Journal of Intelligent and Robotic Systems manuscript No.
(will be inserted by the editor)

Francisco Bonin-Font, Alberto Ortiz and Gabriel Oliver

Department of Mathematics and Computer Science, University of the Balearic Islands, Palma de Mallorca, Spain

Visual Navigation for Mobile Robots: a Survey

Abstract Mobile robot vision-based navigation has been the source of countless research contributions, from the domains of both vision and control. Vision is becoming more and more common in applications such as localization, automatic map construction, autonomous navigation, path following, inspection, monitoring or risky situation detection. This survey presents those pieces of work, from the nineties until nowadays, which constitute a wide progress in visual navigation techniques for land, aerial and autonomous underwater vehicles. The paper deals with two major approaches: *map-based navigation* and *mapless navigation*. Map-based navigation has been in turn subdivided in *metric map-based navigation* and *topological map-based navigation*. Our outline to mapless navigation includes reactive techniques based on qualitative characteristics extraction, appearance-based localization, optical flow, features tracking, plane ground detection/tracking, etc... The recent concept of *visual sonar* has also been revised.

1 Introduction

Navigation can be roughly described as the process of determining a suitable and safe path between a starting and a goal point for a robot travelling between them [18, 72]. Different sensors have been used to this purpose, which has led to a varied spectrum of solutions. In particular, in the last three decades, visual navigation for mobile robots has become a source of countless research contributions since navigation strategies based on vision can increase the scope of application of autonomous mobile vehicles. Among the different proposals, this paper surveys the most recent ones. In many cases, the performance of a good navigation algorithm is deeply joined to an accurate robot localization in the environment. Therefore, some vision-based localization solutions applied and developed for autonomous vehicles have also been included in this survey.

Traditionally, vision-based navigation solutions have mostly been devised for *Autonomous Ground Vehicles* (AGV), but, recently, visual navigation is gaining more and more popularity among researchers developing *Unmanned Aerial Vehicles* (UAV). UAVs offer great perspectives in many applications, such as surveillance, patrolling, search and rescue, outdoor and indoor building inspection, real-time

This work is partially supported by DPI 2005-09001-C03-02 and FEDER funding

Francisco Bonin-Font
Tel.: +34-971-172813
E-mail: francisco.bonin@uib.es

Alberto Ortiz
Tel.: +34-971-172992
E-mail: alberto.ortiz@uib.es

Gabriel Oliver
Tel.: +34-971-173201
E-mail: goliver@uib.es

monitoring, high risk aerial missions, mapping, fire detection or cinema recording. Since UAVs move in 3D space they do not have the limitations of ground robots, which usually cannot overcome rocks, climb stairs or get access to ceilings. Nevertheless, UAVs need to exhibit a notable degree of awareness and exactness to accomplish their navigation and obstacle avoidance tasks successfully. Besides, the typically reduced size of UAVs limits their payload capabilities so that they cannot carry sensors available for ground vehicles, such as lasers or certain brands of sonars. In contrast, cameras used in robot vision-based navigation strategies are light and provide a perception of the environment in a single shot. However, the image resolution can be restricted due to the fact that UAVs fly at high altitude.

For underwater environments, there is still a preference for more traditional navigation solutions (i.e. acoustic-based) because of the special characteristics of light propagation undersea. Dalgleish *et al* [22] considered that, according to the state of the art in underwater autonomous navigation solutions, sonar based systems are limited in resolution and size. These limits are imposed by the acoustic frequency used and the need of accommodation space. Vision systems reduce space and cost and increase the resolution, although their range dramatically decreases in muddy or turbid waters. At present, a number of solutions for *Autonomous Underwater Vehicles* (AUV) can already be found for many undersea critical applications: undersea infrastructures or installations inspection and maintenance, for any of power, gas or telecommunications transport cases, sea life monitoring, military missions, sea bed reconstruction in deep waters, inspection of sunken ancient ships, etc. Vision has become essential for all these applications, either as a main navigation sensor or as a complement of sonar. Consequently, there exists a good motivation to improve AUVs navigation techniques by expanding their autonomy, capabilities and their usefulness.

Regardless of the type of vehicle, systems that use vision for navigation can be roughly divided in those that need previous knowledge of the whole environment and those that perceive the environment as they navigate through it. Systems that need a map can be in turn subdivided in *metric map-using systems*, *metric map-building systems* and *topological map-based systems* [28]. *Metric map-using navigation systems* need to be provided with a complete map of the environment before the navigation task starts. *Metric map-building navigation systems* build the whole map by themselves and use it in the subsequent navigation stage. Other systems that fall within this category are able to self-localize in the environment simultaneously during the map construction. Other sorts of *map-building navigation* systems can be found, as for example *visual sonar-based systems* or *local map-based systems*. These systems collect data of the environment as they navigate, and build a local map that is used as a support for on-line safe navigation. This local map includes specific obstacle and free space data of a reduced portion of the environment, which is usually a function of the camera field of view. Finally, *topological map-based systems* build and/or use topological maps which consist of nodes linked by lines where nodes represent the most characteristic places of the environment, and links represent distances or time between two nodes.

Mapless navigation systems mostly include reactive techniques that use visual clues derived from the segmentation of an image, optical flow, or the tracking of features among frames. No global representation of the environment exists; the environment is perceived as the system navigates, recognizes objects or tracks landmarks.

As for sensors, the different visual navigation strategies proposed in the literature make use of several configurations to get the required environmental information to navigate. Most systems are based on monocular and binocular (stereo) systems, although systems based on trinocular configurations also exist. Another structure that is gaining popularity because of its advantages is that of omnidirectional cameras. Omnidirectional cameras have a 360° view of the environment, and are usually obtained combining a conventional camera with a convex conic, spherical, parabolic or hyperbolic mirror. With this kind of cameras it is easier to find and track features, since they stay longer in the field of view.

The progress made in vision-based navigation and localization for mobile robots up to the late 90's was widely surveyed by DeSouza and Kak in [28]. After the late 90's, some authors have hardly surveyed this area: examples are Kak and DeSouza [66], whose work is restricted to *navigation in corridors*, and Abascal and Lazcano [1], whose work is restricted to *behaviour-based indoor navigation*. A remarkable outline of navigation and mosaic-based positioning solutions for Autonomous Underwater Vehicles (AUV) can be found in [22, 23] and a wide list of underwater vision tracking techniques was surveyed in [141]. Our survey mostly covers the work performed from the late nineties until the present day, and includes all the topics related to visual navigation. The scope of robotics as a discipline and the huge number of existing contributions make almost impossible to make a complete account, so that only the ones that had a higher impact (to the authors' view) have been included in the survey. Furthermore, instead of grouping navigation strategies in *indoor* and *outdoor* categories, as DeSouza and Kak [30] did, this survey distinguishes between *map-based* and *mapless navigation*, since some navigation systems proposed for indoor could also be properly adapted to work in outdoor environments and vice versa.

The rest of the paper is organized as follows: first, section 2 revises the most prominent approaches until the late 90's, mostly coincident with the ones surveyed in [30]; second, section 3 reviews new approaches presented after the submission of [30]; finally, section 4 concludes the paper.

2 From the primary techniques to the advances in the late 90's

De Souza and Kak in [28] structure robot visual navigation in two main subjects: *indoor navigation* and *outdoor navigation*. *Outdoor navigation* is in turn subdivided in *structured* and *unstructured environments*, while *indoor navigation* is subdivided in *map-building-based navigation* and *mapless navigation*. This section of the paper summarizes visual navigation techniques until the late 90's. Therefore, it is a selection of the most outstanding contributions surveyed in the work by De Souza and Kak, although some references not considered there have also been included. Table 1 shows the references surveyed in this section. (In order to make compatible this section of the survey with the one by De Souza and Kak, the rest of this section is structured in the same way as De Souza and Kak's survey, maintaining thus the same category labels.)

2.1 Indoor Navigation

From the first robot developments in 1979 by Giralt [44], many control systems have incorporated, in a lesser or greater extent, some information about the environment where the robot had to navigate. The navigation and localization systems proposed fall mainly within one of the following three groups:

- *map-based navigation* systems
- *map-building-based navigation* systems
- *mapless navigation* systems

2.1.1 Map-based Navigation

These techniques are based on providing the robot with models of the environment, with different degrees of detail depending on the study.

The first approaches made use of an occupancy map with a 2D projection of each prominent feature situated in the environment. Later, the Virtual Force Fields [10, 68] associated every cell containing an obstacle with a repulsive force towards the robot. Other authors incorporated uncertainties in occupancy maps to account for sensor errors [12, 103].

The combination of different sensors has also been employed in other approaches to increase the robustness and reliability of the map building procedure. In [16], range finders and cameras work in collaboration to create occupancy grids [1]. Once the robot has acquired the map, it can navigate in the environment, matching the landmarks found in the on-line image with the expected landmarks of a database. This process is known as self-localization and is fundamental for a correct navigation. The main steps are:

- acquire image information,
- detect landmarks in current views (edges, corners, objects),
- match observed landmarks with those contained in the stored map according to certain criteria, and
- update the robot position, as a function of the matched landmarks location in the map.

[1] An *occupancy grid* represents an observed region and each cell of the grid is labeled with the probability of being occupied by an object.

To solve the localization problem, *absolute localization methods* contrast with *relative localization strategies*. In *absolute localization* methods, the initial position of the robot is unknown. This self-localization problem has been solved either using deterministic triangulation [130], Markov [129] or Montecarlo [34] localization. Atiya and Hager [5] presented in 1993 a remarkable method based on the recognition, in the on-line images, of those features (or connecting lines between features) that stay invariant with respect to the moving robot. This is sufficient to set up correspondences between the environment and the images. These authors also propose to define the sensor errors as a tolerance measurement.

In *relative localization*, it is assumed that, at the beginning of the navigation, the position of the robot is approximately known. Matthies and Shafer [91] used stereo vision to reduce errors. Tsubouchi and Yuta [144] used a CAD model for environment representation. Later on, Christensen *et al* [19] also used CAD models for space representation combined with stereo vision to reduce errors. FINALE [69] self-localizes using a geometrical representation of the environment and a Gaussian model for location uncertainty. Position uncertainty equations prove that location at the end of motion depends on previous positions. A model-based Kalman filter is used to compute landmark position and to project robots location uncertainty into the image.

NEURO-NAV [94, 95] is a representative example of *map-based navigation strategies* based on *topological space representations*. These navigation techniques use a nodes-and-lines graph that layouts the most representative points of the hallway. Both, nodes and lines, are attributed with information about what they represent (central corridor, door, corner, junctions for nodes, and distances between linked nodes for lines). NEURO-NAV has two main modules built up with neural networks: a hallway follower module and a landmark detector module. These two modules compute edges, detect walls and output the proper steering commands to drive the robot at a distance of a wall or centered in a corridor. Most of those neural network outputs are vague and have a degree of confidence. FUZZY-NAV [104] sophisticates the NEURO-NAV system using fuzzy functions that work with blurred variables.

Finally, in order to self-locate, *landmark tracking* algorithms determine the position of the robot, detect landmarks on the camera image and track them in the consecutive scenes. Landmarks can be artificial or natural. In both cases the robot needs to know the identity of the landmarks to be able to track them. This method has been used in map-based navigation systems and in some reactive navigation architectures. Kabuka and Arenas [65] were the first using artificial landmark tracking. An example of natural landmark tracking-based navigation system can be found in [57]. This approach selects landmarks, uses correlation techniques to track them, computes their 3D position using stereo vision information and selects new landmarks to keep on moving towards the goal point.

2.1.2 Map-building-based Navigation

This section includes all the systems that can explore the environment and build its map by themselves. The navigation process starts once the robot has explored the environment and stored its representation. The first to consider this possibility was Moravec with his Stanford Cart [96]. This system was improved by Thorpe for the robot FIDO [134], and was used to extract features from images. These features were then correlated to generate their 3D coordinates. The features were represented in an occupancy grid of two square meter cells. Although this technique provided a representation of obstacles in the environment, it was not good enough to model the world. Occupancy grid-based strategies can be computationally inefficient for path planning and localization, specially in complex and great indoor environments. Furthermore, the validity of these grids for navigation depends on the accuracy of the motion detection robot sensors during the grid construction process. A *topological representation* of the environment is an alternative to an occupancy grid. These systems are based on generating a graph of nodes representing the space, and storing metrical information for every node recognized during the navigation process. The different approaches differ from what constitutes a node, how a node may be distinguished from others, the possibility of using sensor uncertainties or how these uncentainties are computed. One of the major difficulties is the recognition of previously visited nodes.

Thrun [137] went one step further with a remarkable contribution, combining the best of occupancy grids and topological maps for navigation.

2.1.3 Mapless Navigation

This category includes all navigation approaches that do not need knowledge of the environment to run. The movements of the robot depend on the elements observed in the environment (walls, features, doors, desks, etc...). Two main techniques should to be cited: optical-flow- and appearance-based navigation.

Optical-flow-based solutions estimate the motion of objects or features within a sequence of images. Researchers compute optical flow mostly using (or improving) pioneering techniques from Horn [59] and Lucas and Kanade [79].

An interesting approach developed by Santos-Victor [114] emulates the bees' flying behavior. The system moves in a corridor using two cameras to perceive the environment, one camera on each side of the robot, pointing to the walls. Bees keep flying centered in a corridor by measuring the difference of velocities respect to both walls. If both velocities are equal, bees fly straight ahead in the center of the corridor. If velocities are different, they move to the wall whose image changes with minor velocity. The robot calculates the differences in optical flow computed from the images of both sides. The robot always moves in the direction of the optical flow minor amplitude. The main problem of this technique is that the walls need to be textured enough to present an optimum optical flow computation.

Appearance-based matching techniques are based on the storage of images in a previous recording phase. These images are then used as templates. The robot self-locates and navigates in the environment matching the current viewed frame with the stored templates. Examples of these approaches are:

- Matsumoto *et al* [88] VSRR (*View Sequenced Route Representation*), where a sequence of images is stored to be used as a memory. The robot repeats the same trajectory comparing the on-line scene with all stored images using correlation. This approach basically focuses on how to memorize the views.
- Jones *et al* [64], where a sequence of images and associated actions are stored in the robot memory. During navigation, the robot recovers the template that best matches the on-line frame. If the match is above a threshold, the robot runs the action associated to that template.
- Ohno *et al's* [101] solution is similar but faster than Jones': it only uses vertical lines from templates and on-line images to do the matching. It saves memory and computation time.

2.2 Outdoor Navigation

2.2.1 Outdoor navigation in structured environments

Outdoor navigation in structured environments refers to *road following*. *Road following* is the ability to detect the lines of the road and navigate consistently.

Pioneer on these techniques was Tsugawa [143], where a pair of stereo cameras were used to detect obstacles in an automatic car driving approach. One of the most outstanding efforts in *road following* is the NAVLAB project, by Thorpe [135,136]. The NAVLAB road following algorithm has three phases: in the first phase, a combination of color and texture pixel classification is performed defining a Gaussian distribution for each road and non-road pixels; in the second phase, a Hough transform and a subsequent votin process is applied to road pixels, to obtain the road vanishing point and orientation parameters; finally, pixels are classified again according to the determined road edges, and this classification is used for the next image in order to have a system adaptable to changing road conditions.

VITS [142] is a road following framework for outdoor environments equipped with an obstacle detection and avoidance sub-system. This system was firstly developed for the autonomous land vehicle ALVIN, which used a CCD color camera combined with a laser range scanner to gather images of the environment. The vision module of VITS generated a description of the road, either from the image data, from range information, or from both. This road description is transformed by the reasoning module into world coordinates to calculate the trajectory of the robot. The most robust element of the VITS system was the module to segment road pixels and non-road pixels.

Later, Pomerleau *et al* developed ALVINN, a new neural network-based navigation system, used also in the NAVLAB navigation architecture. There are several versions, from the initial one [106] dated from 1991 to the last one [63] dated from 1995 and known as ALVINN-VC. ALVINN is trained watching a human drive during several minutes with the aim at learning his/her reactions when driving on roads with varying properties. The human movements and turns when driving were incorporated to the robot behavior. The architecture consisted of a pre-trained network made of three inter-connected layers. Each layer had a pre-defined function. Mainly: the first one was a 30×32 node layer that contained recorded images, the third one was a 30 node layer that contained all steering angles represented by nodes, and finally the second one was a 5 node layer that was used as an interface to combine the first and second layer nodes. Instead of assigning specific node outputs for robot steering, values were determined from Gaussian distributions centered in each node associated to the road orientation. This guaranteed finer steering angles and slight changes in the output activation levels if the orientation of the road changed slightly.

A prominent and successful project for road-following was the EUREKA project Prometheus [47–50]. The goal of this project was to provide trucks with an automatic driving system to replace drivers in monotonous long driving situations. The system also included a module to warn the driver in potential or imminent dangerous situations.

2.2.2 Outdoor navigation in unstructured environments

In unstructured environments there are no regular properties that can be tracked for navigation. In these cases, two kind of situations can be found:

- The robot randomly explores the vicinity, like planetary vehicles. An example can be found in Wilcox's *et al* vehicle [146].
- The robot executes a mission with a goal position. In this case, a map of the areas in which the robot moves has to be created and a localization algorithm is also needed. A remarkable example of a mapping and positioning system is RATLER (*Robotic All-Terrain Lunar Explorer Rover*), proposed by Krotkov and Herbert [71] in 1995.

Another highly notable development is the planetary vehicle Mars Pathfinder [90], launched in December 1996 and landed in July 1997. The Mars Pathfinder consists of two components, a lander and a rover. The lander is a static component in which a stereo camera is fitted to shoot images of the Mars surface, while the rover is the mobile component which explores the environment. The rover mission path is determined by human operators in the Earth control station by selecting the goal point in 3D representations of previously captured images of the terrain. The position is determined using *dead reckoning* techniques and, to avoid cumulative errors, the rover navigation is limited to 10 m/day. Computation of differences between highest and lowest points of the inspected terrain permits cliff detection.

3 Visual Navigation: from the late 90's up to present

In the last decade, the techniques mentioned so far have matured into more refined versions, or have evolved into other more accurate and efficient systems. This variety of old and new techniques have extended the amount and quality of research in this area and their applications. This section surveys most of these studies distinguishing between *map-based* and *mapless solutions*.

3.1 Map-based Systems

This section considers techniques that build and/or use metric or topological maps. Navigation techniques which need a certain knowledge of the environment included in this paper are: *metric map-using navigation systems*, *metric map-building navigation systems* and *topological map-based navigation systems*. Systems that are able to build maps by themselves can perform this function from the complete environment or just from a portion of it. Therefore, this section also includes *local map-based navigation systems* and *visual sonar techniques*, given their potential relationship with producing metric maps, although some authors use them just to reactively avoid obstacles.

Table 1 Summary of the most outstanding visual navigation studies from 1987 to late 1990's

Authors	Indoor-Outdoor	Category	Method
[10, 68]	Indoor	Map based	Force Fields
[12, 103]	Indoor	Map based	Occupancy Grids
[16]	Indoor	Map based	Occupancy Grids
[34, 129, 130]	Indoor	Map based	Absolute Localization
[5]	Indoor	Map based	Absolute Localization
[91]	Indoor	Map based	Incremental Localization
[144]	Indoor	Map based	Incremental Localization
[19]	Indoor	Map based	Incremental Localization
[69, 94, 95, 104]	Indoor	Map based	Topological Map. Incremental Localization
[65]	Indoor	Map based	Landmark Tracking
[57]	Indoor	Map based	Landmark Tracking
[96]	Indoor	Map building	stereo 3D reconstruction
[134]	Indoor	Map building	Occupancy Grid
[11]	Indoor	Map building	Occupancy Grid
[137]	Indoor	Map building	Grid and Topological Representation
[114]	Indoor	Mapless	Optical Flow
[9]	Indoor	Mapless	Optical Flow
[29]	Indoor	Mapless	Optical Flow
[88]	Indoor	Mapless	Appearance-based Navigation
[64]	Indoor	Mapless	Appearance-based Navigation
[101]	Indoor	Mapless	Appearance-based Navigation
[143]	Outdoor	Structured Environments	Road Following
[47–50]	Outdoor	Structured Environments	Road Following
[142]	Outdoor	Structured Environments	Road Following
[136] , [135]	Outdoor	Structured Environments	Road Following
[106], [63]	Outdoor	Structured Environments	Road Following
[146]	Outdoor	Unstructured Environments	Random Exploration
[71]	Outdoor	Unstructured Environments	Given Mission Exploration
[90]	Outdoor	Unstructured Environments	Random Exploration

3.1.1 Metric Map-using and -building Navigation Systems

This group includes systems that need a complete map of the environment before the navigation starts. There are systems that are unable to map the environment and need to be equipped with it *(map-using systems)*. Other systems explore the environment and automatically build a map of it *(map-building systems)*. The navigation phase starts only if the map of the environment is available for the robot or after the map has been built. The map information can be directly used for navigation, or it can be post-processed to improve the map accuracy, and thus, achieve a more precise localization. This is the navigation technique that requires more computational resources, time and storage capability. Since outdoor environments can be large in size and extremely irregular, visual navigation techniques based on maps are in most occasions applied to indoor environments.

Map building and *self-localization* in the navigation environment are two functionalities that non-reactive systems tend to incorporate. In *map-building* standard approaches, it is assumed that the localization in the environment can be computed by some other techniques, while in pure *localization* approaches, the map of the environment is presumably available. Robots using this navigation approach need to track their own position and orientation in the environment in a continuous way.

This section focuses on metric map-based systems. Metric maps include information such as distances or map cell sizes with respect to a predefined coordinate system, and, in general, are also more sensible to sensor errors. Accurate metric maps are essential for good localization, and precise localization becomes necessary for building an accurate map.

If the exploration and mapping of an unknown environment is done automatically and on-line, the robot must accomplish three tasks: safe exploration/navigation, mapping and localization, preferably in a simultaneous way. *Simultaneous Localization and Mapping* (SLAM) and *Concurrent Mapping and Localization* (CML) techniques search for strategies to explore, map and self-localize simultaneously in unknown environments. This paper surveys those SLAM and CML systems which use only vision sensors to perform their task. Davison and Kita discuss in [27] about sequential localization and map building, review the state of the art and expose the future directions that this research domain should take. Furthermore, they present a tutorial of first-order relative position uncertainty propagation and a software to perform sequential mapping and localization.

Sim and Dudek propose in [121] a framework to learn a set of landmarks and track them across the sequence of images maximizing the correlation of the local image intensity. Landmark features are characterized with position parameters and subsequently used by the robot for self-localization. Sim and Dudek [122] extended their previous work with a new strategy for environment exploration and map building that maximizes coverage and accuracy and minimizes the odometry uncertainties. This proposal maps image features instead of performing a geometrical representation of the environment, operating and managing a framework presented in [121] and adapting an *Extended Kalman Filter* localization framework described in [125] and [74]. In the following stage, exploration policies are chosen among a great number of possibilities: (1) *seed spreader*, by which the robot follows a predefined navigation pattern throughout the environment; (2) *concentric*, where the robot follows concentric circular trajectories, with their center in the starting point, and the direction of movement changes at every circle; (3) *figure eight*, by which the robot follows eight-shaped concentric trajectories; (4) *random*, where the robot moves randomly; (5) *triangle*, by which the robot moves in concentric closed triangular trajectories; (6) *star*, where the robot moves along a set of rays that emanate from the starting point. Experimental results in [122] show that exploration efficiency, measured in observed images definitely inserted in the map divided by the total number of processed images, was maximum for the *concentric* policy, and minimum for the *star* policy. Besides, the mean error in odometry was maximum for the *random* policy and minimum for the *concentric* policy.

Sim *et al* [123, 124] outstanding work solves the SLAM problem with a stereo pair of cameras and a Blackwellised particle filter. The system implements a hybrid approach consisting of 3D landmark extraction for localization, and occupancy grid construction for safe navigation. AQUA is a visually guided amphibious robot developed by Dudek *et al* [30, 43]. This system runs on land and swims into the water. Using a stereo trinocular vision system, it is capable of creating 3D maps of the environment, locate itself and navigate.

In [25], Davison reports a new Bayesian framework that processes image information of a single standard camera to perform localization. Weak motion modeling is used to map strong distinguishable features, which are used to estimate the camera motion.

Wide angle cameras present a much wider field of view than standard lens cameras. Therefore, features are visible longer and are present in more frames. Due to the distortion introduced by a wide angle camera, a previous calibration process has to be performed in order to get corrected images from original frames. In [26], Davison *et al* extend their previous work by substituting the 50° standard camera with a 90° calibrated wide angle camera, leading to a significative improvement in movement range, accuracy and agility in motion tracking. Camera calibration improves the calculation of relative positions, and consequently improves the accuracy of the localization process. On the other hand, the Shi and Tomasi algorithm [119] is adopted in [26] to extract the position of the image features, which are used as landmarks to guide the navigation process. Experimental results prove that with a wide angle camera some aspects are improved: camera motion can be better identified, with particular improvements on rotational and translational movements estimation, the range of movements increase, and large motions or motions with great acceleration are better dealt with, since they appear much less abrupt. Therefore, the cases with trackable acceleration increase. Schleicher *et al* [115] use a top-down Bayesian method-based algorithm to perform a vision-based mapping process consisting in the identification and localization of natural landmarks from images provided by a wide-angle stereo camera. Simultaneously, a self-localization process is performed by tracking known features (landmarks) of the environment. The position of these features is determined through the combination of the *epipolar line* concept, characterisic from stereo theory, and the calculation of the *fundamental matrix*. The authors prove that using the redundancy of the information extracted from the images of both cameras increases the robustness and accuracy, and decreases the processing time of the procedure. The system with a wide-angle stereo camera is compared with a SLAM system that uses an single wide-angle camera to prove the improvements of the stereo-based system.

As for map-building systems that do not explore the possibility of simultaneous localization, we report systems that build 3D metric environment representations or, in some cases, use already existing maps in order to perform a safe navigation. Some researchers have focused their work on approaches to recover 3D environment structures and/or estimate robot motion models from vision information [92, 140].

Manessis *et al* address the 3D environment reconstruction problem using image sequences captured from n different camera views [82]. The two main contributions of this proposal are a new geometric theory for surface recovery from 3D sparse data and an algorithm based on a recursive *structure from motion* (SFM) method, which is used to estimate the location of 3D features and then to reconstruct the scene.

The classic process of building a 3D map using stereo images was refined by Wooden [148] under the DARPA-sponsored project Learning Applied to Ground Robots (LAGR), and particularly applied on its robot LAGR. The map building process consisted of four main steps:

- the captured stereo images were transformed into a three-dimensional representation by matching small patches in the two images,
- the real possible position of image points were deduced from the geometrical characteristics of the camera,
- a derivative was applied to the 3D map points to detect abrupt changes in slope, as for example, trees, rocks, etc..., and,
- in order to decrease the resolution of the map and smooth some variations, the result of the derivative was transformed into a cost map, where every point value was the average of the values over a defined 1.2 m $\times$ 1.2 m region.

Once the map had been created, a process of path planning is used to navigate through the environment.

When a robot explores an environment and constructs an occupancy grid, it makes approach of where the free space is. In this case, the object shape is not important, only the certainty that a fixed location is occupied by an object. In some cases, it is important to recognize the objects because they have to be picked up or manipulated, and, in other cases, it is paramount to recognize if the objects are on a table or lying on the floor. Following this trend, Tomono [139] proposed a high density indoor map-based visual navigation system on the basis of on-line recognition and shape reconstruction of 3D objects, using stored object models. A laser range finder was also used to complement the information provided by the camera. The proposed method contemplated three main issues:

- advanced objects model creation, before the navigation starts,
- on-line object recognition and localization, during the navigation stage and,
- placement of recognized objects in the 3D map of the environment.

Other *map-based navigation techniques* are those that impose a human-guided pre-training phase. Kidono *et al* [67] developed an approximation to this type of systems. In their contribution, a human guides the robot through an environment and during this guided route, the robot records images with a stereo camera and constructs the 3D map on-line, incrementally, frame by frame. After the map is built, the robot can repeat the same route from the starting point to the goal point, tracking features and computing the closest safe path. In this solution, odometry is used to support the stereo vision sensor.

An outstanding evolution of this technique using a calibrated wide angle camera came up from Royer *et al* [111]. The robot was guided by a human in a pre-training navigation stage, recording images from the trajectory. A complete 3D map of the environment was constructed off-line, using the information extracted from the pre-recorded images. A collection of useful landmarks and their 3D position in a global coordinate system were used for localization purposes, during the navigation stage. In the beginning of navigation, the robot had to self-localize in the starting point where it had been left, by comparing the current image to all stored key frames to find the best match. The selected subsequent images had to present a certain movement perception between them, to provide the system with trackable feature information. Losing perceptual movement caused problems to the algorithm. In these terms, the robot was able to follow the same complete pre-recorded trajectory, saving a lot of time in the positioning process. This approximation was basically directed to city navigation, rich in visual features, and where kinematic GPS can present a lot of places with hidden visibility.

Several undersea map construction techniques combined with a proper and accurate algorithm of position estimation can also be considered to belong to the CML category. In major cases, undersea bottom mosaics can be used by AUVs for navigation purposes. Haywood designed a system to mosaic underwater floors using images attached with accurate position coordinates [58]. Marks *et al* [83] developed a technique to implement real-time mosaics using correlation between on-line images and stored images. In a subsequent work, and following the same trend, Fleischer *et al* [36] improved the previous work [83] focusing on dead-reckoning error reduction. Previous systems often assumed that the seafloor was plane and static, and that the camera was facing it, making the image plane almost parallel to the seafloor plane. Gracias *et al* [46] proposed a method for mosaicing and localization that did not make any assumption on the camera motion or its relative position to the sea bottom. The system was based on motion computation by matching areas between pairs of consecutive images of a video sequence. Finally, an interesting contribution to underwater mosaicing and positioning was that by Xu and Negahdaripour in [149]. The vehicle position was computed integrating the motion of the camera from consecutive frames using Taylor series of motion equations, including the second order terms, which in previous research was usually ignored.

3.1.2 Topological Map-based Navigation Systems

A topological map is a graph-based representation of the environment. Each node corresponds to a characteristic feature or zone of the environment, and can be associated with an action, such as turning, crossing a door, stopping, or going straight ahead. Usually, there are no absolute distances, nor references to any coordinate frame to measure space. This kind of maps are suitable for long distance qualitative navigation, and specially for path planning. In general, they do not explicitly represent free space so that obstacles must be detected and avoided on line by other means. Topological maps are simple and compact, take up less computer memory, and consequently speed up computational navigation processes.

Winters and Santos-Victor [147] use an omnidirectional camera to create a topological map from the environment during a training phase. Nodes are images of characteristic places and links are sequences of various consecutive images between two nodes. During the navigation, the position is determined matching the online image with previously recorded images. The matching process is performed with an appearance-based method which consists in projecting every online image onto an eigenspace defined by the covariance matrix of a large image training set.

More recently, Gaspar *et al* use [147] to map indoor structured environments and emulate insect vision-based navigation capabilities [42]. The robot must be able to advance along corridors, recognize their end, turn into the correct directions and identify doors. The division of the map into nodes allows splitting the navigation task along an indoor environment into sub-goals. Every sub-goal is recognizable with landmarks and covers the movement between two nodes; for instance, two doors joined by a corridor. Navigation between two nodes works through detection of the corridor parallel sides and generation of the adequate control signals.

Another topological map-based navigation strategy for indoor environments comes from Košecka *et al* [70]. In a previous exploration stage, video is recorded and, for each frame, a gradient orientation histogram is computed. After that, a set of view prototypes are generated using Learning

Vector Quantization over the set of histograms gathered. Each histogram corresponds to a node in the topological map. During the navigation phase, the gradient orientation histogram of each frame is compared with the view prototypes to determine the location it most likely comes from using the nearest neighbour classification. In case the quotient of the distances with the nearest and the second closest histograms/views is below a certain threshold, the classification is considered correct and a location is obtained; otherwise, the classification is refined by comparing sub-images of the new image and the images in the database closest to the view prototypes.

In recent years, Remazeilles *et al* propose a system based on environment topological representation and a qualitative positioning strategy [108]. Nodes are represented by views captured in a training phase and edges represent the possibility of moving from one scene towards another. The robot navigates tracking landmarks over consecutive frames and keeping them inside the field of view. The localization strategy used in this approach is qualitative since it informs that the robot is in the vicinity of a node, instead of giving exact world coordinates.

One of the map-building robot applications that has proved to be greatly useful is that of *museum guiding robots* (in contrast to other solutions that need the museum map to navigate). These robots need to be autonomous in their missions, recognize people, guide them through different environments and also avoid static and dynamic obstacles, such as chairs, bookcases or other people. Because of the growing interest on this application, two relevant contributions are reviewed in the following. Thrun *et al* [138] developed MINERVA, a robot that uses two cameras combined with a laser sensor to build a complete map of the environment for the navigation process. Shen and Hu [118] presented ATLAS, a museum guiding robot that combines topological map building and appearance-based matching algorithms for localization. ATLAS also incorporates a human face detection algorithm [8] used to actively approach to new visitors.

3.1.3 Local Map-building Navigation Systems and Obstacle Avoidance

The strategies seen so far base their strength in a global description of the environment. This model can be obtained automatically by the robot, or in a previous human guided stage, but it has to be acquired before the robot begins the navigation. Since the early nineties, some authors have developed solutions where visual navigation processes are supported by the on-line construction of a local occupancy grid. In vision-based navigation, the local grid represents the portion of the environment that surrounds the robot and the grid size is determined by the camera field of view. This local information can be used for a subsequent complete map construction or simply updated frame by frame and used as a support for on-line safe navigation. Since robot decisions depend, to a large extent, on what the robot perceives at every moment in the field of view, these navigation techniques arise a debate about what can be considered *deliberative* and what can be considered *reactive* vision-based navigation techniques.

Badal *et al* reported a system for extracting range information and performing obstacle detection and avoidance in outdoor environments based on the computation of disparity from the two images of a stereo pair of calibrated cameras [6]. The system assumes that objects protrude high from a flat floor that stands out from the background. Every point above the ground is configured as a potential object and projected onto the ground plane, in a local occupancy grid called Instantaneous Obstacle Map (IOM). The commands to steer the robot are generated according to the position of obstacles in the IOM.

Gartshore *et al* [39] developed a map building framework and a feature position detector algorithm that processes images on-line from a single camera. The system does not use matching approaches. Instead, it computes probabilities of finding objects at every location. The algorithm starts detecting the objects boundaries for the current frame using the Harris edge and corner detectors [56]. Detected features are back projected from the 2D image plane considering all the potential locations at any depth. The positioning module of the system computes the position of the robot using odometry data combined with image feature extraction. Color or gradient from edges and features from past images help to increase the confidence of the object presence in a certain location. Experimental results tested in indoor environments set the size of the grid cells to 25 mm $\times$ 25 mm. The robot moved 100 mm between consecutive images.

Goldberg *et al* [45] introduced a stereo vision-based navigation algorithm for the rover planetary explorer MER, to explore and map locally hazardous terrains. The algorithm computes epipolar lines between the two stereo frames to check the presence of an object, computes the Laplacian of both images and correlates the filtered images to match pixels from the left image with their corresponding pixels in the right image. The work also includes a description of the navigation module GESTALT, which packages a set of routines able to compute actuation, direction, or steering commands from the sensor information.

Gartshore and Palmer presented in [40] a novel approach for complete unknown environment visual exploration and map construction with a limited field-of-view vision system. Afterwards they extended this work to more complex environments [41]. No landmarks or way-markers are used, and once the navigation has started, there is no human interaction. The exploration agent has to act as a human might do, observing the current view of the environment, exploring it, and deciding in which direction to advance to explore new areas. The main issues of the incremental map building process are:

- Vertical edges are extracted from the current frame to define obstacle boundaries. In some cases, these edges do not correspond to obstacles, but to shadows or specularities.
- To discriminate shadows or specularities from real obstacles, a confident measure is assigned to every edge point. Such a measure is a function of the number of times the object has been seen and the number of times the same area has been viewed.
- Features are connected with lines. These lines could either correspond to objects or just be connecting lines traced for triangulation purposes.
- Lines are also labeled with a confidence of being an obstacle. According to [81]: a candidate point to be labeled as an obstacle can not intersect the line that joins the camera with a real obstacle. The confidence measures are recalculated when points labeled as obstacles are viewed from another point of view as occluding other real obstacles.
- Obstacles and triangulation information are stored in discrete grids.

3.1.4 Visual Sonar

In recent years, *visual sonar* has become an original idea to provide range data and depth measurements for navigation and obstacle avoidance using vision in an analogous way to ultrasound sensors. Therefore, the originality of the concept is not in the navigation process itself, but in the way the data is obtained.

Martens *et al* were pioneers in using the concept of visual sonar for navigation and obstacle avoidance [84]. Their ARTMAP neural network combined sonar data and visual information from a single camera to obtain a more veridical perception of indoor environments. Real distance to obstacles was calculated from distances measured in pixels between obstacle edges and the bottom of the image. This distance computation is based on Horswill's idea [60]: the image is divided in eight columns, and the distance, measured in pixels from the bottom of the image to the object edge in every column, is proportional to the real world distance from the robot to the detected object.

Lenser and Veloso exposed a new visual sonar-based navigation strategy for the ROBOCUP competition and the AIBO robots [32, 73]. AIBOs are dog-shaped robots that have a single camera mounted on their heads. The system segments color images to distinguish floor, other robots, goals, the ball and other undefined objects. Once objects are defined, lines are radiated from the center of the image bottom, every 5°. An object is identified if there exists a continuous set of pixels in a scan line which corresponds to the same item class. Distance from object edges to the focus of the radial lines defines the real distance from the robot to the obstacle. The system builds a local grid of the environment with the robot centered on it, and avoids obstacles using contour following techniques. Since error increases with distance, anything separated more than 2 m can not be measured properly and, consequently, the algorithm only considers obstacles closer than 0.6 m.

Choi and Oh detect obstacle boundaries in images where the diagonal Mahalanobis color distance changes abruptly over points situated in radial lines, emanating from the calibrated camera to the rest of the image [17]. The system assumes that floor color and lighting conditions are constant. Odometry information is used to transform position coordinates on the image plane into world coordinates over a local occupancy grid, whose cells are labeled with a probability of being occupied by an obstacle. Experimental tests have been performed on cluttered offices and the local grid is constructed to support safe navigation. The paper also introduces the idea of omni-directional observation with a standard camera.

Martin computes depth from single camera images of indoor environments using also the concept of visual sonar [85]. The novelty of this method is the use of genetic programming to automatically discover the best algorithm to detect the ground boundaries in a training phase. These algorithms are then combined with reactive obstacle avoidance strategies, initially developed for sonar, and later adapted.

3.2 Mapless Navigation

This section includes a representative collection of mainly reactive visual navigation techniques. Reactive systems usually do not need any previous knowledge of the environment but they make navigation decisions as they perceive it. Those strategies process video frames as they gather them, and are able to produce enough information about the unknown and just perceived environment to navigate through it safely.

Prominent mapless visual navigation techniques here included are classified in accordance with the main vision technique or clue used during navigation: optical flow, feature detection and tracking, environment appearance, and extraction of qualitative information from an image.

3.2.1 Optical Flow-based Navigation Systems

Optical flow can be roughly defined as the apparent motion of features in a sequence of images. During navigation, the robot movement is perceived as a relative motion of the field of view, and, in consequence, it gives the impression that static objects and features move respect to the robot. To extract optical flow from a video stream, the direction and magnitude of translational or rotational scene feature movement must be computed at every pair of consecutive camera frames. Optical flow between two consecutive frames is usually represented by a vector for every pixel, where its norm depends on the motion speed and its direction represents the movement of the corresponding pixel in consecutive images. In some cases, the execution time and the computation resources required can be optimized by first extracting the image prominent features, such as corners or edges [56, 119], and then computing the optical flow only for these features. Image optical flow has been used by some researchers to implement reactive mobile robot navigation strategies, either for indoor or for outdoor environments. Object boundaries appear as regions with significant optical flow, and thus as regions to be avoided. Specularities or irregularities on the floor and textured floors also appear as regions with optical flow and therefore can be wrongly considered as obstacles causing errors during navigation.

Variations in the optical flow pattern or direction are used by Santos-Victor and Sandini to detect obstacles in a reactive, fast and robust approach for plane ground environments using a single camera [113]. Objects that arise from the ground plane cause variations in its normal flow pattern. To analyze and determine the presence of obstacles, the image flow field must be projected inversely onto the horizontal world plane. For translational motion, the projected flow must be constant for every point on the ground plane. Obstacles alter this assumption, presenting higher magnitudes or changes in the vector direction.

Camus *et al* [14] compute on-line the optical flow divergence from sequential wide angle frames to detect and avoid obstacles. Flow divergence is used for computing time to contact to obstacles in a qualitative way. To command the robot safely, one-dimensional maps are computed, where every heading direction is labeled with a potential risk of encountering obstacles.

Talukder *et al* [131] implemented a novel and robust optical flow-based solution to detect the presence of dynamic objects inside the camera field of view . It is applicable to robots with translational and/or limited rotational movement. The algorithm assumes that moving objects cause a discontinuity in optical flow orientation and changes in its magnitude with respect to the background pixels optical flow direction and magnitude. The system is developed and first tested using a single camera, and then using a stereo camera which provides depth information.

Some authors have proved that the combination of stereo vision, to obtain accurate depth information, and optical flow analysis provides better navigation results. Talukder and Matties extended [131] combining the stereo disparity field and optical flow to estimate depth, to model the robot egomotion and to detect moving objects of the scene [132]. In [13], stereo information is combined with the optical flow from one of the stereo images, to build an occupancy grid and perform a real-time navigation strategy for ground vehicles.

A simple and preliminary qualitative visual-based navigation system was proposed in [133] by Temizer and Kaelbling, under the DARPA-Mobile Autonomous Robot Software (MARS) program.

Although, to the best of the authors knowledge, this work does not represent a real progress in the field, it deserves to be included in a survey due to its simplicity and efficiency. The starting point for this strategy is the computation of image edge maps by detecting Laplacian of Gaussian (LOG) zero crossings. A patch matching procedure is subsequently applied using the edge maps of consecutive frames to compute the corresponding optical flow. Finally, the system turns away from zones of high optical flow, since they likely correspond to obstacles.

Visual navigation techniques based on optical flow have proved to be specially useful for *Unmanned Aerial Vehicles* (UAV) because optical flow gives the scene qualitative characteristics that can not be extracted in detail from single low quality images. Within this research trend, an important effort has been devoted to imitate animal behavior as far as the use and processing of apparent motion is concerned. Particularly, insects present a high degree of precision in their navigation and guidance systems, despite the simplicity of their nervous systems and small brains. Many authors have studied the way honeybees and other insects use optical flow to avoid obstacles and/or to navigate centered in the middle of corridors or narrow long ways. Experimental results found by Srinivasan *et al* [126] proved that bees fly balancing the path in the middle of tunnels, evaluating the apparent motion of images that perceive from both sides.

Van der Zwaan and Santos-Victor [145] implemented a UAV with a camera eye equivalent to an insect compound eye. The camera eye consisted of an array of photoreceptors each one connected to an electronic Elementary Motion Detector (EMD), which was able to calculate the local optical flow at its particular position. Contrast on optical flow calculations determined the presence of obstacles, while identifying the EMD polar coordinates that gave the changes on optical flow measures permitted to construct a local map with the location of the obstacles.

Netter and Franceschini [100] also implemented a UAV with a camera eye assembled with an array of photosensors and their corresponding EMDs. The information given by the set of EMDs was used to determine the presence of obstacles. Furthermore, when the UAV flew at a constant speed and altitude, a reference optical flow distribution was calculated from the equation that models the velocity of the artificial retina. To follow the terrain, the system varied thrust and rudders position to adjust the online computed optical flow with the optical flow reference.

Nonetheless, the use of optical flow information in terrain following applications for UAV presents limitations if the aircraft flies at low altitude, at high speed or if it is landing, even more if the camera is facing the ground. In these cases, optical flow estimation loses accuracy. Recently, Srinivasan *et al* [127] presented a new system to increase accuracy in the optical flow estimation for insect-based flying control systems. A special mirror surface is mounted in front of the camera, which is pointing ahead instead of pointing to the ground. The mirror surface decreases the speed of motion and eliminates the distortion caused by the perspective. Theoretically, the image should present a constant and low velocity everywhere, simplifying the optical flow calculation and increasing its accuracy. Consequently, the system increases the speed range and the number of situations under which the aircraft can fly safely. Particularly interesting is the work developed by Green *et al* [53], which describes the design of a UAV prototype called Closed Quarter Aerial Robot (CQAR) that flies into buildings, takes off and lands controlled by an insect-inspired optical flow-based system. This aerial vehicle incorporates a microsensor which weighs 4.8 grams, and is able to image the environment and compute the optical flow. The minimum flying speed of CQAR is 2 m/s, the turning radius is about 2.5 m and to avoid a detected obstacle it needs to turn about 5 meters before. Later, Green *et al* again emphasized the relevance of insect-based navigation strategies in an optical flow-based navigation system for UAVs that fly in near ground environments such as tunnels, caves, inside buildings or among trees [52]. The navigation principles applied in both [52] and [53] come from equation 1:

$$F = (v/d)\sin(\theta) - \omega, \tag{1}$$

where F is the optical flow, v is the translational velocity, d is the distance between the robot and an object, w is the angular velocity, and θ is the angle between the direction of travel and the aforementioned object. Equation 1 models the fact that optical flow of close obstacles has greater magnitude than the optical flow of obstacles that are at longer distances. Furthermore, optical flow magnitude is maximum for obstacles situated orthogonally to the robot motion direction.

To finish with this research line, Srinivasan *et al* presented an overview of illustrative insect-inspired navigation strategies for different situations, and the implementation of those strategies in several robots to test their feasibility [128].

Following a different research line, Cornall and Egan pointed out preliminary results corresponding to the analysis of optical flow patterns. Images were recorded during the UAV flight and transmitted to a ground station to be stored and analyzed. Optical flow example images of translation, pitch, roll to left/right and yaw motion were computed off-line and primary conclusions presented in [20].

In urban missions, UAVs have to fly usually among buildings and at low altitude, avoid obstacles situated at both sides or at the front, and make very steep turns or even U-turns at dead ends. This increases the possibility of crashing, thus the need of a very precise and safe navigation strategy. Hrabar *et al* present in [62] a novel navigation technique for UAVs to fly in between of urban canyons. The authors report a high degree of effectiveness of the system combining stereo forward-looking cameras for obstacle avoidance and two sideways-looking cameras for stable canyon navigation. Since the method is applied on UAVs, everything detected at the front is considered an obstacle. The system projects 3D stereo data onto a 2D map and performs a growing region process to extract obstacles. The robot stops keeping constant the altitude or simply changes direction depending on its distance to the obstacle. Besides, the robot always tries to balance the optical flow from both sides, moving to the direction of larger optical flow magnitude. The system implements a hierarchical architecture. Collisions with obstacles in the front are more probable than on the sides, therefore the stereo output is given priority over the optical flow output. The authors also expose an alternative for implementing this kind of hybrid systems, using two forward-facing fisheye cameras that have lenses with a 190° field of view. In this last case, the central part of an image can be used for stereo front obstacle avoidance, and the peripheral part can be used for computing the optical flow.

3.2.2 Appearance-based Navigation

Appearance-based strategies consist of two procedures. First, in a pre-training phase, images or prominent features of the environment are recorded and stored as model templates. The models are labeled with a certain localization information and/or with an associated control steering command. Second, in the navigation stage, the robot has to recognize the environment and self-localize in it by matching the current on-line image with the stored templates. The main problems of appearance-based strategies are finding an appropriate algorithm to create the environment representation and defining the on-line matching criteria.

Deviations between the route followed in the guided pre-training phase and the route navigated autonomously yield different sets of images for each case, and thus differences in the perception of the environment. Main researchers have focused their contributions on improving the way how images are recorded in the training phase, as well as on the subsequent image matching processes. There are two main approaches for environment recognition without using a map [87]:

– *Model-based Approaches.* They utilize pre-defined object models to recognize features in complicated environments and self-localize in it.
– *View-based Approach.* No features are extracted from the pre-recorded images. The self-localization is performed using image matching algorithms.

Matsumoto *et al* presented in [87–89] research results focusing on indoor route construction with standard or omnidirectional images, definition of correlation equations to model the concept of distance between images, and view creation using stereo divergence for outdoor environments where light conditions change most often.

Zhou *et al* [150] utilize histograms to describe the appearance of pre-recorded indoor images. Color, gradient, edge density and texture histograms are extracted from images, and stored in a multi-dimensional histogram database. The recognition of the environment during the navigation stage is reached by matching the multi-dimensional histogram of the current image with the multi-dimensional histogram of the stored templates. Working with histograms has two main advantages: it saves computation resources and it is simpler and quicker than entire images-based correlation processes.

Borenstein and Koren presented one of the first navigation and obstacle avoidance strategies for mobile robots based on building certainty grids and using the concept of potential fields [10]. Pioneers on applying potential fields in vision-based navigation and obstacle avoidance strategies were Haddad *et al* in [54]. Remazeilles *et al* use the concept of potential fields integrated in an appearance-based navigation method [107]. This system differs from typical appearance-based navigation strategies in the way that navigation is performed. The method defines an image database, which is a set of views built off-line, representing the whole navigable environment. When a navigation mission is defined, an image sequence corresponding to what the robot camera should see during the motion is extracted from the image database. The robot motion is the result of the on-line detection and matching process between the models included in the sequence and the perceived scenes. To navigate the environment, the robot tracks recognizable previously cataloged features. To fit these scene features in its field of view it uses the attractive potential fields to approximate them.

Morita *et al* reported in [97] a novel appearance-based localization approach for outdoor navigation. They extended their *Support Vector Machine* (SVM) -based algorithm, proposed in [98], to a novel SVM-based localization architecture that uses vision information from panoramic images. The SVM localization process consists of two main stages: feature or object learning, recognition and classification, and scene locations learning based on the previous feature classification. In this work, the authors show how panoramic images improve considerably training, matching and localization procedures, since the scenes are less dependent on the variation of the robot heading.

3.2.3 Image Qualitative Characteristics Extraction for Visual Navigation

Reactive visual techniques for robot navigation and obstacle avoidance are often devised around the extraction of image qualitative characteristics and their interpretation. There are two main types of reactive visual obstacle avoidance systems: *model-based* obstacle avoidance systems, which need predefined models of known objects, and *sensor-based* obstacle avoidance systems, which process every on-line sensor information to determine what could be an obstacle or what could be free space. These strategies can be included in what is known as qualitative navigation. Reactive navigation systems based on qualitative information avoid as much as possible using, computing or generating accurate numerical data such as distances, position coordinates, velocity, projections from image plane onto real world plane, or contact time to obstacles. In general, a coordinated behavior-based architecture is needed to manage all qualitative image information and the subsequent reactions [4].

Of particular relevance to this sort of navigation systems, due to their critical dependence on unprocessed sensorial data, is the change of the imaging conditions: illumination intensity, position of light sources, glossiness of the scene materials, etc. As a consequence, and mostly for outdoor applications, depending on time, weather conditions, season, etc. the performance of certain visual navigation systems can be seriously limited. One of the earliest solutions to these problems came from [135]. In 1997, Lorigo *et al* proposed a very low resolution vision-based obstacle avoidance system for unstructured environments [76]. The novelty of the solution was the construction of three simple modules that based the object detection criteria on brightness gradients, RGB color and HSV (hue, saturation, value) information. The goal of this approach was to navigate safely, with no destination point or pre-designed mission. The method assumed that all objects stayed on the plane ground, and that closer objects were in the bottom of the image while further objects were on the top of the image. Apart from the three modules working on brightness, RGB and HSV, a fourth one analyzed simultaneously their results to extract possible object boundaries. Afterwards, this information was used to generate motion commands.

The combination of a camera and other sensors such as laser or sonar has been applied in some reactive approaches to increase safety and the capabilities of the navigation process. CERES [15] is a behavior-based architecture that combines seven ultrasound transducers and a single grayscale camera. The vision module applies a Canny filter to extract edges from images. Edges are a clear evidence of the presence of obstacles. However, the floor carpet texture of the author's test environment generates edges that could be wrongly considered as obstacles. To avoid this misbehaviour, a threshold is imposed to eliminate false edges. The system transforms distances over images to real world distances using a rough camera calibration algorithm. For this particular case, the authors knew that the first fifth portion of the image, from bottom to top, corresponded to the closest 20 cm of the scene and that the other four fifths portion corresponded to the next real world 26 cm. Consequently, all those edges found in the first fifth of the image (bottom) were considered as obstacles to be avoided while the edges on the rest of the image (top) were considered to be far enough so as to be taken into account. Sonar is used to keep distance to the walls.

Other authors prefer to use a bi-level image segmentation process to segregate floor from objects [80]. Floor detection permits determining where the free navigable space is. In the ROBOCUP competition, the detection of the opponent robot and the ball becomes a challenging task to play the game properly. Fasola and Veloso [33] propose to use image color segmentation techniques for object detection, and gray-scale image processing for detecting the opponent robots.

The concept of fuzzy navigation, and particularly using visual sensors, has been used by several authors, combining the extraction of qualitative information from video frames with qualitative navigation algorithms based on fuzzy rules. One example of these techniques comes from Howard *et al* [61]. Their system is focused basically on ensuring a safe navigation through irregular terrains. It is assumed that the terrain can present rocks and variations on its slope. A region growing method based on edge detection and obstacle identification is used to detect rocks on the ground, while the terrain slope is calculated using existing techniques to retrieve 3D information and real Cartesian coordinates from a stereo pair of images. The size and number of rocks and the slope of the terrain are then classified by an algorithm that uses fuzzy terms such as big, small, rocky terrain, flat, sloped, steep, etc... Since the final goal of this system is to mimic as much as possible the human criteria used to classify a terrain, the system is trained by an expert which evaluates images taken from the robot point of view and judges the ability of the robot to navigate through the terrain. The difference between the human classification and the one done by the robot is an indication of the optimality of the system.

3.2.4 Navigation Techniques Based on Feature Tracking

Techniques for tracking moving elements (corners, lines, object outlines or specific regions) in a video sequence have become robust enough so as to be useful for navigation. Many times, the systems divide a tracking task into two sub-problems [141]: first, *motion detection*, which, given a feature to be tracked, identifies a region in the next frame where it is likely to find such a feature, and second, *feature matching*, by which the feature tracked is identified within the identified region.

In general, feature tracking-based navigation approaches do not comprise an obstacle avoidance module, but this task has to be implemented by other means. Although video tracking and mobile robot navigation belong to separate research communities, some authors claim to bridge them to motivate the development of new navigation strategies. Some authors center their research in detecting and tracking the ground space across consecutive images, and steering the robot towards free space.

Pears and Liang use homographies to track ground plane corners in indoor environments, with a new navigation algorithm called *H-based Tracker* [105]. The same authors extend their work in [75] using also homographies to calculate height of tracked features or obstacles above the ground plane during the navigation process.

The accuracy of the navigation strategy must be a strategic point in aerial motion where the speed is high, the processing time must be reduced and the tracking process needs to be more accurate. In [102], Ollero *et al* propose a new image tracking strategy that computes and uses a homography matrix to compensate the UAV motion and detect objects. This system improves their previous work [35] maintaining the tracking success despite the number of attempts is reduced. Zhou and Li [151], and Dao [24] compute and use the homography matrix to detect and track the ground plane over previously tracked image corners or edges using the Harris corner detector [56]. In a more recent work, other authors prefer to combine the concept of feature tracking with stereo 3-D environment reconstruction. In [112], stereo vision is used in a novel navigation strategy applicable to unstructured indoor/outdoor environments. This system is based on a new, faster and more accurate corner detector. Detected features are 3D positioned and tracked using normalized mean-squared differences and correlation measurements.

Support vision information with GPS data in outdoor environments is another possibility of increasing reliability in position estimation. Saripalli and Sukhatme combine a feature tracking algorithm with GPS positioning to perform a navigation strategy for the autonomous helicopter AVATAR [93]. The vision process combines image segmentation and binarization to identify pre-defined features, such as house windows, and a Kalman filter-based algorithm to match and track these windows.

The *scale invariant feature transform* (SIFT) method, developed by Lowe [78], stands out among other image feature or relevant points detection techniques, and nowadays has become a method commonly used in landmark detection applications. SIFT-based methods extract features that are invariant to image scaling, rotation, and illumination or camera view-point changes. During the robot navigation process, detected invariant features are observed from different points of view, angles, distances and under different illumination conditions and thus become highly appropriate landmarks to be tracked for navigation, global localization [117] and robust vision-based SLAM performance [116].

Several techniques have been developed for underwater environments. Some of them are of general application, such as image mosaicing systems, and others are more application oriented, such as the systems for pipeline or cable tracking. Mosaicing of the sea floor based on feature identification and tracking using texture-based operators and correlation-based procedures permits the robot to self-localize and thus identify its motion model [38]. *Pipeline or cable tracking* is an essential issue for accurate maintenance of thousands of kilometers of telecommunication or power cables between islands, countries and continents. In particular, unburied cables can be tracked using vision techniques. The first approaches to cable tracking were based on edge detectors and Hough transform, but they were unable to perform real-time cable tracking at video rates [55, 86, 110] by that time. Grau *et al* [51] propose a system that generates different texture groups and segment images in regions with similar textural behavior to track underwater cables or pipes. Foresti and Gentili [37] implemented a robust neural-based system to recognize underwater objects. Balasuriya and Ura [7] increased and improved the robustness of the existing systems by solving the eventual loss of the cable with dead-reckoning positioning prediction combined with 2D models of the cable. In a recent work, Antich and Ortiz [2] present a control architecture for AUV's navigation based on a cable tracking algorithm that looks for edge alingments related with the cable sides. Finally, the same authors include a new sonar-based algorithm in the vision-based cable tracking architecture to escape from trapping zones [3].

Moving-target vision-based tracking strategies have also become a motivating research trend, specially to improve current fish shoal detection and tracking techniques. Between 2000 and 2001 some relevant solutions were presented by Silpa-Anan *et al* [120] and Fan and Balasuriya [31] respectively. Fan and Balasuriya [31] presented a process based on two parallel stages: object speed calculation represented with optical flow, and moving objects positioning with template-matching techniques. Rife and Rocks went a step forward implementing a system capable of recognizing and tracking only jellyfish [109].

Estimation of camera motion in underwater unstructured environments, where there are no pipes or cables to track, in other words, an environment with no defined references, becomes another complicated and challenging navigation problem. In this type of navigation strategies, references have to be found in the image, defined and tracked. There are fundamentally three methods that are used for this purpose: optical flow, feature tracking or gradient methods. Optical flow-based methods and feature tracking-based methods can cause failure in algorithms due to scattering effects, bad image quality or deficient illumination under the sea. Gradient methods use scene properties such as depth, range, shapes or color intensity, that are computationally more efficient and more accurate [23]. *Station keeping* is one of the problems that can be solved estimating the motion of the camera. Station keeping consists in holding the robot around a fixed position on the undersea floor that has a special interest at that moment. The AUV will hover around the point maintaining the center of the camera pointing on it. Examples of outstanding related solutions can be found in [77, 99] and [21].

4 Conclusions

In the last decades, vision has become one of the most cheap, challenging and promising via for robots to perceive the environment. Accordingly, the number of prominent navigation approaches based on vision sensors have increased exponentially. Visual navigation techniques have been applied on almost all environments and in all kind of robots. The most outstanding pieces of work related with visual navigation from the early nineties until nowadays have been included in this paper to be used as a reference for novel and experienced researchers that want to first explore the possibilities of this domain. Map-based navigation techniques have been contrasted with those systems that do not need a map for navigation in an attempt to proceed gradually from the most deliverative navigation techniques to the most pure reactive solutions.

Tables 2 and 3 show an overview of the most outstanding publications referenced in this survey, from the late nineties to present. The list has been sorted by type of vehicle to facilitate analysis and comparison of the different strategies used in each of these vehicles during the last decade. The following conclusions can be drawn from the aforementioned tables:

– Ground robots do not span the whole amount of applications revised in this survey but cover almost all the strategies considered. Apparently, some strategies seem to be exclusive of ground robots because they are rarely found in aerial or underwater vehicles. This is the case of:

Table 2 Summary of the most outstanding visual navigation studies from the late 90's to the present

Authors	Type of vehicle	Category	Strategy	Type of visual sensor
[121, 122]	Ground	Map building	Visual SLAM. Landmarks localization and tracking	Single standard camera
[123, 124]	Ground	Map building	Visual SLAM. Landmarks extraction and occupancy grids	Stereo cameras
[25]	Ground	Map building	Visual SLAM. Map feature extraction	Single standard camera
[26]	Ground	Map building	Visual SLAM. 3D sparse mapping of interest points	Single wide angle camera
[81]	Ground	Map building	3D construction of an occupancy grid	Single standard camera
[139]	Ground	Map building	3D high density map and object recognition	Single standard camera
[67]	Ground	Map building	Human guided pre-training	Stereo cameras
[111]	Ground	Map building	Human guided pre-training	Single wide angle camera
[147]	Ground	Map building	Topological map	Omnidirectional camera
[42]	Ground	Map building	Topological map	Omnidirectional camera
[70, 108]	Ground	Map building	Topological map	Single standard camera
[6]	Ground	Map building	Local occupancy grid	Stereo cameras
[39]	Ground	Map building	Local occupancy grid	Single standard camera
[45]	Ground	Map building	Local occupancy grid	Stereo cameras
[40]	Ground	Map building	Local occupancy grid	Single standard camera
[17,32,73,84,85]	Ground	Map building	Visual sonar	Single standard camera
[113]	Ground	Mapless	Optical flow	Single standard camera
[14]	Ground	Mapless	Optical flow	Single wide angle camera
[131, 132]	Ground	Mapless	Optical flow combined with stereo information	Stereo cameras
[133]	Ground	Mapless	Optical flow	Single standard camera
[87, 89, 107]	Ground	Mapless	Appearance-based method	Standard or omnidirectional single camera
[98]	Ground	Mapless	Appearance-based method	Panoramic camera

  - visual SLAM systems, because the computation of the environtment model seems to be feasible only for indoor scenarios,
  - homography-based navigation systems, because of their dependency on floor detection and tracking and finally,
  - visual sonar systems and human pre-guided map building systems.

- The use of UAVs has generalized during the last decade and, as a consequence, navigation solutions for this kind of vehicles have improved in safety, accuracy and scope. The vast majority of UAVs use mapless navigation systems. We should highlight here the insect-inspired solutions for optical

Table 3 Summary of the most outstanding visual navigation studies from the late 90's to the present: continuation

Authors	Type of vehicle	Category	Strategy	Type of visual sensor
[138]	Ground	Map building	Museum guiding robot: complete 3D map	Single standard camera
[118]	Ground	Map building	Museum guiding robot: topological map	Single standard camera
[15, 76]	Ground	Mapless	Image qualitative characteristics extraction	Single standard camera
[80]	Ground	Mapless	Image qualitative characteristics extraction	Single standard camera
[61]	Ground	Mapless	Image qualitative characteristics extraction	Stereo cameras
[24, 75, 105, 151]	Ground	Mapless	Features tracking; homography	Single standard camera
[112]	Ground	Mapless	Features tracking; homography	Stereo cameras
[78, 116]	Ground	Mapless	Features tracking: SIFT	Single standard camera
[100, 145]	UAV	Mapless	Optical flow: insect inspired (EMD)	Camera eye
[127]	UAV	Mapless	Optical flow: insect inspired (EMD)	Single standard camera
[52, 53]	UAV	Mapless	Optical flow: insect inspired (EMD)	Single mini wireless camera
[62]	UAV	Mapless	Optical flow: insect inspired	Stereo cameras looking forward combined with two sideways looking cameras
[93]	UAV	Mapless	Features tracking	Single wide angle camera
[30, 43]	Amphibious	Map building	Visual SLAM. 3D total map building	Trinocular stereo cameras
[102]	AUV	Mapless	Features tracking; homography	Single standard camera
[2, 3, 51, 55, 86, 110]	AUV	Mapless	Cable tracking	Single standard camera
[21, 77, 99]	AUV	Mapless	Station keeping	Single standard camera
[36, 46, 58, 83, 149]	AUV	Map building	Underwater floor mosaicing	Single standard camera

flow processing as well as for feature tracking and detection. Some of these aerial robots have also gained in accuracy, operativity and robustness incorporating compound cameras or camera eyes.
– Visual navigation systems for AUVs have to cope with the special characteristics of light propagation undersea. Researchers have mostly focused on developing and/or evolving general visual navigation techniques based on feature tracking, mainly for mosaicing applications. Researchers have also focused on devising application-oriented navigation strategies, in many cases for tracking underwater cables or pipelines.
– Finally, very few amphibious solutions have been proposed.

To conclude this review, it is convenient to note that during the last decade new vision techniques have been applied to vision-based navigation systems, such as those systems based on homographies, visual sonar or visual SLAM. But, it is also important the impulse and progress of other techniques only used in the past by a minority, and that have revealed to be essential for UAVs navigation. Examples of these last techniques are those inspired on insect behavior.

Acknowledgements

The authors are grateful to the anonymous reviewers for their helpful and insightful comments which have certainly improved this paper. We also thank Lucrecia Rallo Fabra for her diligence and for her expert corrections and suggestions.

References

1. J. Abascal, E. Lazkano, and B. Sierra. Behavior-Based Indoor Navigation. *Ambient Intelligence for Scientific Discovery. Foundations, Theories, and Systems*, LNAI(3345):263–285, 2005.
2. J. Antich and A. Ortiz. Development of the Control Architecture of a Vision-guided Underwater Cable Tracker. *International Journal of Intelligent Systems*, 20(5):477–498, 2005.
3. J. Antich, A. Ortiz, and G. Oliver. A Control Strategy for Fast Obstacle Avoidance in Troublesome Scenarios: Application in Underwater Cable Tracking. In *Proc. of 7th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC)*, 2006.
4. R.C. Arkin. *Behaviour Based Robotics.* The MIT Press, 1998.
5. S. Atiya and G.D. Hager. Real Time Vision-Based Robot Localization. *IEEE Transactions on Robotics and Automation*, 9(6):785–800, 1993.
6. S. Badal, S. Ravela, B. Draper, and A. Hanson. A Practical Obstacle Detection and Avoidance System. In *Proc. of 2nd IEEE Workshop on Applications of Computer Vision*, pages 97–104, 1994.
7. B. Balasuriya and T. Ura. Underwater Cable Following by Twin-Burger 2. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, 2001.
8. N. Bellotto and H. Hu. Multisensor Integration for Human-Robot Interaction. *Journal of Intelligent Cybernetic Systems*, 1, 2005.
9. A. Bernardino and J. Santos-Victor. Visual Behaviours for Binocular Tracking. *Robotics and Autonomous Systems*, 25(3-4):137–146, 1998.
10. J. Borenstein and Y. Koren. Real-time Obstacle Avoidance for Fast Mobile Robots. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):1179–1187, 1989.
11. J. Borenstein and Y. Koren. Real Time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 572–577, 1990.
12. J. Borenstein and Y. Koren. The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots. *IEEE Transactions on Robotics and Automation*, 7(3):278–288, 1991.
13. C. Braillon, K. Usher, C. Pradalier, J.L. Crowley, and C. Laugier. Fusion of Stereo and Optical Flow Data Using Occupancy Grid. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 2302–2307, 2006.
14. T. Camus, D. Coombs, M. Herman, and T.H. Hong. Real-Time Single-Workstation Obstacle Avoidance Using Only Wide-Field Flow Divergence. In *Proc. of 13th International Conference on Pattern Recognition (ICPR). Applications and Robotic Systems*, 1996.
15. M.T. Chao, T. Braunl, and A. Zaknich. Visually-Guided Obstacle Avoidance. In *Proc. of 6th Int'l Conf. on Neural Information Processing ICONIP*, volume 2, pages 650–655, 1999.
16. R. Chatila and J.P. Laumond. Position Referencing and Consistent World Modeling for Mobile Robots. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 138–145, 1985.
17. Y.H. Choi and S.Y. Oh. Visual Sonar Based Localization Using Particle attraction and Scattering. In *Proc. of IEEE International Conference on Mechatronics and Automation*, pages 449–454, July 2005.
18. H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, , and S. Thrun. *Principles of Robot Motion.* The MIT Press, 2005.
19. H.I. Christensen, N.O. Kirkeby, S.Kristensen, and L.Knudsen. Model-Driven Vision for Indoor Navigation. *Robotics and Autonomous Systems*, 12:199–207, 1994.
20. T. Cornall and G. Egan. Optic Flow Methods Applied to Unmanned air Vehicles. *Academic Research Forum, Department of Electrical and Computer Systems Engineering, Monash University*, 2003.
21. F.R. Dalgleish, S.W. Tetlow, and R.L. Allwood. Hammerhead: an AUV with an Integral Laser Imaging Sensor. *Oceanology*, 2004.
22. F.R. Dalgleish, S.W. Tetlow, and R.L. Allwood. Vision-Based Navigation of Unmanned Underwater Vehicles: a Survey. Part I: Vision Based Cable-, Pipeline- and Fish Tracking. *Proc. of the Institute of Marine Engineering, Science and Technology. Part B, Journal of Marine Design and Operations*, B(7):51–56, 2004.
23. F.R. Dalgleish, S.W. Tetlow, and R.L. Allwood. Vision-Based Navigation of Unmanned Underwater Vehicles: a Survey. Part II: Vision Based Station Keeping and Positioning. *Proc. of the Institute of Marine Engineering, Science and Technology. Part B, Journal of Marine Design and Operations*, B(8):13–19, 2004.
24. N.X. Dao, B. You, and S. Oh. Visual Navigation for Indoor Mobile Robots Using a Single Camera. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 1992–1997, 2005.

25. A.J. Davison. Real Time Simultaneous Localisation and Mapping with a Single Camera. *Proc. of International Conference on Computer Vision (ICCV)*, 2003.
26. A.J. Davison, Y. González, and N. Kita. Real-Time 3D SLAM with Wide-Angle Vision. In *Proc. of IFAC Symposium on Intelligent Autonomous Vehicles*, 2004.
27. A.J. Davison and N. Kita. Sequential Localization and Map Building for Real Time Computer Vision and Robotics. *Robotics and Autonomous Systems*, 36(4):171–183, 2001.
28. G.N. DeSouza and A.C. Kak. Vision for Mobile Robot Navigation : A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):237–267, 2002.
29. A. Dev, B. Krse, and F. Groen. Navigation of a Mobile Robot on the Temporal Development of the Optic Flow. In *Proc. of IEEE Int'l Conf. of Intelligent Robots and Systems (IROS)*, pages 558–563, 1997.
30. G. Dudek, M. Jenkin, C. Prahacs, A. Hogue, J. Sattar, P. Giguère, A. German, H. Liu, S. Saunderson, A. Ripsman, S. Simhon, L.A. Torres-Mendez, E. Milios, P. Zhang, and I. Rekleitis. A Visually Guided Swimming Robot. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, 2005.
31. Y. Fan and B. Balasuriya. Optical Flow Based Speed Estimation in AUV Target Tracking. *Proc. of IEEE Oceans*, 2001.
32. J. Fasola, P.E. Rybski, and M. Veloso. Fast Goal Navigation with Obstacle Avoidance Using a Dynamic Local Visual Model. In *Proc. of SBAI'05, The VII Brazilian Symposium of Artificial Intelligence*, 2005.
33. J. Fasola and M. Veloso. Real-Time Object Detection using Segmented and Grayscale Images. In *Proc. of IEEE Int'l Conf. on Robotics and Automations (ICRA)*, pages 4088–4090, 2006.
34. F.Dellaert, D.Fox, W.Burgard, and S.Thrun. Monte carlo localization for mobile robots. In *Proc. of IEEE Int'l Conference on Robotics and Automation (ICRA)*, pages 1322–1328, 1999.
35. J. Ferruz and A. Ollero. Real-time Feature Matching in Image Sequences for Non-structured Environments. Applications to Vehicle Guidance. *Journal of Intelligent and Robotic Systems*, 28:85–123, 2000.
36. S.D. Fleischer, R.L. Marks, and S.M. Rock. Improving Real-Time Video Mosaicing of the Ocean Floor. In *Proc. of IEEE Oceans*, 1995.
37. G.L. Foresti and S. Gentili. A Hierarchical Classification System for Object Recognition in Underwater Environments. *IEEE Journal of Oceanic Engineering*, 1(27):66–78, 2002.
38. R. Garcia, X. Cufi, and Ll. Pacheco. Image Mosaicking for Estimating the Motion of an Underwater Vehicle. In *Proc. of 5th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC)*, 2000.
39. R. Gartshore, A. Aguado, and C. Galambos. Incremental Map Buildig Using an Occupancy Grid for an Autonomous Monocular Robot. In *Proc. of Seventh International Conference on Control, Automation, Robotics and Vision ICARCV*, pages 613–618, 2002.
40. R. Gartshore and P. Palmer. Exploration of an Unknown 2D Environment Using a View Improvement Strategy. *Towards Autonomous Robotic Systems*, pages 57–64, 2002.
41. R. Gartshore, P. Palmer, and J. Illingworth. A Novel Exploration Algorithm Based on a Improvement Strategy. *International Journal of Advanced Robotic Systems*, 2(4):287–294, 2005.
42. J. Gaspar, N. Winters, and J. Santos-Victor. Vision-Based Navigation and Environmental Representations with an Omni-directional Camera. *IEEE Transactions on Robotics and Automation*, 16(6):890–898, 2000.
43. C. Georgiades, A. German, A. Hogue, H. Liu, C. Prahacs, A. Ripsman, R. Sim, L.A. Torres, P. Zhang M. Buehler, G. Dudek, M. Jenkin, and E. Milios. An aquatic Walking Robot. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, 2004.
44. G. Giralt, R. Sobek, and R. Chatila. A Multi-Level Planning and Navigation System for a Mobile Robot; A Firts Approach to Hilare. In *Proc. of Sixth Intl Joint Conf. on Artificial Intelligence*, pages 335–337, 1979.
45. S.B. Goldberg, M. W. Maimone, and L. Matthies. Stereo Vision and Rover Navigation Software for Planetary Exploration. In *Proc. of IEEE Aerospace conference Proceedings*, pages 5–2025,5–2036, 2002.
46. N. Gracias and J. Santos-Victor. Underwater Video Mosaics as Visual Navigation Maps. *Computer Vision and Image Understanding*, 1(79):66–91, 2000.
47. V. Graefe. Driveless Highway Vehicles. In *Proc. of Int'l Hi Tech Forum*, pages 86–95, 1992.
48. V. Graefe. Vision for Autonomous Mobile Robots. In *Proc. of IEEE Workshop on Advanced Motion Control*, pages 57–64, 1992.
49. V. Graefe. Vision for Intelligent Road Vehicles. In *Proc. of the IEEE Symposium of Intelligent Vehicles*, pages 135–140, 1993.
50. V. Graefe and K.Kuhnert. Vision-Based Autonomous Road Vehicles. *Vision Based Vehicle Guidance, Springer-Verlag New York*, pages 1–29, 1992.
51. A. Grau, J. Climent, and J. Aranda. Real-time Architecture for Cable Tracking Using Texture Descriptors. In *Proc. of IEEE Oceans Conference*, volume 3, pages 1496–1500, 1998.
52. W.E. Green, PY. Oh, and G.L. Barrows. Flying Insects Inspired Vision for Autonomous Aerial Robot Maneuvers in Near-earth Environments. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 2347–2352, 2004.
53. W.E. Green, PY. Oh, K. Sevcik, and G.L. Barrows. Autonomous Landing for Indoor Flying Robots Using Optic Flow. In *Proc. of IMECE International Mechanical Engineering Congress*, volume 2, pages 1341–1346, 2003.
54. H. Haddad, M. Khatib, S. Lacroix, and R. Chatila. Reactive Navigation in Outdoor Environments Using Potential Fields. In *Proc. IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 1232–1237, 1998.
55. J. Hallset. Testing the Robustness of an Underwater Vision System. *Laplante P, Stoyenko A (eds) Real-time imaging: theory, techniques, and applications. IEEE Press*, pages 225–260, 1996.
56. C. Harris and M. Stephens. Combined Corner and Edge Detector. In *Proc. of Fourth Alvey Vision Conference*, pages 147–151, 1988.

57. M. Hashima, F. Hasegawa, S. Kanda, T. Maruyama, and T. Uchiyama. Localization and Obstacle Detection for a Robot for Carrying Food Trays. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 345–351, 1997.
58. R. Haywood. Acquisition of a Micro Scale Photographic Survey Using an Autonomous Submersible. In *Proc. of IEEE Oceans*, volume 18, pages 1423– 1426, 1986.
59. B.K.P. Horn and B.G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.
60. I. Horswill. Visual Collision Avoidance by Segmentation. In *In Proc. of ARPA94*, pages II:1135–1141, 1994.
61. A. Howard, E. Tunstel, D. Edwards, and A. Carlson. Enhancing Fuzzy Robot Navigation Systems by Mimicking Human Visual Perception of Natural Terrain Traversability. In *IFSA World Congress and 20th NAFIPS International Conference*, volume 1, pages 7–12, 2001.
62. S. Hrabar, G.S. Sukhatme, P. Corke, K. Usher, and J. Roberts. Combined Optic-Flow and Stereo-Based Navigation of Urban Canyons for a UAV. In *Proc. of IEEE Int'l Conf. of Intelligent Robots and Systems (IROS)*, pages 3309–3316, 2005.
63. T.M. Jochem, D.A. Pomerleau, and C.E. Thorpe. Vision Based Neural Network Road and Intersection Detection and Traversal. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, volume 3, pages 344–349, 1995.
64. S.D. Jones, C. Andresen, and J.L. Crowley. Appearance Based Processes for Visual Navigation. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 551–557, 1997.
65. M. Kabuka and A.E. Arenas. Position Verification of a Mobile Robot Using Standard Pattern. *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, 3(6):505–516, 1987.
66. A.C. Kak and G.N. de Souza. Robotic Vision: What Happened to the Visions of Yesterday? In *Proc. of 16th International Conference on Pattern Recognition*, pages 839–847, 2002.
67. K. Kidono, J. Miura, and Y. Shirai. Autonomous Visual Navigation of a Mobile Robot Using a Human-Guided Experience. *Robotics and Autonomous Systems*, 40(2-3):121–130, 2002.
68. D. Kim and R. Nevatia. Simbolic Navigation with a Generic Map. In *Proc. of IEEE Workshop on Vision for Robots*, pages 136–145, 1995.
69. A. Kosaka and A.C. Kak. Fast Vision-Guided Mobile Robot Navigation Using Model-Based Reasoning and Prediction of Uncertainties. *Computer Vision, Graphics, and Image Understanding*, 56(3):271–329, 1992.
70. J. Kosecka, L. Zhou, P. Barber, and Z. Duric. Qualitative Image Based Localization in Indoors Environments. In *Proc. of IEEE Intl Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 3–8, 2003.
71. E. Krotkov and M. Herbert. Mapping and Positioning for a Prototype Lunar Rover. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 2913–2919, 1995.
72. J. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.
73. S. Lenser and M. Veloso. Visual Sonar: Fast Obstacle Avoidance Using Monocular Vision. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 886–891, 2003.
74. J. Leonard and H. F. Durrant-Whyte. Simultaneous Map Building and Localization for an Autonomous Mobile Robot. In *Proc. of IEEE Intl Workshop on Intelligent Robots and Systems*, pages 1442–1447, 1991.
75. B. Liang and N. Pears. Visual Navigation Using Planar Homographies. In *Proc. of IEEE Int'l Conf. on Robotics and Automations (ICRA)*, volume 1, pages 205–210, 2002.
76. L.M. Lorigo and R.A. Brooks nad W.E. Grimson. Visually-Guided Obstacle Avoidance in Unstructured Environments. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 373–379, 1997.
77. J-F. Lots, D.M. Lane, and E. Trucco. Application of a 2 1/2 D Visual Servoing to Underwater Vehicle Station-Keeping. In *Proc. of IEEE Oceans*, 2000.
78. D.G. Lowe. Object Recognition From Local Scale Invariant Features. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 1150–1157, 1999.
79. B.D. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stero Vision. In *Proc. of DARPA Image Uderstanding Workshop*, pages 121–130, 1981.
80. JM. Maja, T. Takahashi, Z.D. Wang, and E. Nakano. Real-Time Obstacle Avoidance Algorithm for Visual Navigation. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 925–930, 2000.
81. A. Manessis and A. Hilton. Scene Modelling from Sparse 3D Data. *Journal of Image and Vision Computing*, 23(10):900–920, 2005.
82. A. Manessis, A. Hilton, P. Palmer, P. McLauchlan, and X. Shen. Reconstruction of Scene Models from Sparse 3D Structure. In *Proc. of IEEE Intl Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2666–2673, 2000.
83. R.L. Marks, S. Rocks, and M.J. Lee. Real-Time Video Mosaicing of the Ocean Floor. In *Proc. of Symposium on Autonomous Underwater Vehicle Technology*, pages 21–27, 1994.
84. S. Martens, P. Gaudian, and G.A. Carpenter. Mobile Robot Sensor Integration with Fuzzy ARTMAP. In *Proc. of the 1998 IEEE ISIC/CIRA/ISAS Joint Conferenc*, pages 307–312, 1998.
85. M. C. Martin. Evolving Visual Sonar: Depth from Monocular Images. *Pattern Recognition Letters*, 27:1174–1180, 2006.
86. S. Matsumoto and Y. ito. Real-time Based Tracking of Submarine Cables for AUV/ROV. In *Proc. of IEEE Oceans*, volume 3, pages 1997–2002, 1995.
87. Y. Matsumoto, K. Ikeda, M. Inaba, and H. Inoue. Visual Navigation Using Omnidirectional View Sequence. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 317–322, 1999.
88. Y. Matsumoto, M. Inaba, and H. Inoue. Visual Navigation Using View Sequenced Route Representation. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, volume 1, pages 83–88, 1996.

89. Y. Matsumoto, K. Sakai, M. Inaba, and H. Inoue. View-Based Approach to Robot Navigation. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)* , pages 1702–1708, 2000.
90. L. Matthies, E. Gat, R. Harrison, B. Wilcox, R. Volpe, and T. Litwin. Mars Microrover Navigation: Performance Evaluation and Enhancement. In *Proc. of IEEE Int'l Conf. of Intelligent Robots and Systems (IROS)*, volume 1, pages 433–440, 1995.
91. L. Matthies and S.A. Shafer. Error Modeling in Stereo Navigation. *IEEE Journal of Robotics and Automation*, 3(3):239–248, 1987.
92. P. McLauchlan and D. Murray. A Unifing Framework for Structure and Motion Recovery from Image Sequences. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 314–320, 1995.
93. L.O. Mejias, S. Saripalli, G.S. Sukhatme, and P.C. Cervera. Detection and Tracking of External Features in an Urban Environment Using an Autonomous Helicopter. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 3972–3977, 2005.
94. M. Meng and A.C. Kak. Mobile Robot Navigation Using Neural Networks and Nonmetrical Environment Models. In *Proc. of IEEE Control Systems*, pages 30–39, 1993.
95. M. Meng and A.C. Kak. NEURO-NAV: A Neural Network Based Architecture for Vision-Guided Mobile Robot Navigation Using non-Metrical Models of the Environment. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, volume 2, pages 750–757, 1993.
96. H.P. Moravec. The Stanford Cart and the CMU Rover. In *Proceedings of IEEE*, volume 71, pages 872–884, 1983.
97. H. Morita, M. Hild, J. Miura, and Y. Shirai. Panoramic View-Based Navigation in Outdoor Environments Based on Support Vector Learning. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 2302–2307, 2006.
98. H. Morita, M. Hild, J. Miura, and Y. Shirai. Panoramic View-Based Navigation in Outdoor Environments Based on Support Vector Learning. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 1885–1890, 2006.
99. S. Negahdaripour, X. Xu, and L. Jin. Direct Estimation of Motion From Seafloor Images for Automatic Station-Keeping of Submersible Platforms. *IEEE Journal of Oceanic Engineering*, 24(3):370–382, 1999.
100. T. Netter and N. Franceschini. A Robotic Aircraft that Follows Terrain Using a Neuromorphic Eye. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, volume 1, pages 129–134, 2002.
101. T. Ohno, A. Ohya, and S. Yuta. Autonomous Navigation for Mobile Robots Referring Pre-Recorded Image Sequence. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, volume 2, pages 672–679, 1996.
102. A. Ollero, J. Ferruz, F. Caballero, S. Hurtado, and L. Merino. Motion Compensation and Object Detection for Autonomous Helicopter Visual Navigation in the COMETS System. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 19–24, 2004.
103. G. Oriolo, G. Ulivi, and M. Vendittelli. On-Line Map Building and Navigation for Autonomous Mobile Robots. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 2900–2906, 1995.
104. J. Pan, D.J. Pack, A. Kosaka, and A.C. Kak. FUZZY-NAV: A Vision-Based Robot Navigation Architecture Using Fuzzy Inference for Uncertainty-Reasoning. In *Proc. of IEEE World Congress Neural Networks*, volume 2, pages 602–607, 1995.
105. N. Pears and B. Liang. Ground Plane Segmentation for Mobile Robot Visual Navigation. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 1513–1518, 2001.
106. D.A. Pomerleau. ALVINN: an Autonomous Land Vehicle in a Neural Network. Technical Report CMU-CS-89-107, Carnegie Mellon University, 1989.
107. A. Remazeilles, F. Chaumette, and P. Gros. Robot Motion Control from a Visual Memory. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 4695–4700, 2004.
108. A. Remazeilles, F. Chaumette, and P. Gros. 3D Navigation Based on a Visual Memory. In *Proc. of IEEE Int'l Conf. on Robotics and Automations (ICRA)*, pages 2719–2725, 2006.
109. J. Rife and S. Rock. A Low Energy Sensor for AUV-Based Jellyfish Tracking. In *Proc. of 12th International Symposium on Unmanned Untethered Submersible Technology*, 2001.
110. P. Rives and J.J Borelly. Visual Servoing Techniques Applied to an Underwater Vehicle. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 20–25, 1997.
111. E. Royer, J. Bom, M. Dhome, B. Thuillot, M. Lhuillier, and F. Marmoiton. Outdoor Autonomous Navigation Using Monocular Vision. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 3395–3400, 2005.
112. P. Saeedi, P.D. Lawrence, and D. G. Lowe. Vision-Based 3-D Trajectory Tracking for Unknown Environments. *IEEE Transactions on Robotics*, 22(1):119–136, 2006.
113. J. Santos-Victor and G. Sandini. Visual-Based Obstacle Detection: a Purposive Approach Using the Normal Flow. In *Proc. of International Conference on Intelligent Autonomous Systems*, 1995.
114. J. Santos-Victor, G. Sandini, F. Curotto, and S. Garibaldi. Divergence Stereo for Robot Navigation: Learning from Bees. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 1993.
115. D. Schleicher, L. M. Bergasa, R. Barea, E. Lopez, and M. Ocaa and. Real-Time Simultaneous Localization and Mapping Using a Wide-Angle Stereo Camera. In *Proc. of IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, pages 55–60, 2006.
116. S. Se, D.G. Lowe, and J.Little. Mobile Robot Localization and Mapping with Uncertainty Using Scale-Invariant Visual Landmarks. *The International Journal of Robotics Research*, 21(8):735–758, 2002.
117. S. Se, D.G. Lowe, and J. Little. Vision-Based Global Localization and Mapping for Mobile Robots. *IEEE Transactions on Robotics*, 21(3):364–375, 2005.
118. J. Shen and H. Hu. Visual Navigation of a Museum Guide Robot. In *The Six World Congress on Intelligent Control and Automation (WCICA)*, volume 2, pages 9169–9173, 2006.

119. J. Shi and C. Tomasi. Good Features to Track. In *Proc. of IEEE Intl Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, 1994.
120. C. Silpa-Anan, T. Brinsmead, S. Abdallah, and A. Zelinsky. Preliminary Experiments in Visual Servo Control for Autonomous Underwater Vehicle. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, volume 4, pages 1824–1829, 2001.
121. R. Sim and G. Dudek. Learning Generative Models of Scene Features. In *Proc. of IEEE Intl Conf. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 406–412, 2001.
122. R. Sim and G. Dudek. Effective Exploration Strategies for the Construction of Visual Maps. In *Proc. of 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles*, volume 3, pages 3224–3231, 2003.
123. R. Sim, P. Elinas, M. Griffin, A. Shyr, and J. J. Little. Design and Analysis of a Framework for Real-Time Vision-Based SLAM Using Rao-Blackwellised Particle Filters. In *Proc. of the 3rd Canadian Conference on Computer and Robotic Vision*, 2006.
124. R. Sim and J. J. Little. Autonomous Vision-Based Exploration and Mapping Using Hybrid Maps and Rao-Blackwellised Particle Filters. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, 2006.
125. R. Smith, M. Self, and P. Cheeseman. Estimating Uncertain Spatial Relationship in Robotics. *Autonomous Robot Vehicles (I. Cox and G.T. Wilfong eds)*, pages 167–193, 1990.
126. V. Srinivasan, M. Lehrer, W.H. Kircner, and S.W. Zhang. Range Perception Through Apparent Image Speed in Freely Flying Honeybees. *Visual neuroscience*, 6(5):519–35, May 1991.
127. V. Srinivasan, S. Thurrowgood, and D. Soccol. An Optical System for Guidance of Terrain Following in UAVs. In *Proc. of the IEEE International Conference on Video and Signal Based Surveillance (AVSS)*, pages 51–56, 2006.
128. V. Srinivasan, S. W. Zhang, J. S. Chahl, G. Stange, and M. Garratt. An Overview of Insect-Inspired Guidance for Application in Ground and Airborne Platforms. *Proc. of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 218(6):375–388, 2004.
129. S.Thrun. Probabilistic Algorithms in Robotics. Technical Report CMU-CS-00-126, Carnegie Mellon University, 2000.
130. K. Sugihara. Some Location Problems for Robot Navigation Using a Single Camera. *Computer Vision, Graphics, and Image Processing*, 42:112–129, 1988.
131. A. Talukder, S. Goldberg, L. Matties, and A. Ansar. Real-time Detection of Moving Objects in a Dynamic Scene from Moving Robotic Vehicles. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 1308–1313, October 2003.
132. A. Talukder and L. Matties. Real-time Detection of Moving Objects from Moving Vehicles Using Dense Stereo and Optical FLow. In *Proc. of IEEE Int'l Conf. on Intelligent Robots and Systems (IROS)*, pages 3718–3725, October 2004.
133. S. Temizer and L. P. Kaelbling. Optical Flow Based Local Navigation. Web Page. Massachusetts Institute of Technology, 2003. http://people.csail.mit.edu/lpk/mars/.
134. C. Thorpe. FIDO: Vision and Navigation for a Mobile Robot. In *PhD dissertation, Dept Computer Science, Carnegie Mellon University*, 1983.
135. C. Thorpe, M.H. Herbert, T. Kanade, and S.A. Shafer. Vision and Navigation for the Carnegie-Mellon Navlab. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):362–372, 1988.
136. C. Thorpe, T. Kanade, and S.A. Shafer. Vision and Navigation for the Carnegie-Mellon Navlab. In *Proc. of Image Understand Workshop*, pages 143–152, 1987.
137. S. Thrun. Learning Metric-Topological Maps for Indoor Mobile Robot Navigation. *Artificial Intelligence*, 99(1):21–71, 1998.
138. S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, and D. Fox. MINERVA: A Second-Generation Museum Tour-Guide robot. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, volume 3, pages 1999–2005, May 1999.
139. M. Tomono. 3-D Object Map Building Using Dense Object Models with SIFT-Based Recognition Features. In *Proc. of IEEE Int'l Conf. of Intelligent Robots and Systems (IROS)*, pages 1885–1890, 2006.
140. P. Torr, A.W. Fitzgibbon, and A. Zisserman. Maintaining a Multiple Motion Model Hypotheses Over Many Views to Recover Matching and Structure. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 485–491, 1998.
141. E. Trucco and K. Plakas. Video Tracking : A Concise Survey. *IEEE Journal of Oceanic Engineering*, 31(2):520–529, 2006.
142. M.A. Truk, D.G. Morgenthaler, K.D. Gremban, and M. Marra. VITS - A Vision System for Autonomous Land Vehicle Navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):342–361, 1988.
143. S. Tsugawa, T. Yatabe, T. Hisrose, and S. Matsumoto. An Automobile With Artificial Intelligence. In *Proc. of Sixth International Joint Conference on Artificial Intelligence*, pages 893–895, 1979.
144. T.Tsubouchi and S. Yuta. Map-Assisted Vision System of Mobile Robots for Reckoning in a Building Environment. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 1978–1984, 1987.
145. S. van der Zwaan and J. Santos-Victor. An Insect Inspired Visual Sensor for the Autonomous Navigation of a Mobile Robot. In *Proc. of the Seventh International Sysposium on Intelligent Robotic Systems (SIRS)*, 1999.
146. B. Wilcox, L. Matties, D. Gennery, B. Copper, T. Nguyen, T. Litwin, A. Mishkin, and H. Stone. Robotic Vehicles for Planetary Exploration. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 175–180, 1992.
147. N. Winters and J. Santos-Victor. Omnidirectional Visual Navigation. In *Proc. of IEEE Int'l Symposium on Intelligent Robotic Systems (SIRS)*, pages 109–118, 1999.

148. D. Wooden. A Guide to Vision-Based Map Building. *IEEE Robotics and Automation Magazine*, 13:94–98, 2006.
149. X. Xu and S. Negahdaripour. Mosaic-based positioning and improved motion estimation methods for automatic navigation of sumersible vehicles. *IEEE Journal of Oceanic Engineering*, 27(1):79–99, 2002.
150. C. Zhou, Y. Wei, and T. Tan. Mobile Robot Self-Localization Based on Global Visual Appearance Features. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 1271–1276, 2003.
151. J. Zhou and B. Li. Homography-Based Ground Detection for a Mobile Robot Platform Using a Single Camera. In *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA)*, pages 4100–4101, 2006.