that combines both kind of information. Depth information is obtained by applying the distant region classifier described in Chapter 3, while stationary regions are detected by considering the absence of large variations in the object's boundaries and color pixels discrepancies between frames. Our approach selects the appropriate pixels to compose the background from a set of images by minimizing a cost function that penalizes deviations from our background model. This method requires a set of aligned images, and to do this accurately, we propose an image registration process that also takes advantage of our distant region segmentation method.

The chapter is organized as follows. In Section 5.2, we introduce the main important related works. Our proposal is described in Section 5.3. Section 5.4 shows the experimental results, which are analyzed throughout the discussion carried out in Sec. 5.5.

5.2 Related work

Background estimation from a set of images has been widely studied in many areas of computer vision. In general, most of techniques to background estimation are based on processing images taken from a static monocular camera. These techniques discard the use of depth information since, traditionally, it has been considered as relying exclusively in stereoscopic information (e.g., Kalman filtering [93], mixtures of Gaussians [104], optic flow [50], among others). In [53], the authors exploit depth information recovered from stereo cameras to remove the background. However, the use of stereo cameras is often an unusual configuration in most systems.

Our approach is related to different proposals that pose the background estimation as an energy minimization problem. These methods construct a background from a set of images whose pixels have an associated cost assigned using some penalization criteria such that an energy functional is minimized to obtain a composite image of low-cost pixels.

Here, we review some of them, and state the novelty of our proposal. In [2], the authors define a cost function that includes a likelihood term penalizing the color values of pixel with low probability. Per-pixel probability distributions are estimated using histograms with fixed intervals. This method is integrated in a framework that allows assisted interactive retouch.

In [19], background estimation is formulated as a labeling problem, where a cost function penalizing low color stationariness and motion boundary inconsistencies is minimized by graph cuts. Low color stationariness occurs at pixels with large color variance in a small time interval. Motion boundary inconsistencies occurs at pixels where the image starts to differ from others.

In [47], the authors propose a method for a set of images taken from the same viewpoint with no restrictions on the time interval between them (i.e., non-time sequences). In essence, they redefine the motion boundary penalty from [19] to a term that does not require temporal coherence. Additionally, they also include the object likelihood term from [2] but in a non-parametric form.

Recently, in [18], the background is composed by minimizing a function that penalize image areas where the pixels are not stationaries and also a new predicted term obtained using an image inpainting technique.

Motivated by the previous works, we also consider the background estimation as a labeling problem. We use a similar cost function as in [19], applying graph cuts to minimize it. However, to the best of our knowledge, any previous work has taken into account that depth information can be extracted from single images. Then, we propose to use depth information to penalize deviations from our background model. This depth information is extracted by applying a classifier to identify distant image regions, described in Chapter 3. If images are taken from a moving camera, our method, in a similar way that all previously reviewed methods, requires an initial image alignment before applying the proposed solution. This is relevant to a precise penalties computation and, consequently, a good pixel selection. We solve this

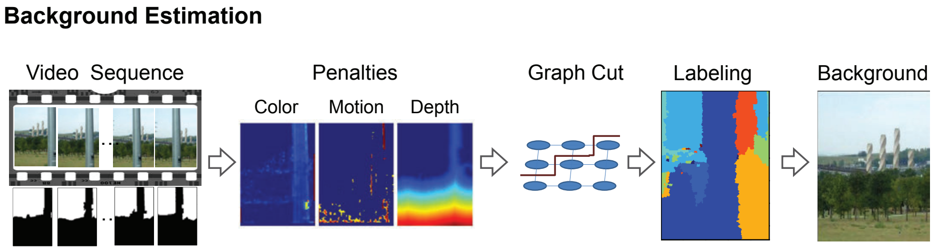


Figure 5.1: Overview of our background estimation approach. Given a sequence of aligned images, we compute color, motion, and depth penalties for each pixel to compose an energy function. This function is minimized by using graph cuts. The minimization result is a labeling where each pixel correspond to an input image in which the background is visible. Finally, the scene background is reconstructed by copying pixels from the appropriate input image.

problem by aligning the input images relying also on our distant region segmentation. Distant regions are used for alignment due to they provide a reliable information about camera motion, leading the images aligned with respect to the background.

5.3 Proposed background estimation approach

To formulate the background extraction problem, we first assume that the input of our method is a sequence of aligned images of a scene. If that were not the case, the image should be aligned by following the procedure described in Sec. 5.3.1. Our objective is to estimate the background by finding, for each pixel, an input image in which the background is visible. Then, the scene background reconstruction combines the pixels from the appropriate input image. Each pixel has assigned a labeling corresponding to a frame number, and each possible labeling has an associated cost. The goal is obtaining a labeling that minimizes that cost. Figure 5.1 illustrates this approach.

Formally, let $\mathcal{I} = \{I_1, \ldots . I_N\}$ be a set of N input images. $\mathcal{P}$ denotes the set of pixels in an image. $I_n(p)$ denotes the color value at pixel position $p \in \mathcal{P}$ for n-th image I_n. Let $\mathcal{L} = \{1, \ldots, N\}$ be a set of labels, each one corresponding to an image in $\mathcal{I}$. A labeling is a mapping $f : \mathcal{P} \to \mathcal{L}$, that means that a pixel $p \in \mathcal{P}$ has assigned the label $f_p \in \mathcal{L}$. Each labeling f generates an image $I_f : p \to I_{f_p}(p)$. Then, the background estimation problem is posed as finding a labeling f^* to construct the background $I_B = I_{f^*}$ such that f^* is a minimum cost labeling. In Sec. 5.3.2, we formalize the cost function to be minimized.

5.3.1 Image registration

The described cost minimization process can be applied as long as the set of images have been acquired from an static camera. If the camera is not static, first images have to be registered. To do that, we align each two consecutive images by using Lucas-Kanade algorithm. To perform such alignment between the current image I_{f_p} and next image I_{f_p+1}, we use as template T the distant regions of I_{f_p+1}. Distant regions are used to align images since they behave as an infinity plane providing accurate information about the camera motion, leading the images aligned with respect to the background. This plays a significant role during penalties computation because a precise alignment reduces the probability of selecting wrong pixels values to compose the background. Lucas-Kanade algorithm iteratively minimizes the difference between T and I_{f_p} under the following goal objective

$$\sum_q \left(I_{f_p}(W(q,\mathbf{a})) - T(q)\right)^2 \quad , \tag{5.1}$$

with respect to $\mathbf{a} = \{a_i\}_{i=1\ldots6}$, where $W(q,\mathbf{a})$ is an affine warp

$$W(q,\mathbf{a}) = \begin{bmatrix} (1+a_1)q_x & a_3q_y & a_5 \\ a_2q_x & (1+a_4)q_y & a_6 \end{bmatrix} \quad .$$

5.3.2 Energy function

The energy function $E(f)$ of a labeling f is defined as [15]

$$E(f) = \sum_{p\in\mathcal{P}} D_p(f_p) + \sum_{p,q\in\mathcal{N}} V_{p,q}(f_p, f_q) \quad , \tag{5.2}$$

where D is the data term, and V is the smoothness term. The data term defines the cost of assigning the label f_p to pixel p. The smoothness term is the cost of assigning labels f_p and f_q to neighboring pixels p and q, such that $p, q \in \mathcal{N}$, being $\mathcal{N}$ the set of adjacent pixels in $\mathcal{P}$.

A labeling that minimize the energy E is found by using the α-expansion algorithm proposed in [26]. For details about the optimization algorithm, please refer to [15]. Briefly, the algorithm performs an iteration for every label $\alpha \in \mathcal{L}$. Given the current labeling f, in each iteration, the algorithm finds a labeling $\hat{f}$ that minimizes E over all labelings within one α-expansion of f. An α-expansion step consists in generating a new labeling f' by replacing some pixels of f with α label. This step is performing by graph cuts. The previously step is repeated in a cycle which is successful if a strictly better labeling is found at any iteration. The algorithm stops after the first unsuccessful cycle, since no further improvement is possible.

Smoothness term penalizes the intensity differences between neighboring regions [71], giving a higher cost when f_p and f_q do not match well

$$V_{p,q}(f_p, f_q)=\frac{(\|I_{f_p}(p)-I_{f_q}(p)\|+\|I_{f_p}(q)-I_{f_q}(q)\|)}{2} \tag{5.3}$$

The data term penalizes the labelings that do not hold the background model. Our data term accounts the color stationariness D^S, motion boundary consistency

D^M, and proximity/distantness information D^P

$$D_p(f_p) = \alpha D_p^S(f_p) + \beta D_p^M(f_p) + \gamma D_p^P \ . \tag{5.4}$$

Since, the three components have different units, we first normalize each one between 0 and 1, and then we introduce different weights for each component to balance the contribution of each one. The first two terms in D were introduced by [19], and the last term corresponds to our approach. In the next sections, we detail each term.

Stationariness

This term penalizes image regions whose color varies significantly along time. The stationariness cost $D_p^S(f_p)$ assigns a high cost to pixels with large color variance in a small time interval [19]. Formally,

$$D_p^S(f_p) = min\{Var_{f_p-1,f_p}(p), Var_{f_p,f_p+1}(p)\} \ , \tag{5.5}$$

where $Var_{i,j}(p)$ is the mean of the variance of each color channel from image I_i to I_j at pixel p, and $f_p \pm r \in \mathcal{L}$ denotes the r-th image posterior or previous to the current one, respectively.

Motion boundary consistency

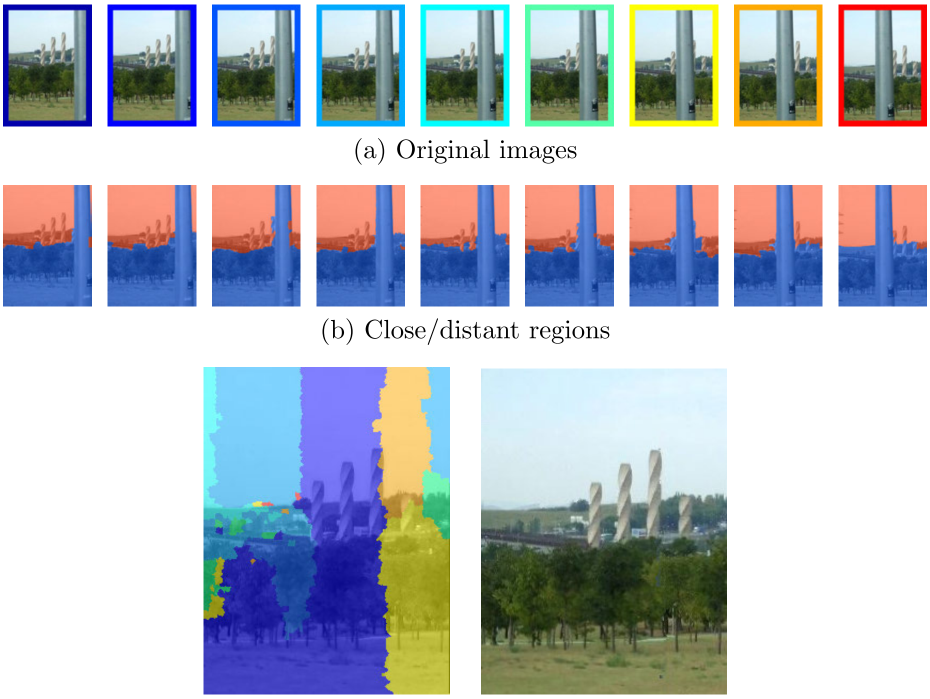
We use motion boundaries to penalize pixels corresponding to moving objects. Motion boundaries occur in adjacent image regions having different image velocities due to motion parallax or independent moving objects [14]. Based on that, the motion boundaries can be approximated as the gradient of the difference between an image and the background, which is justified since the boundary of a moving object occurs at locations where the images start to differ. Assuming that I_{f_p} is the background image and I_i is an input image containing moving objects, the difference image $F_{f_p,i} = \parallel I_{f_p} - I_i \parallel$ has a large gradient magnitude $\parallel \triangledown F_{f_p,i} \parallel$ when I_{f_p} and I_i are poorly matched. Likewise, $\parallel \triangledown I_i \parallel$ has large values at intensity edges. Motion boundary consistency penalizes motion boundaries that do not occur at background's intensity edges [19]

$$D_p^M(f_p) = \frac{1}{N} \sum_{i \in \mathcal{L}} \frac{\parallel \triangledown F_{f_p,i}(p) \parallel^2}{\parallel \triangledown I_i(p) \parallel^2 + \epsilon} \ , \tag{5.6}$$

where ϵ is a small value to avoid zero-division.

Proximity/distantness information

This term penalizes those image regions which are close in the scene, since we assume that the background is composed by distant regions. This implies that we require at least rough information about scene depths. For computing such segmentation, we



(a) Original images

(b) Close/distant regions

(c) Labeling

(d) Our result

Figure 5.2: (a) Nine (of eleven) images of the *Towers* sequence, (b) Close/distant regions estimation for such frames (red corresponds to distant regions, blue to close regions), (c) Computed labeling where each label corresponds to a region of an image, and (d) Estimated background using our method.

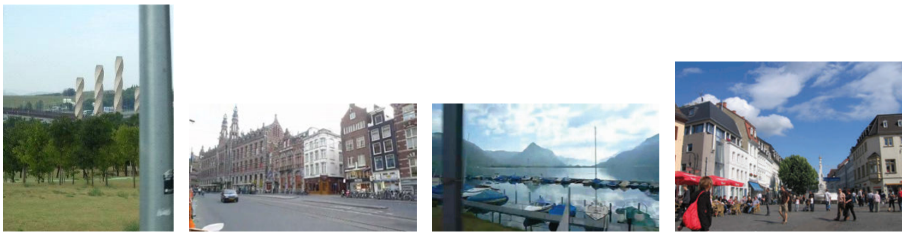
take as base our approach to distinguish far regions. In this case, we have established the threshold to distinguish what is a distant region at 30 m since, for the camera used, beyond that distance the moving objects in the scene just show a scarce motion in the image, and most of them can be considered as part of the background.

We assign a cost to each pixel belonging to a close region R_c, which is the Euclidean distance between that pixel and the nearest pixel belonging to a distant region R_d

$$D_p^P = \begin{cases} 0 & \text{if } p \in R_d \\ min\{d(p,q) \mid q \in R_d\} & \text{if } p \in R_c \end{cases} , \tag{5.7}$$

where R_c is the set of pixels belonging to close regions, R_d is the set of pixels in distant regions, and $d(p,q)$ is the Euclidean distance between two pixels coordinates.

Basically, we are stating that a close region has a higher associated cost when it is further away from any distant region. We also penalize those regions that being distant in the previous frame become closer in the current frame, because they probably belong to moving objects approaching to the camera.



(a) Towers #frames: 11 (b) City #frames: 7 (c) Train #frames: 3 (d) Market #frames: 8

Figure 5.3: Sequences used to evaluate our method.

Figure 5.2 illustrates an example result of our approach. Figure 5.7(a) shows some frames used to compose the background of *Tower* scene. This kind of sequence is frequent in videos recorded from moving vehicles like cars or trains, where an object disturbs the background. The disturbing object is classified as close during the sequence (see Fig. 5.7(b)). Then, the background is composed by a set of pixels with low-cost from different images as we can observe in Fig. 5.7(c) for the computed labeling. Our method removes those objects that snail the scene as we depict in Fig. 5.7(d).

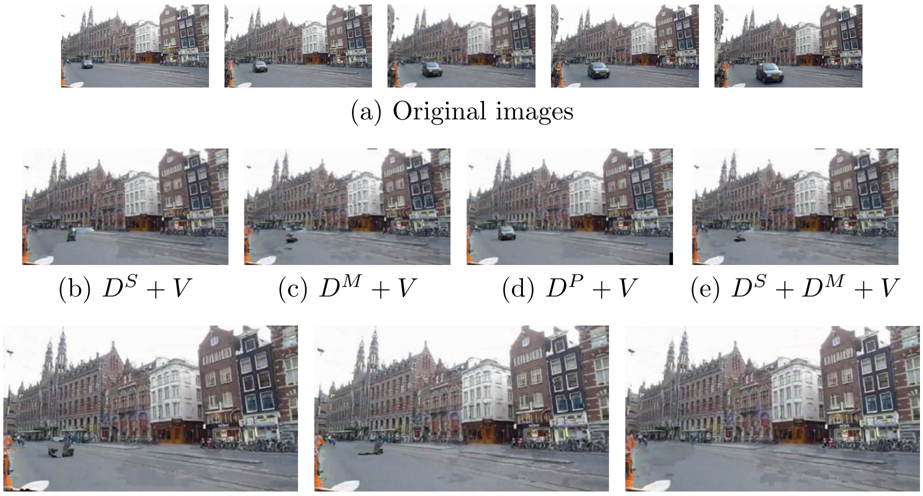
5.4 Experimental results

For evaluating our method we build a test set of four sequences depicted in Fig. 5.3. *Towers* sequence is taken by us using a hand held consumer camera, requiring alignment between frames. *City* and *Train* are also taken using consumer camera, but both sequences were extracted from Youtube. These sequences have a low-quality due to the compression applied. However, our method shows good results in obtaining background even under this quality. *Market* sequence is used in [47]. It is taken fixing the camera with a tripod, without temporal coherence between frames.

The parameters values to control the effect of each term in the data term are experimentally defined as $\alpha = .3$, $\beta = .4$, and $\gamma = .3$, which gave us good results.

5.4.1 Analysis of energy terms effect

The effect of each term in the energy function is depicted in Fig. 5.4. First, terms are considered in isolation (see Fig. 5.4(b)-(d)). All of them contribute to reduce the transient objects. As Fig. 5.4(d) shows, the proximity/distantness term in isolation keeps the car which is located further away from the camera. This occurs since we are not considering color or motion changes. Thus, the far-away car has a low-cost due to it has a high probability of belonging to the background. Then, a progressive improvement of the background estimation is obtained by combinations between the



(a) Original images

(b) $D^S + V$ (c) $D^M + V$ (d) $D^P + V$ (e) $D^S + D^M + V$

(f) $D^S + D^P + V$ (g) $D^M + D^P + V$ (h) $D^S + D^M + D^P + V$

Figure 5.4: (a) *City* sequence, (b)-(h) Interaction between terms. Including our term on the cost function allow us to reach a better background estimation results. This implies that all terms are complementary.

data term components (see Fig. 5.4(e)-(g)). Finally, using all terms a significantly improvement is reached (see Fig. 5.4(h)). This implies that the different terms in E are complementary. Note that by combining the depth information our method overcome the results of [19], which are the ones in Fig. 5.4(e).

5.4.2 Comparison with respect to state-of-the-art

We compare our proposal against the popular median filtering algorithm [79] and the approach of [2], which is in the state-of-the-art of background estimation. Figures 5.5-5.9 show the results of applying our approach to different sequences.

In Fig. 5.5, we show the result of our method in a scene with an independent moving object (i.e., the car approaching to the camera). Fig. 5.5(b) depicts how the car is penalized since it is moving, and how the penalization increases as the car become closer. Note that distant regions have a low-cost due to our depth-based term. Our method effectively removes the car while median filter method keeps some ghost of it, as Fig. 5.5(c) shows. Fig. 5.5(d) shows the result of Agarwala et al. In many cases, as in Fig. 5.6, this method requires a user interaction to remove some artifacts that are still present in the obtained result. After that step, Agarwala et al. method reaches a comparable performance with respect to our method. In the rest of experiments, such manual processing is performed when it is necessary to ensure a comparable result. Another example of applying our method to compose the background of *Tower* scene is depicted in Fig. 5.7. Unlike median filter and Agarwala et al. methods, our method effectively removes the object that snail the scene as we

Table 5.1: Norm of absolute difference RGB obtained between our results against the manually refined results of Agarwala et al.

Algorithms	Sequences			
	Towers	City	Train	Market
Our	0.0551	0.0804	0.0479	0.0603
Median filter	0.0942	0.1023	0.0483	0.0715

show in Fig. 5.7(b).

Figure 5.8 shows an experiment done to evaluate our method under low-quality images. This kind of videos are not intentionally captured for extracting the background, however it can be obtained without transient objects. Even under low-quality videos our proposal correctly estimates the background.

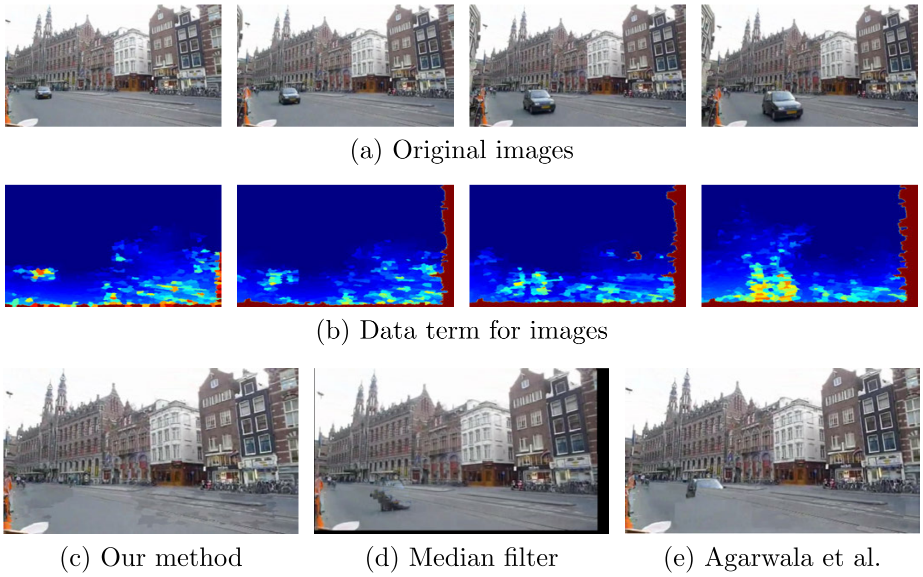
Figure 5.9 shows the performance our method in a scene without temporal coherence between frames. However, our approach behaves reasonably well under this setting. Although some ghosts are present in shadows, our results are comparable with respect to Agarwala et al. The remaining artifacts can be removed by using a gradient-domain fusion as in [2, 47].

From a quantitative viewpoint, we compute the norm of absolute difference in RGB channels between our results against the manually refined results of Agarwala et al. to measure how much the results of both methods differs. Table 5.1 shows the obtained values for each sequence. According to that, we can see that both methods are close one to another. However, our approach is fully automatic while the method of Agarwala et al. requires user interactions for refinement. For instance, when the estimated background is still containing foreground objects, the user must selects these regions which will be replaced by new ones offered by the system. In some cases, this interactive step must be repeatedly performed to achieve an acceptable result. Moreover, Agarwala et al. apply additional steps as, for instance, gradient-domain fusion to remove image artifacts. By contrast, our method is remarkably simpler and straightforward.

5.5 Conclusions

In this chapter, we have presented a method to background estimation containing moving/transient objects, which take advantage of monocular depth information for such purpose. Our segmentation method is used to found the background by penalizing close regions in a cost function, which integrates color, motion, and depth terms. The cost function is minimized by using a graph cuts approach.

We have tested our approach with sequences taken under different conditions (e.g., moving/static camera, temporal/non-temporal coherence, low/high-quality). Experimental results shown that our method significantly outperforms the median filter approach. Also, our approach is comparable to state-of-the-art methods that require



(a) Original images

(b) Data term for images

(c) Our method

(d) Median filter

(e) Agarwala et al.

Figure 5.5: (a) Images of the *City* sequence, (b) Data term for each image (red corresponds to high-cost values, blue to low-cost values). Estimations using: (c) Our method, (d) Median filter, and (e) Agarwala et al., 2004.

user intervention. Unlike Agarwala et al., we perform this task automatically.

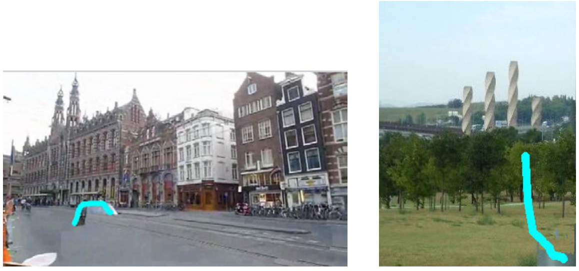
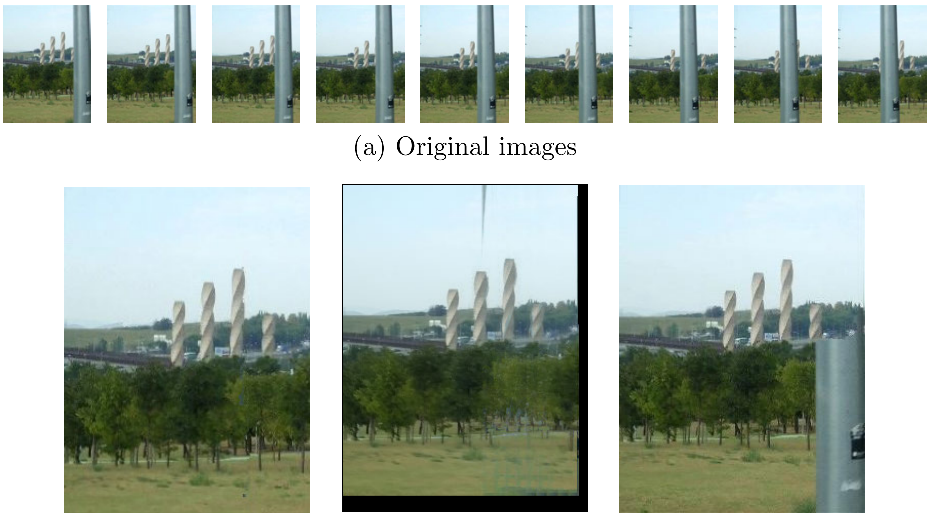


Figure 5.6: Examples of interactive step required by Agarwala et al. method to remove some artifacts for *City* and *Towers* sequences.



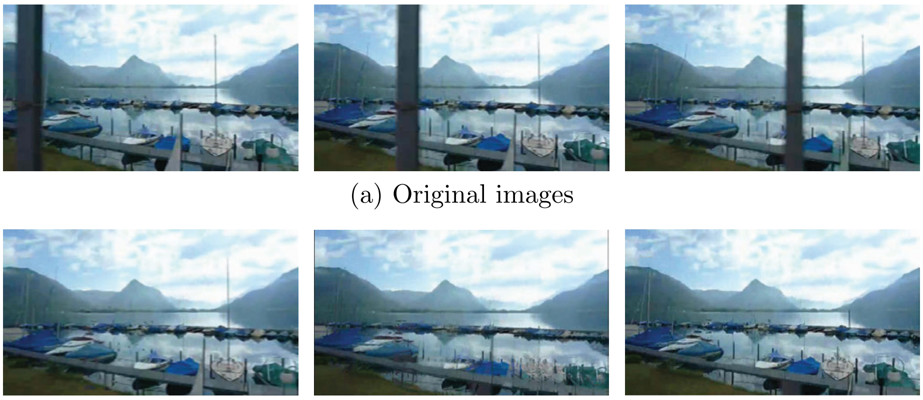
(a) Original images

(b) Our method

(c) Median filter

(d) Agarwala et al.

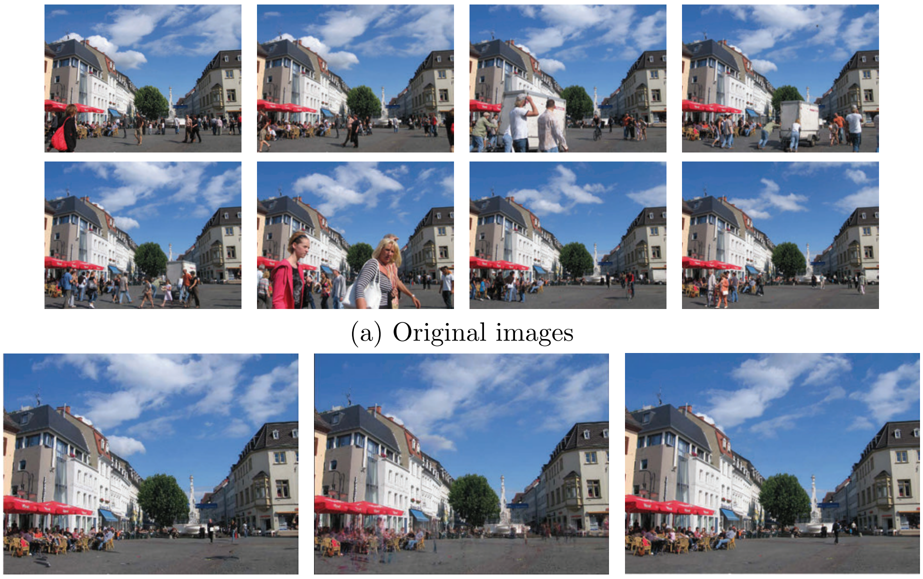
Figure 5.7: (a) Nine (of eleven) images of the *Towers* sequence. Estimations using: (b) Our method, (c) Median filter, and (d) Agarwala et al., 2004.



(a) Original images

(b) Our method (c) Median filter (d) Agarwala et al.

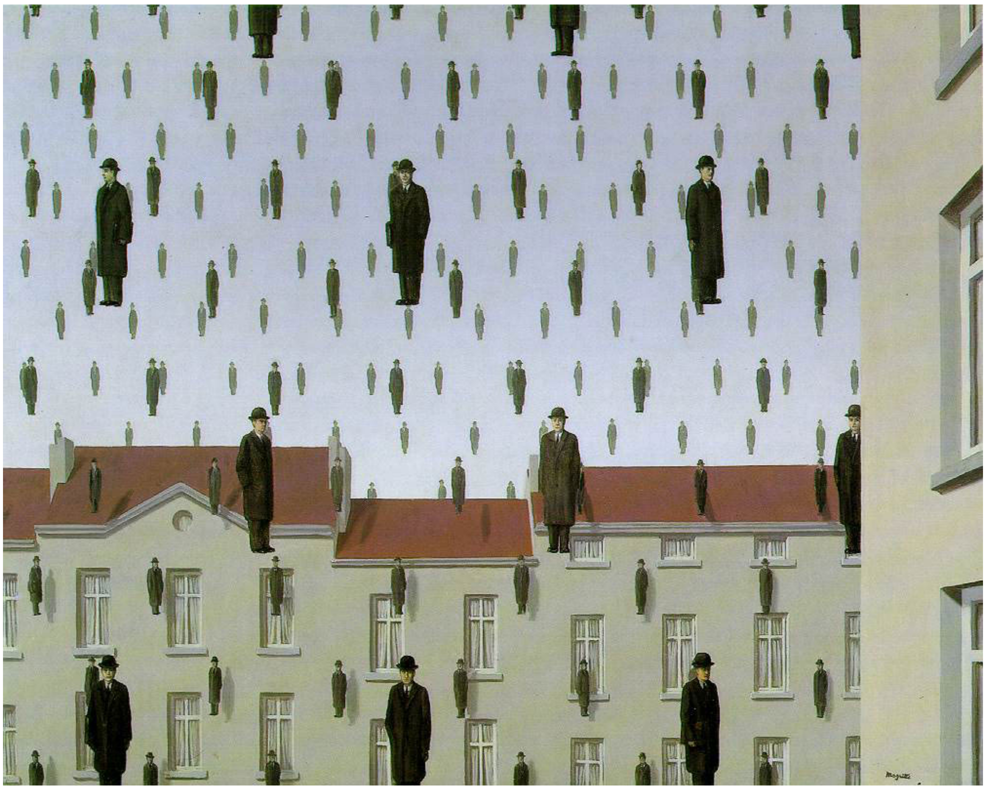
Figure 5.8: (a) The *Train* sequence. Estimations using: (b) Our method, (c) Median filter, and (d) Agarwala et al., 2004.



(a) Original images

(b) Our method (c) Median filter (d) Agarwala et al.

Figure 5.9: (a) Images of the *Market* scene. Estimations using: (b) Our method, (c) Median filter, and (d) Agarwala et al., 2004.



Golconda, René Magritte, 1953.

René Magritte (1898-1967) was a Belgian surrealist painter known for his witty and provocative images. His art is based on absurd, paradoxical, and unusual images, trying to change preconceived perceptions of reality and forcing the viewer to notice the reality of his/her environment through their works.

Golconda is featured by bowler hatted men in raincoats floating weightlessly in a blue sky in front of houses. Men are placed over the whole scene in a symmetrical distribution and challenging any physical law. In this painting, we see two parallelisms with respect to the current chapter. On the one hand, men are placed in a hexagonal grid with scale progression, which resembles the behavior of a common pedestrian candidate generation method. On the other hand, in opposition to the rules that govern the earth: a real person can not appear in all places on a scene. Typical pedestrian detection systems classify interesting image regions to decide if they contains a pedestrian or not. Under this idea, we want to leverage information from our structured world to generate windows, with a high likelihood of containing a pedestrian.

Chapter 6

Pedestrian candidates generation based on coarse depth maps

What we see is a stable world. **Art and Illusion, E. H. Gombrich, 1960.**

Most pedestrian detection systems are based on applying a previously trained classifiers in a candidate window established on an image region of interest. Common techniques for candidate generation (e.g., sliding window approaches) are based on an exhaustive search over the image. This implies that the number of windows produced is huge, which translates into a significant time consumption in the classification stage. In this chapter, we propose a method that significantly reduces the number of windows to be considered by a classifier to detect a pedestrian. Our method is a monocular one that exploits geometric and depth information available on single images. Both representations of the world are fused together to generate pedestrian candidates based on an underlying model which is focused only on objects standing vertically on the ground plane and having certain height, according with their depths on the scene. We evaluate our algorithm on a challenging dataset and demonstrate its utility for pedestrian detection. Our algorithm generates a reduced amount of candidate windows to be evaluated by a pedestrian classifier, keeping a high probability of detecting the real pedestrian in the image.

6.1 Introduction

The main objective of Advanced Driver Assistance Systems (ADAS) is increasing driver safety and comfort. ADAS systems require a full understanding of the scenarios where the vehicle is evolving, including detection of moving and stationary objects that determine the free space available for driving. In that case, the vehicle's surroundings is perceived and monitored by sensors to avoid unsafe situations (e.g., collisions). Although systems employing active sensors (e.g., radar, lidar, etc.) have shown promising results in object detection, they have several drawbacks, such as high cost, high consumption, and interference caused by sensors of the same type installed in different vehicles. However, passive sensors based on visual information (like cameras) receive a rich representation of the environment, that can be used to identify objects on the scene, as well as to detect lanes and recognize traffic signs. Indeed, due to the low cost of camera sensors, vision-based systems will be present as standard equipment on mid/low-priced vehicles providing information to ADAS applications.

A fundamental stage in scene understanding is the recognition of objects which are present in the scene (e.g., pedestrian [38], vehicles [89], signals [110]). To warn the driver in time of potential dangers, this step must be performed efficiently. Analyzing the whole image to locate potential objects locations is not feasible due to this constraint. What many object detection proposals do is follow two steps: First, hypothesize about object locations in an image, and then test this hypothesis to verify the presence of the object. Often these steps are executed multiple times on an image to recognize different objects by using independent methods.

In the context of pedestrian detection, the number of hypothesis to be evaluated can be drastically reduced by assuming that interesting objects are approximately vertical and their height is into a limited range. A reduced number of scanning windows has two advantages for an object detection module: on the one hand, speeding up the detection by discarding large image portions that not provide relevant information; on the other hand, reducing the false positive detections by focusing on specific regions with high probability of having the presence of pedestrians.

In this chapter, we propose a novel method to generate a set of candidate hypothesis for pedestrian detection based on just analyzing a medium-level representation of the scene. Based on different cues and context information which are available in single images, it is possible to obtain important clues for pedestrian detection as scene layout, object dependencies, surfaces, and occlusions. Our method fuses two complementary mid-level scene representations to select the image region where applying an object recognition algorithm has sense.

Basically, we use geometric information obtained from a single image that allow us to distinguish between three main classes of surfaces: horizontal, vertical and those that belong to sky regions in the image [58]. From these information, we are able to know what regions are potentially supporting surfaces (i.e., the ground), and what are vertical objects. In the ADAS context, interesting objects are vertical and located over the road plane; so we have a first clue where selectively searching them.

Another useful information is the distance of objects in the scene, since it constrains the pedestrian detector's scale to be used. To take advantage of that, we propose to use the multiclass depth segmentation approach presented in Chapter 3, which is useful to determine an approximated distance of objects receding into depth. Although depth information extracted from our approach is rough, it is useful. Both kind of information are combined to select regions of interest (ROI) containing possible stationary or moving vertical objects located in front of the vehicle. Figure 6.1

depicts our approach to pedestrian candidates generation.

The chapter is organized as follows. In the next section, we introduce some related work. Next, we detail our method to monocularly compute the medium-level information used in our proposal. Then, we describe how we fuse both representations to generate candidate windows. Finally, we measure the performance of our approach for pedestrian candidates generation and conclude.

6.2 Related work

Many approaches concerning our purpose have been proposed. Here, we refer some of them. A detailed description of the following strategies can be found in [37].

The simplest candidate generation method for pedestrian detection is the sliding window approach [25] which is an exhaustive scan over the input image with windows of different scales at all the possible positions. The drawback of this approach is that requires generating a big number of candidates to reach an acceptable performance. A big number of candidates implies a higher computation time during classification, which is undesirable. Additionally, many of these candidates are false positives since this method does not use any prior knowledge.

Other technique is based on the so-called flat world assumption [12, 33]. In this case, pedestrians are assumed to be on a planar road. This is a strong constraint which implies that the road geometry and its position with respect to the camera is known and remains constant along time. Under these conditions, the algorithm generates candidates over the presupposed road plane with pedestrian-sized windows.

However, due to road imperfections, car accelerations, and changes in the road slope, the camera pose changes, and the image is scanned sub-optimally. Then, road geometry and camera pose cannot be assumed as constant. The limitations of flat world assumption-based method can be overcome by adjusting the scanning grid to a road surface estimated dynamically, as it proposes in [37]. The algorithm estimates the road surface based on 3D points provided by a stereo camera, and then performs a road scanning in the same way that the previous method.

Using stereo cameras, in [5], a compact description of the world for autonomous vehicles, called “stixel world”, is introduced. It is offering a strong simplification of the data, but preserving the information of interest. The underlying model focuses only on objects standing vertically on the ground plane and having certain height. The main purpose of the achieved simplification is to reduce the amount of data that scene understanding algorithms must manage. To build a stixel world model of a scene, existing approaches are based on (dense or sparse) depth maps from a stereo rig [88].

Unlike previous pedestrian candidate approaches, our work take into account contextual information to reduce the number of candidates, assuming that our world has certain regularities with respect to how pedestrian are commonly located in an image. With respect to [5], our work differs in the fact that we use single images, which are

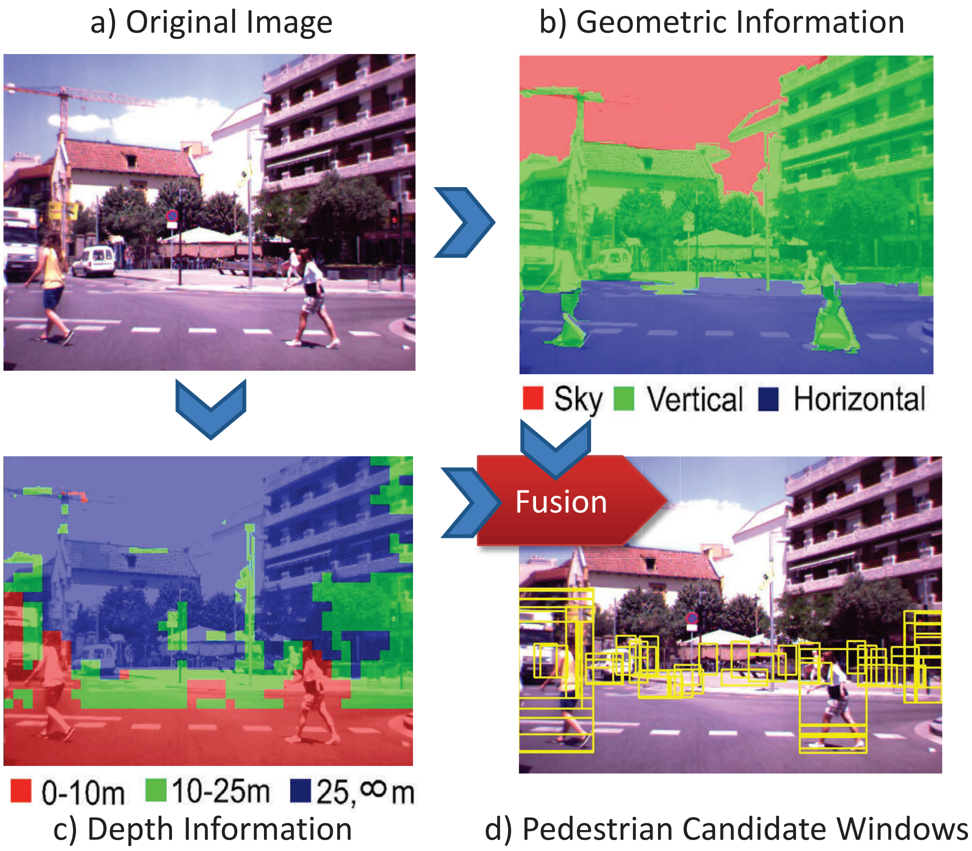


Figure 6.1: Our method is based on exploiting information extracted from an image. First, pixels corresponding to vertical objects and a rough depth map are obtained. Then, we generate candidate windows from this information.

very informative of both the overall scene structure and the object distances, as many recently works have demonstrated [58, 96].

6.3 Medium-level representations

In this section, we describe our method to build a compact representation of a scene. Figure 6.1 shows a schema of our approach. Briefly, the first step is extracting useful information from the image regarding geometric and depth clues.

Geometric information about the scene is recovered by using the approach proposed by Hoiem et al. [58]. This method segments the image into three geometric classes that depend on the orientation of the surfaces in the scene. Each region in the image is classified as horizontal, vertical or sky. Horizontal surfaces are approx-

Table 6.1: Pedestrian Sizes

Distance (m)	Minimum Size (pixels)	Maximum Size (pixels)
0 - 10	70 × 140	120 × 240
10 - 25	30 × 60	70 × 140
25 - 50	12 × 24	30 × 60

imately parallel to ground plane and objects can be supported by them (e.g., road surface). Vertical surfaces are roughly perpendicular to ground plane (e.g., buildings, pedestrian, cars, trees, etc.). The sky is usually located on top regions of the image, corresponding to the air and clouds. Basically, an image is over-segmented into superpixels [92], each of which belongs to a particular geometric class. Each superpixel is described by depth cues, including color, location, perspective, and texture. Then, from a logistic regression form of AdaBoost previously trained, the geometric class of each superpixel is inferred. An example of the result of this process is shown in Fig. 6.1(b).

To complement this information, we classify the image pixels into three depth regions, following the proposal described in Chapter 3. An example of the result of our approach is shown in Fig. 6.1(c). In this case, each depth range is properly selected taking into account the object's scale variability, whose image projection onto image plane is affected due to perspective effects. We define the following distance ranges: 0-10 m, 10-25 m, and more than 25 m [37]. The first two ranges are high-risk areas in case of vehicle collision against an object. The last range are a low-risk areas, where pedestrians are less vulnerable to suffer the consequences of an accident. Table 6.1 shows the minimum and maximum size of a pedestrian at certain distance from the camera for the considered configuration. We use these pedestrian sizes as size constraints in our candidate windows generation process.

6.4 Candidate window generation approach

From the previous intermediate results, we hypothesize about which vertical objects at different depths are interesting in the ADAS context. Basically, we start by dividing an image into superpixels, which is an attempt to divide the image such that boundaries coincide with image edges, grouping similar pixels into regions. Then, we combine geometric and depth information by an agglomerative hierarchical clustering [102] over the computed superpixels until the bounding box enclosing a set of superpixels has a coherent size with respect to the object size to be detected.

Our hierarchical clustering is based on the following set of physical/spatial assumptions:

- Gestalt constraints: We take into account two grouping principles of Gestalt school. On the one hand, the principle of good continuation which states that regions which are connected have smooth boundaries. On the other hand, the principle of similarity which states that the elements in a region are similar, including similar color, brightness, and texture [43]. To fulfill the Gestalt principles, the image is over-segmented into superpixels by using Turbopixels approach [75].

- Gravity constraint: Elements in the driving environment should stand on the ground plane.
- Depth constraint: All superpixels belonging to an object are located at the same depth region, and must be grouped together.
- Size constraint: In our context, the size of an interesting object is constrained to certain range according with its depth, taking into account the camera calibration properties (i.e., focal length, and image size).

Inspired by [114], we use an agglomerative clustering method on the Euclidean distances between the coordinates of the superpixels centroids. The algorithm is composed of the following steps:

1. Start with two sets of superpixels: $\mathcal{G}$ containing vertical superpixels whose distance to the ground plane is below a threshold, and $\mathcal{V}$ containing the rest of vertical superpixels.
2. Find the nearest pair of superpixels, that is the pair (g, v), where $g \in \mathcal{G}$ and $v \in \mathcal{V}$, whose Euclidean distance $d(g, v)$ between their centroids is minimal.
3. Combine g and v to form the superpixel $g = g \cup v$ if the following conditions hold:
   (a) Both g and v are located at the same depth range, and
   (b) The size of the merged superpixel $g = g \cup v$ is within the considered pedestrian size, according with its depth range.
4. If the size of g is within the minimum and maximum sizes of a pedestrian, generate a new candidate window for g. Remove v from $\mathcal{V}$.
5. While g fulfills the size condition, repeat from step 2. Otherwise, select a new $g \in \mathcal{G}$ and start again from step 2, until $\mathcal{V}$ is empty.

An example of how the clustering algorithm works is shown in Fig. 6.2. Fig. 6.2(a) shows the information sources used during clustering, as we described above. In this case, we are devoted to pedestrian candidates generation, considering pedestrian's sizes for merging. In Fig. 6.2(b), we can observe how the superpixels are fused together into a single one as the algorithm progresses. Superpixels in blue are gradually merged until reaching a size limited to their depth. In Fig. 6.2(c) each region is enclosed into a bounding box to conform a candidate window.

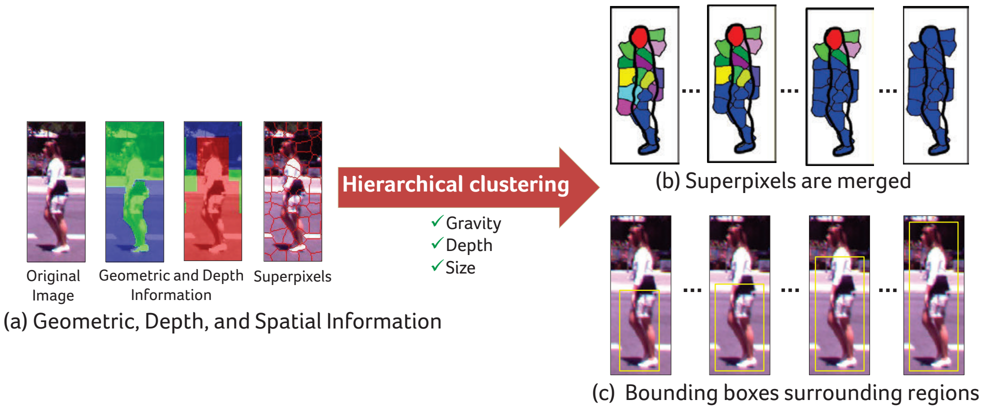

Original Image
Geometric and Depth Information
Superpixels
(a) Geometric, Depth, and Spatial Information
Hierarchical clustering
✓Gravity
✓Depth
✓Size
(b) Superpixels are merged
(c) Bounding boxes surrounding regions

Figure 6.2: Hierarchical Clustering. (a) Information sources used in our clustering algorithm, (b) An example of how superpixels are merged as the clustering algorithm progresses, and (c) Bounding boxes surrounding regions.

6.5 Experimental results

In this section, we evaluate the performance of our algorithm for candidate windows generation with respect to state-of-the-art methods, using a public available dataset.

The dataset consists of 15 sequences taken from a stereo-rig rigidly mounted in a car while it is driving on an urban scenario[1]. Each image has an associated depth map computed from stereo images. In total, there are 4364 frames, which correspond to 7983 manual annotated pedestrians visible at less than 50 meters.

We train a multiclass classifier following our approach described in Chapter 3. During the training phase of our depth-based segmentation method, we use a training set consisting of 700 images randomly taken from different sequences. The corresponding stereo depth maps are used to label a set of positive/negative examples for each distance range.

To judge the benefits of our approach, we compare how well the generated candidates are related to ground truth pedestrian annotations. Based on the evaluation protocol proposed in [37], we measure the performance of our approach in terms of the following criteria:

1. Minimizing the amount of pedestrian candidates generated $\mathrm{PC} = \mathrm{TP} + \mathrm{FP}$.
2. Maximizing the True Positive Rate $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$.

Each candidate window c is compared against the ground truth annotation a using

[1] `http://www.cvc.uab.es/adas/index.php?section=other_datasets`

the area of overlap between both bounding boxes by the formula

$$\text{overlap}(c, a) = \frac{\text{area}(a \cap c)}{\text{area}(a \cup c)} \ , \tag{6.1}$$

A candidate is classified as TP, FP or FN using the overlapping measure proposed by Everingham et al. [29] for object detection evaluation in the PASCAL Challenge,

$$\text{classify}(c, a) = \begin{cases} \text{TP} & \text{if overlap}(c, a) > \Gamma \\ \text{FP} & \text{if overlap}(c, a) \leq \Gamma \\ \text{FN} & \text{if } a \text{ does not have any associated} \\ & \text{candidate } c \ . \end{cases} \tag{6.2}$$

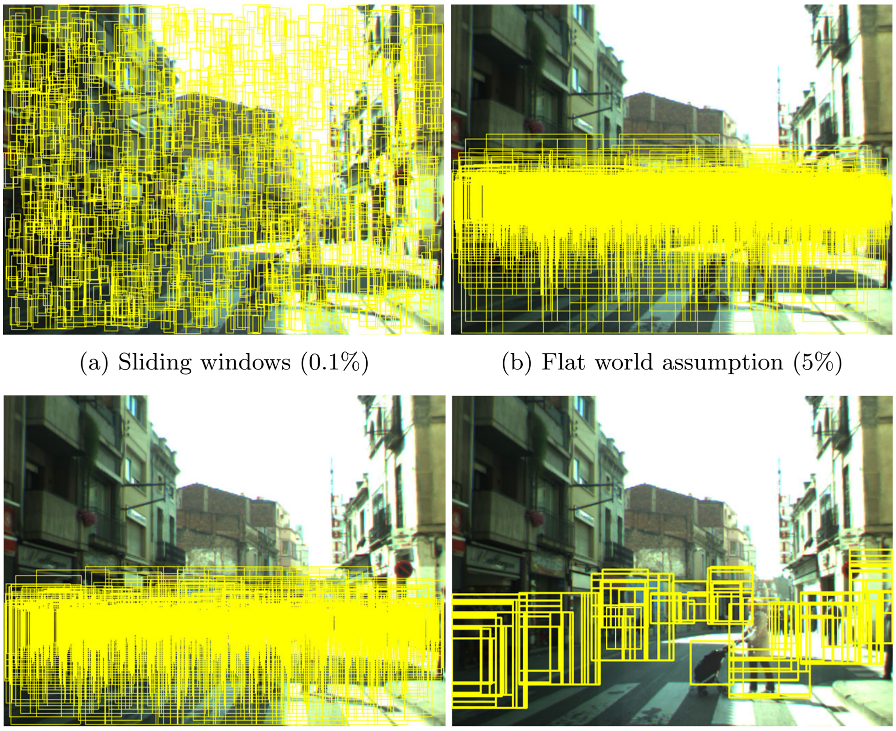
In our case, for a candidate c to be a TP, we require that this overlap exceeds a threshold $\Gamma = 50\%$.

6.5.1 Comparison with other strategies

In this section we compare our candidate generation proposal with respect to sliding windows, flat world assumption and adaptive road scanning approaches. Regarding sliding windows, we use three configurations with different amount of candidates: perfect, dense and sparse. The difference between these configurations lies in the parameters chosen to achieve a trade-off between TPR and PC.

Figure 6.3 depicts a qualitative comparison between our proposal and the methods described above. We can observe that our approach selects a reduced number of candidate windows with respect to the rest of considered methods.

Figure 6.4 shows the results in terms of TPR and PC per frame of each algorithm. Although sliding window has the best performance with respect to TPR (100% and 98%), the number of candidates to classify is huge, which affects the time consumption in a posterior classification stage. Note that the configuration of both perfect and dense have a high computational cost to be used in practice. Assuming a flat world the search space is significantly reduced, but the TPR is low (35%). This implies that many pedestrians will be lost during this process. The TPR drops due to the camera motion (mainly, pitch angle variations) produced by road slopes, which produce that in many cases the fixed plane does not coincide with the real one, and hence the generated candidate windows are not correct. A trade-off between TPR and the number of candidates is reached using adaptive road scanning, where the road plane is adjusted in each frame. However, the TPR is not perfect (74%). Finally, our method reduces remarkably the search space but, at same time, maintains a high performance with respect to TPR (84%). The obtained reduction in the number of candidates is very significant because we combine strong clues about the physical world for filtering the search space. The used priors regarding vertical surfaces and its depths, coupled with our spatial restrictions, allow us to focus on image regions where the probability of having pedestrians is relatively high.



(a) Sliding windows (0.1%)

(b) Flat world assumption (5%)

(c) Adaptive road scanning (5%)

(d) Our approach (100%)

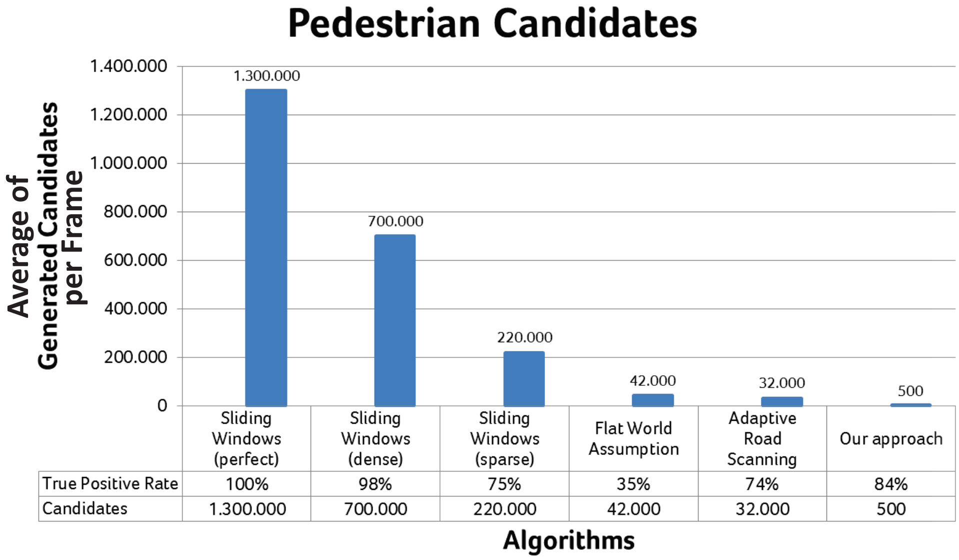
Figure 6.3: Qualitative evaluation. Here, we can see the candidate windows generated by the considered approaches versus our results. The number in parenthesis indicates the percentage of displayed windows. The number of candidate windows is significantly reduced by applying our approach. Figures (a)-(c) were taken from [37].

To reach a similar performance to our method, sliding window approach requires generating approximately 300.000 windows, which is an amount 600 times bigger than the candidates generated by our method.

Figure 6.5 shows qualitative results obtained with our method. Figures 6.5(b) and (c) show geometric and depth information used in our candidates generation process. As we depict in Fig. 6.5(d), the windows are posed only over interesting regions for our context, while large image portions are discarded.

6.5.2 Analysis of False Negatives

We analyze the characteristics of the pedestrians not included by the hypothesis generated by our proposal. To do so, we build an histogram counting the FN according with its depth. We can see in Fig. 6.6 that nearly 80 % of them are distant pedestrians. This is because the far pedestrians have smaller sizes (i.e., very few pixels) due to the sensor resolution. Then, they are hard to be segmented into their constituents parts



	Sliding Windows (perfect)	Sliding Windows (dense)	Sliding Windows (sparse)	Flat World Assumption	Adaptive Road Scanning	Our approach
True Positive Rate	100%	98%	75%	35%	74%	84%
Candidates	1.300.000	700.000	220.000	42.000	32.000	500

Figure 6.4: Comparison between different approaches to generate candidate windows.

by the superpixel algorithm and to agglomerate by our approach. However, these pedestrians are outside the high-risk area, and will be further detected with very high probability when the car approaches to them, since our proposal includes 99.4% of pedestrian in the range 0-10 m. From our opinion, this candidate distribution with respect to distances is preferable since closer pedestrians are the most vulnerable ones and require special efforts.

6.6 Conclusions

In this chapter we have presented a novel monocular method for generating pedestrian candidates for the sake of pedestrian detection from images taken from a moving car. Our method is based on cues which involve two relevant sources of information about a scene: geometric relationships and depth. Geometric information is extracted by using the approach proposed by Hoiem et al. [58], whereas depth is roughly computed by our multiclass classifier approach. Both clues are combined to generate pedestrian candidates by a hierarchical clustering. This clustering agglomerates pixels which are related through physical properties like appearance, gravity, proximity, and size constraints that we impose.

We have evaluated our model for pedestrian candidates generation and compared it with respect to other approaches in the state-of-the-art. The results shown that our method overcome all considered methods because significantly reduces the number of

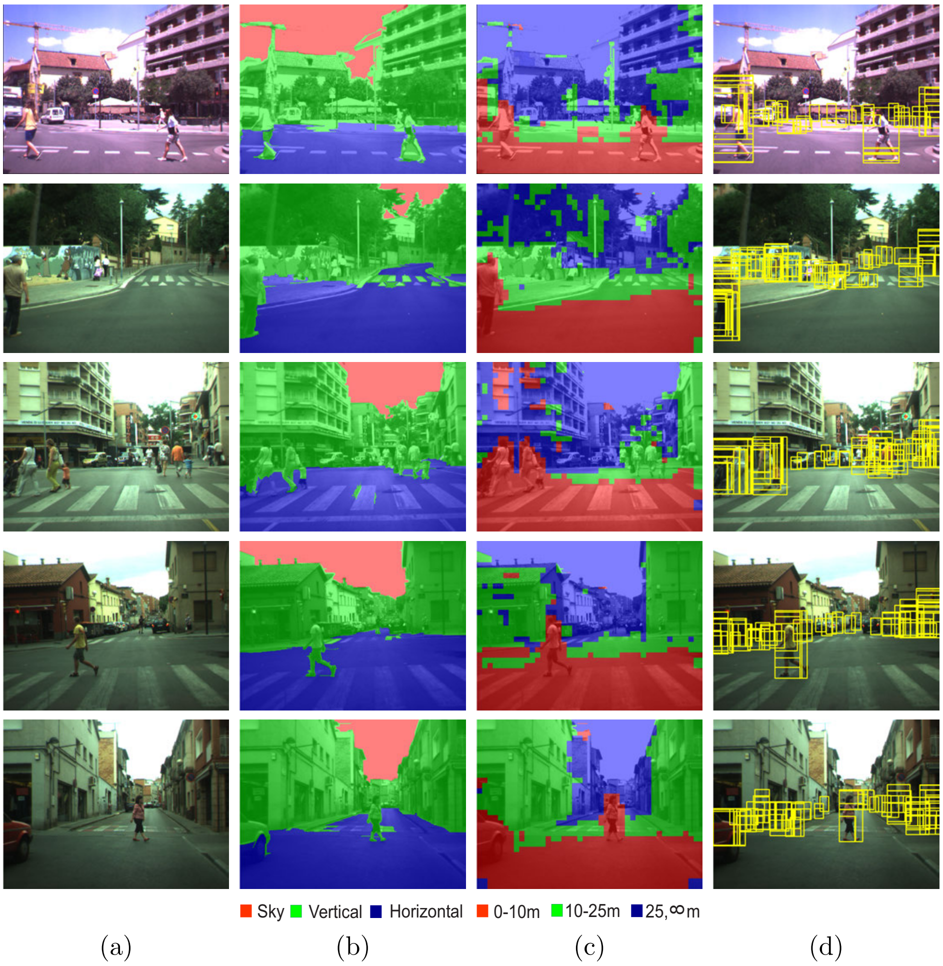


Figure 6.5: Results of our approach to build a compact representation of the world: (a) Original image, (b) Geometric information, (c) Depth information, and (d) Candidate windows.

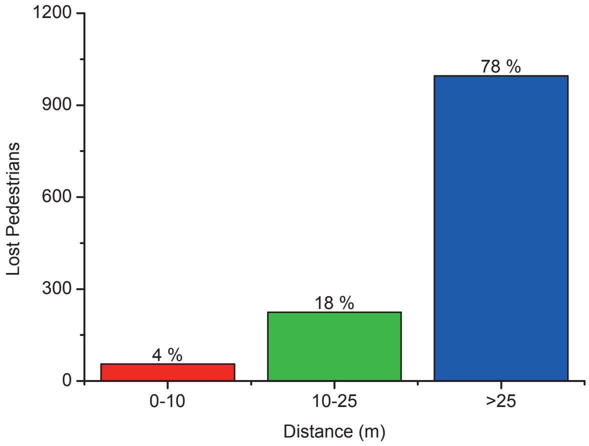
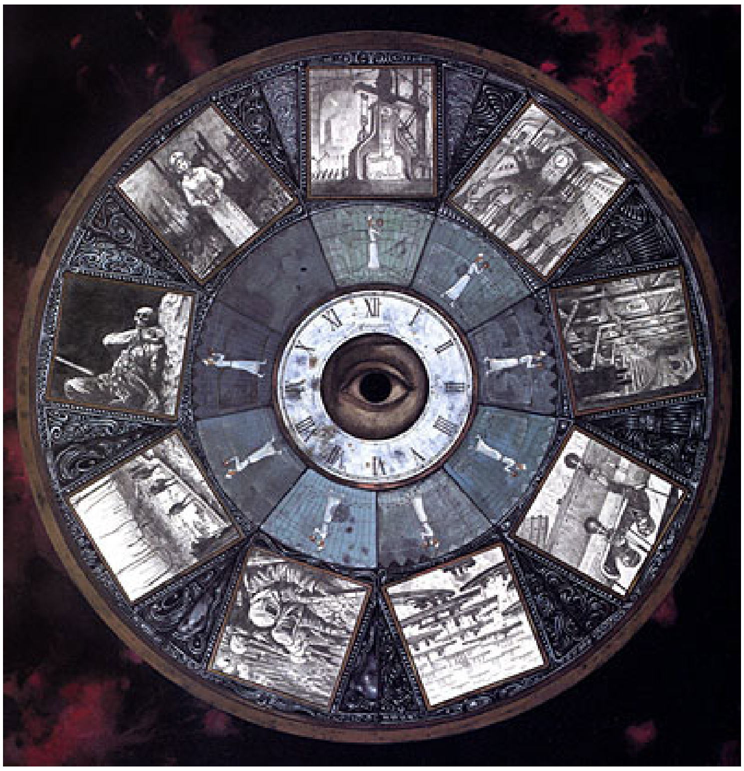


Figure 6.6: Histogram of pedestrians which are lost during our process. We can see that mainly they are pedestrian located at far distances from the car, and they will be further detected.

candidates to be evaluated by a posterior pedestrian classifier, as the high value for TPR shows. Additionally, our method looses few pedestrians, but there are mainly located at non-risk areas.



The wheel of industry, Shaun Tan, 2011.

The Viewer, written by acclaimed horror writer Gary Crew and illustrated by Shaun Tan, tells the story of a boy whose obsession with curious artifacts leads him to discover an strange box at a dump site. It proves to be an ancient chest full of optical devices. One is an intricate viewing device which reveals images of significant historical ages, including the present and the future.

Regarding our present work, we have shown how computer vision problems can be posed on different and novel perspectives to lead better results. However, we are aware that there are still a lot of future work, where new ways of address problems could be discovered. As in “The Viewer”, the future, the novelty is there, in somewhere, waiting to be found.

Chapter 7

Conclusions

All thinking is sorting, classifying. All perceiving relates to expectations and therefore to comparisons. **Art and Illusion, E. H. Gombrich, 1960.**

In this chapter, we highlight overall conclusions of our work, which are intimately linked with the contribution of the thesis. After that, we describe some possible further lines to extend our approaches.

7.1 Summary and contributions

In this thesis, we have studied how the human perceive depth using monocular cues. Inspired on this fact, we have focused on how, by means of simple image features is possible obtaining a representation of a scene depth from a single outdoor image taken from a camera. We have developed a low-cost method to compute coarse depth maps by applying the learned relation between a set of visual features extracted from a scene image and scene depth. The resulting depth representation is sufficiently informative to tackle many computer vision problems from a different perspective, as we have demonstrated through the different explored applications. Specifically, we have studied the use of depth information in three distinct computer vision problems: camera rotation estimation, background estimation, and pedestrian candidate generation. For each one, we have proposed new methods in which there are an evident, but unexploited, relations between problem definition and depth. In the following, we summarize the major contributions of this work.

Monocular coarse depth map estimation We have proposed a supervised learning approach to classify the pixels of outdoor images in just four categories: near, medium-distance, far and very-far. Based on how humans perceive depth at far distances, we have analyzed and selected a reduced set of low-level features to infer depth information. These features have been used to describe image regions, which are processed by Adaboost classifiers whose combined output is used to segment the image into depth categories. Segmentation results have been combined by using conditional random field approach to provide spatial coherence.

In the quantitative evaluation, we have quantified the performance achieved by of our depth segmentation approach for different configurations of the image oversegmentation process, varying the number of regions for regular grid and superpixels. As result of these experiments, we have concluded that the performance of both settings is quite similar. Then, the rest of experiments have performed using a regular grid configuration.

We have analyzed the significance of the selected visual features for the proposed coarse depth segmentation. Region descriptions including color, texture and location lead to a reliable performance. Our results have shown that the usage of all selected features is the best performing configuration.

Additionally, we have compared the performance of our proposal against the one achieved using a more ambitious depth map estimation method. Our method overcomes the performance of it, using a remarkable inferior number of low-level features.

Egomotion estimation methods based on distant regions We have used distant regions obtained from a single camera to obtain accurate camera motion estimations, relying on the fact that at distant scene regions, image flow is dominated by mainly rotational motion component. Distant regions have been selected by applying a classifier from our segmentation approach, which allow us to distinguish between close/distant regions in a image. Here, we have proposed two novel algorithms to rotation estimation. The first algorithm uses corresponding distant points (DP) to estimate the camera rotation by solving an overdetermined set of equations. The second one tracks distant regions (DR) between two frames, avoiding point matching. Once rotation is computed, we have estimated translation (up to a scale factor) by canceling rotational effect on the images, leaving the resulting motion just explained by translation velocities.

Real experiments in different sequences have shown that rotations are accurately estimated, since distant points/regions provide strong indicators of camera rotation. Our algorithms outperforms other monocular state-of-the-art methods, and have a comparable performance with respect to a selected stereo algorithm.

Despite our two algorithms are very precise, the accuracy of DR outperforms DP because DR exploits valuable information for motion estimation, which is available on these distant regions. DP requires distant points that commonly are located at the sky or other less textured image regions. These kind of points are hard to be detected and tracked in successive frames. Additionally, DR avoids the feature extraction and matching, which is valuable since small errors in the estimated image flow usually bring to large perturbations in the motion estimation.

Moreover, from the experimental results, we have concluded that our methods are also robust to segmentation errors, leading to accurate results even under a large number of outliers.

Background estimation method exploiting close/distant regions We have presented a novel method to background estimation containing moving/transient objects, which uses information about proximity/distantness of a region for such purpose. Our approach penalizes close regions in a cost function, which integrates color, motion, and depth terms. The cost function is minimized by using a graph cuts approach.

We have tested our approach with sequences taken under different conditions (e.g., moving/static camera, temporal/non-temporal coherence, low/high-quality). Experimental results shown that our method significantly outperforms the median filter approach, and is also comparable to state-of-the-art methods that take advantage of human interaction.

Pedestrian candidates generation exploiting coarse depth maps We have presented a novel monocular method for generating pedestrian candidates. Our method is based on cues which involve two relevant sources of information about a scene: geometric relationships and depth. Both clues have been combined to generate pedestrian candidates by a hierarchical clustering. This clustering agglomerates pixels which are related through physical properties like appearance, gravity, proximity, and size constraints that we impose.

We have evaluated our model for pedestrian candidates generation and compared it with respect to other approaches in the state-of-the-art. The results shown that our method reduces the number of candidates to be evaluated by a posterior pedestrian classifier very significantly, while keeping a high true positive rate. Missdetections are mostly at far distance, in non-risk areas.

7.2 Future work

Here, we describe several ideas for future work that extend our current framework. Additionally, we briefly describe new areas that can benefit by using coarse depth information.

Coarse depth map estimation In our work, we just focused on the use of low-level features to our purpose. However, there are several monocular cues supporting human depth perception that can be used in depth estimation. One important depth cue is the occlusion that has the property of providing depth ordering without attenuating with distance [24]. In computer vision, occlusion has been addressed has the problem of distinguishing edges in images as occlusion boundaries corresponding to 3D scene structures. Approaches using appearance-based features for occlusion detection has been defined [105] and can be explored to extend our framework in such direction.

Another powerful visual cue is the relative size of objects. In our case, including such kind of information require an additional processing to detect and identify a known object to further determine its relative position regarding the observed object size. Perhaps this cue can be useful in the case of pedestrian and vehicle detection systems, where the information about approximately sizes is available.

Egomotion estimation methods based on distant regions In this case, a future work is the integration of our method into different ADAS systems. For instance, some pedestrian and vehicle detection systems consider a flat road, and the camera parameters are assumed as constant and known. Under these assumptions, windows with different positions and scales are generated over the road plane to be further analyzed by a classifier. However, just small variations in the pitch angle due to the variable road geometry and the vehicle dynamics significantly alter the road projection in the image, leading to poor results. From the egomotion information, the camera pitch angle can be obtained and a more precise image scanning can be done. As long as the planar road assumption holds, a better detection will be achieved.

Background estimation method exploiting close/distant regions Our approach to background estimation in some situations produce artifacts, which are not removed from the final image. To overcome this problem, it can be complemented with techniques to remove those artifacts that are still present in very dissimilar images. For instance, a gradient-domain fusion method can be used to remove temporal incoherences [82].

Additionally, another interesting topic is the selection of appropriate frames to compose the background since many frames in a sequence do not contribute to the final estimation. This can be performed by analyzing a video sequence under certain inter-frame overlapping criteria to discover what are the best frames to estimate the background, in a similar way as in [78].

Pedestrian candidates generation exploiting coarse depth maps An immediate future work is integrating our method in a pedestrian detection system to evaluate how it benefits the overall performance of such system.

We also are interested in improving pedestrian classifiers by using depth information. Due to the high variability of the pedestrian at different distances, different classifier can be trained depending on the target distance provided by our approach to compute rough depth maps.

Other possible applications of depth information We believe that estimating coarse depth maps has a wide variety of applications. For instance, we can refine the search area in the problem of tracking. To track a vehicle, the maximum displacement between frames could be bounded in each depth zone. Other applications can be improved by using an approximated depth map. For example, some vanishing point estimation methods for road detection [69] are based on searching it over a possible space of solutions. This search space can be significantly reduced by knowing where are the most distant regions in a scene which probably contains the vanishing point. A rough depth map can also be valuable to initialize 3D reconstruction algorithms, which usually start with a random disparity map and progressively refine the 3D model [55].

Appendix A

Theories of visual perception

¿Cómo puede una masa informe cerebral —exclama Newton— acabar pensando y sintiendo la gloria de los colores?" "El mundo que vemos —le contesta J. Allan Hobson, psiquiatra y neurocientífico del Instituto de Salud Mental de Massachusetts— no es más que una secuencia de estructuras de activación de neuronas que representan imágenes. Se acabó el juego. **El viaje a la felicidad, E. Punset, 2010.**

In this appendix, we introduce a historical and background summary from the perspective of vision research, focusing mainly on the mechanism by which humans can perceive depth. First, we briefly review the most relevant milestones in vision research from the different fields of science viewpoint's, including art, optics, biology and psychology. Then, we define perception, visual perception and state those facts and hypotheses that our work holds.

A.1 Introduction

Undoubtedly, the present state of our understanding in vision is influenced by the advances reached at the early stages of its research. In this appendix, we briefly examine an historical perspective of vision research, focusing on the disciplines that have influenced it. We mainly address some relevant discoveries and hypothesis on optics, art, and biology, which were intimately related regarding the way that images are formed in the eye. Under this interdisciplinary context, the psychology of vision emerged as an attempt to understand how vision perception works, which is essential to comprehend how we derive our knowledge about the world. Nowadays, despite the progress made, even many facts still are a mystery and some of these theories remain matters of debate. After that, we mainly focus on the commonly accepted mechanism by which humans can perceive depth from visible light. The study of this has inspired centuries of research, creating a vast wealth of ideas and techniques that make our own work possible.

A.2 Theories of visual perception

Early studies and discussions on vision involved optics, art, biology, philosophy and psychology, which were treated jointly by Greek thinkers. As the study of these areas progressed, the difference between them and the specialization of each one became notorious. In this section, we briefly review the most relevant milestones in vision research. The quoted text introducing this section reveals that the mysteries about how our visual system works and its relation with our brain are still under discussions in the research world, but also that they captured the interest of people since its inception.

Optics

Currently, optics is a branch of physics which study the behavior and properties of light, including its interactions with matter. Now, there is a clear distinction between optics and vision, but formerly such difference did not exist.

In the ancient Greek, which is the origin of the Western vision traditions, we find discussions and speculations about light and the visual process in different treatises. Plato (ca. 424-347 B.C.) articulated the emission theory based on the idea that visual perception is accomplished by rays emitted by the eyes. However, in line with modern conceptions, Aristotle (ca. 384-322 B.C.), in his treatise *On Sense and the Sensible*, guesses that

> *If the visual organ proper really were fire, which is the doctrine of Empedocles, a doctrine taught also in the Timaeus [of Plato], and if vision were the result of light issuing from the eye as from a lantern, why should the eye not have had the power of seeing even in the dark? It is totally idle to say, as the Timaeus does, that the visual ray coming forth in the darkness is quenched.*

The emission theory was shared by thinkers such as Euclid (ca. 325-265 B.C.) and Ptolemy (ca. 100-170). Later, in the Arab world, Ibn al-Haytham (in Latin, Alhazen) (ca. 965-1039) rejected the emission theory of vision based on reasoned idea that a ray could not proceed from the eyes and reach the stars at the instant after we open our eyes. He also showed that vision is not merely a phenomenon of pure sensation (namely what results from the introduction of light rays into the eyes), but that it involves the faculties of judgment, imagination and memory.

The belief that the eye is an active organ that emits ray of light instead of a receiver of it, existed in scientific circles until Kepler's work on the retinal image early in the 17th century [117]. Even today there are misunderstandings about visual perception, as some works reveal [120].

Art

Greek traditions about vision from medical, philosophical and mathematical viewpoints were later transmitted to Islam and Latin Christendom, laying the basis for medieval and Renaissance theories of vision [76].

During the Renaissance, one of the distinguishing features was the attempts to represent the world in a convincing manner by applying techniques like linear perspective. From the rediscovery of Euclid's geometry and Alhazen's optics, linear perspective was developed and formalized by Fillipo Brunelleschi (1377-1446) and Leon Battista Alberti (1404-1472) [67]. It is interesting to note that the principles of reducing a 3D scene to a 2D picture were formulated before the image-forming properties of the eye had been described.

Leonardo Da Vinci (1452-1519) was among the first to take a scientific approach towards understanding how our world works and how we see it. In a systematic way, he focused on four main themes: painting, architecture, mechanics, and human anatomy. In one of these works, he noted how an object looked when it was close and when it was farther away, considering the atmosphere effects on the colors, and under different light conditions. He meticulously measured the apparent sizes of objects at different distances to know why distant objects looked smaller and what size to paint them if he knew exactly how far away they were. These observations allowed him to develop the perspective drawing, which was reflected on the greater realism of paintings, giving the illusion of depth and distance to flat walls and canvases.

Biology

One can not discard the importance of the study about the ocular anatomy and its interrelation with the brain as part of vision understanding. However, we limit here to refer just to two events which radically changed the perspective about vision process.

On the one hand, Johannes Kepler (1571-1630) proposed the image projection onto retinal, which was a pivotal point in the vision research. He is best known for his eponymous laws of planetary motion, however he also focused on the optical theory, extending his study of optics to the human eye, and he is generally considered by neuroscientists to be the first to correctly describe the formation of the retinal image in the eye [76].

On the other hand, Charles Wheatstone (1802-1875) offered the explanation of stereopsis, that "the mind perceives an object of three dimensions by means of the two dissimilar pictures projected by it on the two retina" [119].

Psychology

From a psychological point of view, diverse theories to explain how the human beings perceive their environment have been proposed in the later two centuries. Here, we only briefly describe the most relevant for us.

The structuralism school was founded by Wilhem Wundt (1832-1920) as an attempt to explore the mind by means of a careful study of its basic elements and their relationships. According with this school of thought, each stimulus element in a scene yields its own sensation and the totality of these sensations forms the percept [45].

However, some basic facts are difficult to explain based uniquely on sensations. For instance, when we move away from an object, the experienced sensation is that the object reduces its size, arising from the shrinking retinal image; however, we known that the object remains the same size, which shows some learning based on experience. Due to that, Hermann von Helmholtz (1821-1894), in a constructivist matter, believed that external world cannot be directly perceived, but our visual system interprets the sensory evidence in order to construct percepts. This is the result of an "unconscious inference" based on assumptions and conclusions from our long history of visual experiences and interactions with the world [48]. This suggests that learning plays a fundamental role for perceiving.

Gestalt theory emerged as a reaction against previous approaches to psychology, based on the work of three pioneers: Max Wertheimer (1880-1943), Wolfgang Köhler (1887-1967), and Kurt Koffka (1886-1941). Basically, they state that "the whole differs from the sum of its parts" because they focused on the global and holistic processes involved in perceiving the environment. Gestalt psychologist proposed a number of rules or principles used by the human to organize percepts during perception, emphasizing that such process is indivisible. On the one hand, they describe grouping laws, explaining how smaller objects are grouped to form larger ones. On the other hand, they explain the perceptual segregation of one object from another, i.e. to distinguish between the figure and the ground. In some scholarly communities, such as cognitive psychology and computational neuroscience, Gestalt theories of perception were criticized for being descriptive rather than explanatory in nature. Moreover, the evidence for the principles of organization is based upon the manner in which 2D pictures are perceived rather than real 3D situations.

Unlike previous theories, James Gibson (1904-1979) believed that perception is a direct process, without the need of an extra inference processing [40]. According to Gibson, a person perceives the world as an active observer who is constantly moving his or her eyes, head and body relative to the environment. The information responsible for perception exists in the environment and it can be used immediately, without being transformed, processed or manipulated in any way. One source of this information is the optical flow produced by our motion in the environment. Optical flow information, according to Gibson, provides information about the observer speed and heading. Another kind of source information is those that remains invariant as the observer moves. For example, an observer can perceive the size of an object directly seeing the number of units of ground's texture gradient covered by the base of that object [42].

By the 1970s, the study of vision from a computational point of view was driven by the development of the information theory, cybernetics, and computers. From a more empiric perspective, the aim of this discipline is understanding visual processes by building computer models of these processes. The work of David Marr (1945-1980) contributed significantly to that goal. In a modular way, he defined a set of the stages of visual perception. Basically, the first one, called the primal sketch, takes the visual image and make explicit certain forms of information contained therein (e.g., edges, regions, etc.). Next, in the $2\frac{1}{2}$D sketch, the orientation and rough depth of visible surfaces are made explicit. Finally, objects are organized in a the 3D model representation. Marr's approach is still influential as we can see in many of the recent works described in Chapter 2.

A.3 Perception, visual perception and its relation to this work

Perception refers to processes by which an organism interprets and organizes sensations received from the world in order to understand the surrounding environment and to behave effectively within it. In this definition, we can observe two main components: on the one hand, the organism receives certain physical stimuli from the world by means a sensory receptor system; and, on the other hand, these sensations are further processed.

In humans, the sense organs (eyes, ears, nose, tongue, and skin) are composed by specialized cellular structures that have receptors for specific stimuli. These cells are linked to the nervous system and, in the later instance, to the brain, where the sensory information is selected, organized, and interpreted (and sometimes distorted). Human beings possess a multitude of senses (including the traditionally five senses of sight, hearing, smell, taste, and touch) that provide information about the physical world. The information coming in from the different modalities of the senses is integrated to give perceptual capabilities in the task that the organism is trying to perform, involving higher-level brain processing [36]. Despite that our perceptions are a completely subjective experiences, the descriptions of what we perceive can be shared and contrasted with others. Then, this communicational possibility plays an important role in checking the existence of close correspondences between the perception of the world and other descriptions of it (as, for instance, by physical measurements).

One of the most richest source of information about the environment come from our visual system, which allow us to perceive color, distance, depth, motion, and form. Obviously, vision has played a vital role in the survival of human as specie. In ecological terms, the design of the visual system has highly adaptation to the environment in which men have been evolved. For instance, the trichromatic color vision is commonly related with the ability of our early ancestors for distinguishing nutritious food, or the stereoscopic vision which provided important cues for calculating distances to safely move amongst treetops.

We state that visual perception is the ability to interpret information from the effects of visible light through the visual system. In a more computational viewpoint, the visual system is the responsible of a number of complex tasks, including the reception of light and the formation of monocular representations; the construction of a binocular perception from a pair of two dimensional projections; the identification and categorization of visual objects; assessing distances to and between objects; and guiding body movements in relation to visual objects.

From previous paragraphs, we derive those facts and hypothesis that our work satisfies:

- Optics reception theories about light and Kepler's image formation, since these theories are intimately related with the camera image formation process, which is our source of information.
- The attempts of Renaissance artists to represent 3D scenes in 2D canvas in a realistic way by means of linear perspective, use of colors, textures, aerial effects, etc., i.e. pictorial cues which are the base of our visual features used to build our models.
- The Helmholtz's idea of inference by which the visual experience is fundamental to learn models for understanding the surrounding world. We learn models from training sets (that can be assumed as "experience") by using a set of visual features.
- The Gestalt's principles of grouping and segregation that allow figure/background differentiating, allowing us to differentiate objects from background.
- The Gibson's hypothesis about that information to percept the environment is available in it. We extract information to distinguish roughly depths from visual features computed from images.
- Marr's $2\frac{1}{2}$D sketch since we are focus on obtaining rough depth of visible surfaces by applying our models.

A.4 Summary

In this appendix, we have briefly examined an historical perspective of vision research, focusing on the disciplines that have influenced it. We have mainly addressed some relevant discoveries and hypothesis on optics, art, biology, and psychology, which were intimately related regarding the way that images are formed in the eye. Under this interdisciplinary context, the psychology of vision emerged as an attempt to understand how vision perception works, which is essential to comprehend how we derive our knowledge about the world. After that, we have posed the work of this thesis in relation to relevant perception facts and hypothesis.

Appendix B

Publications

This thesis take as bases the following publications:

Conference Papers

- Camera Egomotion Estimation in the ADAS Context, D. Cheda, D. Ponsa and A. M. López, IEEE Conf. Intell. Transp. Syst., 2010.
- Monocular Egomotion Estimation based on Image Matching, D. Cheda, D. Ponsa and A. M. López, Int. Conf. Pattern Recognit. Appl. and Methods, 2012.
- Monocular Depth-based Background Estimation, D. Cheda, D. Ponsa and A. M. López, Int. Conf. Comput. Vision Theory Appl., 2012.
- Pedestrian Candidates Generation using Monocular Cues, D. Cheda, D. Ponsa and A. M. López, IEEE Intell. Vehicles Symposium, 2012.

Bibliography

[1] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. Seitz, and R. Szeliski, “Building Rome in a Day,” *Commun. ACM*, vol. 54, no. 10, pp. 105–112, 2011.

[2] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen, “Interactive Digital Photomontage,” *ACM Trans. Graph.*, vol. 23, pp. 294–302, 2004.

[3] 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 Recognit.*, vol. 36, no. 12, pp. 2927–2944, 2003.

[4] K. S. Arun, T. S. Huang, and S. D. Blostein, “Least-squares Fitting of Two 3-D Point Sets,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 698–700, 1987.

[5] H. Badino, U. Franke, and D. Pfeiffer, “The Stixel World - A Compact Medium Level Representation of the 3D-World,” in *DAGM Symp. Pattern Recognit.*, 2009, pp. 51–60.

[6] R. Bailey, C. Grimm, and C. Davoli, “The Effect of Warm and Cool Object Colors on Depth Ordering,” in *Proc. 3rd Symp. Appl. Percept. Graphics Visualization*, 2006, p. 161.

[7] T. Bailey and H. Durrant-Whyte, “Simultaneous Localization and Mapping: Part I,” *IEEE Rob. Autom Mag.*, vol. 13, no. 2, pp. 99–110, 2006.

[8] ——, “Simultaneous Localization and Mapping (SLAM): Part II,” *IEEE Rob. Autom Mag.*, vol. 13, no. 3, pp. 108–117, 2006.

[9] S. Baker and I. Matthews, “Lucas-Kanade 20 Years On: A Unifying Framework,” *Int. J. Comput. Vision*, vol. 56, no. 1, pp. 221–255, 2004.

[10] S. Battiato, S. Curti, M. La Cascia, M. Tortora, and E. Scordato, “Depth Map Generation by Image Classification,” in *Soc. of Photo-Opt. Instrum. Eng. Conf. Ser.*, vol. 5302, 2004, pp. 95–104.

[11] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, “Hierarchical Model-Based Motion Estimation,” in *European Conf. Comput. Vision*, 1992, pp. 237–252.

[12] M. Bertozzi, A. Broggi, R. Chapuis, F. Chausse, A. Fascioli, and A. Tibaldi, “Shape-Based Pedestrian Detection and Localization,” in *IEEE Conf. Intell. Transp. Syst.*, 2003, pp. 328–333.

[13] I. Biederman, *On the Semantics of a Glance at a Scene*, 1981, ch. 8, pp. 213–263.

[14] M. J. Black and D. J. Fleet, “Probabilistic Detection and Tracking of Motion Boundaries,” *Int. J. Comput. Vision*, vol. 38, pp. 231–245, 2000.

[15] Y. Boykov, O. Veksler, and R. Zabih, “Efficient Approximate Energy Minimization via Graph Cuts,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1222–1239, 2001.

[16] D. Burschka and E. Mair, “Direct Pose Estimation with a Monocular Camera,” in *Robot Vision: Int. Workshop*, 2008, pp. 440–453.

[17] H. Chen and T. Liu, “Finding Familiar Objects and their Depth from a Single Image,” in *IEEE Int. Conf. Image Proc.*, 2007, pp. 389–392.

[18] X. Chen, Y. Shen, and Y. H. Yang, “Background Estimation using Graph Cuts and Inpainting,” in *Proc. of Graphics Interface Conf.*, 2010, pp. 97–103.

[19] S. Cohen, “Background Estimation as a Labeling Problem,” in *IEEE Int. Conf. Comput. Vision*, 2005, pp. 1034–1041.

[20] A. Comport, E. Malis, and P. Rives, “Real-time Quadrifocal Visual Odometry,” *Int. J. Rob. Res.*, vol. 29, pp. 245–266, 2010.

[21] K. Crammer and Y. Singer, “On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines,” *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, 2002.

[22] ——, “On the Learnability and Design of Output Codes for Multiclass Problems,” *Mach. Learn.*, vol. 47, no. 2-3, pp. 201–233, 2002.

[23] A. Criminisi, “Single-View Metrology: Algorithms and Applications,” in *DAGM Symp. on Pattern Recognit.*, 2002, pp. 224–239.

[24] J. Cutting and P. Vishton, “Perceiving Layout and Knowing Distances: The Integration, Relative Potency, and Contextual Use of Different Information About Depth,” in *Percept. of Space and Motion*, 1995, ch. 3, pp. 69–117.

[25] N. Dalal, “Finding People in Images and Videos,” Ph.D. dissertation, Institut National Polytechnique de Grenoble, 2006.

[26] A. Delong, A. Osokin, H. Isack, and Y. Boykov, “Fast Approximate Energy Minimization with Label Costs,” *Int. J. Comput. Vision*, pp. 1–27, 2011.

[27] M. Dimiccoli, J. Morel, and P. Salembier, "Monocular Depth by Nonlinear Diffusion," *Indian Conf. Comput. Vision, Graphics & Image Process.*, pp. 95–102, 2008.

[28] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results."

[29] M. Everingham, A. Zisserman, C. K. I. Williams, L. V. Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dorkó, S. Duffner, J. Eichhorn, J. D. R. Farquhar, M. Fritz, C. Garcia, T. Griffiths, F. Jurie, D. Keysers, M. Koskela, J. Laaksonen, D. Larlus, B. Leibe, H. Meng, H. Ney, B. Schiele, C. Schmid, E. Seemann, J. Shawe-taylor, A. Storkey, O. Szedmak, B. Triggs, I. Ulusoy, V. Viitaniemi, and J. Zhang, "The 2005 PASCAL Visual Object Classes Challenge," in *First PASCAL Challenges Workshop*, 2006.

[30] C. Fermüller and Y. Aloimonos, "Ambiguity in Structure from Motion: Sphere versus Plane," *Int. J. Comput. Vision*, vol. 28, no. 2, pp. 137–154, 1998.

[31] Y. Freund and R. E. Schapire, "A Short Introduction to Boosting," *J Jpn. Soc. Artif. Intell.*, vol. 14, no. 5, pp. 771–780, 1999.

[32] J. Friedman, T. Hastie, and R. Tibshirani, "Additive Logistic Regression: a Statistical View of Boosting," *Annals of Statistics*, vol. 28, pp. 337–374, 2000.

[33] D. Gavrila and S. Munder, "Multi-cue Pedestrian Detection and Tracking from a Moving Vehicle," *Int. J. Comput. Vision*, vol. 73, pp. 41–59, 2007.

[34] A. Geiger and B. Kitt, "ObjectFlow: A Descriptor for Classifying Traffic Motion," in *IEEE Intell. Veh. Symp.*, 2010.

[35] A. Geiger, J. Ziegler, and C. Stiller, "StereoScan: Dense 3D Reconstruction in Real-time," in *IEEE Intell. Veh. Symp.*, 2011.

[36] W. Geisler, "Visual Perception and the Statistical Properties of Natural Scenes," *Annual Review of Psychology*, vol. 59, pp. 167–92, 2008.

[37] D. Gerónimo, "A Global Approach to Vision-based Pedestrian Detection for Advanced Driver Assistance Systems," Ph.D. dissertation, Computer Vision Center, Universitat Autònoma de Barcelona, 2010.

[38] D. Gerónimo, A. López, A. Sappa, and T. Graf, "Survey of Pedestrian Detection for Advanced Driver Assistance Systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1239 –1258, 2010.

[39] J. Geusebroek and A. Smeulders, "A Six-Stimulus Theory for Stochastic Texture," *Int. J. Comput. Vision*, vol. 62, pp. 7–16, 2005.

[40] J. Gibson, *The Ecological Approach to Visual Perception.* Lawrence Erlbaum Associates, 1986.

[41] ——, *Motion Picture Testing and Research*, ser. Aviation Psychology Program Research Reports. U.S. Govt. Print. Off., 1947.

[42] B. Goldstein, “The Ecology of J.J. Gibson’s Perception,” *Leonardo*, vol. 14, no. 3, pp. 191–195, 1981.

[43] ——, *Sensation and Perception.* Wadsworth Cengage Learning, 2010.

[44] H. Goldstein, C. P. Poole, and J. L. Safko, *Classical Mechanics (3rd Edition).* Addison Wesley, 2002.

[45] I. Gordon, *Theories of Visual Perception.* Psychology Press, 2004.

[46] S. Gould, R. Fulton, and D. Koller, “Decomposing a Scene into Geometric and Semantically Consistent Regions,” in *IEEE Int. Conf. Comput. Vision*, 2009, pp. 1–8.

[47] M. Granados, H.-P. Seidel, and H. P. A. Lensch, “Background Estimation from Non-Time Sequence Images,” in *Proc. of Graphics Interface Conf.*, 2008, pp. 33–40.

[48] R. Gregory, “Knowledge in Perception and Illusion,” *Philosophical Transactions of the Royal Society of London - Series B: Biological Sciences*, vol. 352, no. 1358, pp. 1121–1127, 1997.

[49] C. Guibal and B. Dresp, “Interaction of Color and Geometric Cues in Depth Perception: When Does “Red” Mean “Near”?” *Psych. Res.*, vol. 69, pp. 30–40, 2004.

[50] D. Gutchess, M. Trajkovics, E. Cohen-Solal, D. Lyons, and A. Jain, “A Background Model Initialization Algorithm for Video Surveillance,” in *IEEE Int. Conf. Comput. Vision*, vol. 1, 2001, pp. 733–740.

[51] D. Hanes, J. Keller, and G. McCollum, “Motion Parallax Contribution to Perception of Self-motion and Depth,” *Biol. Cybern.*, vol. 98, no. 4, pp. 273–293, 2008.

[52] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2004.

[53] M. Harville, G. Gordon, and J. Woodfill, “Adaptive Video Background Modeling using Color and Depth,” in *Int. Conf. Image Process.*, vol. 3, 2001, pp. 90–93.

[54] K. He, J. Sun, and X. Tang, “Single Image Haze Removal Using Dark Channel Prior,” in *IEEE Conf. Comput. Vision and Pattern Recognit.*, 2009, pp. 1956 –1963.

[55] H. Hirschmuller, “Stereo processing by semiglobal matching and mutual information,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 328–341, 2008.

[56] D. Hoiem, A. Efros, and M. Hebert, “Automatic Photo Pop-Up,” *ACM Trans. on Graphics*, vol. 24, no. 3, pp. 577–584, 2005.

[57] ——, “Geometric Context from a Single Image,” in *Int. Conf. Comput. Vision*, 2005, pp. 654 – 661.

[58] ——, “Recovering Surface Layout from an Image,” *Int. J. Comput. Vision*, vol. 75, no. 1, pp. 151–172, 2007.

[59] B. K. P. Horn and B. G. Schunck, “Determining Optical Flow,” *Artif. Intell.*, vol. 17, pp. 185–203, 1981.

[60] J. Huang and C. X. Ling, “Using AUC and Accuracy in Evaluating Learning Algorithms,” *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 3, pp. 299–310, 2005.

[61] J. Huang and B. Cowan, “Simple 3D Reconstruction of Single Indoor Image with Perspective Cues,” in *Can. Conf. on Comput. and Rob. Vision*, 2009, pp. 140–147.

[62] M. Irani and P. Anandan, “About Direct Methods,” in *Int. Workshop on Vision Algorithms: Theory and Pract.*, 2000, pp. 267–277.

[63] M. Irani, P. Anandan, and M. Cohen, “Direct Recovery of Planar-Parallax from Multiple Frames,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1528 – 1534, nov 2002.

[64] M. Irani, B. Rousso, and S. Peleg, “Recovery of Ego-motion Using Region Alignment,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 3, pp. 268–272, 1997.

[65] B. Kitt, A. Geiger, and H. Lategahn, “Visual Odometry based on Stereo Image Sequences with RANSAC-based Outlier Rejection Scheme,” in *IEEE Intell. Veh. Symp.*, 2010, pp. 486 – 492.

[66] D. C. Knill, “Mixture Models and the Probabilistic Structure of Depth Cues,” *Vision Res.*, vol. 43, no. 7, pp. 831–854, 2003.

[67] W. R. Knorr, “On the Principle of Linear Perspective in Euclid’s Optics,” *Centaurus*, vol. 34, no. 3, pp. 193–210, 1991.

[68] P. Kohli, L. Ladicky, and P. Torr, “Robust Higher Order Potentials for Enforcing Label Consistency,” in *IEEE Conf. Comput. Vision Pattern Recognit.*, june 2008, pp. 1–8.

[69] H. Kong, J. Audibert, and J. Ponce, “General Road Detection from a Single Image,” *Trans. Img. Proc.*, vol. 19, no. 8, pp. 2211–2220, 2010.

[70] K. Konolige, M. Agrawal, and J. Solà, “Large-Scale Visual Odometry for Rough Terrain,” in *Rob. Res.*, ser. Springer Tracts in Advanced Robotics, 2011, vol. 66, pp. 201–212.

[71] V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick, “Graphcut Textures: Image and Video Synthesis using Graph Cuts,” *ACM Trans. Graph.*, vol. 22, pp. 277–286, July 2003.

[72] V. Labatut and H. CherifiLaC2012, “Accuracy Measures for the Comparison of Classifiers,” *Computing Research Repository*, 2012.

[73] T. S. Lee, “Image Representation using 2D Gabor Wavelets,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 959–971, 1996.

[74] R. Lerner and E. Rivlin, “Direct Method for Video-Based Navigation Using a Digital Terrain Map,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 406 –411, 2011.

[75] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, “TurboPixels: Fast Superpixels Using Geometric Flows,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, pp. 2290–2297, 2009.

[76] D. C. Lindberg, *Theories of Vision from al-Kindi to Kepler*, ser. University of Chicago history of science and medicine. University of Chicago Press, 1981.

[77] B. Liu, S. Gould, and D. Koller, “Single Image Depth Estimation from Predicted Semantic Labels,” in *IEEE Conf. Comput. Vision Pattern Recognit.*, 2010, pp. 1253–1260.

[78] F. Liu, Y.-h. Hu, and M. L. Gleicher, “Discovering Panoramas in Web Videos,” in *ACM Int. Conf. on Multimedia*, 2008, pp. 329–338.

[79] W. Long and Y.-H. Yang, “Stationary Background Generation: An Alternative to the Difference of Two Images,” *Pattern Recogn.*, vol. 23, pp. 1351–1359, November 1990. [Online]. Available: http://dl.acm.org/citation.cfm?id=92641.92648

[80] B. Lucas and T. Kanade, “An Iterative Image Registration Technique with an Application to Stereo Vision,” in *Int. Joint Conf. Artif. Intell.*, 1981, pp. 674–679.

[81] P. Mamassian, D. C. Knill, and D. Kersten, “The Perception of Cast Shadows,” *Trends in Cognitive Sci.*, vol. 2, no. 8, pp. 288–295, 1998.

[82] J. T. J. K. C. T. Miguel Granados, Kwang In Kim, “Background Inpainting for Videos with Dynamic Objects and a Free-moving Camera,” in *European Conf. Comput. Vision*, 2012.

[83] V. Nedovic, A. Smeulders, A. Redert, and J. M. Geusebroek, “Stages As Models of Scene Geometry,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1673–1687, 2010.

[84] D. Nistér, “An Efficient Solution to the Five-Point Relative Pose Problem,” *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 26, no. 6, pp. 756–777, 2004.

[85] D. Nistér, O. Naroditsky, and J. R. Bergen, “Visual Odometry for Ground Vehicle Applications,” *J. Field Robotics*, vol. 23, no. 1, pp. 3–20, 2006.

[86] S. Obdrzalek and J. Matas, “A Voting Strategy for Visual Ego-motion from Stereo,” in *IEEE Intell. Veh. Symp.*, june 2010, pp. 382 –387.

[87] J. Oliensis, “A Critique of Structure-from-Motion Algorithms,” *Comput. Vision Image Understanding*, vol. 80, no. 2, pp. 172–214, 2000.

[88] D. Pfeiffer and U. Franke, “Modeling Dynamic 3D Environments by Means of the Stixel World,” *IEEE Intell. Transp. Syst. Mag.*, vol. 3, no. 3, pp. 24–36, 2011.

[89] D. Ponsa, J. Serrat, and A. M. López, “On-board Image-based Vehicle Detection and Tracking,” *Trans. Inst. Meas. Control*, vol. 33, no. 7, pp. 783–805, 2011.

[90] R. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, “Image Change Detection Algorithms: A Systematic Survey,” *IEEE Trans. Image Process.*, vol. 14, no. 3, pp. 294–307, 2005.

[91] F. Remondino and C. Fraser, “Digital Camera Calibration Methods: Considerations and Comparisons,” in *Int. Arch. of Photogramm., Remote Sens. and Spatial Inf. Sci.*, vol. XXXVI, no. part 5, 2006, pp. 266–272.

[92] X. Ren and J. Malik, “Learning a Classification Model for Segmentation,” in *IEEE Int. Conf. Comput. Vision*, 2003, pp. 10–17.

[93] C. Ridder, O. Munkelt, and H. Kirchner, “Adaptive Background Estimation and Foreground Detection using Kalman-Filtering,” in *Int. Conf. Recent Adv. Mechanotronic*, 1995, pp. 193–199.

[94] R. Rifkin and A. Klautau, “In Defense of One-Vs-All Classification,” *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, 2004.

[95] A. Saxena, J. Schulte, and A. Ng, “Depth Estimation using Monocular and Stereo Cues,” in *Int. Joint Conf. Artif. Intell.*, 2007, pp. 2197–2203.

[96] A. Saxena, M. Sun, and A. Ng, “Make3D: Learning 3D Scene Structure from a Single Still Image,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 824 –840, 2009.

[97] D. Scaramuzza and F. Fraundorfer, “Visual Odometry,” *IEEE Rob. Autom. Mag.*, vol. 18, no. 4, pp. 80–92, 2011.

[98] R. Schapire and Y. Singer, “Improved Boosting Algorithms Using Confidence-rated Predictions,” *Mach. Learn.*, vol. 37, no. 3, pp. 297–336, 1999.

[99] R. E. Schapire and Y. Freund, *Boosting: Foundations and Algorithms.* MIT Press, 2012.

[100] P. H. Schönemann, “A Generalized Solution of the Orthogonal Procrustes Problem,” *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[101] F. Y. Shih, T. Klotz, and W. Brockmann, “A System for Rotational Velocity Computation from Image Sequences,” *Image Vision Comput.*, vol. 24, no. 4, pp. 357 – 362, 2006.

[102] R. Sibson, "SLINK: An Optimally Efficient Algorithm for the Single-Link Cluster Method," *Comput. J.*, vol. 16, no. 1, pp. 30–34, 1973.

[103] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo Tourism: Exploring Photo Collections in 3D," in *SIGGRAPH Conf.*, 2006, pp. 835–846.

[104] C. Stauffer and W. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking," in *IEEE Conf. Comput. Vision Pattern Recognit.*, vol. 2, 1999, pp. 637–663.

[105] A. Stein, "Occlusion Boundaries: Low-Level Detection to High-Level Reasoning," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, 2008.

[106] G. Temple, *Cartesian Tensors : an Introduction.* Dover Publications, 2004.

[107] T. Thanh, H. Nagahara, R. Sagawa, Y. Mukaigawa, M. Yachida, and Y. Yagi, "Robust and Real-Time Egomotion Estimation Using a Compound Omnidirectional Sensor," in *IEEE Int. Conf. Rob. Autom.*, 2008, pp. 492–497.

[108] T. Thanh, Y. Kojima, H. Nagahara, R. Sagawa, Y. Mukaigawa, M. Yachida, and Y. Yagi, "Real-Time Estimation of Fast Egomotion with Feature Classification Using Compound Omnidirectional Vision Sensor," *IEICE Trans. Inf. Syst.*, vol. 93D, no. 1, pp. 152–166, 2010.

[109] T. Tian, C. Tomasi, and D. J. Heeger, "Comparison of Approaches to Egomotion Computation," in *IEEE Int. Conf. Comput. Vision and Pattern Recognit.*, 1996, pp. 315–320.

[110] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view Traffic Sign Detection, Recognition, and 3D Localisation," in *IEEE Workshop App. Comput. Vision*, 2009, pp. 1–8.

[111] A. Torralba and A. Oliva, "Depth Estimation from Image Structure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, pp. 1226–1238, 2002.

[112] ——, "Statistics of Natural Image Categories," *Network*, vol. 14, no. 3, pp. 391–412, 2003.

[113] T. Troscianko, R. Montagnon, J. L. Clerc, E. Malbert, and P.-L. Chanteau, "The Role of Colour as a Monocular Depth Cue," *Vision Res.*, vol. 31, no. 11, pp. 1923–1929, 1991.

[114] K. van de Sande, J. Uijlings, T. Gevers, and A. Smeulders, "Segmentation as Selective Search for Object Recognition," in *IEEE Int. Conf. Comput. Vision*, 2011, pp. 1879–1886.

[115] A. V. van den Berg and E. Brenner, "Humans Combine the Optic Flow with Static Depth Cues for Robust Perception of Heading," *Vision Res.*, vol. 34, no. 16, pp. 2153–2167, 1994.

[116] ——, "Why Two Eyes are Better than One for Judgements of Heading," *Nature*, vol. 371, pp. 700–702, 1994.

[117] N. Wade and M. Swanston, *Visual Perception: An Introduction.* Psychology Press, 2001.

[118] G. Wang, Z. Hu, F. Wu, and H. Tsui, "Single View Metrology from Scene Constraints," *Image Vision Comput.*, vol. 23, no. 9, pp. 831–840, 2005.

[119] C. Wheatstone, "Contributions to the physiology of vision. part the first. on some remarkable, and hitherto unobserved, phenomena of binocular vision," *Philosophical Transactions of the Royal Society of London*, vol. 128, no. 1, pp. 371–394, 1838.

[120] G. A. Winer, J. E. Cottrell, V. Gregg, J. S. Fournier, and L. A. Bica, "Fundamentally Misunderstanding Visual Perception. Adults' Belief in Visual Emissions." *The American Psychologist*, vol. 57, no. 6-7, pp. 417–424, 2002.

[121] P. Yin, A. Criminisi, J. Winn, and I. Essa, "Bilayer Segmentation of Webcam Videos Using Tree-Based Classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 30–42, 2011.

[122] L. Zelnik-Manor and M. Irani, "Multi-Frame Estimation of Planar Motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 1105–1116, 2000.

[123] R. Zhang, P. Tsai, J. Cryer, and M. Shah, "Shape from Shading: A Survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 690–706, 1999.

[124] J. Zhu, H. Zhou, S. Rosset, and T. Hastie, "Multi-class Adaboost," *Stat. and Its Interface*, vol. 2, pp. 349–360, 2009.