Magno Edgar da Silva Guedes

Vision Based Obstacle Detection for All-Terrain Robots

Lisboa
2010

UNIVERSIDADE NOVA DE LISBOA

Faculdade de Ciências e Tecnologia

Departamento de Engenharia Electrotécnica

e de Computadores

Vision Based Obstacle Detection for All-Terrain Robots

Magno Edgar da Silva Guedes

Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa para obtenção do grau de Mestre em Engenharia Electrotécnica e de Computadores.

Orientador: Prof. José António Barata de Oliveira

Lisboa
2010

Acknowledgements

First of all, I would like to express my sincere gratitude to my dissertation supervisor, Prof. José Barata, for the opportunity, motivation and support, and to Pedro Santana for all the support, fruitful comments and valuable help.

I would also like to thank my parents Aurélio and Florinda and my brother Quim for giving me strength to never give up. Finally, I thank my friends and colleagues for all the support.

Eles não sabem, nem sonham,
que o sonho comanda a vida,
que sempre que um homem sonha
o mundo pula e avança
como bola colorida
entre as mãos de uma criança.

António Gedeão

Resumo

Esta dissertação apresenta uma solução para o problema da detecção de obstáculos em ambientes todo-o-terreno, com particular interesse para robôs móveis equipados com visão estereoscópica. Apesar das vantagens da visão, sobre outros tipos de sensores, tais como o custo, peso e consumo energético reduzidos, a sua utilização ainda apresenta uma série de desafios. Tais desafios incluem a dificuldade em lidar com a considerável quantidade de informação gerada e a robustez necessária para acomodar níveis altos de ruído. Estes problemas podem ser atenuados por pressupostos rígidos, tal como considerar que o terreno à frente do robô é planar. Apesar de permitir um menor custo computacional, estas simplificações não são necessariamente aceitáveis em ambientes mais complexos, onde o terreno pode ser mais irregular. Esta dissertação propõe a extensão de um conhecido detector de obstáculos que, por relaxar a assumpção do plano é mais adequado para ambientes não estruturados. As extensões propostas são: (1) a introdução de um mecanismo de saliência visual para focar a detecção em regiões mais prováveis de conter obstáculos; (2) filtros de votação para diminuir a sensibilidade ao ruído; e (3) a fusão do detector com um método complementar por forma a criar um sistema híbrido e, portanto mais robusto. Resultados experimentais obtidos com imagens de ambientes todo-o-terreno mostram que as extensões propostas permitem um aumento de robustez e eficiência computacional sobre o algoritmo original.

Abstract

This dissertation presents a solution to the problem of obstacle detection in all-terrain environments, with particular interest for mobile robots equipped with a stereo vision sensor. Despite the advantages of vision, over other kind of sensors, such as low cost, light weight and reduced energetic footprint, its usage still presents a series of challenges. These include the difficulty in dealing with the considerable amount of generated data, and the robustness required to manage high levels of noise. Such problems can be diminished by making hard assumptions, like considering that the terrain in front of the robot is planar. Although computation can be considerably saved, such simplifications are not necessarily acceptable in more complex environments, where the terrain may be considerably uneven. This dissertation proposes to extend a well known obstacle detector that relaxes the aforementioned planar terrain assumption, thus rendering it more adequate for unstructured environments. The proposed extensions involve: (1) the introduction of a visual saliency mechanism to focus the detection in regions most likely to contain obstacles; (2) voting filters to diminish sensibility to noise; and (3) the fusion of the detector with a complementary method to create a hybrid solution, and thus, more robust. Experimental results obtained with demanding all-terrain images show that, with the proposed extensions, an increment in terms of robustness and computational efficiency over the original algorithm is observed.

List of Symbols and Notations

Symbol	Description
OD	Obstacle Detector
OOD	Original Obstacle Detector [Manduchi et al., 2005]
EOD	Extended Obstacle Detector [Santana et al., 2008]
SalOD	Saliency Based Obstacle Detector [Santana et al., 2009]
ESalOD	Extended Saliency Based Obstacle Detector [Santana et al., 2010]
ROC	Receiver Operating Characteristic
TPR	True Positive Rate
FPR	False Positive Rate
θ	minimum slope a surface must have to be considered as an obstacle
H_{min}	minimum height an object must have to be considered an obstacle
H_{max}	maximum allowed height between two points to be considered compatible with each other
p	generic 3-D data point
p'	projection of a generic 3-D data point p onto the image plane
C_U	upper truncated cone (for compatibility test)
C'_U	upper truncated triangle (result from projecting C_U onto the image plane)
C_L	lower truncated cone (for compatibility test)
C'_L	lower truncated triangle (projection of C_L onto the image plane)

Symbol	Description
r	range for ground plane estimation
t	minimum area a triangle, defined by three 3-D points, must have in order to the points be considered collinear
d_{plane}	maximum distance to a potential ground plane a 3-D point must have to pertain to the same plane
n_{hypo}	number of generated plane hypoteses for ground plane estimation
g	scalling factor for ground plane estimation
α	scalling factor for ground plane estimation
n	base resolution for space-variant resolution
m	base resolution for rough analysis in space-variant resolution
n_{slide}	maximum number of consecutive pixels skipped in SalOD
n_{max}	maximum number of consecutive pixels skipped in ESalOD
w	radius of scan window for the region growing process
d	maximum distance between two points to be aggregated in the region growing process
v	voting filter threshold
a	area filter threshold

Contents

List of Figures

Chapter 1

Introduction

Over the past few years, we have been observing an increasing interest on unmanned all-terrain vehicles [Matthies et al., 2007]. From military operations [Bellutta et al., 2000] to interplanetary exploration [Maimone et al., 2006], from rescue missions [Birk and Carpin, 2006] to wide-area surveillance and humanitarian demining [Santana et al., 2007], their presence is increasingly noticeable. Although the man-in-the-loop is often required for the control of high level operations, service robots must integrate autonomous capabilities, such as obstacle detection and avoidance, in order to move safely [Kim et al., 2006], [Matthies et al., 2007].

Despite the long research history in obstacle detection, a set of hard challenges are still to be tackled if the targeted environments are unstructured. This dissertation contributes to this line of research by proposing a reliable and computationally efficient obstacle detector for off-road environments. Efficiency here is particularly interesting to enable consumer robotics, which must be cheap. Therefore, expensive sensors or computational units must be avoided.

In order to perceive their surroundings, robots are normally equipped with a wide variety of sensors [Thrun et al., 2006], including Global Positioning Systems (GPS), Inertial Measurement Units (IMU), Radio Detection and Ranging Systems (RADAR), Laser Scanners (LADAR) or Stereoscopic Cameras. However, integrating all this equipment in low cost or small robots, where space and energy storage are limited, is quite challenging. From the aforementioned sensory modalities, stereo vision and laser scanners are the better suited to enable a proper

characterisation of unknown scenarios [Lalonde et al., 2006], [Manduchi et al., 2005]. These two types of sensors are complementary and consequently desirable. In particular, stereo vision overcomes several limitations of laser scanners, such as sensitivity to vibration due to mechanical components, active interaction with the environment, slow 3D reconstruction, low resolution, and absence of colour information. Moreover, vision systems aggregates a set of important features for all-terrain service robots, such as general purpose capabilities, small energetic footprint, light weight, small size and low cost. Due to these reasons, stereo vision has been selected in the context of this work. However, despite the mentioned advantages, the large amount of generated data, which is also noisy, allied to the unstructured nature of off-road environments makes the use of stereo vision for robust and fast obstacle detection a still open problem.

A common approach to reduce its computational cost is to introduce some assumptions, such as some form of structure. A typical one is that obstacles are 3-D points standing above a flat ground [Konolige et al., 2006], [Broggi et al., 2006]. However, off-road terrains are often rough and hardly flat, which makes this approach quite unsuitable. A more comprehensive technique is the fitting of several planes to several parts of the environment, and use their residual as a measure of traversability [Singh et al., 2000], [Goldberg et al., 2002]. One limitation of this method is the computational cost associated to the multiple fitting processes. Another limitation is related to its heuristic nature, which complicates the task of specifying the proper size of the planes and what an obstacle is, taking into account a specific robot's physical apparatus. A more complete, yet too expensive solution, is to generate a digital elevation map, upon which trajectories are planned according to models of interaction between the robot and the terrain [Lacroix et al., 2002]. Another known approach is to characterise obstacles in terms of the statistics governing patches of accurate 3-D point clouds [Lalonde et al., 2006]. However, such accuracy is only attainable with laser scanners. A more useful technique for stereo vision and, in particular, for outdoor environments is to define obstacles in terms of geometrical relationships between their composing 3-D points, as proposed by Manduchi et al. [Manduchi et al., 2005]. However, this approach still lacks robustness and computational efficiency.

In order to reduce both noise sensitivity and computational cost, the detector was recently extended by Santana et al. [Santana et al., 2008]. However, the proposed extensions are too rigid, i.e. noise and computational time are reduced at the expense of a reduction in terms of true positive rate. Bearing what have been said, this dissertation describes the research work that was carried out to improve robot obstacle detection. In particular, the work was concentrated on improving the previously mentioned algorithm in order to increase accuracy, robustness and computational efficiency.

1.1 Problem Statement

As previously stated, this dissertation intends to develop a pure vision-based obstacle detector for all-terrain service robots. In order to achieve this, three main problems must be taken into consideration:

1. The proposed model must be suitable for off-road environments, where both structured and unstructured surfaces must be correctly identified either as obstacles or free-space. Obstacles here are defined as anything that can block the passage of a wheeled robot.

2. The proposed model must be robust to noisy data, typical of a stereo vision-based system. Noise may be induced by several different factors (e.g. insufficient illumination, excessive light exposure, dirty lens, uncalibrated cameras, etc.) and inevitably affects the 3-D reconstruction process.

3. The proposed model must be computationally efficient and cope with real-time constraints, so that the robot can drive safely in demanding environments. As 3-D stereo reconstruction generates dense point clouds, the analysis of such data is often expensive. Thus, maintaining low complexity and improved efficiency can be particularly challenging.

1.2 Solution Prospect

This dissertation proposes the following solutions for the identified problems:

1. Obstacles will be defined according to the geometrical relationships between their composing 3-D points, as in the model proposed by Manduchi et al. [Manduchi et al., 2005]. By adopting this model, hard assumptions about the terrain's topology are discarded and surfaces will be considered as obstacles if their slope or height are higher than the maximum a wheeled robot can climb or step over. Additionally, a hybrid architecture is presented where the proposed model is integrated with a detector that makes the plane assumption. This architecture allows the exploitation of the complementary role exhibited by both detectors, i.e. to increase in true positive rate and to reduce in computation time.

2. Noise will be reduced by means of a robust filtering mechanism. This mechanism consists on a remodelled version of the *voting filters* introduced by Santana et al. [Santana et al., 2008]. Briefly, each time a pair of related 3-D data points are considered to pertain to an obstacle, each one of them cast a *vote*. By the end of the analysis, the more *votes* a potential obstacle point have, less likely is to be an outlier erroneously computed from 3-D reconstruction. This dissertation adds scale invariance to the voting mechanism. Additionally, potential obstacles are segmented and the number of points each segment contains is thresholded by a novel *area filter*, eliminating sparse noisy data.

3. Computational efficiency will be improved by a saliency-based space-variant resolution mechanism. The space-variant resolution mechanism was also proposed by Santana et al. [Santana et al., 2008] in order to reduce the number of pixels being analysed, thus saving computation time. This dissertation uses visual saliency in order to modulate the space-variant resolution, so that the most important regions, i.e. regions that are prone to contain obstacles, are analysed with more detail. This not only improves efficiency, but also reduces false positive rate.

1.3 Dissertation Outline

This dissertation is organised as follows:

Chapter 2 gives a brief overview of the state of the art regarding obstacle detection techniques for off-road environments;

Chapter 3 describes the supporting mechanisms for the developed algorithm, such as the disparity calculation, saliency computation and ground plane estimation.

Chapter 4 describes a full obstacle detector suited for autonomous mobile robots equipped with a stereo vision sensor and operating in rough and unstructured outdoor environments;

Chapter 5 presents an improved obstacle detection architecture that integrates two different techniques in an efficient way;

Chapter 6 presents a set of experimental results, which encompasses a comparative analysis between the developed model and its predecessors.

Chapter 7 gives some conclusions about the developed work and future work possibilities.

1.4 Further Readings

The work developed in this dissertation had as its starting point the algorithm proposed by Santana et al. [Santana et al., 2008]. Part of the concepts proposed in this dissertation, with the goal of extending this algorithm, had been published in:

[Santana et al., 2010] Santana, P., Guedes, M., Correia, L., and Barata, J. (2010). A saliency-based solution for robust off-road obstacle detection. *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA 2010)*.

[Santana et al., 2009] Santana, P., Guedes, M., Correia, L., and Barata, J. (2009). Saliency-based obstacle detection and ground-plane estimation for off-road vehicles. *Proceedings of the 7th Intl. Conf. on Computer Vision Systems (ICVS 2009)*, volume 5815 of *LNCS Series*, pages 275-284. Springer.

Chapter 2

State of the Art

This chapter surveys the state of the art in off-road obstacle detection algorithms. In the past 10 years, there has been extensive research on obstacle detection techniques for all-terrain service robots, mainly due to the challenging problem of properly distinguish obstacles in rough and unstructured natural environments. This problem have promoted several approaches in the search for better solutions.

Hence, herein is presented an overview of the most successful approaches to solve the problem previously mentioned.

The flat terrain assumption or flat world approach (section 2.1) is a well known method that takes advantage of simplifications in order to reduce complexity and accelerate the detection process. However, as the name suggests, work well when the terrain is relatively plane but fails in rough areas. Next, a similar approach is taken so as to improve detection in curved terrains by assuming that the plane show smooth slope variations (section 2.2), which is not necessarily the case in off-road. A more comprehensive approach is to characterise obstacles by means of a traversability cost (section 2.3). However, this technique requires the construction of local maps containing the terrain's topology, which is computationally expensive and needs high storage capability. Loosing the necessity for topology understanding of the terrain, obstacles can be characterised directly from a statistical analysis of its composing 3-D points (section 2.4). However, this method requires accurate 3-D point registration, which is only attainable with

laser scanners, thus is more applicable with data obtained by a laser scanner than the more noisy data obtained by stereo vision. Finally, a more useful technique for vision systems is to define obstacles in terms of geometric relationships between 3-D data points (section 2.5). This method is more robust, still the involved complexity requires a comprehensive solution in order to reduce computational cost.

The expensive computation of the referred techniques lies mainly in the large amount of data that have to be analysed. This fact can be circumvented if the region of analysis is reduced. However, the problem is to know which regions should be analysed. An efficient method capable of selecting interest regions is the application of visual attention mechanisms. Section 2.6 survey some of the attention mechanisms that are prone to be applied in mobile robotics.

2.1 Flat Terrain Assumption

Although off-road environments are quite marked by their irregularity, it is often possible to determine a dominant ground plane in the robot's surroundings. The presence of a ground plane is of major importance for reducing the complexity of the assumptions needed to be made in order to characterise obstacles. Moreover, assuming that the robot is navigating in a relatively flat terrain, obstacles can be simply characterised as prominent surfaces standing above the ground.

In the model proposed by Konolige et al. [Konolige et al., 2006], the robot is assumed to navigate on a locally flat ground. In this case, obstacles near to the robot's location can be detected by thresholding the height of 3-D points standing above the ground plane. Fig. 2.1 depicts the proposed model to detect and map obstacles in the robot's surroundings. First, disparity and colour images are obtained from a stereo camera. Then, the 3-D point cloud is computed from the disparity image and the ground plane is extracted using a RANSAC technique [Fischler and Bolles, 1981]. 3-D points that lie too high above the ground plane, but lower than the robot's height, are labelled as obstacles and sight lines, i.e. columns of ground plane pixels in the disparity image that lead up to a distant obstacle, determines if there is free-space. The

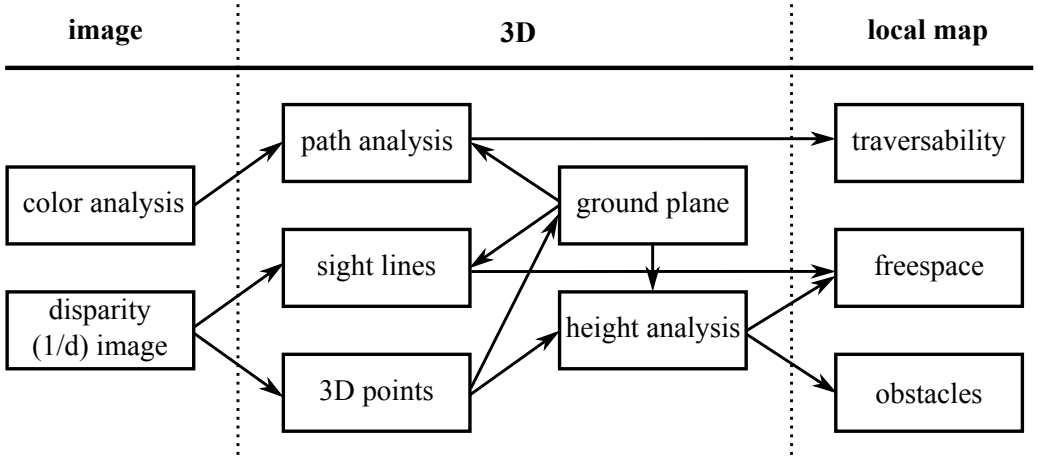


Figure 2.1: Visual processing diagram for obstacle detection in flat terrains as proposed by Konolige et al. [Konolige et al., 2006].

colour image is used by the algorithm to learn traversable paths.

Alternatively, Broggi et al. [Broggi et al., 2006] proposes to detect obstacles directly in the disparity images, rather than performing a 3-D reconstruction of the environment. Hence, detection is made by applying several filters to the disparity image in order to confine disparity concentrations that are eligible to be obstacles. With this, computational time is saved but the model are limited to specific types of obstacles. In fact, in this particular case, the model can only successfully detect thin tall obstacles or large obstacles' edges while untextured obstacles are detected with a laser scanner.

Although these approaches work well for flat terrains, they fail in rougher ones, which is typically the case in off-road.

2.2 OD in Terrains with Smooth Slope Variations

In the previous section, a flat terrain was assumed. However, apart from taking into account the presence or not of a dominant ground plane, natural terrains are hardly flat. An approach that deals with non-flat terrains is proposed by Batavia and Singh [Batavia and Singh, 2002]. Their model is suited for cases where the terrain has significant curvature but is smooth enough to consider obstacles as discrete discontinuities in the terrain. Although the basic idea behind this

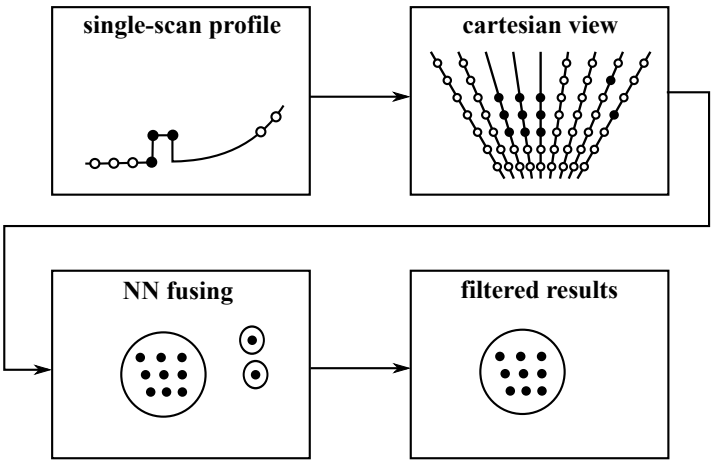


Figure 2.2: Overview of the obstacle detection algorithm for curved terrains as proposed in [Batavia and Singh, 2002].

model is similar to the one presented earlier, i.e. to find prominent surfaces above the ground, the difference lies in the ground's topology. In their work, Batavia and Singh use a 2-axis laser scanner as the input sensor. The obstacle detection algorithm is depicted in Fig. 2.2 and consists of two stages: classification and clustering (fusion). In the classification stage, each range point scanned is classified as *obstacle* or *freespace* if it represents or not a discontinuity across the ground curve. The ground curvature is estimated by converting the laser data into Cartesian coordinates and calculate the resulting gradient. Scans are then accumulated in a time window in order to determine the amount of data that will be fused in the next stage. In the fusion stage, pixels classified as obstacles are clustered using a nearest-neighbour (NN) criterion, and then candidate obstacles are filtered based on their mass and size.

Being more generic than the previous assumption, this approach still doesn't fit well with the reality of off-road environments.

2.3 Traversability and Elevation Maps

Rather that assuming a typical geometry for the terrain (flat or curved), a more comprehensive solution have also been studied, which consist on fitting several planes to different parts of the environment, and use the residual of the process as a measure of traversability, as proposed

by Goldberg et al. [Goldberg et al., 2002] and Hamner et al. [Hamner et al., 2008]. Differently from deciding if a certain region in the environment corresponds to an obstacle that the robot should avoid or a free space where the robot may navigate freely, traversability offers the possibility for the robot to negotiate a trajectory over that region trading off between the potential cost of passing through versus opt for a longer route.

The concept consists on creating grid-based local traversability maps, where obstacles are represented in terms of the level of hazardness associated to each cell. Briefly, the system starts by collecting, for each cell, the first and second moment statistics about all the range points (acquired from the input sensor and expressed in the coordinate frame of the local map) that falls inside that cell. Then, for each cell, the moment statistics from a robot-sized patch of surrounding cells are merged in order to find the best-fit plane. The resulting plane parameters are used to compute an hazard level that corresponds to the traversability cost of the cell.

Following the idea of traversability measures, a similar procedure is also presented in Lacaze et al. [Lacaze et al., 2002]. In the later, vehicle masks are placed along potential trajectories in an elevation map in order to predict pitch and roll along the paths. Thus, plane fitting is only done along the estimated paths. The pitch and roll measures are used to estimate the cost of traversing each path.

One limitation of these methods, however, is the computational cost associated to the multiple fitting processes and storage requirements. Another limitation concerns with its heuristic nature, which complicates the task of specifying the proper size of the planes and what an obstacle is, taking into account a specific robot's physical apparatus.

Lacroix et al. [Lacroix et al., 2002] proposes a model to predict the chassis attitude and the internal configurations of the robot for several positions along a trajectory arc over a digitalised elevation map. Prediction is made by a geometric placement function that modulate the interaction between the robot and the terrain. The predicted configurations are used to compute the level of *dangerousness* for each position, which is considered for choosing the best trajectory. However, this method is still too expensive to cope with real-time constraints.

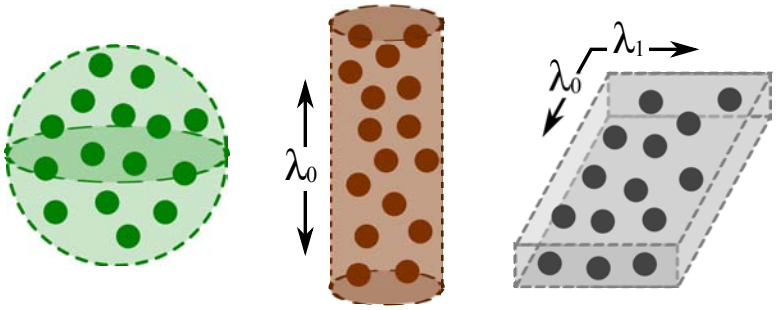


Figure 2.3: The three types of structure that 3-D data can be classified into (left: scattered regions, middle: linear structures, right: planar surfaces) [Lalonde et al., 2006].

2.4 Statistic Analysis of 3-D Data

All previous approaches focused on finding prominent surfaces on the ground or estimate trajectories based on the ground's features. However, such approaches are best applied in smooth terrains. For rougher or vegetated terrain, a different analysis criteria is needed.

A well known solution for this aspect interpret the statistics governing 3-D point clouds in order to classify visible surfaces in the environment. This solution, proposed by Vandapel et al. [Vandapel et al., 2004] and later by Lalonde et al. [Lalonde et al., 2006] uses the spatial distribution of the generated 3-D point cloud to classify regions into surfaces, linear structures and vegetation. The classification is done by first computing the eigenvalues of the covariance matrix for all the points within a neighbourhood of the point of interest and inspect the relative magnitudes of those eigenvalues. The vegetation, typically scattered points have no dominant eigenvalue, linear structures have one dominant eigenvalue and solid surfaces have two dominant eigenvalues (see Fig. 2.3). Also, the estimated ground plane of the local area can be recovered by this method as it is the eigenvector corresponding to the smallest eigenvalue of the covariance matrix.

However, this method requires accurate 3-D point registration, thus is more applicable with data obtained by a laser scanner than the more noisy data obtained by stereo vision

2.5 Geometrical Relationships in 3-D Point Clouds

An approach that have been attracting particular interest, mainly due to its correctness is the one that define obstacles in terms of geometrical relationships between their composing 3-D points. By inspecting such relationships it is possible to efficiently consider visible surfaces whose geometric properties represent a real obstacle to the robot, because they are too high or too inclined for the robot to pass through.

In the work of Manduchi et al. [Manduchi et al., 2005], an outstanding definition of obstacle is presented by introducing the concept of compatibility between pairs of 3-D points.

Briefly, a visible surface is considered part of an obstacle if its slope is larger than a certain value θ, representing the higher slope a robot can climb, and if it spans a vertical interval larger than a threshold H, representing the height an obstacle must have to block the robot from passing through. In order to apply this concept to arbitrarily shaped surfaces, slope and height measures are taken from pairs of 3-D points and those who have the conditions to pertain to the same surface and to be considered as obstacles are denoted compatibles. A more extensive explanation about compatibility will be given in section 4.1.

By this model, any surface point is considered an obstacle if there is at least one other point pertaining the same surface whose distance is within a certain interval and the line connecting them presents a slope higher than θ. The geometrical interpretation that can be made from this is that spanning an inverted truncated cone, with its vertex in a surface point and aligned upwards or downwards, if it encompasses another surface point, then both the vertex point and all the points encompassed by the cones are considered compatible and therefore, obstacle points (see Fig. 2.4).

In spite of its applicability for detecting obstacles in rough terrains, this approach suffers from a high sensitivity to noisy data and a excessive computational cost.

Several improvements over the original approach have been made in order to attenuate those issues. In [van der Mark et al., 2007] computation time was reduced by extensive use of lookup tables. Also, the inclusion of distance uncertainty measures for stereo computation have made

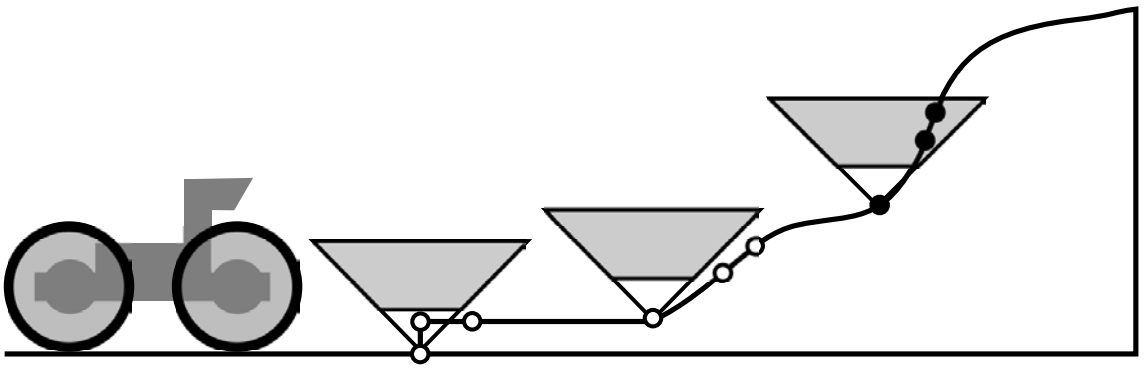


Figure 2.4: Example in side-view for the process of classifying obstacle points (filled dots) by the method proposed in [Manduchi et al., 2005]. For the sake of readability, only the upper truncated cones are represented.

possible to better detect far obstacles besides the problem that accuracy in 3-D reconstruction decreases with distance and hence, obstacle points appear more sparse. However, trying to detect far obstacles in a bad signal-to-noise ratio is not mandatory. Instead, reliable detection of obstacles with a low false positive rate, in the robot surroundings, its a more important issue that still have to be solved

In the work of Santana et. al [Santana et al., 2008], efforts where made in order to reduce computational cost of the original method by fastening the process of checking for compatible points, applying a space-variant resolution mechanism that will be described in section 4.2. Also, Santana et al. introduced two voting filters embedded in the original obstacle detection algorithm so as to reduce its sensitivity to noise. These voting filters will be also described in section 4.3.

In the later, the voting filters are too strong, influencing negatively the true positive rate. Also, they are not scalable, which means that far obstacles, represented by fewer 3-D points are more prone to be eliminated than closer ones. On the other hand, the space-variant resolution operates to great extent blindly, i.e. in order to reduce computational cost, many pixels are skipped. However, it would be useful to known whose pixels should be skipped and whose should be analysed. This selection process can be carried out by a visual attention mechanism capable of detach obstacle regions from the background.

2.6 Visual Attention Mechanisms

Recent research in the field of mobile robotics have shown the applicability of visual attention mechanisms in order to guide expensive tasks such as object detection and characterisation, reducing the region of interest and, consequently, saving computation time.

Hong et al. [Hong et al., 2002] uses prediction to focus a colour-based detector of puddles and road signs. Prediction, in this case, is no more than collecting laser and colour camera data into a world model and, given the actual position of the robot in the world model and the information previously obtained, estimate which regions of future images shall be analysed.

In a more active way, visual saliency has been used to control the gaze of a humanoid head (e.g. [Vijayakumar et al., 2001], [Orabona et al., 2005], [Morén et al., 2008]), or to detect objects in domestic environments (e.g. [Meger et al., 2008], [Yu et al., 2007]).

Besides object detection, visual saliency has also been used to select strong landmarks for visual self-localisation and mapping in urban environments (e.g. [Newman and Ho, 2005], [Frintrop et al., 2007]).

Except for the first case, all the previous applications are restricted to indoor or urban environments. In the unstructured off-road environments, obstacles are not necessarily the most salient in the image and do not belong to well specified classes of objects.

Chapter 3

Supporting Mechanisms

This chapter summarises the mechanisms that will serve as support for the obstacle detection algorithm, namely the calculation of stereo disparity, the computation of saliency maps and the estimation of the dominant ground plane. Stereo disparity allows the computation of 3-D point clouds, i.e. a three-dimensional representation of the scene captured by the stereo sensor. Saliency maps will be used to guide the detector through regions where obstacles detach more significantly from the background. Finally, ground plane estimation determines the orientation of the dominant ground plane next to the robot location.

3.1 Stereo Vision

The model proposed in this dissertation uses dense 3D point clouds (see Fig. 3.1) in order to detect obstacles. Thus, an efficient method for calculating such point clouds is required. The use of stereo vision was adopted for this purpose.

Briefly, a pair of cameras internally and externally calibrated, and displaced horizontally from one another, gives a right and left image. Both images are then used to find matching elements, i.e. elements in the right image that have high similarities with elements in the left image. Calculating the disparity of the matched elements enables the estimation of their three-dimensional positions.

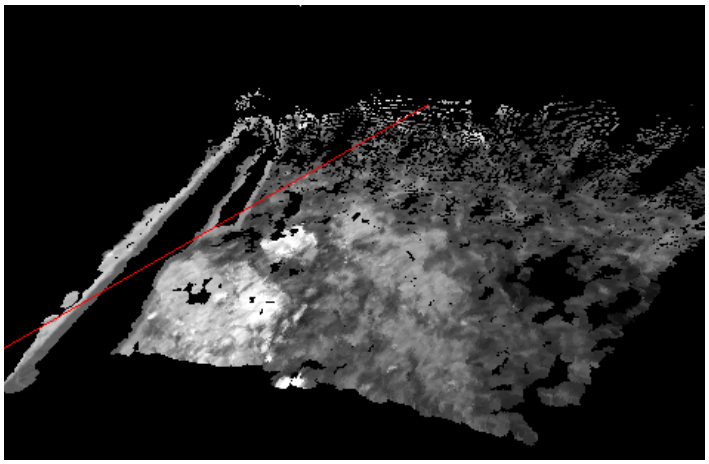


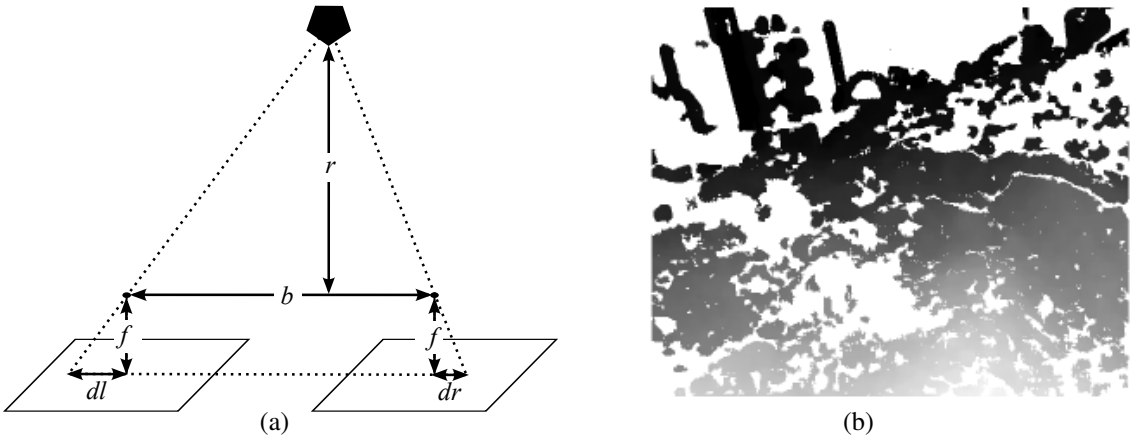
Figure 3.1: Example of a 3D point cloud obtained using the Small Vision System framework [Konolige and Beymer, 2007].

3.1.1 Disparity

Disparity is defined by the difference in image location of an object. In stereo vision, disparity is used to estimate the range of objects captured by the stereo sensor. The distance from the camera to a given object is calculated by triangulation. Fig 3.2(a) exemplifies the process assuming that both images are embedded within the same plane. This can be achieved by precise camera alignment. In order to calculate disparity, it's needed to find the objects location in both images. With this setup, disparity is only observed horizontally, i.e. a point projected within a given row of the left image must project in the same row of the right image, which reduce the search space. Fig 3.2(b) depicts a typical disparity map where pixel intensities are related to computed range.

3.2 Saliency Computation

The following describes the biologically inspired saliency model. It is a specialisation for off-road environments of the one proposed by Itti et al. [Itti et al., 1998]. Let L be the left image, with width w and height h, provided by the stereo vision sensor. To reduce computational cost, saliency is computed on a region of interest (ROI) of L. The ROI is an horizontal strip between rows u and h, where u corresponds to the upper-most row containing more than 100 pixels with



(a) (b)

Figure 3.2: (a) Disparity calculation for an object located at a distance r from a stereo camera with a baseline b and a focal length f. In the left image, the object is projected at a distance dl from the centre of the image and in the right image is projected at a distance dr from the centre. Knowing the disparity value $d = dl - dr$, the range r is given by $r = (b \cdot f)/d$. (b) Disparity map computed with the SVS libraries [Konolige and Beymer, 2007] for the image $\#24$ (see Fig: A.1). The gradient represents the range from nearer points (light gray) to farther ones (black). White pixels represent points with no computed range.

---

an associated depth within the range of interest r. To further reduce computational cost, all image operators are performed over 8-bit images, whose magnitude is clamped to $[0, 255]$ by thresholding.

A dyadic Gaussian pyramid $I(\sigma)$ with six levels $\sigma \in \{0, \ldots, 5\}$ is created from the intensity channel of ROI. The resolution scale of level σ is $1/2^{\sigma}$ times the ROI resolution scale. Intensity is obtained by averaging the three colour channels. Then, four on-off centre-surround intensity feature maps $I^{on-off}(c, s)$ are created, to promote bright objects on dark backgrounds, in addition to four off-on centre-surround intensity feature maps $I^{off-on}(c, s)$, to promote dark objects on bright backgrounds.

On-off centre-surround is performed by across-scale point-by-point subtraction, between a level c with finer scale and a level s with coarser scale (linearly interpolated to the finer resolution), with $(c, s) \in \Omega = \{(2, 4), (2, 5), (3, 4), (3, 5)\}$. Off-on maps are computed the other way around, i.e. subtracting the coarse level from the finer one.

These maps are then combined to produce the intensity conspicuity map,

$$C_I = \sum_{i \in \{on-off, off-on\}} \left(\frac{1}{2} \bigoplus_{(c,s) \in \Omega} I^i(c, s) \right)$$

where the across-scale addition $\bigoplus$ is performed with point-by-point addition of the maps, properly scaled to the resolution of level $\sigma = 3$. Sixteen orientation feature maps, $O(\sigma, \theta)$, are created by convolving levels $\sigma \in \{1, \ldots, 4\}$ with Gabor filters tuned to orientations $\theta \in \{0°, 45°, 90°, 135°\}$. Gabor filters are themselves centre-surround operators and therefore require no across-scale subtraction procedure [Frintrop, 2006]. As before, all orientation feature maps are combined at the resolution of level $\sigma = 3$ in order to create the orientations conspicuity map,

$$C_O = \sum_{\theta \in \{0°,45°,90°,135°\}} \left(\frac{1}{4} \bigoplus_{\sigma \in \{1,\ldots,4\}} O(\sigma, \theta) \right).$$

The saliency map S is obtained by modulating the intensity conspicuity map C_I with the orientations one C_O, $S = \mathcal{M}(\frac{1}{2} \cdot \mathcal{N}(C_I), \frac{1}{2} \cdot \mathcal{N}(C_O))$, where $\mathcal{M}(A, B) = A \cdot sigm(B)$, being $sigm(.)$ the sigmoid operator and $\mathcal{N}(.)$ rescales the provided image's amplitude between $[0, 255]$. Fig. 3.3(a) depicts a saliency map generated by this method.

The proposed saliency model is essentially based on the model proposed by [Itti et al., 1998] but considering both on-off and off-on feature channels separately, which has been shown to yield better results [Frintrop, 2006]. Still, two major innovations are present in the proposed model. First, the normalisation operator $\mathcal{N}(.)$ does not try to promote maps according to their number of activity peaks, as typically done. The promotion of some maps over others according to activity peaks showed to provide poor results in the tasks herein considered. This is because spatially frequent objects, which are inhibited in typical saliency applications, may be obstacles for the robot, and thus must also be attended.

A narrow trail may be conspicuous in the intensity channel if it is, for instance, surrounded by dense and tall vegetation. This contradicts the goal of making obstacles salient, rather than the background, which is why saliency is computed from the weighted product of the conspicuity maps [Hwang et al., 2009] rather than their addition [Itti et al., 1998], [Frintrop, 2006]. It focus the saliency on regions where orientations are strong, i.e. small objects, borders of objects, and on entire objects if considerably textured, which is the most often case off-road.

3.3 Ground Plane Estimation

The used solution to modulate the hypothesis-generation step of a conventional RANSAC [Fischler and Bolles, 1981] robust estimation procedure is composed of the following seven steps.

1. pick randomly a set R of three non-collinear 3-D points within range r, and generate its corresponding ground plane hypothesis, h_R, with some straightforward geometry. Points are considered non-collinear if the area of the triangle defined by them is above a threshold t.

2. the score of the plane hypothesis is the cardinality of the set of its inliers, $score(h_R) = |P_{h_R}|$. An inlier, $j \in P_{h_R}$, is a 3-D point whose distance to plane h_R, $d(j, h_R)$, is smaller than a given threshold d_{plane}.

3. repeat steps 1 and 2 until n_{hypo} hypotheses, composing a set H, have been generated.

4. select for refinement, from H, the hypothesis with the highest score:

   $b = \arg\max_{h \in H} score(h)$.

5. compute b', which is a refined version of b, by fitting the inliers set of the latter, P_b. This fitting is done with weighted least-squares orthogonal regression, via the well known Singular Valued Decomposition (SVD) technique. The weight w_q of an inlier $q \in P_b$ is given by $w_q = 1 - \frac{d(q,b)}{d_{plane}}$. That is, the farther q is from b, the less it weights in the fitting process. Compute the inliers set of b', $P_{b'}$, and substitute the current best ground plane estimate by the refined one, i.e. make $b = b'$ and $P_b = P_{b'}$.

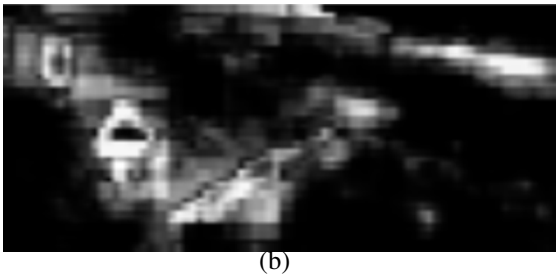
6. iterate step 5 until $|P_b|$ becomes constant across iterations or a maximum number of iterations, m_{refit}, is reached.

7. take b as the ground plane estimate.

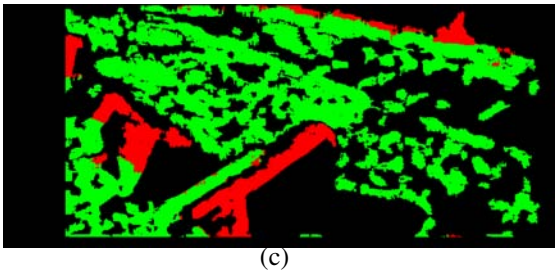
To take saliency into account, each 3-D point p selected to build an hypothesis in step 1 must pass a second verification step. This step reduces the chances of selecting p proportionally to the *local saliency* of its projected pixel p'. The underlying empirical assumption is that saliency is positively correlated with the presence of obstacles. Preferring non-salient points thus raises the chances of selecting ground pixels (see Fig 3.3(e)).

Formally, a 3-D point p is rejected in the second verification step if $s_{p'} > \frac{P(x)}{\alpha \cdot n_l}$, where: $n_l \in [0, 1]$ is the number of pixels with saliency below a given threshold l normalised by the total number of pixels; *local saliency* $s_{p'} \in [0, 255]$ is the maximum saliency within a given sub-sampled chess-like squared neighbourhood of p', with size $g \cdot n_l$, being g the empirically defined maximum size; $P(x) \in [0, 255]$ represents samples from an uniform distribution; and α is an empirically defined scaling factor. The goal of using the normalised number of pixels with a saliency value under a given threshold is to allow the system to progressively fall-back to a non-modulated procedure as saliency reduces its discriminative power, i.e. it is too spread in space. This happens for instance in too textured terrains, in which the sampling procedure is too constrained as a result of the saliency map's cluttering.

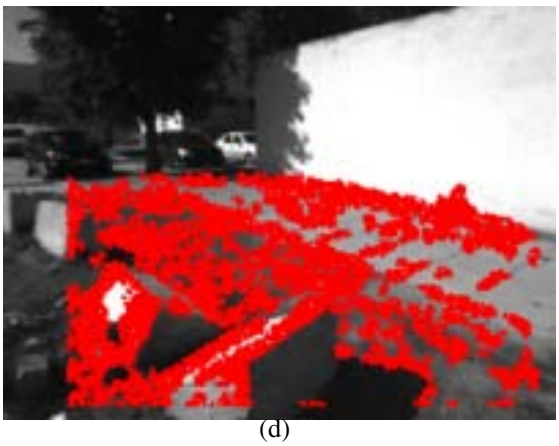
(a)



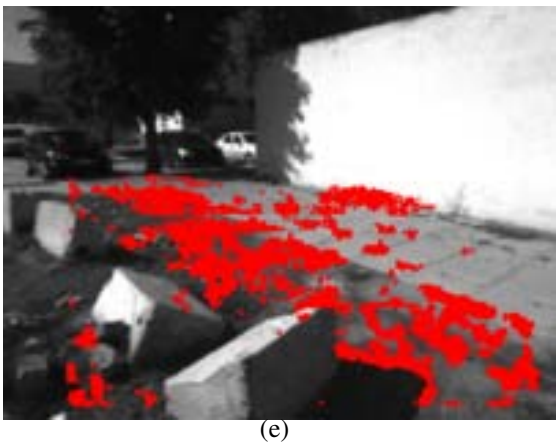
(b)



(c)



(d)



(e)

Figure 3.3: Saliency computation and ground-plane estimation results for a range of 10m (a): left image obtained from a stereo camera; (b): saliency map generated from (a). (c): pixel classification results based on the computed ground-plane (red, green and black pixels correspond to obstacles, ground, and points without computed depth, respectively). (d): pixels (red) corresponding to 5000 3-D points randomly sampled for ground-plane estimation without saliency modulation. (e): same as (d) but with saliency modulation.

Chapter 4

Obstacle Detection Core

This chapter presents a full obstacle detector suited for autonomous mobile robots operating in rough and unstructured outdoor environments, and equipped with a stereo vision sensor. Since, in such environments, little assumptions can be made about the morphology of a typical obstacle, Manduchi et al. [Manduchi et al., 2005] proposed a method for defining local obstacle points based on the relationship between pairs of 3-D data points. Such definition, described in section 4.1 serves as a basis for the work presented in this dissertation in which a space-variant resolution mechanism modulated by saliency, described in section 4.2, and a voting filter, described in section 4.3 are used in order to reduce both computation time and noise sensitivity. Section 4.4 describes a method to isolate the detected objects and discard the ones that are small enough not to be considered as obstacles.

4.1 Obstacle Definition

As previously mentioned, cross-country environments usually don't have large planar surfaces where obstacles just pop out from the ground. In this case, having a model that classify obstacles based on their heights according to the global ground plane is unreliable, thus a more comprehensive approach is needed. In literature a suitable definition of obstacle in typical off-road environments can be found in [Manduchi et al., 2005]. In their model, from now on

denominated Original Obstacle Detector (OOD), obstacles are defined as follows:

Definition 1: Two 3-D points $p_a = (x_a, y_a, z_a)$ and $p_b = (x_b, y_b, z_b)$ are considered *compatible* with each other if the following conditions are both met:

1. $H_{min} < |y_b - y_a| < H_{max}$
2. $\dfrac{|y_b - y_a|}{\|p_b - p_a\|} > \sin\theta$

where, θ is the minimum slope a surface must have to be considered as an obstacle, H_{min} is the minimum height an object must have to be considered an obstacle, and H_{max} is the maximum allowed height between two points to be considered compatible with each other.

Definition 2: Two 3-D points $p_a = (x_a, y_a, z_a)$ and $p_b = (x_b, y_b, z_b)$ pertain to the same obstacle if at least one of the following conditions is met:

1. p_a and p_b are compatible with each other;
2. p_a and p_b are linked by a chain of compatible point pairs.

For a better understanding of how an obstacle can be specified by these conditions, the compatibility relationship expressed in *Definition 1* may be interpreted geometrically as follows: considering a 3-D point p, its compatible points are those who are spatially positioned inside two truncated cones C_U and C_L with vertex in p, both oriented vertically (i.e. along the y-axis) and symmetrical between each other, with an aperture angle of $(\pi - 2\theta)$ and limited by $y = H_{min}$ and $y = H_{max}$ (see Fig. 4.1).

Note that θ and H_{min} are closely related with the robot's technical and physical specifications. In fact, θ can be described as the maximum inclination a surface must have to be climbable by the robot, while H_{min} is actually the height of the free space between the ground and the robot where small objects may be stepped over without harming the robot's structure. On the other hand, H_{max} is more related with the quality of the computed stereo as a sparse 3-D

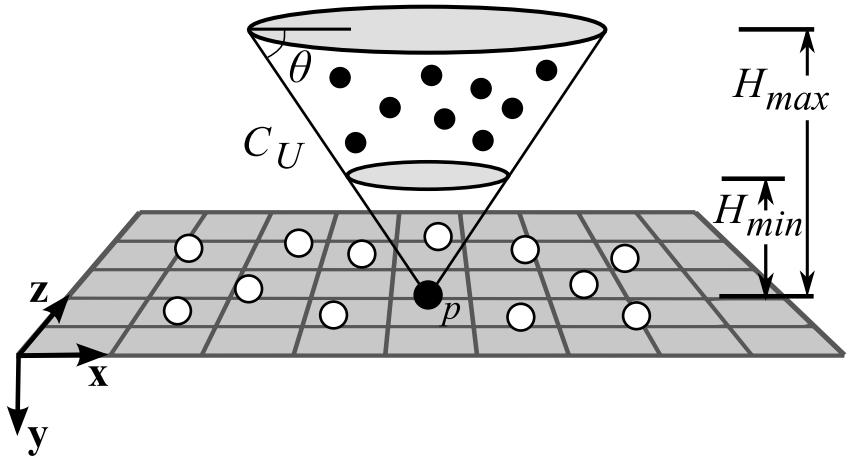


Figure 4.1: Geometric interpretation of the base model [Manduchi et al., 2005] where filled and unfilled circles represent points that are compatible and incompatible, respectively, with p. For readability reasons, C_L is not represented in the figure.

point cloud result on the obstacle points being located far apart from each other which, for a low H_{max} may result in over-segmentation or even having obstacle points not to be considered as such. However, a H_{max} excessively high have the consequence of increasing substantially the number of points to be analysed and, consequently, increase the computation time of the OD algorithm, as will be seen in the next section.

4.2 OD Algorithm

The previous section described the way two 3-D data points must be related in order to be considered as obstacle points.

In this section, we will see how the process of checking the set of 3-D points, given by the stereo sensor, is made. The 3-D point cloud is computed as in [Konolige and Beymer, 2007].

As previously seen, the task of finding obstacles in a 3-D point cloud implies looking at pairs of compatible pixels. Rather than looking to all the possible pairs of points from the point cloud, which would result in a number of $N^2 - N$ tests, with N the total number of computed 3-D points, Manduchi et al. [Manduchi et al., 2005] demonstrated that, actually, only a reduced subset of pixels is needed. That is, being p' the projection of the 3-D point p onto the image plane, the two truncated cones, C_U and C_L, that involves the 3-D points compatible with p must

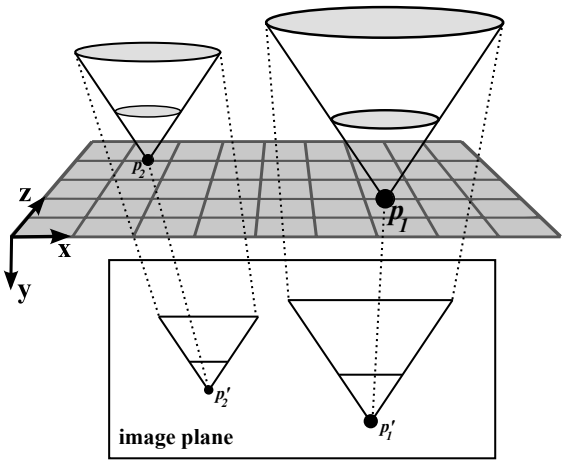


Figure 4.2: Projection of the truncated cones C_U onto the image plane, resulting in the truncated triangles C'_U.

project onto two truncated triangles C'_U and C'_L in the same image plane and with vertex in p' (see Fig. 4.2). Also, has been demonstrated that, by scanning the pixels in the range image starting from bottom to it's top and from left to right, it suffices to consider only the upper truncated triangle C'_U to efficiently detect obstacle points. Bearing this in mind, the task of checking for the compatibility points of p means applying the compatibility test to the points projected inside the C'_U relative to p'. If compatible points are found, all of them as well as p' are labelled as obstacle points.

When projecting the truncated cone onto the image plane, the correspondent truncated triangle's height is given by $\frac{H_{max} f}{p_z}$, where f is the camera's focal length, and base approximately equal to $\frac{2H_{max} f}{\tan\theta_{max} p_z \cos v}$, with $v = \arctan\frac{p_x}{p_z}$.

4.2.1 Tilt-Roll Compensation

All the above geometrical considerations assume that the camera is not tilted nor rolled in respect to the ground-plane. This is an obviously unbearable constraint for all-terrain robots. In the original approach [Manduchi et al., 2005], the authors compensate small variations on the camera's attitude by overestimating the truncated triangle size. This approximation inevitably increases the computational cost, and thus should be discarded.

The following proposes a more exact way of compensating for tilt and roll, whatever their

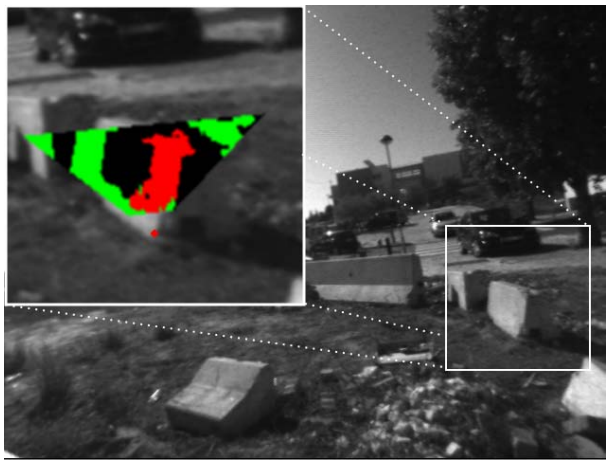


Figure 4.3: Compatibility test on a real image. The zoomed image depicts the results of the compatibility test regarding the pixel in the truncated cone's vertex. Red, green and black pixels overlay on the zoomed image correspond to the compatible, incompatible and without computed range points, respectively. Incompatible points show up in the truncated triangle due to perceptive. Note the rotation of the truncated triangle, which is a result of the compensation for the camera's roll angle, with respect to the ground-plane.

magnitude. First, the dominant plane, assumed to be the ground one, is computed according to the method proposed in section 3.3. Then, the 3-D point cloud is rotated in order to align the world's reference frame, given by the normal to the computed ground-plane, with the camera's reference frame. The projected truncated triangle is also rotated accordingly, by projecting to the image plane the normal vector to the estimated ground plane so as the truncated triangle be oriented along the same normal vector (see Fig. 4.3. This way, the pixels scanned in the image plane correspond to the 3-D points that are actually encompassed by the correspondent truncated cone.

4.2.2 Space-Variant Resolution

Despite the advantages of using a truncated triangle in order to greatly reduce the compatibility tests, the computational cost of the method remains too expensive. Space-variant resolution is thus essential to further reduce the computational load. A successful model implementing it, from now on denominated of Extended Obstacle Detector (EOD) [Santana et al., 2008], can be summarised as follows.

For a given pixel p' (sequentially sampled from $1/n$ of the full resolution, from the image's bottom to its top and from left to right), its C'_U is first scanned for compatible points with $1/m$ of the maximum resolution in a chess-like pattern, where $m > n$. If a compatible point with p' is found, C'_U is rescanned as usual with $1/n$ of the maximum resolution (see Fig. 4.5(c) for a graphical interpretation of this method). Then, finished the scanning procedure, the full resolution is recovered with the following region growing method. For each obstacle pixel p', all its neighbours within a distance w in the image plane are also labelled obstacles if their corresponding 3-D points are closer than d from p.

Despite its considerably achievements in reducing computational cost [Santana et al., 2008], the EOD method operates to great extent blindly. That is, in order to reduce computational cost n and m are increased and consequently the number of skipped pixels as well. Visual saliency (section 3.2) is known to be an important asset in many search tasks and thus it is a powerful candidate to guide the space-variant resolution mechanism in an informed way. Bearing this in mind, a Saliency based Obstacle Detector (SalOD) [Santana et al., 2009] is herein proposed.

Rather than applying the compatibility test along the whole scan, as performed by the OOD [Manduchi et al., 2005], a pixel p', sampled from $1/n$ of the full resolution is tested iff:

1. n_{slide} consecutive pixels in the same row of p' have not been tested so far; or
2. n consecutive pixels, after a pixel that has been tested and labelled as obstacle in the same row of p', have not been tested so far; or
3. there is a 10% increment between the *local saliency* of p' and the one of its preceding scanned pixel, provided that both share the same row; or
4. the last scanned pixel had no computed 3-D, and hence no information could be obtained from it.

Local saliency is computed by taking the maximum saliency from the set of pixels within the same column of p', including itself, and contained in its truncated triangle. This diminishes

the effects of poor light conditions, which in some situations make the top of large obstacles to appear more salient than their bottom. If only the saliency of p' were used instead, many object bottom pixels would be inappropriately skipped.

Roughly speaking, the described process slides along rows for n_{slide} pixels unless an increase in saliency is observed, or something has been detected. While sliding, the compatibility test is not performed, and consequently computational cost is saved and the chances of generating false positives is reduced. However, some additional features can be added in order to enhance the both system's performance and accuracy. An extended version for the SalOD is also proposed and coined as Extended Saliency based Obstacle Detector (ESalOD).

First, in the extended version of the algorithm, p' is sampled from the full resolution input image, so any pixel in a scanning row as a non-null probability of being assessed and, therefore be considered as obstacle, which improve the reliability of the method.

Secondly, rather than having a fixed n_{slide}, a dynamical one is used instead,

$$n_{slide} = \begin{cases} k \cdot n & k < \frac{n_{max}}{n} \\ n_{max} & otherwise \end{cases}$$

where k is the number of consecutive times a pixel was tested and labelled as non-obstacle, since the last time a pixel was considered an obstacle, and n_{max} is an empirically defined scalar. The application of this method results in skipping progressively more pixels as obstacles are not found. Since the sliding process start with small jumps, the chances of failing to detect the borders of objects are reduced.

Additionally, every time a pixel p' is labelled as obstacle, the saliency of all pixels in C_U, i.e. within its truncated triangle, are increased in 10% (empirically defined). This mechanism is used to reinforce the presence of an obstacle by increasing the chances of a subsequent analysis of all pixels associated with it. This is an atypical interaction between the task-specific detector and the saliency map, as it allows the detection results to modulate the saliency map, which is in turn guiding the detector. Typically (e.g. [Itti et al., 1998], [Frintrop, 2006]), the influence is unidirectional and flows from the saliency map to the task-specific detector.

Also, instead of analysing every rows that are multiple of n, as in the first approach (SalOD), the extended (ESalOD) instead skips $n+k$ rows, where k is incremented every time an analysed row does not contain any obstacle pixel. Whenever an obstacle pixel is found k is zeroed. This procedure, which mimics to some extent the columns sliding process, is extremely useful in the reduction of the computation load in environments with few obstacles, or when obstacles are mostly in the far-field. Since the truncated triangle for points in the near-field is quite large, skipping image bottom rows when no obstacle is found there, greatly reduces the computational cost.

A graphical representation of the space-variant resolution herein proposed is depicted in Fig. 4.5(d).

4.3 Voting Filter

In addition to performance, both accuracy and robustness are likewise important. These two additional ingredients, in the form of voting filters, are integral part of EOD. These filters intend to diminish the effects caused by artifacts introduced during the 3-D reconstruction process. In this formulation, a given point p is said to cast a number of *votes* equal to the number of compatible points with p, and is also said to be *voted* by those points whose upper truncated cone C_U include p (see Fig. 4.4). Only points that cast more than min_{votes} votes *and* are voted more than min_{voted} times, simultaneously, are considered obstacles. Thus, rather than the one-to-one mapping considered in the OOD, where compatibility is sufficient to define an obstacle, in EOD a many-to-many mapping is necessary. This naturally results in higher levels of robustness.

A careful observation reveals that the *and* operation (see above) is too strong, and consequently influences negatively the true positive rate. Empirical observations led to use the conjunction operator instead, as it has been shown to foster parametrisation flexibility. This slight change allows reducing the false positives rate by pushing further the voting thresholds, with minimum impact on the true positives.

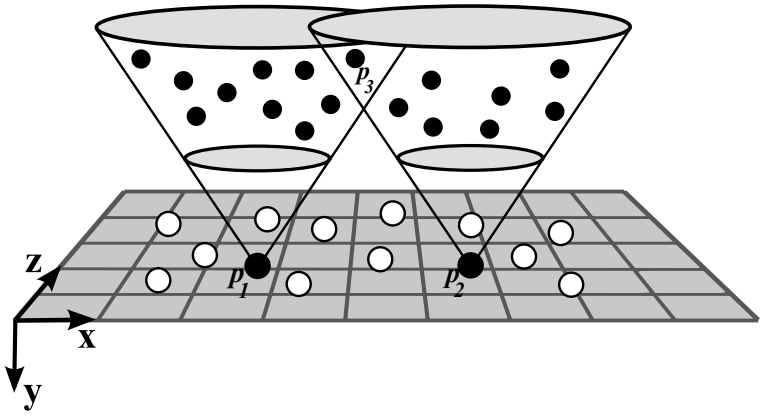


Figure 4.4: Voting mechanism [Santana et al., 2008]. Considering that, for the sake of this example, only p_1 and p_2 were tested for compatibility, p_1 is said to cast 10 votes whereas p_3 is said to be voted 2 times.

A mechanism to normalise the number of votes associated to each point, according to the theoretically maximum number of possible *votes*, is also missing in EOD. The relevance of this issue stems from the fact that farther obstacles are represented by fewer pixels than closer obstacles. In the algorithm herein proposed, the amount of $votes$ and $voted$ is normalised as follows.

- Let p be a 3-D point and p' its projection in the input image.
- Let A'_p be the set of pixels, with computed range, falling inside the truncated triangle of p'.
- Let R_p be the set of points being voted by p.
- Let B'_p be the set of pixels, with computed range, whose truncated triangles encompass p'.
- Let S_p be the set of points voting in p.

Now, rather than comparing min_{votes} and min_{voted} against R_p and S_p, respectively, as for the OOD case, the comparison in is performed against the normalised scalars, $\frac{|R_p|}{|A'_p|}$ and $\frac{|S_p|}{|B'_p|}$, respectively. For the current implementation, the *votes threshold* is aggregated in a single parameter, $v : v < \frac{|R_p|}{|A'_p|} \vee v < \frac{|S_p|}{|B'_p|}$.

4.4 Obstacle Segmentation

Although the focus of this work is centred in the correct and quick detection of obstacles, an useful extra can be embedded in the algorithm in order to identify distinct obstacles in the image. As being said in section 4.1, two 3-D points p_a and p_b pertain to the same obstacle if they are compatible with each other or linked by a chain of compatible point pairs (*Definition 2*). Bering this in mind, we may segment obstacles by applying an Union-Find algorithm to the *points graph* where compatible points are linked with each other [Manduchi et al., 2005]. Briefly, the algorithm can be explain as follows:

1. Firstly, label all points as *non-obstacles*;

2. When checking the upper truncated triangle, C'_u, of a point p', determine the set S'_p of compatible points with p'.

3. If S'_p is not empty, add p' to the set and find the point within S'_p with the smallest label. If all points are labelled as *non-obstacles*, assign a new label to all points in S'_p, otherwise, assign the smallest label found to all points in S'_p, keeping track of label exchanges in an equivalence table;

4. After all points are checked, make a second pass relabelling all points with their smallest equivalent label.

Applying this connected component labelling to the 3-D point cloud instead of applying it onto the image plane have the advantage of properly distinguish spatially linked obstacles even if their projections presents unconnected regions. On the other and, connected regions in the image plane may correspond to more than one obstacle which is also considered by this approach (the results of this procedure may be seen on Fig. 4.5 (c) and (g)).

4.4.1 Area Filter

Once all obstacles are isolated, we may apply a simple rule in order to eliminate those whose number of pixels are low enough not to be considered as obstacles. In [Manduchi et al., 2005] a *shape-based validation* is proposed to achieve this goal, taking into account some 3-D attributes of the obstacles, as the volume of the 3-D bounding box around an obstacle, the average and maximum slope of an obstacle and also its height. However, determining such attributes correspond to an unnecessary addition of computation. Instead of considering such attributes, an obstacle point p is considered non-obstacle if $L_p < 100 \times \frac{a}{p_z^2}$, where L_p is the total number of points with the same label as p and a is an empirically defined scalar. Although this approach may not be very realistic in the point of view of obstacles' morphologies, due to perspective, experimental results prove its usefulness when in conjunction with the *Voting Filter* (section 4.3). In fact, the *Voting Filter* by itself remove most of the noise but if excessively exploited, may weaken real obstacles. In order to achieve an acceptable trade-off, some punctual noise may remain detected as obstacle. Fortunately, the remaining noise is typically sparse and with low density which is easily removed by this *Area Filter*.

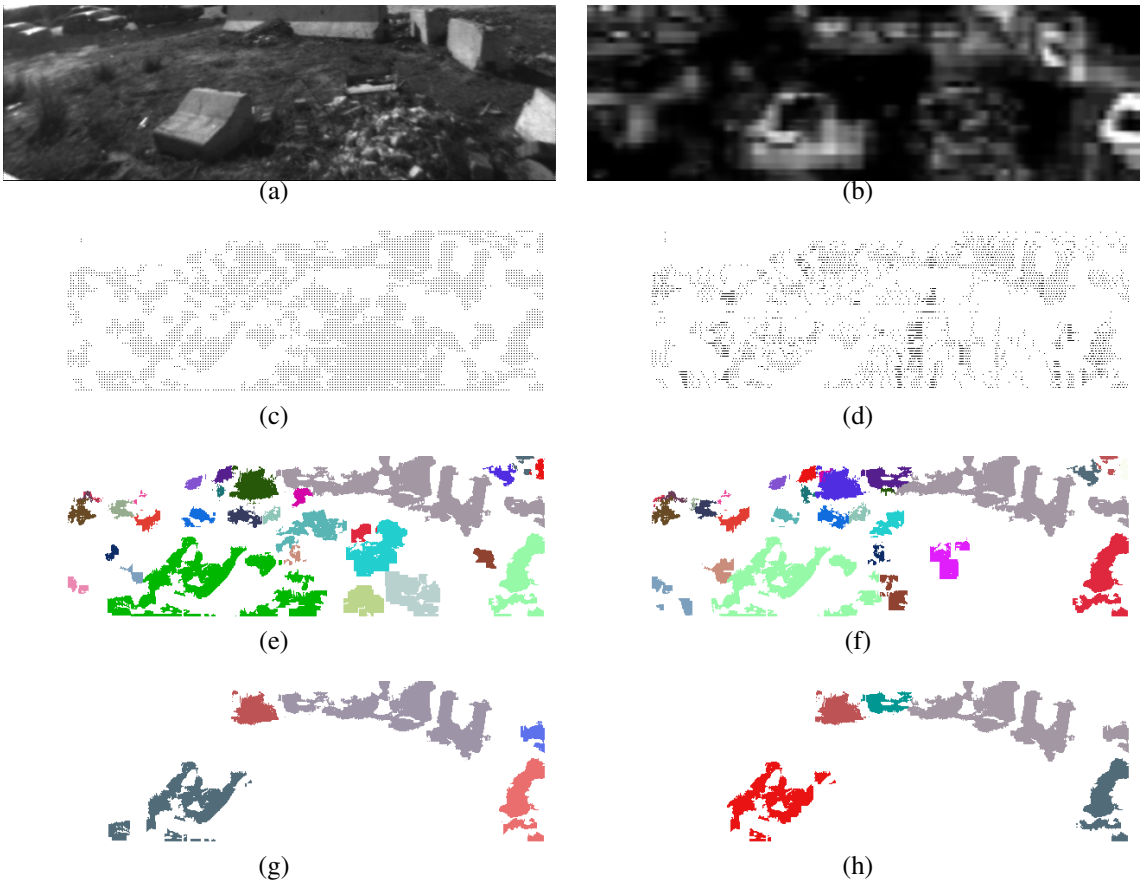


Figure 4.5: Obstacle detection results. (a): Left image acquired from a stereo camera and cropped to range $r = 10$m. (b): Saliency map computed from (a). (c-d) Graphical representation of the space-variant resolution for (a), using, in (c) the original method (EOD) [Santana et al., 2008] and in (d) the saliency modulated method (ESalOD) proposed in this dissertation. White pixels correspond to points that have been skipped by the detector due to the lack of saliency or computed range. Black pixels correspond to points that have been analysed. Note that, instead of analysing equidistant pixels (c), the method herein proposed (d) focus the analysis on salient regions and regions where obstacles are being detected, skipping more pixels, otherwise. This mechanism result in reduced computation, as less analysis are being made, and increased robustness, as analysis are focused in regions of interest. (e-h): Detected segments using the EOD's space-variant resolution and the ESalOD's space-variant resolution with the Voting Filter (section: 4.3) and Area Filter (section: 4.4.1) parameterised as follows: (e) EOD with $v = 0$ and $a = 0$ (f) ESalOD with $v = 0$ and $a = 0$ (g) EOD with $v = 20$ and $a = 25$ (h) ESalOD with $v = 20$ and $a = 25$. Note that, with the filters turned off, the EOD method is significantly more noisy. Also note the advantage of using Votes and Area Filters for noise reduction.

Chapter 5

Hybrid Obstacle Detector

As stated in [Rankin et al., 2005], the variety of objects with different dominant techniques presented in off-road environments requires the use of different detection techniques in parallel for a complete detection of all possible obstacles.

This chapter presents a new obstacle detection architecture, represented in Fig. 5.1, that integrates two different techniques in an efficient way, where saliency is used throughout the system in order to reduce its computational cost and augment its robustness. Briefly, a coarser and faster obstacle detector is used to detect large obstacles and to focus the a finer and slower one on regions of the environment where small obstacles, and consequently harder to detect, may be present. Detectors complementary role aims at the development of a system that properly trade-offs between computational cost and detection accuracy.

5.1 Architecture for Hybrid Obstacle Detection

The following describes the system in a nutshell. First, a stereo vision sensor provides two images, one obtained from the left camera and another from the right one. The saliency map of the left image is computed, and a stereo processing step is carried out in order to provide a dense 3-D points cloud [Konolige and Beymer, 2007]. Saliency information (see section 3.2) is then used to guide a hypothesise-test method for the ground plane estimation step (see section 3.3).

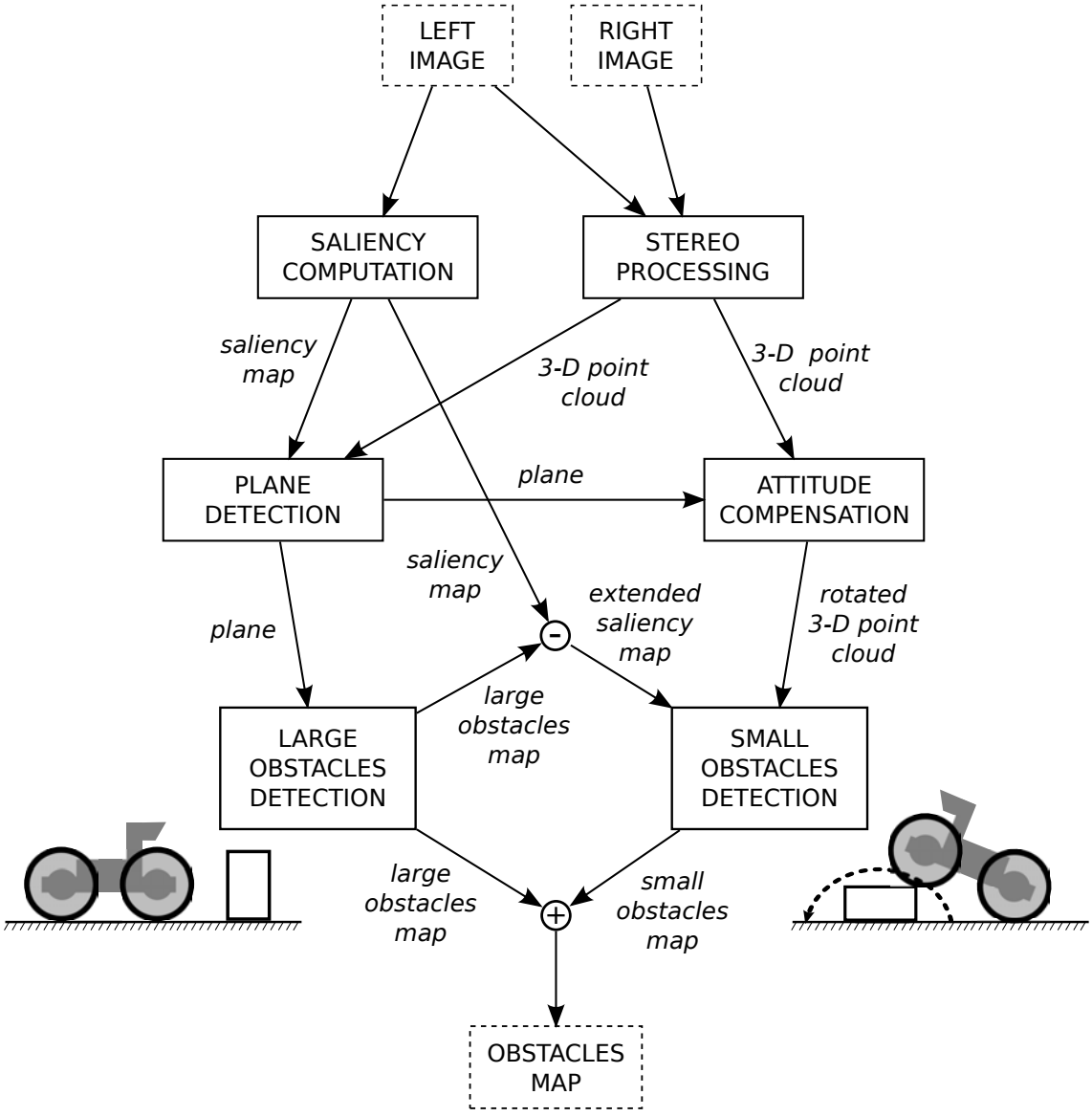
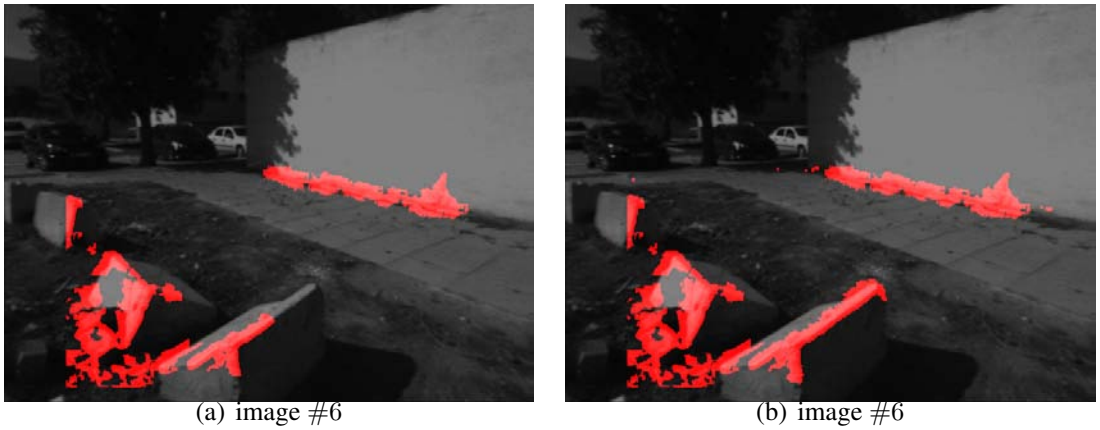
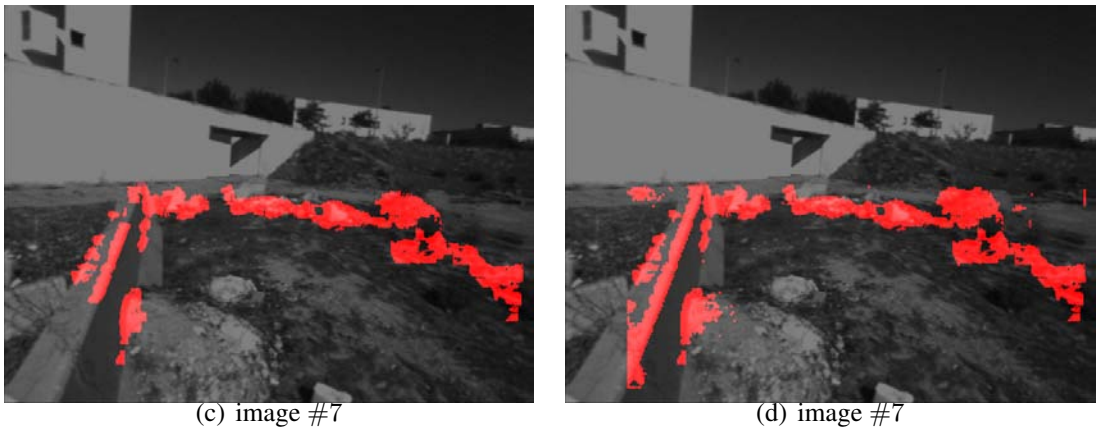


Figure 5.1: Architecture for hybrid obstacle detection.



(a) image #6 (b) image #6



(c) image #7 (d) image #7

Figure 5.2: Hybrid Obstacle Detector results. In the left column are depicted the results for the small obstacles detector alone (as described in chapter 4 and set for its best parameterisation) and in the right column are depicted the results of hybridisation. Note the difficulty for the cone-based detector to classify the top of large homogeneous obstacles, which is resolved by its fusion with the plane based detector.

Subsequently, a large obstacles map is obtained by checking which 3-D points are considerably above or below the ground plane. As the camera's optical axis, normally, isn't parallel to the ground plane, the 3-D points are rotated according to the estimated ground plane and thus compensating for the robot attitude. The large obstacles map is then subtracted from the saliency one in order to focus the accurate obstacle detector (see Chapter 4) on areas that have not been already analysed. Finally, the small and large obstacles maps are merged in order to produce the final obstacles map.

Obstacle detection in all-terrain requires some adaptations to the process of computing visual saliency. The adapted model (see section 3.2) uses the orientation of local patches as

the main information source to assess the conspicuity of an image. Although this procedure showed to reduce the false positives rate in all-terrain, it fails to pop-out large obstacles, where the presence of shadows reduces their texture and consequently any orientation information therein. That is, only the top of those obstacles gets notorious, which inevitably affects the detection process. This is a problem the saliency-based small obstacles detector faces, but that the coarser one can easily overcome due to the fact that it is not guided by saliency.

Fig. 5.2 depicts an example of the results obtained with this method, presenting the complementary role of both detectors.

5.2 Cost Map Representation

The obstacle detector developed for this dissertation was made to detect discrete obstacles which, as stated in [Rankin et al., 2005], it's sufficient for uncluttered terrains where the free space is equally traversable. However, in the architecture herein presented, the accurate obstacle detector described previously (see chapter 4) can be used for detection of small protuberances in the terrain, i.e. small obstacles, that can be traversed by the robot by stepping it over. This small obstacles are better represented with an associated cost of traversability, either because the motors must push harder or the terrain cause significant vibration.

Bearing this in mind, the accurate obstacle detector may be reinterpreted and readjusted in order to detect such surfaces and represent them in terms of their traversability cost. First, one can lower the minimum values of height, H_{min}, and slope, θ. Second, reinterpret the *Voting Mechanism* in a way that, instead of functioning as a binary classification process (obstacle/non-obstacle), it is used for calculating the traversability cost. Finally, each obstacle point will have an associated level of cost, given by the result of the adapted *Voting Mechanism*.

In its original form (see section 4.3) each compatible point contributes with a vote. In this adaptation, each vote is weighted by the relationship between the pairs of compatible points in terms of relative position. The logical idea behind this is that short and less inclined surfaces are less expensive to traverse than taller and more inclined ones.

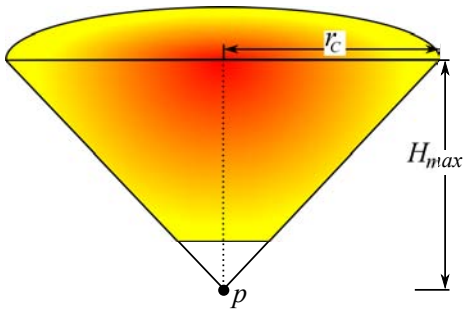


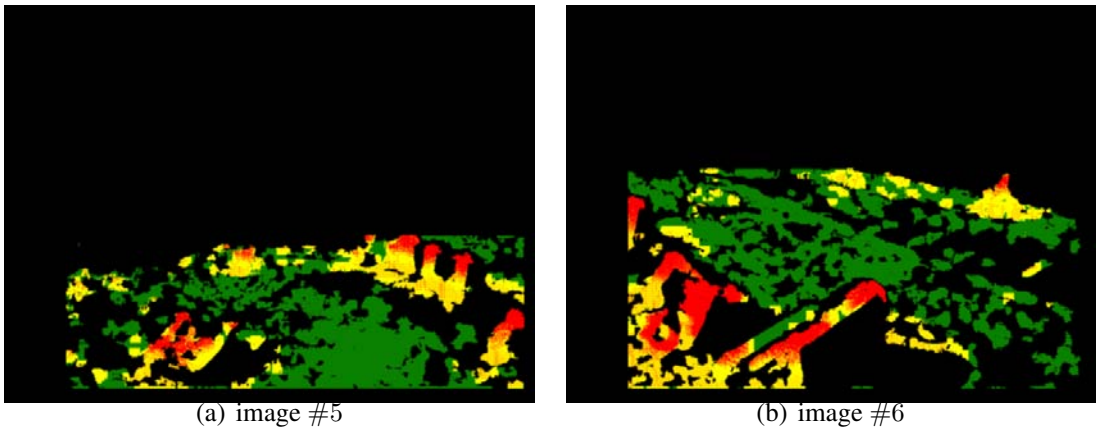
Figure 5.3: Graphical representation of the weighted votes. The gradient represent the vote amount v_a within a truncated cone from the maximum value (red) to the minimum value (yellow). For readability purposes, the cone was chopped by half.

The weighted votes may hence be described as follows. Let p_a and p_b be two 3-D data points compatible with each other (see section 4.1 for details on compatibility). The vote amount v_a each one will give to the other is the result of:

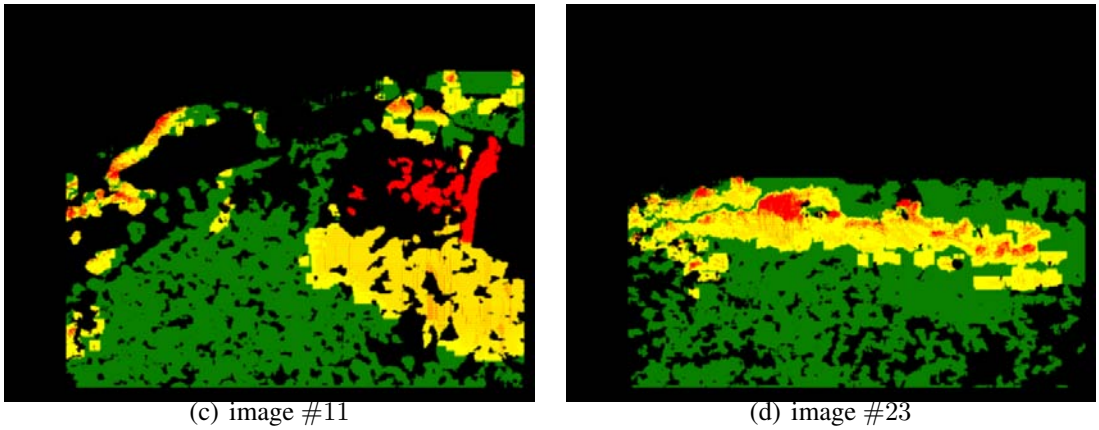
$$v_a = w_1 \cdot \left(1 - \frac{\sqrt{(x_a - x_b)^2 + (z_a - z_b)^2}}{r_C}\right) + w_2 \cdot \left(\frac{|y_a - y_b|}{H_{max}}\right)$$

where r_C is the base radius of the truncated cone C_U and w_1 and w_2 are empirically defined scalars. The practical effect of this equation (see Fig. 5.3) is that when applying the compatibility test to a point p, vote amount is maximum for a point located in the top centre of the truncated cone, and varies inversely with the distance from that point. Fig. 5.4 depicts the generated cost maps for a set of images.

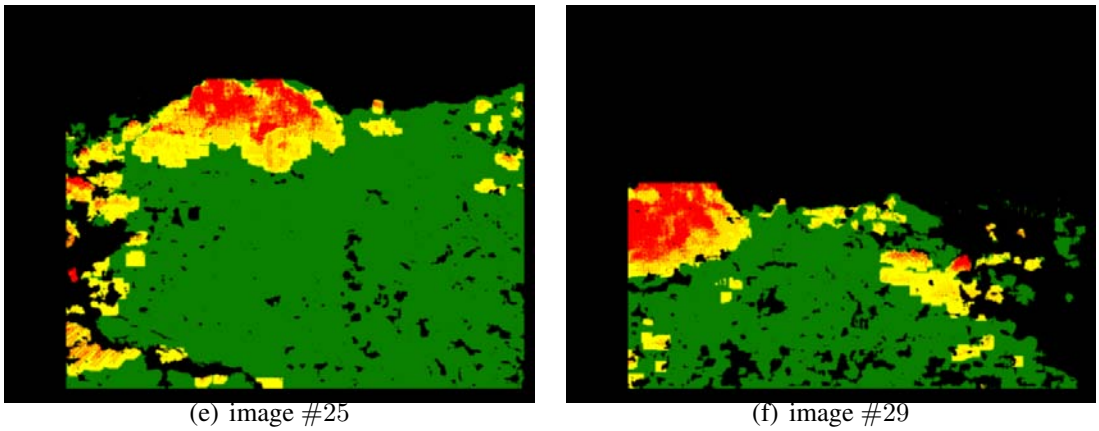
In this model, obstacles detected by the small obstacle detector are represented in terms of its traversability cost while the obstacles detected by the large obstacle detector are logically represented as traversable or non-traversable. Unfortunately, due to schedule constraints, this dissertation does not include rigorous analysis or extensive experimental tests for this model. Nevertheless, the model offer great perspectives for future work.



(a) image #5 (b) image #6



(c) image #11 (d) image #23



(e) image #25 (f) image #29

Figure 5.4: Cost map results for a subset of images obtained from the dataset used for experimental tests (Fig. A.1). Traversability cost is represented as a gradient from the maximum level (red) to the minimum (yellow). Green pixels correspond to free-space (i.e. without cost of traversability) and black pixels correspond to points with no computed range. In order to isolate the contribution of the weight votes, obstacles classified by the large obstacle detector are not represented. Note that, in spite of the votes and area filters are turned off, much of the ground noise is still eliminated. This happens because the points that cause such noise have typically low values for v_a and consequently, in the normalisation process (see section 4.3 for details) their casted votes are approximated to zero, thus being classified as free-space.

Chapter 6

Experimental Results

In order to validate the developed obstacle detector, a series of experiments over the different models (OOD, EOD, and the herein proposed SalOD and ESalOD) were carried out. All experiments used a dataset composed by a set of 36 stereo 640×480 image pairs (see Fig. A.1) acquired with a 9cm baseline Videre Design STOC sensor, at an approximate height of 1.5m. Due to poor light conditions and blur induced by motion, the acquired low contrast images generate noisy 3-D point clouds. These are stringent conditions but quite realistic for outdoor robots. The images have been hand-labelled (obstacle/non-obstacle pixels) for ground truth (see Fig. A.3).

The libraries of Small Vision System (SVS) [Konolige and Beymer, 2007] and OpenCV framework [Bradski and Kaehler, 2008] were used for stereo and low-level computer vision routines, respectively. Tests where made on a Centrino Dual Core 2GHz. When nothing is said otherwise, the ESalOD model has been parametrised for the best performance, $H_{min} = 0.1$m, $H_{min} = 0.4$m, $\theta = 40°$, $(n \times m) = (3 \times 6)$, $a = 25$, $v = 20$, $n_{max} = 30$, $d = 0.4$m, $w = 8$, $r_{min} = 1$m, $r_{max} = 10$m, $n_{hypo} = 500$. Additional parameters for saliency computation and ground plane estimation have been set to their default values: $r = 10$m, $t = 100$, $d_{plane} = 0.15$m, $\alpha = 4$ and $g = 150$ [Santana et al., 2009].

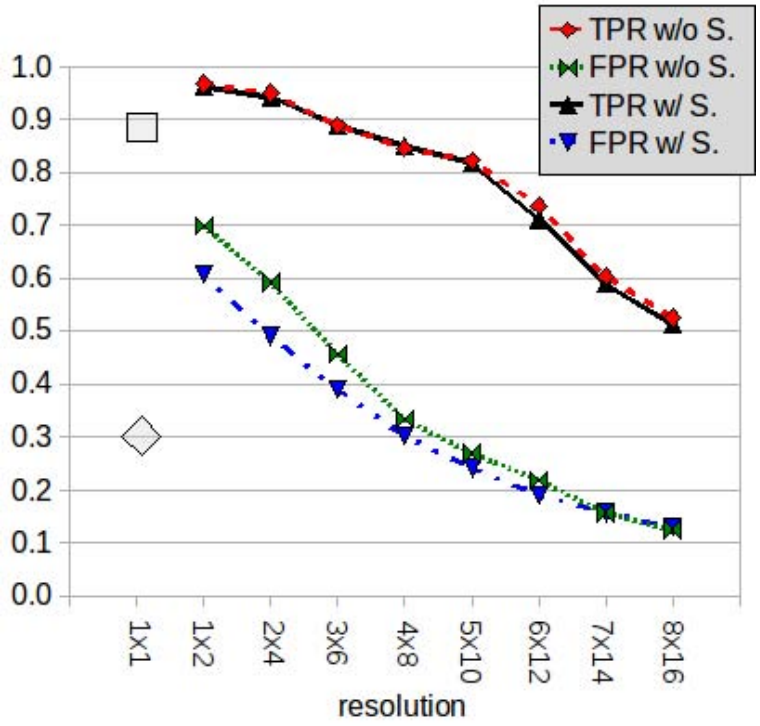


Figure 6.1: Base resolution selection.

6.1 Base Resolution Selection

Defining a proper base resolution for the algorithm to operate is very important in order to allow computational cost to be kept within reasonable levels, without harming the True Positive Rate (TPR).

Fig. 6.1 show the result of testing the system with several base resolutions. The graph plots the average of the True Positive Rate (TPR) and False Positive Rate (FPR) (see Fig. A.5 for a graphical representation of this correlation) of the ESalOD with and without the use of saliency related mechanisms, over all images in the dataset, for a given base resolution. In either case and in order to isolate the impact of saliency in the space-variant resolution mechanism, the voting and area filters were turned off, $v = 0$ and $a = 0$. The grey square and diamond represent the TPR and FPR of the OOD, respectively. Resolution 3×6 has been selected as it is the rightmost, i.e. with lowest resolution, having a TPR similar to the one of OOD. The general increment of both TPR and FPR, when compared to the ones of the ODD, are due to the use of

region growing. This allows selecting lower resolutions without losing TPR. As expected, these plots also confirm that the use of saliency alone, i.e. without the voting and area filters, is able to help on the reduction of FPR, without harming the TPR.

6.2 Votes and Area Filters Testing

Fig. 6.2 show that the area filter, which aims at removing false positives, is pivotal for the overall system improvement. Each plot in the graph is the average of the Receiver Operating Characteristic (ROC) curves of the area filter over all images in the dataset A.1, for a given parametrisation, that is, for a given image and parametrisation of the area filter, $a \in \{0, 5, 25\}$, the ROC curve is built by sliding the threshold of the votes filter over its domain, $v \in \{0, 5, \ldots, 100\}$.

The ROC curves reveal that the absence of the area filter ($a = 0$), results in the poorest curve, i.e. with the lowest area under the curve, showing the usefulness of the filter. The relative performance associated of the other two different values, i.e. $a \in 5, 25$, switches at the intersecting point of the corresponding ROC curves. Nevertheless, $a = 25$ is selected as it is the one performing better for lower values of FPR.

Fig. 6.3 show the impact of the voting mechanism. Each plot in the graph is the average of the ROC curves over all images in the dataset A.1, for six different configurations:

1. OOD with resolution (1×1), which by not having a voting mechanism is limited to a point;

2. EOD with resolution (3×6) and $v \in [0, 15]$;

3. SalOD with resolution (3×6), which by the same reason of OOD is constrained to a point;

4. ESalOD with votes filter on, $v \in \{0, 5, \ldots, 100\}$, and area filter off, $a = 0$;

5. ESalOD with votes filter off, $v = 0$, and area filter on, $a \in \{0, 5, \ldots, 100\}$;

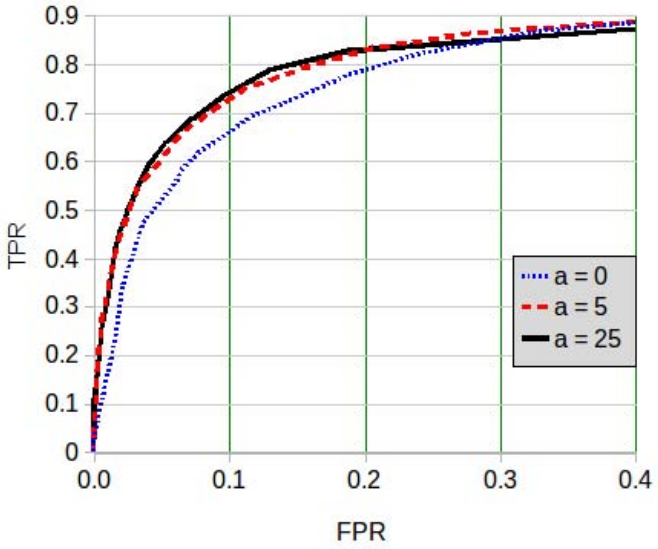


Figure 6.2: Impact of the area filter.

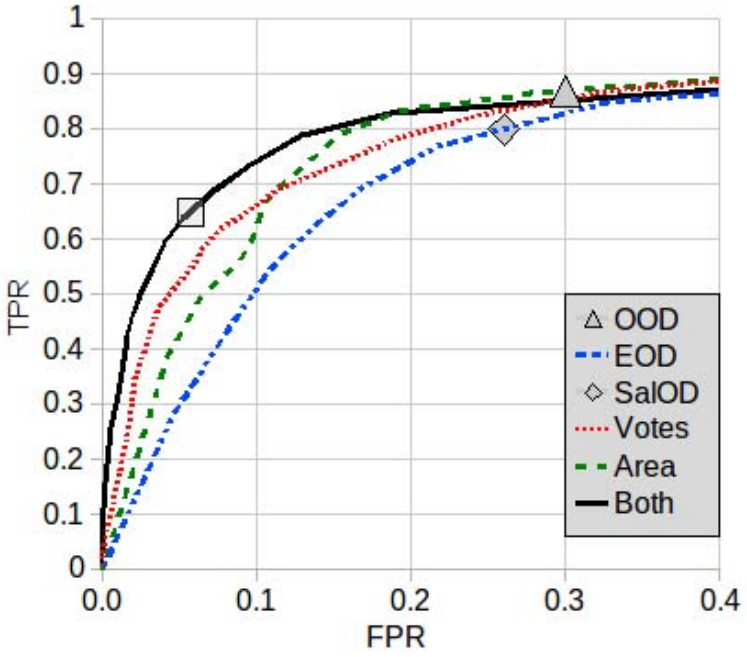


Figure 6.3: Impact of the voting mechanism.

6. ESalOD with both filters on, $a = 25$ and $v \in \{0, 5, \ldots, 100\}$.

The ROC curves associated to using only the voting mechanism or the area filter are below the curve when both are employed. The fact that the voting mechanism and area filter curves switch their relative performance at an interception point show that their relevance depends on the preferred trade-off. The blend of both mechanisms is always better than the two alone, thus showing that only when the two are blended the highest area under the curve is attained, whatever the preferred trade-off.

It is also visible that ESalOD performs better and more robustly than its previous versions, i.e. OOD, EOD and SalOD.

The grey square signals the point associated to the ESalOD chosen configuration, i.e. $a = 25$ and $v = 20$. With this configuration, almost no false positives were visible (see Fig. A.4), and most of the loss in terms of true positives were restricted to obstacles' inner points. A proper representation of obstacles' boundaries are barely untouched.

This configuration thus provides a clean and sufficiently complete environment's representation for obstacle avoidance purposes.

6.3 Computation Time Comparison

After determining the best parameterisation for efficient performance in ESalOD, its computation time was compared with the previous models.

Fig. 6.4 shows that the added complexity of ESalOD, in order to obtain higher levels of accuracy and robustness, does not compromise its performance. As predicted, the use of saliency in SalOD results in a reduction of computation cost when compared to the EOD. Being the ESalOD more comprehensive, i.e. able to operate in full resolution if the saliency map demands it, results in a poorer performance when compared with the SalOD, but still better than the one of the EOD. In sum, ESalOD generated a considerably better ROC curve than its predecessors, without loosing on performance.

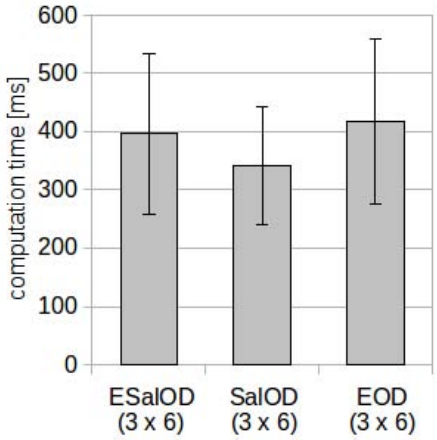


Figure 6.4: Computation time comparison. Each bar corresponds to the average computation time over all images in the dataset, for a given configuration. Error bars correspond to their standard deviation. All configurations operate on resolution (3×6). EOD runs with votes in their best parametrisation, i.e. $v = 7$, space-variant resolution, and without its additional morphological filters [Santana et al., 2008]. ESalOD runs with both voting and area filters, and space-variant resolution mechanisms turned on and parametrised for their best performance, i.e. $v = 20$ and $a = 25$. SalOD runs with space-variant resolution and, differently from its original implementation [Santana et al., 2009], for the sake of a comparison, with the filters configured as for the ESalOD case. Ground-plane compensation is turned on in all cases, despite not originally considered in EOD.

An additional experiment shows that the saliency-based slide mechanism is responsible for 17% saving of computation time. Stereo, saliency, and ground-plane estimate computation take, in average, 40ms, 43ms and 54ms, respectively.

The original obstacle detector (OOD) performs in average significantly slower, roughly $110\times$, than the ESalOD. This is particularly important as the ESalOD is also performing in full-resolution, despite the fact its base resolution is (3×6). The timing information reported by Manduchi et. al [Manduchi et al., 2005] varies between 0.67s and 4s (after conversion to 640×480 images). This is rather different from the timings one obtained with the implementation of their algorithm made for these tests, where an average of 40s was obtained. Possibly, this is mostly due to the fact that a great part of the images in the used dataset (Fig. A.1) cover larger areas of the near-field than the ones used by Manduchi et. al.. In these situations, the truncated triangle is quite large and consequently more expensive to analyse. This happens because the images were taken from a lower height, and many of them with large tilt and roll angles.

Note that all tested algorithms are built from the same backbone, i.e. the OOD implementation. This reduces to the minimum any bias that could benefit any of the models.

6.4 Hybrid OD Testing

This section presents the experimental results for the Hybrid Obstacle Detector (see chapter 5). In order to take advantage of the ground-truth A.3 used in the previous tests, the Hybrid OD was tested in its discrete detection mode, i.e. without cost maps and without the adapted voting mechanism (see section 5.2 for details). Nevertheless, the results obtained show the advantage of joining two different types of obstacle detectors that complement each other. For a visualisation of the results obtained with the dataset (Fig. A.1) refer to Fig. A.6. The correlation of this results with the ground truth is shown in Fig. A.7.

The predicted advantage of fusing two different but complementer detectors in a hybrid model is that large and textureless obstacles, which lack saliency, are more detectable by a plane-based method. In opposition, the linearisation nature of the plane-based method should hinder a proper detection of small obstacles, making the cone-based method more convenient for the task. Furthermore, setting the plane-based method to detect small obstacles should result in an increase in terms of False Positive Rate (FPR).

As predicted, Fig. 6.5 shows a growth in terms FPR as the required height of objects to be considered obstacles, h, in the plane-based method is reduced. Unpredictably however, the growth is small. This phenomenon is due to the fact that the height of the estimated ground-plane is most often slightly overestimated. As a result, the meaning of h changes to an artificially higher value. This can be observed by the larger amount of false negatives in the lower parts of the obstacles, than the ones produced by the cone-based mechanism. Having both methods the same critical threshold, those false negatives should coincide. This leads to the conclusion that, as predicted, in order to have similar FPR, the plane-based method must focus on taller obstacles than the cone-based one. The steady growth of TPR is a reflection of the complementary role of both methods, further supported by Fig. 6.6. While the cone-based

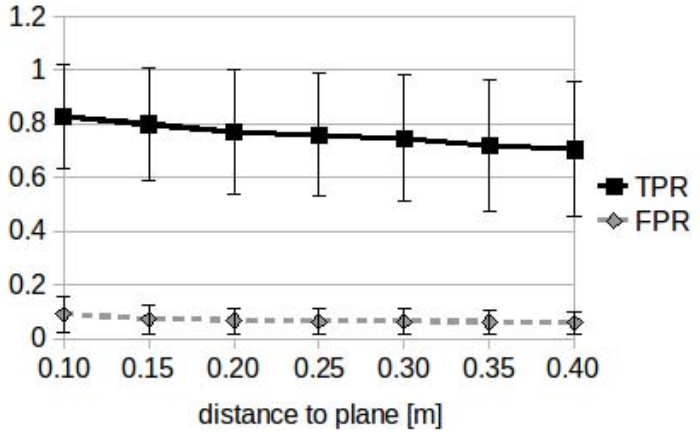


Figure 6.5: Analysis of the best configuration for the plane-based detector in the fusion process. Plots refer to the average True Positive Rate (TPR) and False Positive Rate (FPR) over all images in the dataset, for different values of h in the ground-plane method. Conversely, the cone-based obstacle detector was set static to its best parametrisation.

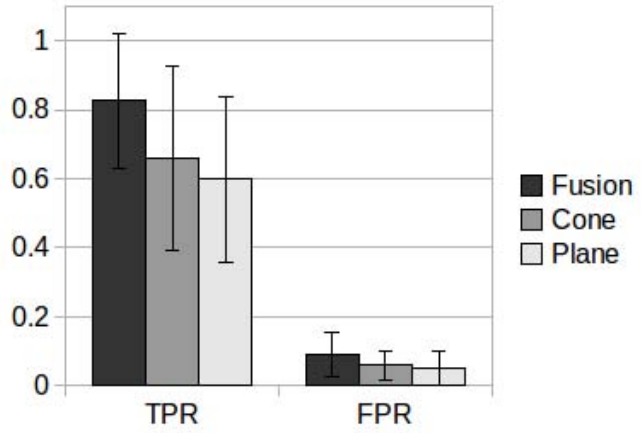


Figure 6.6: Comparison between fusion and isolated obstacle detection methods. Bars correspond to the Average True Positive Rate (TPR) and False Positive Rate (FPR) over all images in the dataset, for three configurations: (1) cone-based detector alone (Hmin=10cm), (2) plane-based detector alone ($h = 10cm$) and (3) fusion of the previous two configurations. The graph shows that the fusion process produces a higher TPR than any of the methods alone. This supports the prediction that both methods could operate in a complementary way. This is further enforced by the reduced standard deviation (error bars) of the fusion process. The opposite trend in terms of FPR is $\approx 6\times$ weaker, and thus insufficient to contradict the conclusions taken so far.

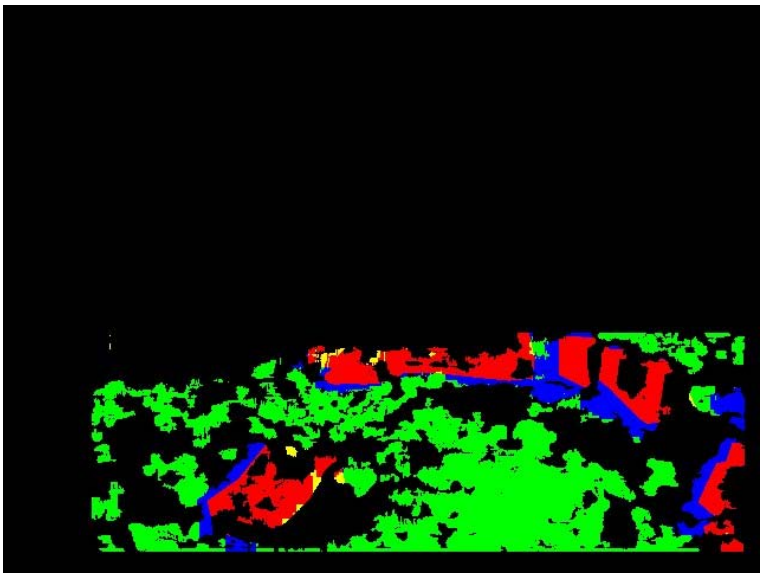


Figure 6.7: Correlation map between the results obtained from applying the ESalOD with its best parameterisation to the image #5 and its ground truth. Red pixels correspond to true positive detection, green pixels to true negative detection, blue pixels to false positive detection and yellow pixels to false negative detection. Note the problem of false positive detection (blue pixels) near dense obstacles.

method detects low obstacles and the lower parts of larger obstacles, the plane-based method detects the upper regions (see Fig. 5.2).

The fact that the cone-based mechanism exhibits higher levels of FPR than the plane-based one (see Fig. 6.6), is easily explained if the position where these false positives emerge is taken into account. In the cone-based method case, false positives are mostly around obstacles, and thus their sole consequence is the enlarged perception of obstacles (see Fig. 6.7). This happens because those false positives are in fact ground points compatible to the obstacles' points. Hence, under the compatibility test assumption, there are not actual false positives. In opposition, the false positives generated by the plane-based method are disperse all over the image, having no correlation with the actual position of obstacles.

Chapter 7

Conclusions, Contributions and Future Work

This chapter summarises this dissertation, discussing the proposed approaches and contributions, as well as the results obtained, followed by some aspects to be taken into consideration for future work.

7.1 Summary of Contributions

In this dissertation, an accurate, robust, computationally efficient and vision-based obstacle detector for all-terrain service robots was proposed.

The following summarises the set of contributions of this dissertation:

1. The well known obstacle detector originally proposed in the work of Manduchi et al. [Manduchi et al., 2005] was enhanced and integrated in a hybrid architecture. The proposed improvements greatly reduce false positive rate but also showed difficulties in correctly detecting the top of large homogeneous, i.e. structured, obstacles. An efficient fusion with a fast, rough but complementary plane-based obstacle detector, allowed to diminish this effect.

2. The *voting filters* proposed by Santana et al. [Santana et al., 2008] where reinterpreted. In its original model, the votes were not invariant to scale, which means that a vote casted by a near 3-D point had the same weight as a vote casted by a 3-D point located far away. The problem associated with this method relies on the fact that, when projected onto the image plane, far away obstacles are represented by less pixels, thus encompassing less data. In this dissertation, a normalisation for this method was proposed, by taking into consideration not only the casted votes but also its maximum possible number. In addition, the proposed *voting mechanism* was integrated with an *area filter*, allowing to further reduce noise without hampering true positive rate.

3. The space-variant resolution mechanism proposed by Santana et al. [Santana et al., 2008] was remodelled. In its original form, the method contributed to computational time reduction by blindly skipping some pixels. In this dissertation, a saliency model that pop out obstacles from the background, in the image plane, was used to focus, in an informed way, the attention of the detector in those salient regions. Thus, skipping more pixels in less salient regions and analysing more data in more salient ones results on a lower probability to loose true positive detection and do detect false positives, as well as an improved computational efficiency.

4. In order to improve the correctness of the method, invariance to the robot posture was introduced in order to enable a robust behaviour in more demanding situations.

7.2 Conclusions

The following conclusions can be drawn.

Experimental results show the ability of the proposed model to properly locate all obstacles in the environment. Although obstacles are not always fully represented (mainly due to lack of 3-D information), in the majority of the cases their borders are correctly detected, which is sufficient for proper obstacle avoidance. Also, fusing the proposed cone-based obstacle detector

with a plane-based one have been shown to increase robustness over any of the methods alone.

It was shown that, with normalisation, the voting mechanism results in a more efficient noise reduction. It was also shown the complementary role of the normalised voting mechanism and the area filter. The integration of both results in a more robust solution than having only one of them alone. Additionally, by an empirically analysis on the dataset images, it was possible to conclude that an adapted version of the normalised voting mechanism can be used to label obstacle points with a traversability cost (see Fig. 5.4), rather than a binary tag (obstacle/free-space).

Finally, testings shown that the proposed model outperforms its predecessors on both robustness and computation time. This fact suggests that saliency have an important role in focusing the analysis. Being faster means that less points are analysed, but being more robust means that the analysis is done where it needs to be done, i.e. in regions containing obstacles.

7.3 Future Work

The following presents some opportunities for future work, based on the obtained results:

- To implement the proposed detector in a physical robot in order to fully assess its applicability for obstacle avoidance purposes.

- To explore new ways of modulating the saliency map with the output generated by the obstacle detector. In the hybrid model, the output of the large obstacles detector modulate the saliency map in order to focus the small obstacles detector in areas that have not been already analysed. In the single model, obstacles' presence is used to reinforce its corresponding saliency. However, a more extensive analysis on this subject would be interesting.

- To exploit the generation of traversability maps. In this dissertation, an adapted version of the voting mechanism was used to represent obstacles in terms of their traversability cost.

However, the adaptation was based on simple heuristics. Further analysis and testing would be required for better understanding the capabilities of this method.

- To improve obstacles' representation. When 3-D point clouds are sparse, obstacles may be poorly represented. Although a good detection of obstacles' borders its sufficient for obstacle avoidance, it would be interesting to use both colour images and saliency maps to recover a full representation of each obstacle (regarding colour intensity or saliency homogeneity). This aspect is important for a proper representation and consequent labelling of different types of objects (e.g. differentiating a rock from a bush would be important to determine object's traversability).

Bibliography

[Batavia and Singh, 2002] Batavia, P. and Singh, S. (2002). Obstacle detection in smooth high curvature terrain. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*, volume 3, pages 3062–3067.

[Bellutta et al., 2000] Bellutta, P., Manduchi, R., Matthies, L., Owens, K., and Rankin, A. (2000). Terrain perception for demo iii. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV 2000)*, pages 326–331.

[Birk and Carpin, 2006] Birk, A. and Carpin, S. (2006). Rescue robotics - a crucial milestone on the road to autonomous systems. *Advanced Robotics*, 20(5):595–605.

[Bradski and Kaehler, 2008] Bradski, G. and Kaehler, A. (2008). *Learning OpenCV: computer vision with the OpenCV library*. O'Reilly Media, Inc.

[Broggi et al., 2006] Broggi, A., Caraffi, C., Porta, P., and Zani, P. (2006). The single frame stereo vision system for reliable obstacle detection used during the 2005 darpa grand challenge on terramax. In *Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC 2006)*, pages 745–752.

[Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

[Frintrop, 2006] Frintrop, S. (2006). Vocus: A visual attention system for object detection and goal-directed search. *PhD thesis, LNCS (LNAI)*, 3899.

[Frintrop et al., 2007] Frintrop, S., Jensfelt, P., and Christensen, H. (2007). Simultaneous robot localization and mapping based on a visual attention system. In *The 4th International Workshop on Attention in Cognitive Systems (WAPCV 2007)*. Springer-Verlag.

[Goldberg et al., 2002] Goldberg, S., Maimone, M., and Matthies, L. (2002). Stereo vision and rover navigation software for planetary exploration. In *Proceedings of the 2002 IEEE Aerospace Conference*, volume 5, pages 5–2025–5–2036.

[Hamner et al., 2008] Hamner, B., Singh, S., Roth, S., and Takahashi, T. (2008). An efficient system for combined route traversal and collision avoidance. *Autonomous Robots*, 24(4):365–385.

[Hong et al., 2002] Hong, T., Rasmussen, C., Chang, T., and Shneier, M. (2002). Fusing ladar and color image information for mobile robot feature detection and tracking. *Intelligent Autonomous Systems*, 7:124.

[Hwang et al., 2009] Hwang, A., Higgins, E., and Pomplun, M. (2009). A model of top-down attentional control during visual search in complex scenes. *Journal of Vision*, 9(5):25.

[Itti et al., 1998] Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259.

[Kim et al., 2006] Kim, S., Roh, C., Kang, S., and Park, M. (2006). A hybrid autonomous/teleoperated strategy for reliable mobile robot outdoor navigation. In *SICE-ICASE, 2006. International Joint Conference*, pages 3120–3125.

[Konolige et al., 2006] Konolige, K., Agrawal, M., Bolles, R., Cowan, C., Fischler, M., and Gerkey, B. (2006). Outdoor mapping and navigation using stereo vision. In *Proceedings of the 10th International Symposium on Experimental Robotics (ISER 2006)*. Springer Verlag.

[Konolige and Beymer, 2007] Konolige, K. and Beymer, D. (2007). SRI Small Vision System: User's Manual. *SRI Intl.(May 2007)*.

[Lacaze et al., 2002] Lacaze, A., Murphy, K., and DelGiorno, M. (2002). Autonomous mobility for the demo iii experimental unmanned vehicles. In *Association for Unmanned Vehicle Systems International Conference on Unmanned Vehicles (AUVSI 2002)*.

[Lacroix et al., 2002] Lacroix, S., Mallet, A., Bonnafous, D., Bauzil, G., Fleury, S., Herrb, M., and Chatila, R. (2002). Autonomous rover navigation on unknown terrains: functions and integration. *International Journal of Robotics Research*, 21(10-11):917–942.

[Lalonde et al., 2006] Lalonde, J., Vandapel, N., Huber, D., and Hebert, M. (2006). Natural terrain classification using three-dimensional ladar data for ground robot mobility. *Journal of Field Robotics*, 23(10):839–861.

[Maimone et al., 2006] Maimone, M., Biesiadecki, J., Tunstel, E., Cheng, Y., and Leger, C. (2006). Surface navigation and mobility intelligence on the mars exploration rovers. *Intelligence for Space Robotics*, pages 45–69.

[Manduchi et al., 2005] Manduchi, R., Castano, A., Talukder, A., and Matthies, L. (2005). Obstacle detection and terrain classification for autonomous off-road navigation. *Autonomous Robots*, 18(1):81–102.

[Matthies et al., 2007] Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., Goldberg, S., Huertas, A., Stein, A., and Angelova, A. (2007). Computer vision on mars. *International Journal of Computer Vision*, 75(1):67–92.

[Meger et al., 2008] Meger, D., Forssén, P., Lai, K., Helmer, S., McCann, S., Southey, T., Baumann, M., Little, J., and Lowe, D. (2008). Curious george: an attentive semantic robot. *Robotics and Autonomous Systems*, 56(6):503–511.

[Morén et al., 2008] Morén, J., Ude, A., Koene, A., and Cheng, G. (2008). Biologically based top-down attention modulation for humanoid interactions. *International Journal of Humanoid Robotics*, pages 3–24.

[Newman and Ho, 2005] Newman, P. and Ho, K. (2005). Slam-loop closing with visually salient features. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005)*, pages 635–642.

[Orabona et al., 2005] Orabona, F., Metta, G., and Sandini, G. (2005). Object-based visual attention: a model for a behaving robot. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005) - Workshops*, page 89, Washington, DC, USA. IEEE Computer Society.

[Rankin et al., 2005] Rankin, A., Huertas, A., and Matthies, L. (2005). Evaluation of stereo vision obstacle detection algorithms for off-road autonomous navigation. In *AUVSI Symposium on Unmanned Systems*.

[Santana et al., 2007] Santana, P., Barata, J., and Correia, L. (2007). Sustainable robots for humanitarian demining. *International Journal on Advanced Robotics Systems*, 4(2):207–218.

[Santana et al., 2009] Santana, P., Guedes, M., Correia, L., and Barata, J. (2009). Saliency-based obstacle detection and ground-plane estimation for off-road vehicles. In *Proceedings of the 7th International Conference on Computer Vision Systems (ICVS 2009)*, pages 275–284. Springer-Verlag New York Inc.

[Santana et al., 2010] Santana, P., Guedes, M., Correia, L., and Barata, J. (2010). A saliency-based solution for robust off-road obstacle detection. In *Accepted for publication in the Proceedings of the International Conference on Robotics and Automation (ICRA 2010)*.

[Santana et al., 2008] Santana, P., Santos, P., Correia, L., and Barata, J. (2008). Cross-country obstacle detection: Space-variant resolution and outliers removal. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008)*, pages 1836–1841.

[Singh et al., 2000] Singh, S., Simmons, R., Smith, T., Stentz, A., Verma, V., Yahja, A., and Schwehr, K. (2000). Recent progress in local and global traversability for planetary rovers. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2000)*, volume 2, pages 1194–1200.

[Thrun et al., 2006] Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., et al. (2006). Stanley: The robot that won the darpa grand challenge: Research articles. *Journal of Robotic Systems*, 23(9):661–692.

[van der Mark et al., 2007] van der Mark, W., van den Heuvel, J., and Groen, F. (2007). Stereo based obstacle detection with uncertainty in rough terrain. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV 2007)*, pages 1005–1012.

[Vandapel et al., 2004] Vandapel, N., Huber, D., Kapuria, A., and Hebert, M. (2004). Natural terrain classification using 3-d ladar data. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2004)*, volume 5, pages 5117–5122.

[Vijayakumar et al., 2001] Vijayakumar, S., Conradt, J., Shibata, T., and Schaal, S. (2001). Overt visual attention for a humanoid robot. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*, volume 4.

[Yu et al., 2007] Yu, Y., Mann, G., and Gosine, R. (2007). A task-driven object-based attention model for robots. In *Proceedings of the International Conference on Robotics and Biomimetics*.

Appendix A

Dataset and Image Results

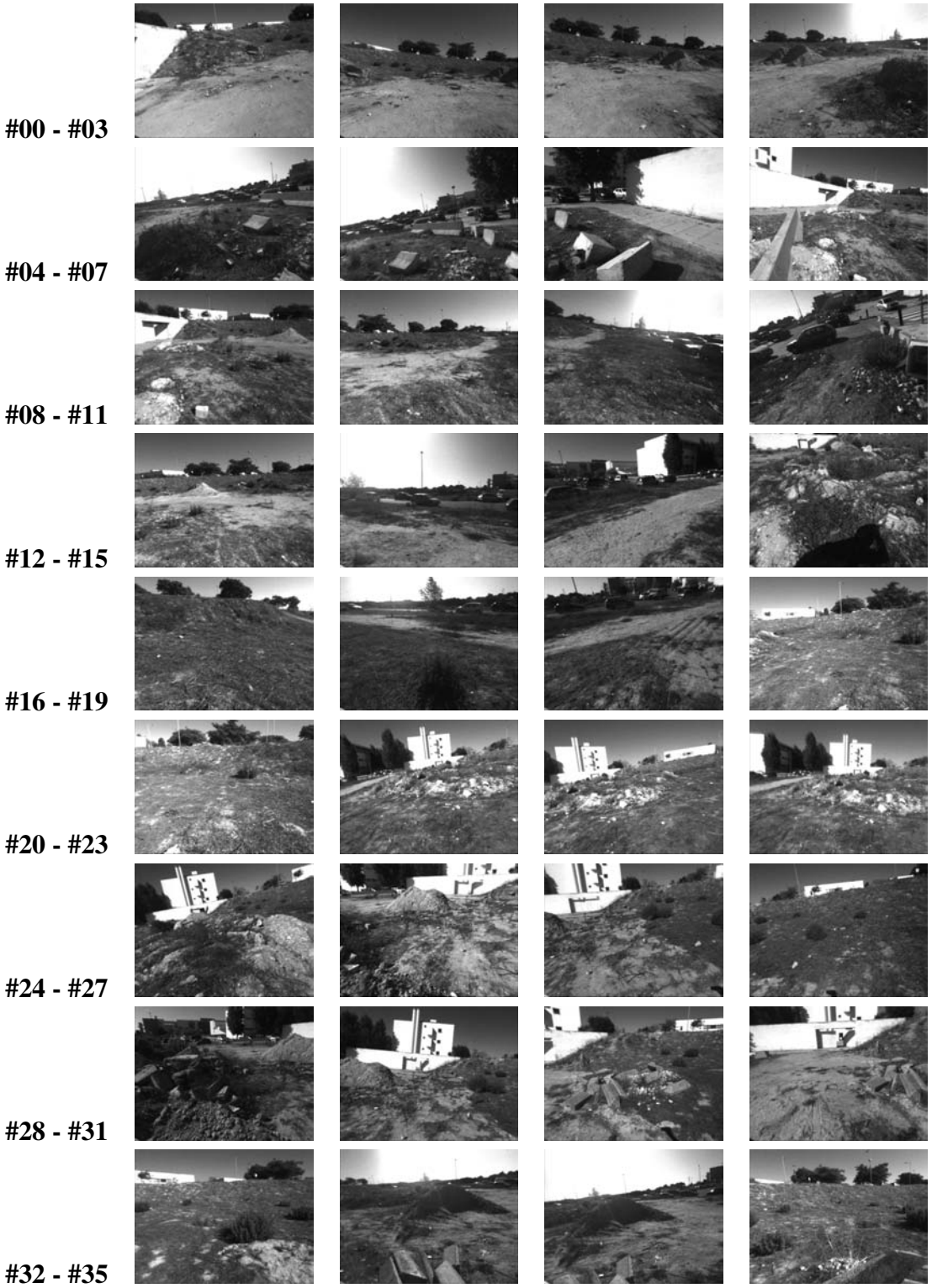


Figure A.1: Left-camera images encompassing the dataset used in all experiments.

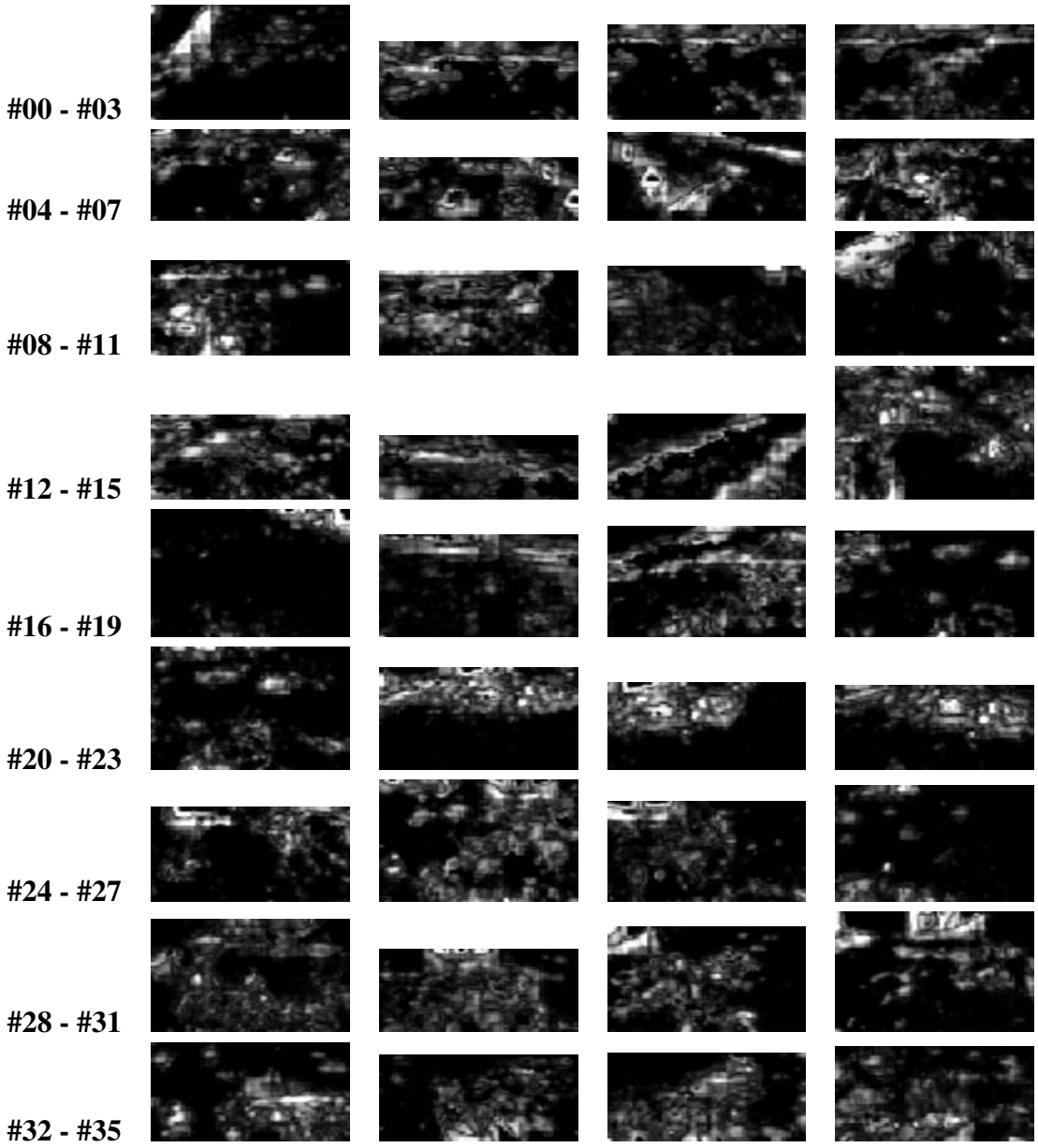


Figure A.2: Saliency maps obtained from each image in the dataset (Fig. A.1) up to a range of 10m.

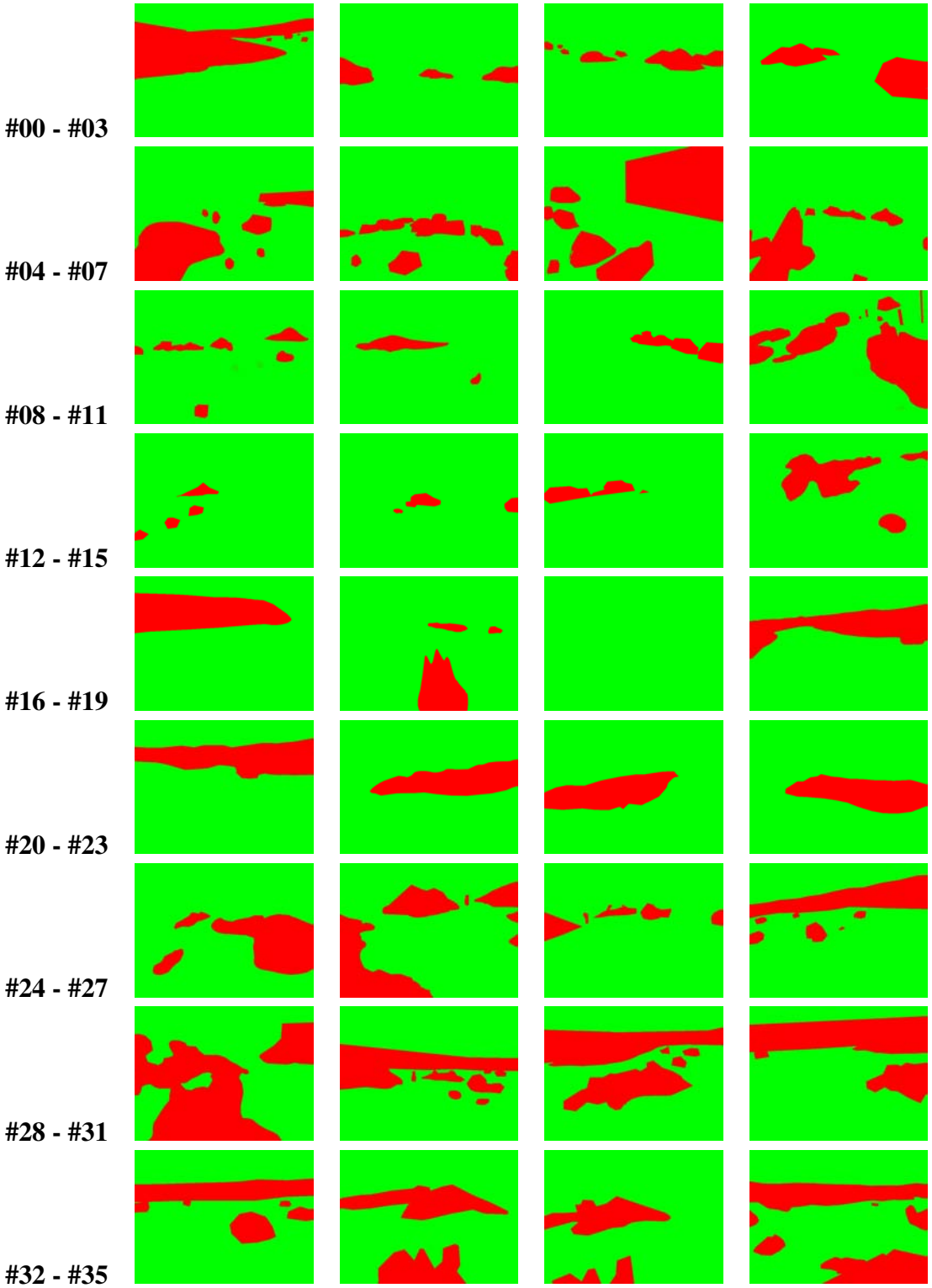


Figure A.3: Obstacle-ground truth hand-drawed for each image in the dataset (Fig. A.1). Obstacles (in red) laying outside the detection range considered in the experiments made, i.e. more than 10m away from the vision sensors, may not be represented.

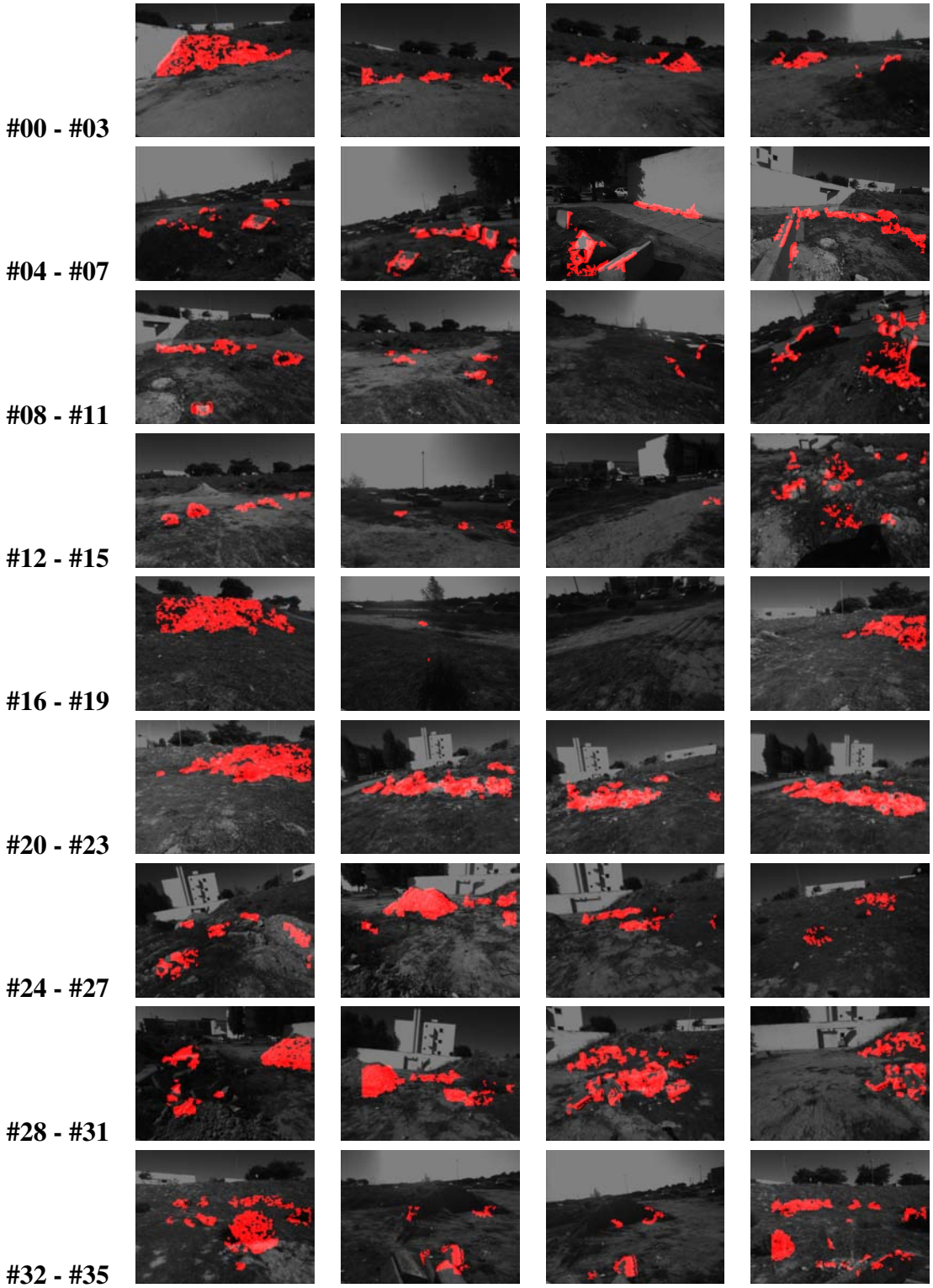


Figure A.4: Detection results obtained from all the images in the dataset (Fig. A.1) for the best parameterisation (see chapter 6 for further details). Points considered as obstacles are represented in red.

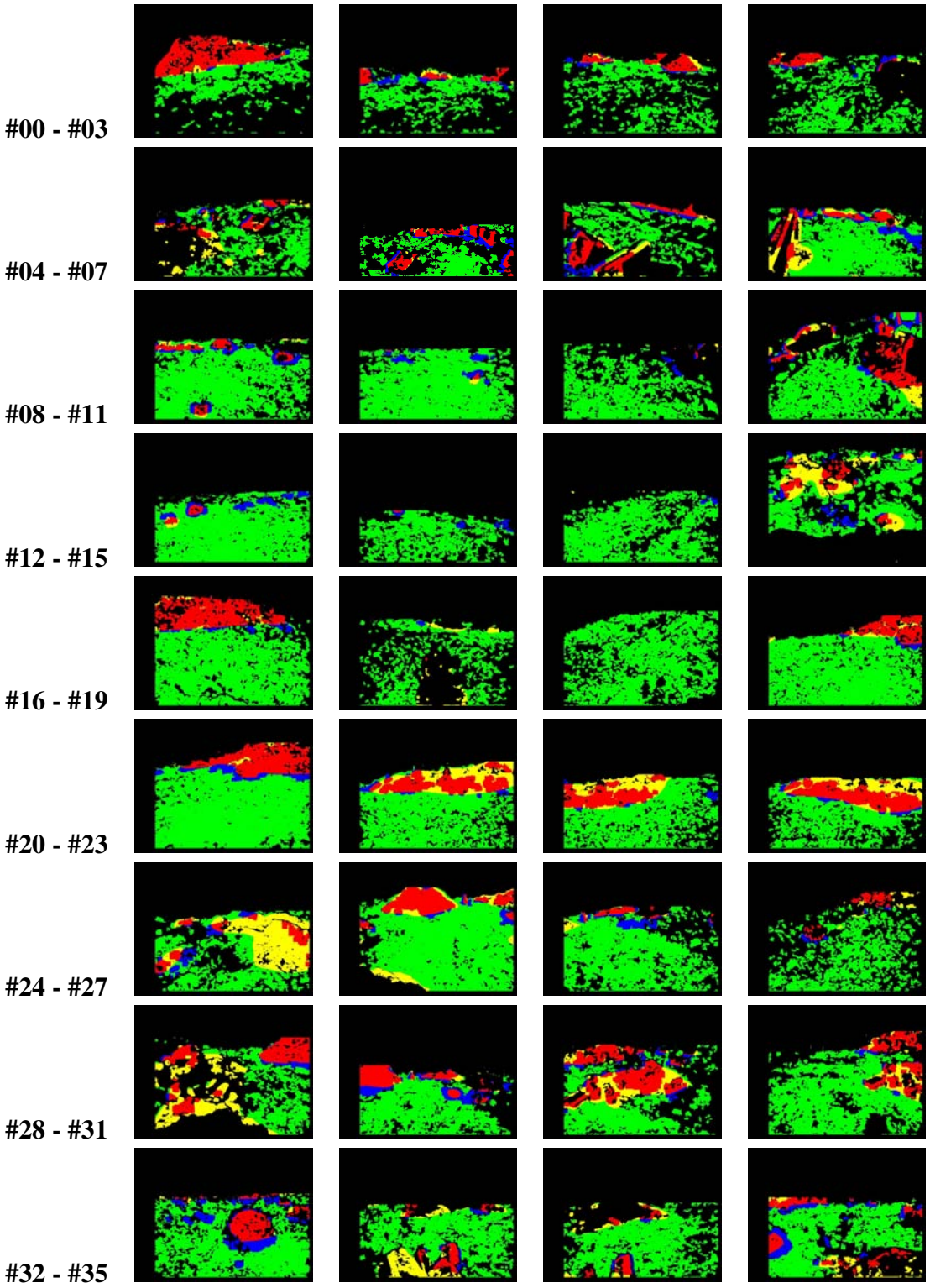


Figure A.5: Correlation maps between the obstacle-ground truth (Fig. A.3) and the system output (Fig. A.4). In red: true possitive detection; in green: true negative detection; in blue: false positive detection; in yellow: false negative detection; black pixels have no computed depth.

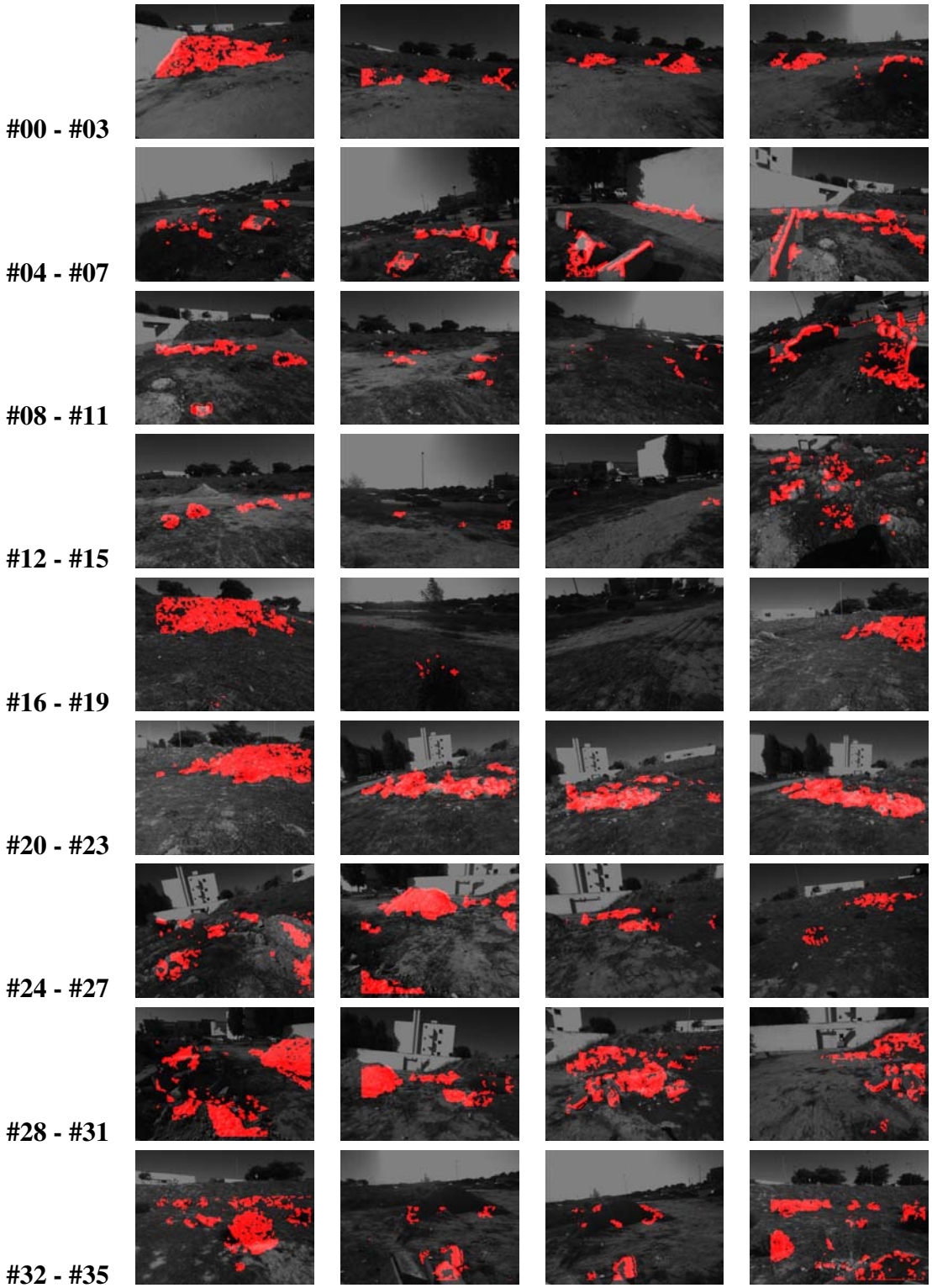


Figure A.6: Detection results obtained from all the images in the dataset (Fig. A.1) for the Hybrid Obstacle Detector. Points considered as obstacles are represented in red.

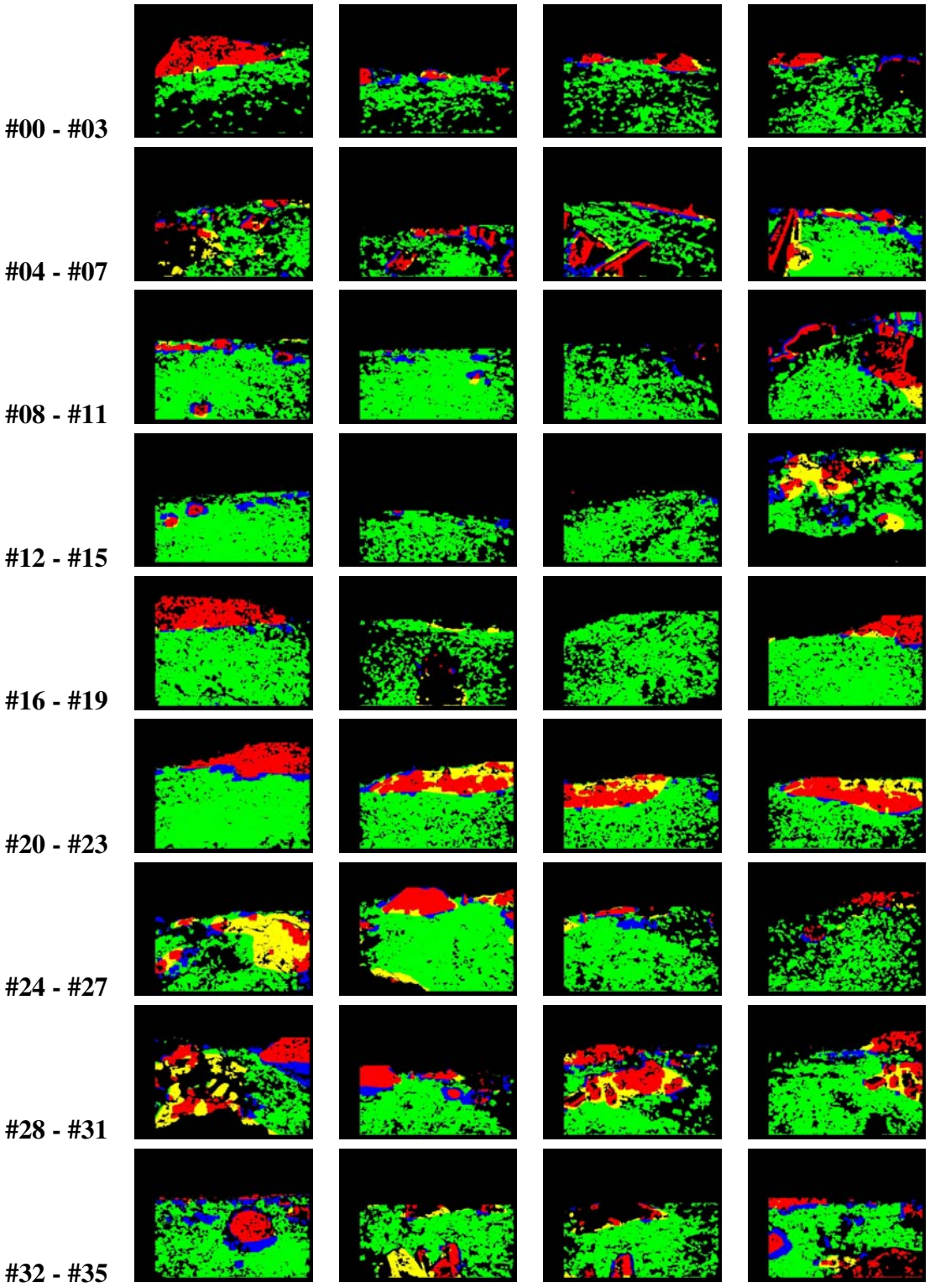


Figure A.7: Correlation maps between the obstacle-ground truth (Fig. A.3) and the system output (Fig. A.6). In red: true possitive detection; in green: true negative detection; in blue: false positive detection; in yellow: false negative detection; black pixels have no computed depth.