Augmented reality user-interface for robot teleguide based on video and laser data

Davide De Tommaso and Marco Macaluso

University of Hertfordshire
School of Electronic Communication and Electrical Engineering
Hatfield, UK

Supervisors: S. Livatino, G. Muscato

October 11, 2009

Contents

List of Figures

List of Tables

Preface

The present work is part of a research project coordinated from the professors Salvatore Livatino, actually at the department of School of Electronic, Communication and Electrical Engineering of the University of Hertfordshire [2] (Hatfield - United Kingdom), and Giovanni Muscato from DIEES (Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi) [3] of the University of Catania (Catania - Italy).
This project involves computer vision enhancements for remote robot teleguiding of the 3MORDUC, a robot built and managed at DIEES Lab in Catania. During the last years, many works have been done by previous students focuses in two main different topics: sinthetic vision and camera vision based. The our main aim has been those to mix sinthetic and camera vision based in a unique system of vision that combines the advantages of the one and the other one to improve the capabilities of previous teleguide system. We have focused our studies *in the field of computer research which deals with the combination of real-world and computer-generated data,* the Augmented Reality (AR) [1].

Chapter 1

Introduction

We discuss follow a base introduction to the work.

1.1 The 3MORDUC

The 3MO.R.D.U.C. (3rd version of the MObile Robot DIEES University of Catania) Figure 5.1 is a wheeled mobile robot with a differential drive kinematic configuration. This open robotic platform was successfully used in localization and navigation experiments. The robot structure has three shelves linked together. On the lowest shelf, two lead batteries (12V/18Ah) provide the power supply. The robot autonomy is about 30-40 min. for continuous working. The on board electronic rack controls each module of the robot (motion, sensors and communication modules). Several sensors on board monitor the workspace and the robot state. A belt of bumpers (16 switches) around the entire perimeter is mounted on the robot base, just over the wheels level. These sensors recognize and reduce damage in case of a collision. The two robot motor axes are equipped with incremental encoders (resolution of 500 pulses per turn). These sensors are useful to calculate heading and position of the robot by using the kinematic model.

Figure 1.1: The 3MORDUC

To detect obstacles on the workspace the robot has an on board Laser Measurement Sensor (LMS) and a sonar belt (8 sonars).The LMS operates by measuring the flight time of a pulsed laser light beam that is reflected by obstacle. An internal rotating mirror deflects the transmitted pulsed laser beam so that a scan is made Stereoscopic Robot Teleguide based on Laser and Video sensors of the surrounding area. The time between transmission and reception of the light pulse is directly proportional to the distance between *the scanner and the object. The sonar sensors measure the distance from an obstacle using the flight time of an ultrasonic signal produced by means of a vibrating piezoelectric sensor. On the robot there are also two high quality stereoscopic cameras; each one has a resolution of 1.3 Megapixel; they are equipped with fixed focus lens of 4.0 mm. The CCD sensors of these cameras have a good noise immunity and sensibility; moreover, it is possible to adjust all the image parameter, e.g. exposure gain, frame rate, resolution. The cameras are mounted on a rigid support; it permits to simply adjust the camera distance in a range 5-20 cm.*

Description of the robot from DIEES Robotics [4]

1.2 The 3MORDUC: brief description of his sensors

In this section we describe briefly the sensors of the 3MORDUC, refer by [8].

Sonar sensors The sonar sensors measures the distance from an obstacle using the flight time of an audio signal produced by a vibration of a piezoelectric sensor. To measure the distance to an object, the time from transmission of a pulse to reception is measured and converted into a range by knowing the speed of sound. The sonar has a field of view that is a cone, so the sensitive area increases proportionately with the distance from the obstacle.

Figure 1.2: The SRF08 sonar

To avoid the false obstacles is needed to wait until the sent waves stop, so an inhibition time is necessary. The introduction of this delay doesnt allow reading distance too short. The sonar used on the robot are SRF08, Figure 5.2 linked to the bus I2C.

Bumbers A belt of bumpers around the entire perimeter is mounted on the base, over the wheel. These sensors have, as aim, to recognize and reduce damage in a collision. The bumpers are switches that send impulses when there is a collision. They are linked to the same bus of the sonar sensors (I2C).

Stereocam The STH-MDCS2-VAR-C 5.3 is a low power compact digital stereo cam system, which can be connected to a PC via IEEE 1394. There are two high quality cameras; each one has a resolution of 1.3 Megapixel, equipped with fixed focus lens (2.5 mm). The CCD sensors of these cameras have a good noise immunity and sensibility; moreover, it is possible to adjust all the image parameter (e.g. exposure gain, frame rate, resolution). The cameras are mounted on a rigid support;

Figure 1.3: The STH-MDCS2-VAR-C stereocam

Figure shows the hardware configuration of the system. The support permits to adjust in simply way the camera distance in a range 5-20 cm.

Scanner Laser The LMS system (Figure 5.4) operates by measuring the time of flight of laser light pulses: a pulsed laser beam is emitted and reflected if it meets an object. The reflection is registered by the scanners receiver.

Figure 1.4: The Sick LMS200

The measurement data is available in real time for further evaluation via RS232/RS422 serial interface. Its possible to configure three separate angular resolutions (0.25/0.5/1) and the maximum scan angle (100/180); moreover each scan is in clockwise mode.

Encoders A digital optical encoder (Figure 5.5) is a device that converts motion into a sequence of digital pulses. By counting a single bit or by decoding a set of bits, the pulses can be converted to relative or absolute position measurements. Encoders have both linear and rotary configurations, but the most common type is rotary.

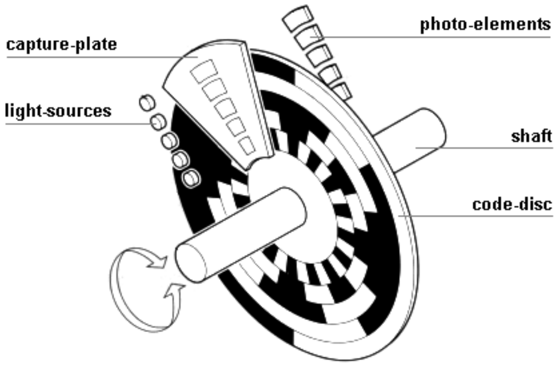

Figure 1.5: Encoder system

Rotary encoders are manufactured in two basic forms: the absolute encoder where a unique digital word corresponds to each rotational position of the shaft, and the incremental encoder, which produces digital pulses as the shaft rotates, allowing measurement of relative position of shaft. On the robot there are placed two incremental encoders with a resolution of 500 pulses/turn. The incremental encoder, sometimes called a relative encoder, is simpler and cheaper than the absolute encoder. It consists of two tracks and two sensors whose outputs are called channels A and B.

1.3 Application Scenario: 3MORDUC Teleguide

The 3MORDUC teleguide system consists in a client/server application. The server run on the robot and provides the services for remote teleguiding and for sending data information from his sensors. The main aim of the client application is to provides a user-friendly interface to the user for making teleguide operations simple and more realistic as possible.

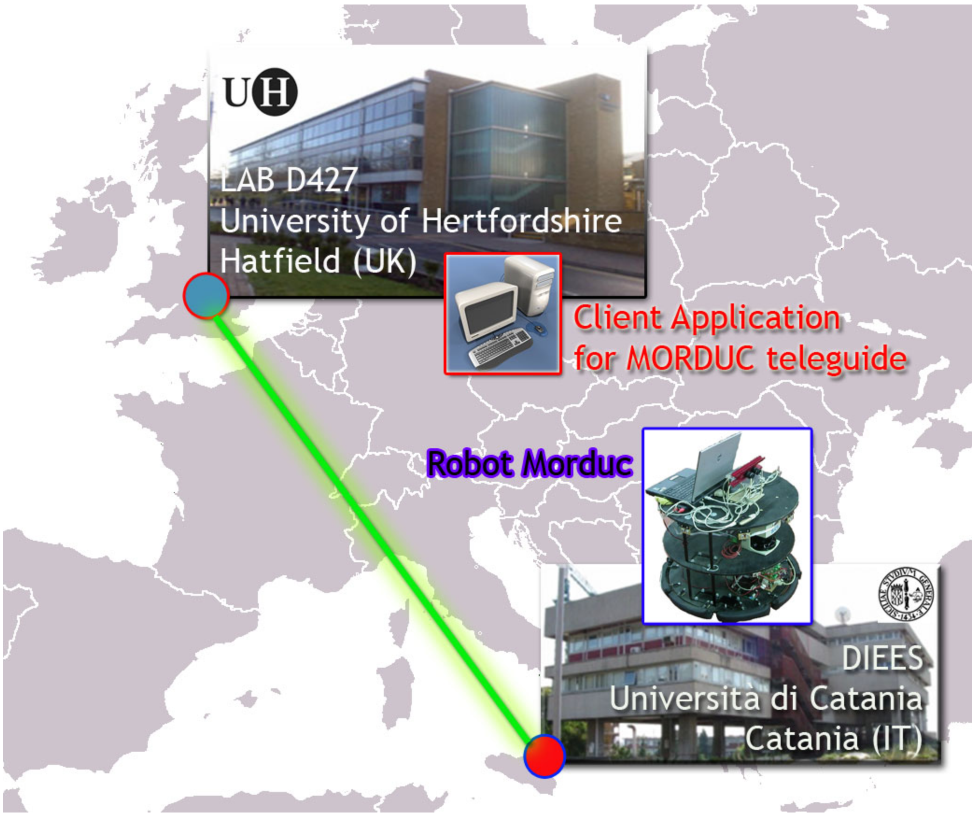

Figure 1.6: The 3MORDUC teleguide system

Client sends move commands to the robot, from driver station (Hatfield - University of Hertforshire) to the robot located in Catania (DIEES - Universita' di Catania), showing into the client interface the enviroment in which robot is moving.

1.4 Previous Works

We briefly describe previous works by students about this project's research.

3D Stereo Visualization advantages for the mobile robot teleguide [9]
In his work, F. Privitera, developed a 3D simulator for 3MORDUC offline teleguiding, using OpenGL libraries. This application builds a 3D enviroment from a static laser map, using the *Map Building* [10] algorithm. The user is able to move the robot in a sinthetic platform, feeling like being in the virtual environment himself, thanks to stereoscopics visualization.

Developing and experimenting on-line mobile robot teleguide [11]
This work represent the first step towards real-time 3MORDUC teleguide. The application is based to a client-server architecture, in which the client sends move commands to the server that reply with real stereo pictures from robot.

Depth enhanced mobile robot teleguide using laser and video sensors [12]
The work of A. Pennisi and C. Arena was that of evaluate the improvements of stereoscopic visualization in two different teleguide mode: laser and video based. They have fixed some bugs and inserted changes into previous application for telecommunication protocol and architecture improving.

Mobile Robot tele-guide based on laser sensors [13]
In her work, V. Neri has analyzed the differences between laser based teleguide and video based teleguide referring previous works.

1.5 Main Goal: Sensor Fusion

The main aim of our work was trying to find a way to sensor fusion information. While previous works were focused over laser or video based teleguide studies separately , we have trying, for the first time in this project, to mix these information to improve the realism of the teleguide experience.
Analyzing previous works, we have thinked that only laser or camera based teleguide was not sufficient to a realistic remote teleguide experience. These teleguide systems have different disadvantages and advantages. The main disadvantage in use laser based teleguide is the sinthetic enviroment in which the driver-user teleguides robot, that losses details from real enviroments; however laser data are lightweight than stereo images. In the other side, camera based teleguide is more realistic than laser based but heavy in terms of bytes sends in the network.

1.6 The Core Idea

On the one hand, the main advantage of a picture from camera is that it represents fully details of a scene, but is very difficult go back to exact information about distance of objects from visual source. On the other hand, the main advantage of the laser is just to know the exact distance to the near objects from the robot. In such a way, we thinked to mix information from these two sensors; starting from stereo pictures, we want add on these, information about distance using color palette for making simple the perception of objects's proximity during teleguide operation.

1.7 Used Tools

We have used the following tools:

OpenCV (Open Source Computer Vision)
OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real time computer vision.
Example applications of the OpenCV library are Human-Computer Interaction (HCI); Object Identification, Segmentation and Recognition; Face Recognition; Gesture Recognition; Motion Tracking, Ego Motion, Motion Understanding; Structure From Motion (SFM); Stereo and Multi-Camera Calibration and Depth Computation; Mobile Robotics. .
We referred by [14]

Adobe Photoshop *Adobe Photoshop, or simply Photoshop, is a graphics editing program developed and published by Adobe Systems. It is the current market leader for commercial bitmap and image manipulation software.*
We referred by [1]

C++ Programming language used for implementation of the application.

Python Programming language used for prototype version of the main application.

1.8 Outline

A brief description of each chapter of this report is given as follows.

Chapter 2

Background knowledge: a main topics overview

2.1 3MORDUC Sensors Specifications and Limits

We previously discussed about 3MORDUC sensors in Section 1.2. In this section a more detailed discussion about sensors and their data is given. This argument will be useful in the next chapters.

2.1.1 The STH-MDCS2-VAR/-C Stereo Head

The STH-MDCS2-VAR/-C is the digital stereo camera mounted up to the 3MORDUC. It consists of two camera modules, each with a 1.3 megapixel, progressive scan CMOS imager and a 1394 peripheral interface module, mounted in a rigid body. We present only technical specifications of our interest for future work, referred by [15].

- Micron MT9M001 Megapixel Sensors 1280 x 960 maximum image size , High sensitivity, low noise, Low pixel cross-talk, Rolling shutter.
- Fully synchronized stereo left and right video streams are synchronized to the IEEE 1394 bus clock.
- Monochrome or Bayer Color
- High frame rates 30 Hz for 640x480, 7.5 Hz for 1280x960
- On-chip decimation full frame 640x480 and 320x240 modes

- Electronic zoom mode center 640x480 subwindow
- Extensive control of video parameters, Automatic or manual control of exposure and gain, Automatic control of black level, Manual control of color balance.
- 50 Hz mode reduces indoor light interference in countries with 50 Hz electrical line frequency
- Stereo calibration information can be stored on the device, and downloaded automatically to the PC
- IEEE 1394 interface to standard PC hardware carries power and commands to device, data to PC
- Standard C/CS mount lenses, interchangeable focal lengths from 3.5 mm to 50 mm
- Variable baseline, 5 20 cm
- Anodized aluminum alloy chassis, high rigidity

Each camera can actually send images at faster frame rates up to 60 Hz at 640x480, and 15 Hz at 1280x960. However, because the IEEE 1394 bus is restricted to 32 MB/s in transferring video data, there is not enough bandwidth to accommodate two video streams at the highest rate.

Focal Length and Field of View

The **focal length** is the distance from the lens virtual viewpoint to the imager. It defines how large an angle the imager views through the lens. The focal length is a primary determinant of the performance of a stereo system. It affects two important aspects of the stereo system: how wide a field of view the system can see, and how good the range resolution of the stereo is. Unfortunately theres a tradeoff here. A wide-angle lens (short focal length) gives a great field of view, but causes a drop in range resolution. A telephoto lens (long focal length) can only see a small field of view, but gives better range resolution. So the choice of lens focal length usually involves a compromise. In typical situations, one usually chooses the focal length based on the narrowest field of view acceptable for an application, and then takes whatever range resolution comes with it.
The **field of view** is completely determined by the focal length. The formulas for the FOV in horizontal and vertical directions are:

$$\text{HFOV} = 2\arctan(3.33/ f\ )$$
$$\text{VFOV} = 2\ \arctan(2.50\ /\ f\ )$$

where f is in millimeters. For example, a 3.5 mm lens yields a horizontal FOV of 87 degrees. This is about the smallest practical focal length for the STH-MDCS2-VAR.

2.1.2 The SICK LMS200 laser scanner

The SICK LMS200 is a 2D laser scanner mounted in the middle of the 3MOR-DUC. We present some technical specifications referred by [16].

- Range - Maximum 80 m (262.5 ft)
- Angular Resolution - 0.25/0.5/1.0 (selectable)
- Response Time - 53 ms/26 ms/13 ms
- Measurement Resolution - 10 mm (0.39 in)
- Data Interface - RS 232/RS 422 (configurable)
- Transfer Rate - 9.6/19.2/38.4/500 kBd

Possible external data processing:

- Evaluation of partial sectors of the 100 or 180- field of vision
- Averaging of the measurement values transferred (increasing accuracy and smoothness)
- Straight line and curve approximations by interpolation of measurement values
- Determination of position/volume of any object

Spot Spacing/Spot Diameter/Range

In a radial field of vision, a light impulse (spot) is emitted every 0.25, 0.5 or 1 (depending on the set variant). As a result of the beam geometry and the diameter of the individual spots, the spots overlap on the target object or up to a certain distance.

Figure 5.8 shows spot spacing in relation to the range and the corresponding spot diameter.

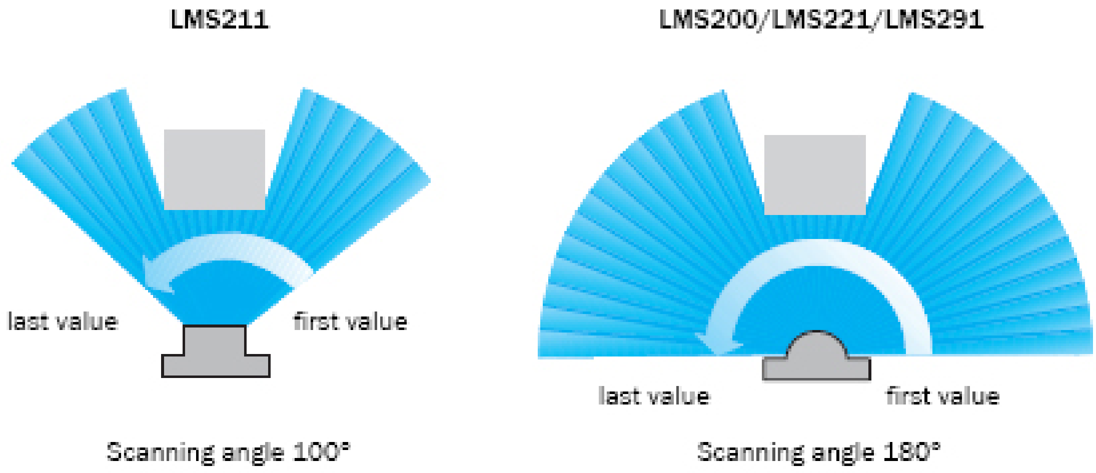

Figure 2.1: Scanning Angle

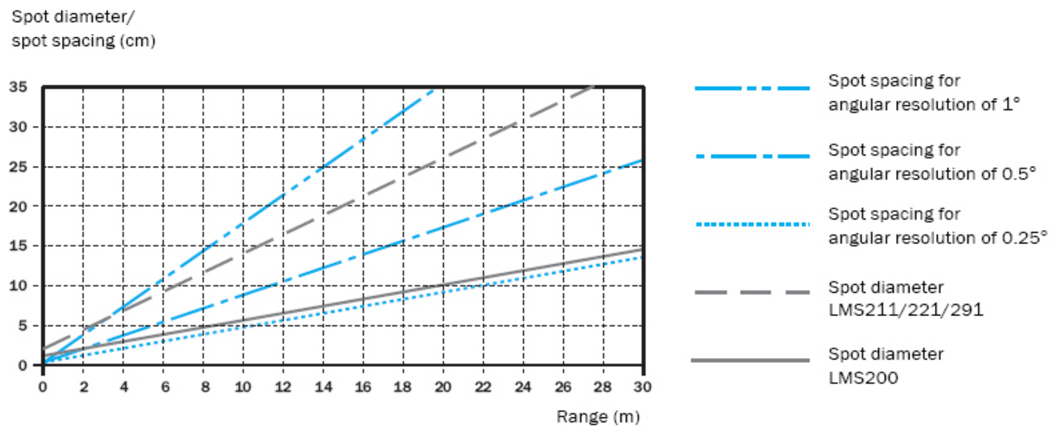

Figure 2.2: Spot sizes/spot spacing

2.2 Image Processing

In electrical engineering and computer science, image processing is any form of signal processing for witch the input is an image, such as photographs or frames of video; the output of image processing can be either an image or a set of characteristics or parameters related to the image.
Digital image processing allows the use of much more complex algorithms for image processing, and hence can offer both more sophisticated performance at simple task, and the implementation of methods which would be impossible by anlog means.
In particular , digital image processing is the only pratical technology for:

- classification
- feature extraction
- pattern recognition
- projection
- multi-scale signal analysis

Some techniques which are used in digital image processing include:

- principal components analysis
- independent components analysis
- self-organizing maps
- hidden Markov models
- neural networks

2.2.1 Edge Detection: Object Recognition

Edges characterize boundaries and are therefore a problem of foundamental importance on image processing. Edges in images are areas with strong intensity contras, a jump in intensity from one pixel to the next [17] . There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories: Gradient and Laplacian. The Gradient method detects the edges by looking for the maximum and minimum in the first derivate of the image. The Laplacian method searches for zero crossing in the second derivate of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivate of the

image can highlight its location.
Suppose we have the the following signal Figure 5.9, with an edge shown by jump in intensity.

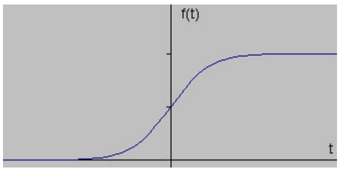

Figure 2.3: Signal with an edge

If we take the gradient of this signal (which, in one dimension, is just the first derivative with respect to t) we get Figure 2.4.

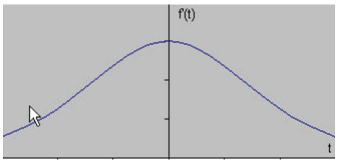

Figure 2.4: Signal with an edge

Clearly, the derivative shows a maximum located at the center of the edge in the original signal. A pixel location is declared an edge location if the value of the gradient exceeds some threshold. Edges will have higher pixel intensity values than those surrounding it. So once a threshold is set, you can compare the gradient value to the threshold value and detect an edge whenever the threshold is exceeded. Furthermore, when the first derivative is at a maximum, the second derivative is zero. As a result, another alternative to finding the location of an edge is to locate the zeros in the second derivative. This method is known as the Laplacian and the second derivative of the signal is shown in Figure 2.5.

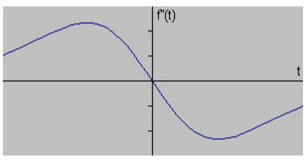

Figure 2.5: Signal with an edge

2.2.2 Canny algorithm

The Canny edge detection algorithm [18] is known to many as the optimal edge detector. Canny's intentions were to enhance the many edge detectors already out at the time he started his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "A Computational Approach to Edge Detection". In his paper, he followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is important that edges occuring in images should not be missed and that there be no responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge. This was implemented because the first 2 were not substantial enough to completely eliminate the possibility of multiple responses to an edge. Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (nonmaximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a nonedge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2.

In order to implement the canny edge detector algorithm, a series of steps must be followed. The first step is to filter out any noise in the original image before trying to locate and detect any edges. And because the Gaussian filter can be computed using a simple mask, it is used exclusively in the Canny algorithm. Once a suitable mask has been calculated, the Gaussian smoothing can be performed using standard convolution methods. A convolution mask is usually much smaller than the actual image. As a result, the mask is slid over the image, manipulating a square of pixels at a time. The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise. The localization error in the detected edges also increases slightly as the Gaussian width is increased.

After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image. The Sobel operator performs a 2-D spatial gradient measurement on an image. Then, the approximate absolute gradient magnitude (edge strength) at each point can be found. The Sobel operator uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). They are shown in Figure 2.8. The

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Figure 2.6: Sobel operator

magnitude, or *edge strength*, of the gradient is then approximated using the formula:

$$|G| = |G_x| + |G_y| \tag{2.1}$$

Finding the edge direction is trivial once the gradient in the x and y directions are known. However, you will generate an error whenever sumX is equal to zero. So in the code there has to be a restriction set whenever this takes place. Whenever the gradient in the x direction is equal to zero, the edge direction has to be equal to 90 degrees or 0 degrees, depending on what the value of the gradient in the y-direction is equal to. If G_y has a value of zero, the edge direction will equal 0 degrees. Otherwise the edge direction will equal 90 degrees. The formula for finding the edge direction is just:

$$\Theta = \tan^{-1}\frac{G_y}{G_x} \tag{2.2}$$

Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image. There are only four possible directions when describing the surrounding pixels - 0 degrees (in the horizontal direction), 45 degrees (along the positive diagonal), 90 degrees (in the vertical direction), or 135 degrees (along the negative diagonal). So now the edge orientation has to be resolved into one of these four directions depending on which direction it is closest to (e.g. if the orientation angle is found to be 3 degrees, make it zero degrees).

After the edge directions are known, nonmaximum suppression now has to be applied. Nonmaximum suppression is used to trace along the edge in the edge direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.

Finally, hysteresis is used as a means of eliminating streaking. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold. If a single threshold, T1 is applied to an image, and an edge has an average strength equal to T1, then due to noise, there will be instances where the edge dips below the threshold. Equally it will also extend above the threshold making an edge look like a dashed line. To avoid this, hysteresis uses 2 thresholds, a high and a low. Any pixel in the image that has a value greater than T1 is presumed to be an edge pixel, and is marked as such immediately. Then, any pixels that are connected to this edge pixel and that have a value greater than T2 are also selected as edge pixels. If you think of following an edge, you need a gradient of T2 to start but you don't stop till you hit a gradient below T1.

Figure 2.7: Example of a Canny Algorithm result

2.3 Augmented Reality for robot teleguiding

Augmented Reality (AR) is a growing area in virtual reality research[19]. The world environment around us provides a wealth of information that is difficult to duplicate in a computer. An augmented reality system generates a composite view for the user. It is a combination of the real scene viewed by the user and a virtual scene generated by the computer that augments the scene with additional information. The application domains reveal that the augmentation can take on a number of different forms. In all those applications the augmented reality presented to the user enhances that person's performance in and perception of the world. The ultimate goal is to create a system such that the user can not tell the difference between the real world and the virtual augmentation of it. To the user of this ultimate system it would appear that he is looking at a single real scene.
Virtual reality was defined as "a computer generated, interactive, three-dimensional environment in which a person is immersed." (Aukstakalnis and Blatner 1992). There are three key points in this definition. First, this virtual environment is a computer generated three-dimensional scene which requires high performance computer graphics to provide an adequate level of realism. The second point is that the virtual world is interactive. A user requires real-time response from the system to be able to interact with it in an effective manner. The last point is that the user is immersed in this virtual environment.
A very visible difference between between virtual reality and augmented reality systems is the immersiveness of the system. Virtual reality strives for a totally immersive environment. The visual, and in some systems aural and proprioceptive, senses are under control of the system. In contrast, an augmented reality system is augmenting the real world scene necessitating that the user maintains a sense of presence in that world. The virtual images are merged with the real view to create the augmented display. There must be a mechanism to combine the real and virtual that is not present in other virtual reality work.
The computer generated virtual objects must be accurately registered with the real world in all dimensions. Errors in this registration will prevent the user from seeing the real and virtual images as fused. The correct registration must also be maintained while the user moves about within the real environment. Discrepancies or changes in the apparent registration will range from distracting which makes working with the augmented view more difficult, to physically disturbing for the user making the system completely unusable. An immersive virtual reality system must maintain registration so that changes in the rendered scene match with the perceptions of the user.

Any errors here are conflicts between the visual system and the kinesthetic or proprioceptive systems. The phenomenon of visual capture gives the vision system a stronger influence in our perception (Welch 1978). This will allow a user to accept or adjust to a visual stimulus overriding the discrepancies with input from sensory systems. In contrast, errors of misregistration in an augmented reality system are between two visual stimuli which we are trying to fuse to see as one scene. We are more sensitive to these errors (Azuma 1993; Azuma 1995). Milgram (Milgram and Kishino 1994; Milgram, Takemura et al. 1994) describes a taxonomy that identifies how augmented reality and virtual reality work are related. He defines the Reality-Virtuality continuum shown as Figure 2.8.

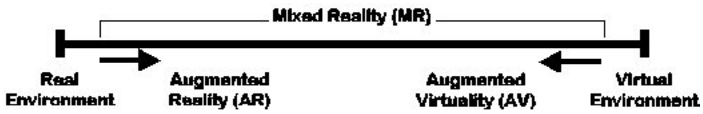

Figure 2.8: Virtuality continuum

The real world and a totally virtual environment are at the two ends of this continuum with the middle region called Mixed Reality. Augmented reality lies near the real world end of the line with the predominate perception being the real world augmented by computer generated data. Augmented virtuality is a term created by Milgram to identify systems which are mostly synthetic with some real world imagery added such as texture mapping video onto virtual objects. This is a distinction that will fade as the technology improves and the virtual elements in the scene become less distinguishable from the real ones.

Milgram further defines a taxonomy for the Mixed Reality displays. The three axes he suggests for categorizing these systems are: Reproduction Fidelity, Extent of Presence Metaphor and Extent of World Knowledge. Reproduction Fidelity relates to the quality of the computer generated imagery ranging from simple wireframe approximations to complete photorealistic renderings. The real-time constraint on augmented reality systems forces them to be toward the low end on the Reproduction Fidelity spectrum. Milgram also places augmented reality systems on the low end of the Extent of Presence Metaphor. This axis measures the level of immersion of the user within the displayed scene.

The third, and final, dimension that Milgram uses to categorize Mixed Reality displays is Extent of World Knowledge. Augmented reality does not simply mean the superimposition of a graphic object over a real world scene. This is technically an easy task. One difficulty in augmenting reality, as de-

fined here, is the need to maintain accurate registration of the virtual objects with the real world image. AThis often requires detailed knowledge of the relationship between the frames of reference for the real world, the camera viewing it and the user. In some domains these relationships are well known which makes the task of augmenting reality easier or might lead the system designer to use a completely virtual environment.
There are different Augmented Reality application domains:

- Medical
- Entertainment
- Military training
- Engineering design
- Manufacturing, Maintenance and Repair
- Consumer design
- Robotics and telerobotics

In the domain of robotics and telerobotics an augmented display can assist the user of the system (Kim, Schenker et al. 1993; Milgram, Zhai et al. 1993). A telerobotic operator uses a visual image of the remote workspace to guide the robot. Annotation of the view would still be useful just as it is when the scene is in front of the operator. There is an added potential benefit. Since often the view of the remote scene is monoscopic, augmentation with wireframe drawings of structures in the view can facilitate visualization of the remote 3D geometry. If the operator is attempting a motion it could be practiced on a virtual robot that is visualized as an augmentation to the real scene. The operator can decide to proceed with the motion after seeing the results. The robot motion could then be executed directly which in a telerobotics application would eliminate any oscillations caused by long delays to the remote site.

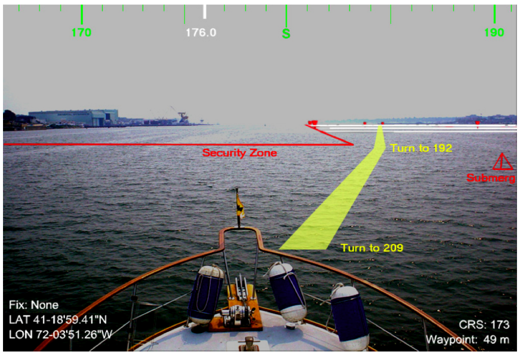

Figure 2.9: An example of augmented reality in guide application

Chapter 3

State of the art

In this chapter are introduced some results of state-of-the-art research on sensor fusion in robot application. Initially, we focus our attention on augmented reality and sensor fusion previous projects, finding three different approaches that matched with our problem. These project's research, shown as follow, has been the reference to carry out our algorithm.

3.1 Colored 2D Maps for Robot Navigation with 3D Sensor Data

The authors: Oliver Wulf, Christian Brenneke and Bernardo Wagner of Institute for Systems Engineering of the University of Hannover proposed an approach based on sensor fusion for veichle teleoperation [6]. The complexity of real world is still a challenging problem and is very hard to represent in a digital format accessible by a robot for real-time navigation in real enviroments. The system combines a 3D laser sensor and with 2D algorithms for path planning and simultaneous localization and mapping (SLAM). The principal idea of this approach is to extract two virtual 2D scans from the 3D point cloud. One scan contains landmarks (used for SLAM) and the other scan contains obstacles (used for path planning). As regular 2D maps are not able to differ between obstacles and landmarks has been introduced a Colored 2D Map 3.1 that is able to carry both types of information in a consistent way. For this reason, results demostrate that teleguide become more easy to perform and it is possbile to discern different objects around robot.

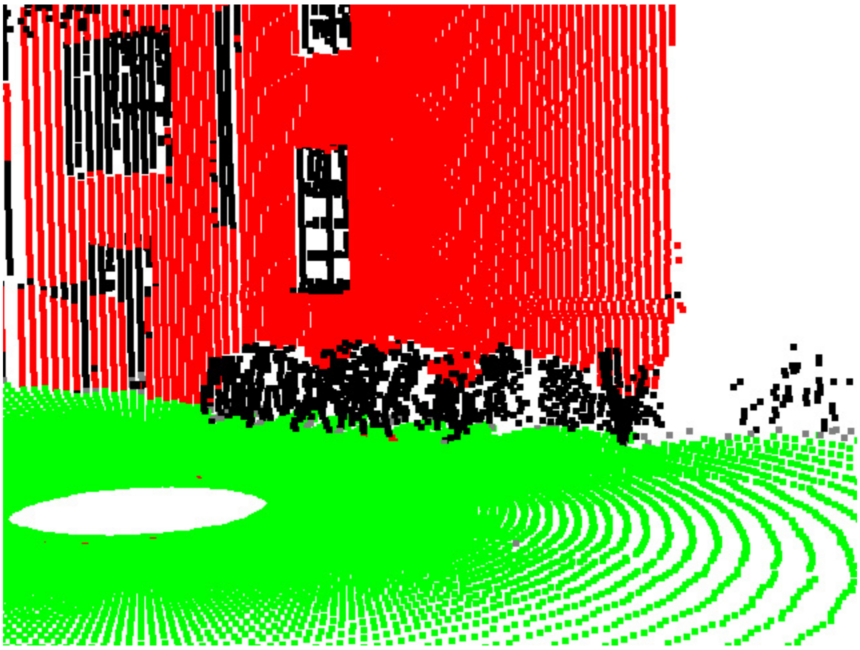

Figure 3.1: Colored 2D Maps.

3.2 A Sensor Fusion Based User Interface for Vehicle Teleoperation

Sensor fusion is commonly used to reduce uncertainty in localization, obstacle detection, and world modeling. In the paper [5], the authors: Roger Meier, Terrence Fong, Charles Thorpe and Charles Baur has been propose a novel software that allow to the operator to teleguide robot in unknown enviroments by sensor fusion. The solution exposed try to enhance the quality of information available to the operator using sensor fusion to create a user interface which that displays multisensor data. Many teleoperation errors can be directly attributed to inaccurate distance estimation and for this reason has been necessary to represent multi-dimensional data on a flat screen. Has been used stereo vision, sonar information, and odometry to create a 2D image overlay which improves estimation of relative distance and spotting of nearby obstacles. By using sensor fusion, the authors demostrate that it's possible to build better user interfaces, combining data from multiple, complementary sensors allows us to increase the quality of the information available to the operator in a typical robot teleguide operation.

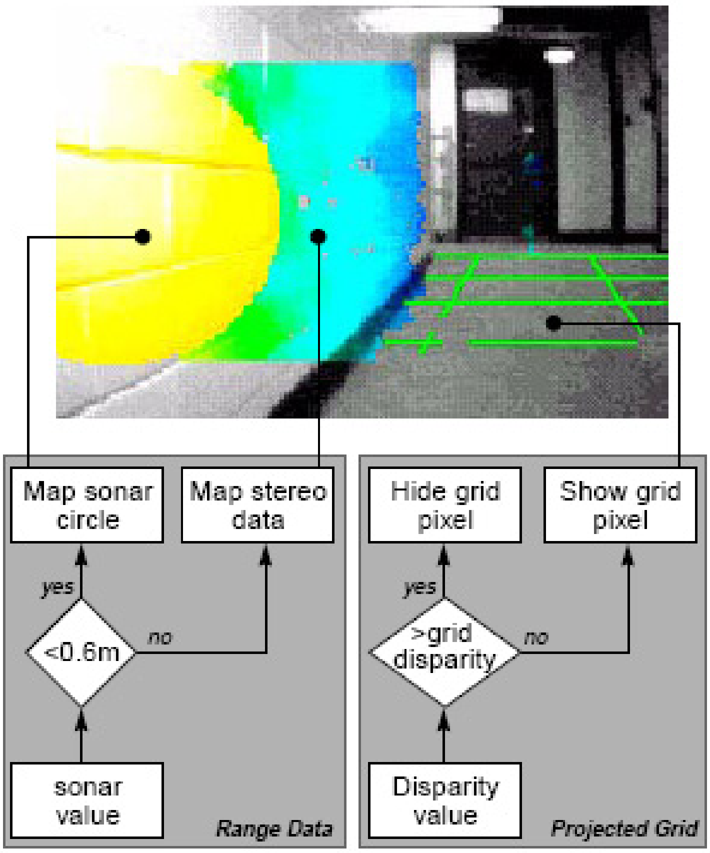

Figure 3.2: A Sensor Fusion Based User Interface.

3.3 Fusion of laser and visual data for robot motion planning and collision avoidance

In the paper [7] the authors: Haris Baltzakis, Antonis Argyros, Panos Trahanias from Greece has been proposed a method for inferring scene structure information based on both laser and visual data is proposed. The quantity of information encapsulated in 2D laser scans may, in certain cases, be insufficient for more crucial and demanding robotic tasks, such as obstacle detection and collision avoidance. This is because various objects common

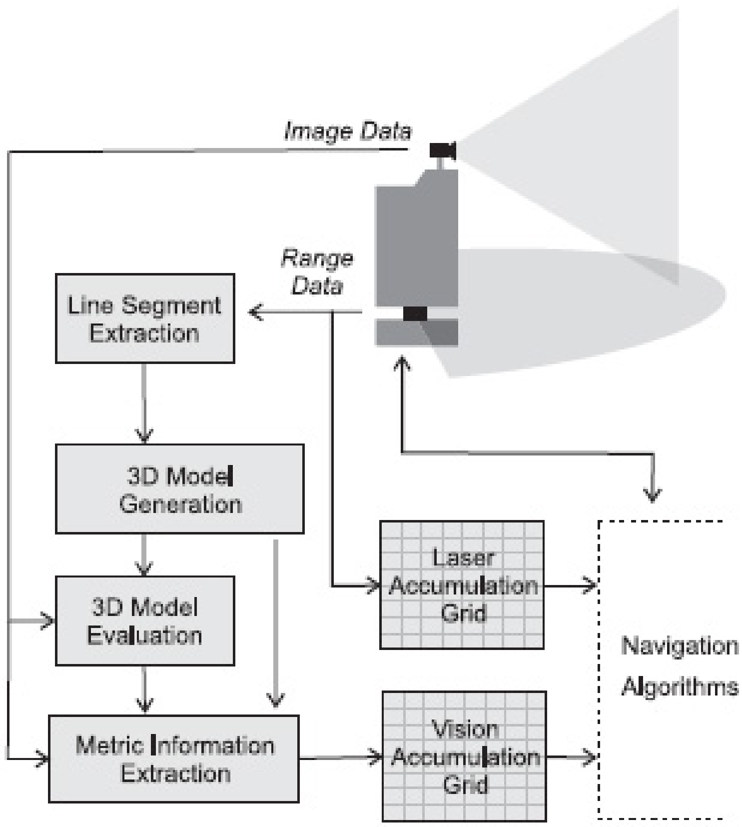

Figure 3.3: Block diagram of the proposed methodology.

even in the simplest indoor environments, such as chairs, tables, or shelves onwalls, are sometimes invisible to laser scanners and thus absent or misinterpreted in the resulting 2D profiles. In this solution has been proposed a methodology for fusing 2D laser with stereo visual information in order to infer accurate 3D information. Based on a single 2D range scan, a local 3D model of the robots environment is constructed. This is based on the assumption that the environment consists of a flat horizontal floor. In order to evaluate the developed 3D model, a pair of images acquired by a calibrated stereo vision rig is used. Points from the first image are ray-traced to the 3D model, and 3D coordinates are estimated. Based on this information, image points are reprojected onto the frame of the second camera.

Figure 3.4: Visual depth information extration example.

Chapter 4

Proposed Investigation

4.1 The Idea

Generally, laser information or camera images, are the information used to teleguide a robot. Main goal is to enrich the teleguide experience merging this two information sources. Information achieved from laser sensor, are mainly information about disance, that let driver have a better conception of the manouvre enviroment and to understand which obstacle are more or less far from the robot.
Either laser or camera information source have their benefits and drawbacks. Camera images provide a complete and detailed scene representation, but the bidimensional nature of this source preclusive to obtain exact information about objects's disatnce. Laser sensor provide exact information about objects's distance in the proximity of robot, but provide a synthesize scene rapresentation trough map.
The idea is to merge laser and camera information exploiting the benefit and remedy the drawbak; enriching camera images whith laser information about distance using colors and visual guide lines that permits a simply and immediate perception of the objects's proximity during the telguide Figure 4.1.

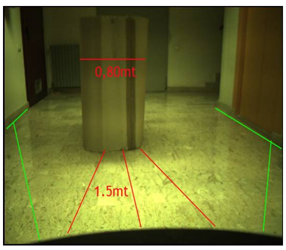

Figure 4.1: The idea of augmented reality

4.2 Image Retrieval

Images from robot's camera are obtained through internet connection using the HTTP protocol. Morduc robot hosts an HTTP server to handle client's requests. Each request include the service intended and if necessary the movement that the robot has to do. Possible robot movement are:

- Forward
- Backward
- Leftward
- Rightward
- No movement

Service available are:

- Stereo image and odometry data
- Laser map and odometry data
- Stero image in grey-scale and laser map
- Laser data
- Stereo image, laser map and odometry data

The client does an HTTP request with GET method specifying a valid resource. The server processes this request and provide the demanded resource if available.
Table 4.2 describes the different service available in Morduc server. For each of them is specified the method name in server application, the name of the URL resource and a set of flag to specify if server reply with laser map, stereo image, laser data or odometry data.

ServerMethodName	URLresource	LsrMap	Img	LsrData	Odometry
ResponseGetImg	stereo.jpg	No	Yes	No	Yes
ResponseGetLsr	laser.jpg	Yes	No	No	Yes
RespnoseGetLsrImg	laserimg.jpg	Yes	Yes	No	Yes
ResponseGetDatiLsr	laser.txt	No	No	Yes	No
ResponseGetDatiLsrandImg	datilaserandimg.jpg	No	Yes	Yes	Yes

As sescribed before different services and joined robot movements are available in Morduc server.
Different URL names allow to sepcify both service and robot movement:

stereo.fow.jpg,laser.fow.jpg,laserimg.fow.jpg,datilaserandImg.fow.jpg
to get video imager or laser image or laser and video image in a one frame or laser data or video image and laser data; making the robot moves forward

stereo.bak.jpg,laser.bak.jpg,laserimg.bak.jpg,laser.bak.txt,datilaserandImg.bak.jpg
to get video imager or laser image or laser and video image in a one frame or laser data or video image and laser data; making the robot moves backward

stereo.rgh.jpg,laser.rgh.jpg,laserimg.rgh.jpg,laser.rgh.txt,datilaserandImg.rgh.jpg
to get video imager or laser image or laser and video image in a one frame or laser data or video image and laser data; making the robot moves rightward

stereo.lft.jpg,laser.lft.jpg,laserimg.lft.jpg,laser.lft.txt,datilaserandImg.lft.jpg
to get video imager or laser image or laser and video image in a one frame or laser data or video image and laser data; making the robot moves leftward

The service we require for our application is the one that provide stereo image, laser map and odometry data:

```
>> HTTP Client Request
>> datilaserandImg.MOV.jpg header:

GET <image.jpg> HTTP/1.1 [CRLF]
Host: <ip address>[CRLF]
User-agent: MorducTeleGuide/0.1 [CRLF]
[blank line here] [CRLF]


>> HTTP Server Response from
>> HTTP Request datilaserandImg.MOV.jpg header:

HTTP/1.1 200 OK [CRLF]
Server:Morduc/<TickCount>\<PoseX>\<PoseY>\<PoseTheta>\<Collisions>\<minlas> [CRL
Data: Laser/<laserdata_1>/<laserdata_2>/...<laserdata_181>/ [CRLF]
```

```
Content-Type: image/jpeg [CRLF]
Content-Lenght: <imglenght> [CRLF]
[blank line here] [CRLF]
<image bytes>
```

So an example of HTTP request to Morduc server is:

"'*http://MoruducIPAddress/URLResourceName*"'

Our application use only mono image, so we have to spilt the stereo image obtained from Morduc in two images (left and right camera) and use only one of them. We have tested however that processing both the images is possibile to view some stereo effect.

4.3 Edge Detetection

To emphasize obstacles and their distance in a scene is neecessary to detect the different objects in the enviroment. To identify this objects we look for their edges using **Canny**'s edge detection algorithm (Section 2.2.2) of OpenCV framework.
Figure 4.2 shows the result of the edge detection algorithm application on an image retrived from Morduc.
The resulting image is a black and white image. White pixels show obstacles's edges after they have been dilated to highlight them.
Based on the enviroment condition (eg. light, shape and number of obstacles) could be necessary to set right Canny's algorithm parameters to detect edges as best one can (Figure 4.3).

Figure 4.2: Edge detection applied on an image from Morduc

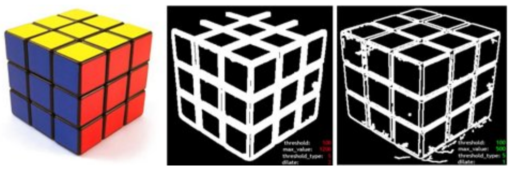

Figure 4.3: Edge detection: result of different Canny's parameters

4.4 Color Map Creation

Information obtained from laser could be either a laser map or laser samples. Laser map is a jpeg image showing the enviroment reconstructed by laser refraction on the base of objects refraction. Laser samples are 181 numeric information representing the distance of the nearest object in the scene for each degree in the upper semicircle with the robot at the center as shown in Figure 4.4. As discussed in Section 4.2 this samples are provided by Morduc including them in the header of HTTP response to the client.
From this sample is possible to reconstruct the laser map, showing a pixel for each degree at a vertical distance on base of the numeric value of the corresponding sample as shown in Figure 4.5.
Is important to notice that the field view of camera is usually shorter than the 181 degrees of a laser scan; so we need to filter the laser samples to obtain only those samples referred to obstacle in the field of view of camera. This is done manually at the biginning of our application.
We want to use this distance information to enrich camera image showing the distance of nearest obstacles. To do that we obtain a color map from laser samples, assigning a color at each pixel of this map on base of the nearest obstacle distance Figure 4.6.
To have a perfect corresponding from the camera image and the color map is necessary that the two images have the same size. Than we devided the horizzontal size of laser map in 181 strips, representing the horizzontal space from two consecutive samples. We assign the same color value at every pixel within each strip on base of function in Figure 4.7. As this figure shows nearer the obstacle is more red will appear the corrisponding pixel, the opposite for the green color. So we choose to rapresent the obstacles's distance in a scene using a color scale from red to highlight nearest objects to green for the far ones.

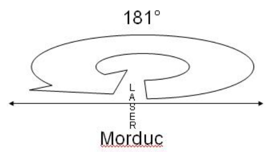

Figure 4.4: Laser sampling

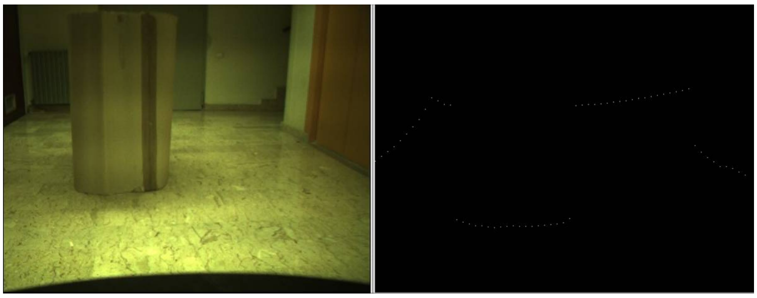

Figure 4.5: Laser Map reconstruction from laser samples

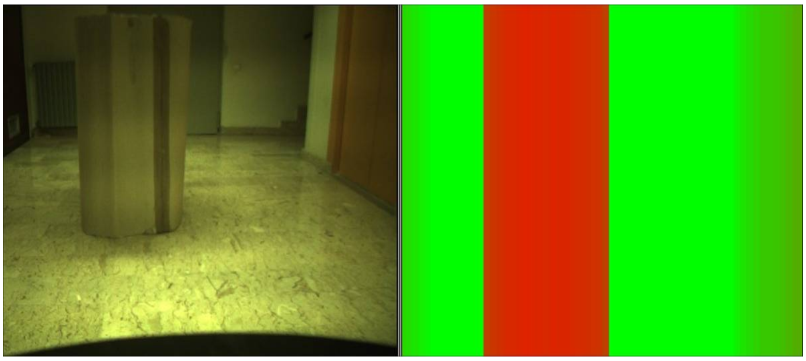

Figure 4.6: Color map obtained from laser samples

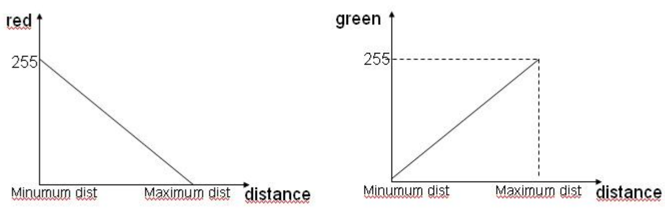

Figure 4.7: Function to obtain red and green value for each pixel based on distance

4.5 The Fusion Algorithm

The goal is now to merge camera image and laser color map developed as in Section 4.4.
We'll suppose that there is a perfect horizzontal alignment between camera image and color map (in our application is possibile to calibrate at the beginning of the teleguide to align these two images).
From the edge dtection algorithm we obtain a balck and white image where white pixels identify obstacles's edges (Section 4.3). Scannering this image beginning from the bottom we identify the white pixels. These pixels will be the edges of the obstacles in the scene.
We look for the corresponding distance at same horizzontal position in color laser map and associate the color founded in map to each pixel of the edge. So we're associating color to border pixels, based on the distance from Morduc of the obstacle, for which, the specify pixels are showing in camera image its border.
A schematic view of each steps is shown in Figure 4.8.

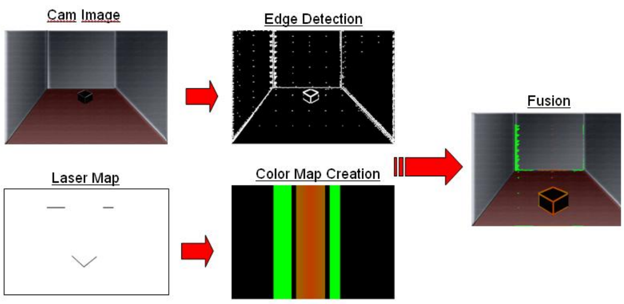

Figure 4.8: Schema of fusion algorithm steps

Objects's Transparency

Once we have founded the color for each border pixels in the camera image we can add these color values to the corresponding pixels in the original image (image received from Morduc's camera). The color value founded are respective to red and green color, so in an RGB image we add to the original pixel only red and green value obtainig a trasparency effect as shown in Figure 4.9.
Adding and not substituting the pixel value we can keep the information about the scene.
Founded border pixels we color all the surface of the obstacle ending this process at a specific height that can be setted.

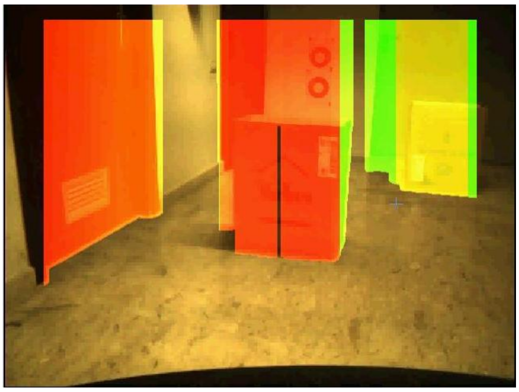

Figure 4.9: Morduc image with transparency applied

Lines Toward Objects

To draw lines in camera image we still use the edge detection result image, tracing a line from the center of Morudc to edges pixels founded.
These lines are then colored with the color of corresponding surface to which they are directed.
We also group together lines directed to same obstacle and subdivide them to the ones directed to different obstacles, based on the color value of the surface to which they ends and the corresponding distance.
Figure 4.10 shows a Morduc camera image enriched witho lines towards obstacles, while Figure 4.11 shows a Morduc camera image enriched with both transaprency and lines.

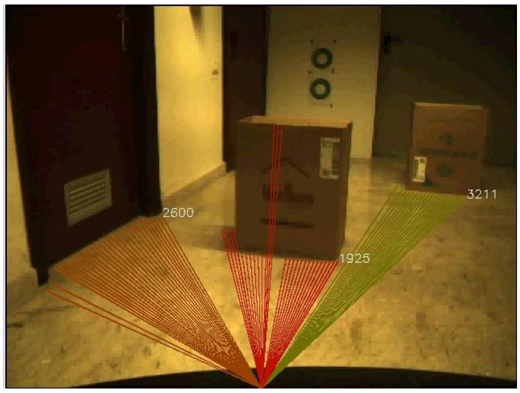

Figure 4.10: Morduc image with lines applied

Numerical Distance Information

As shown in Figure 4.10 or in Figure 4.11 at the end of a group of lines we display the numeric information about distance of the obstacle to which those lines end. To obtain this kind of information we process with an inverse formla of withc shown in Figure 4.7. Beginning from color value of a surface and knowing formula used to obtain that colour from a distance information gived from laser; appling inverse formula we can obtain numeric information about distance from color value.

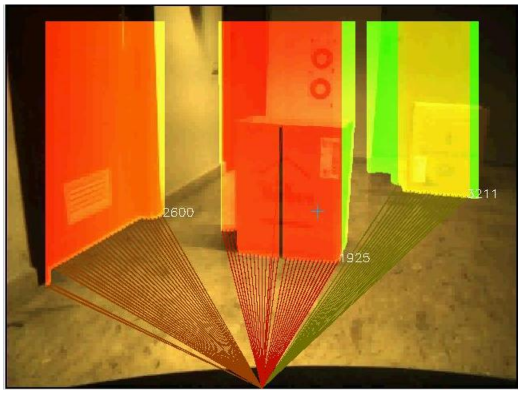

Figure 4.11: Morduc image with transparency and lines applied

Chapter 5

AR4MORDUC

5.1 Introduction

AR4MORDUC is the application that implements our algorithm for fusion of the laser and camera data from Morduc. This application enables a driver to remote teleguide Morduc, watching images from stereo cameras in which distance information over objects is shown. Before to start teleguide, the user is able to choice settings about laser and object distance information. The next paragraphs will be presents the application interface and how operates in remote teleguide.

5.2 AR4MORDUC: A brief description

The application 5.1 enables the user to configure the following parameters:

- **Network Options**
- **Laser Options**
- **Teleguide Options**
- **Laser Map Aligment Options**
- **Edge Detection Options**

5.3 Network Options

These are the network parameters for configuring remote teleguiding:

Figure 5.1: AR4MORDUC Application

- **IP Address:** This is the destination address in which is on the Morduc HTTP web server. In the server there are services to request data from Morduc sensors (i.e. Camera, Laser, Bumbpers, Sonar, etc.).

- **Port:** This is the port in which listen Morduc HTTP web server.

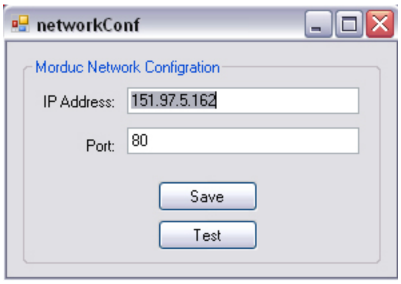

Figure 5.2: Network Options

5.4 Laser Options

These are the Laser Map Parameters to associate colors to distances.

- **MIN LASER DIST:** The Minimum Laser Distance is the smallest distance that laser might capture. A recognized object at this distance (NEAR OBJECT) will be associated the RED color in the laser color map.

- **MAX LASER DIST:** The Maximum Laser Distance is the greater distance that laser might capture. A recognized object at this distance (FAR OBJECT) will be associated the GREEN color in the laser color map.

- **Laser Disparity:** This parameter enforces difference beween FAR or NEAR objetcs, by the following formula:

$$GREEN = 255 - RED/disparity$$

5.5 Teleguide Options

These are the Teleguide Options to setup the augmented reality:

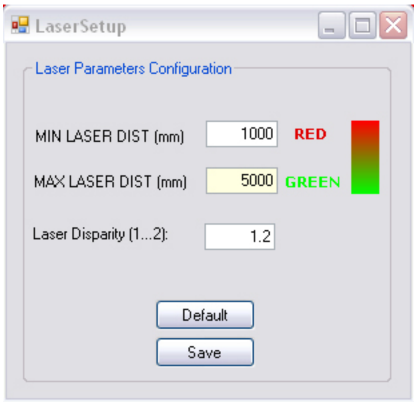

Figure 5.3: Laser Options

- **Threshold for Different Object Recognition:** When an object is at least this distance from another object, the application will recognizes two different objects by lines mode.
- **MAX Distance from lines:** When an object is at least this distance from Morduc will be point from lines.
- **Space from top image to Color Map:** This is the space (in pixel) from the upper border of the image to the color map.

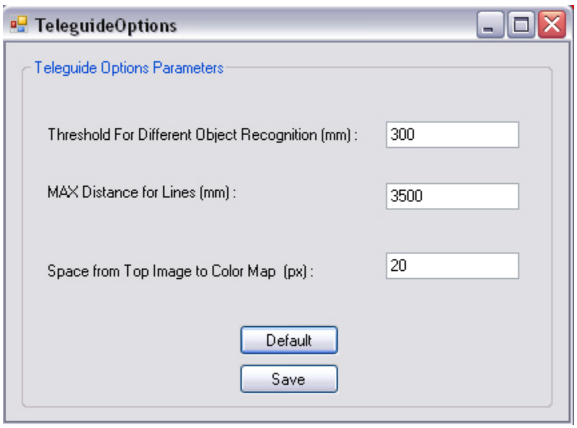

Figure 5.4: Teleguide Options

5.6 Laser Map Alignment

This window form enables the user to align the Laser Color Map filtering the Laser data from Morduc to the field of view of the camera, by the following options:

- **Lft Laser:** How much degree, from the 90 to the left of the 181 Morduc's Laser data, have to filter.
- **Rgt Laser:** How much degree, from the 90 to the right of the 181 Morduc's Laser data, have to filter.
- **Alpha:** Alpha level for the Laser Color Map.

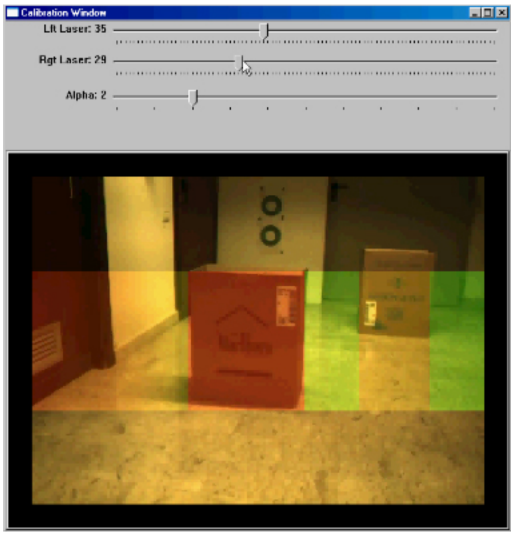

Figure 5.5: Laser Map Alignment

5.7 Edge Detection Options

This window form enables the user to setup the edge detection parameters of Canny's Algorithm.

5.8 Teleguiding Modes

The AR4MORDUC teleguide system is able to operates in three different modes of teleguide:

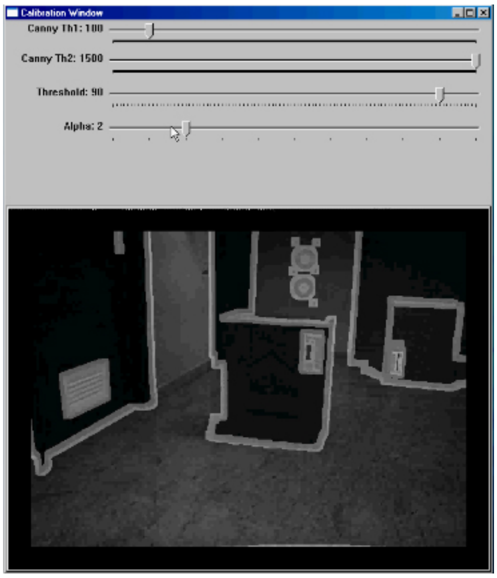

Figure 5.6: Laser Map Alignment

- Colors and Lines 5.7
- Lines 5.8
- Colors 5.9
  Each mode can be selected, in a while user teleguides Morduc, by pressing particular keyboard keys.

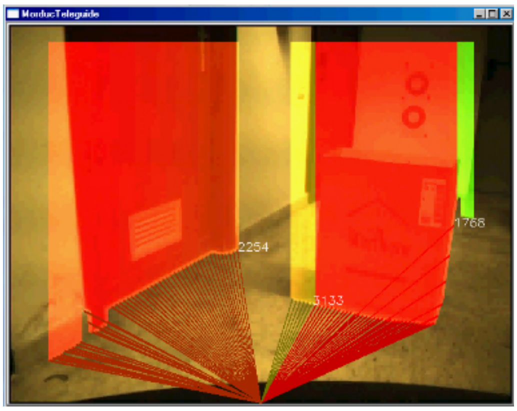

Figure 5.7: Colors and Lines Mode

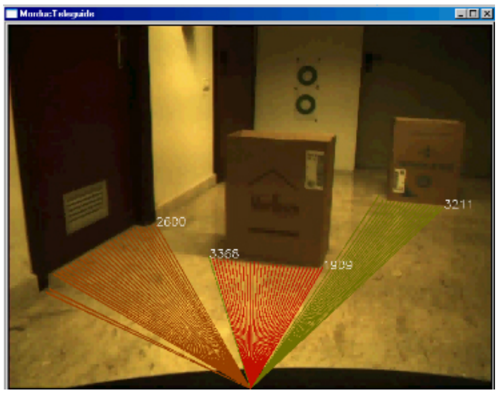

Figure 5.8: Lines Mode

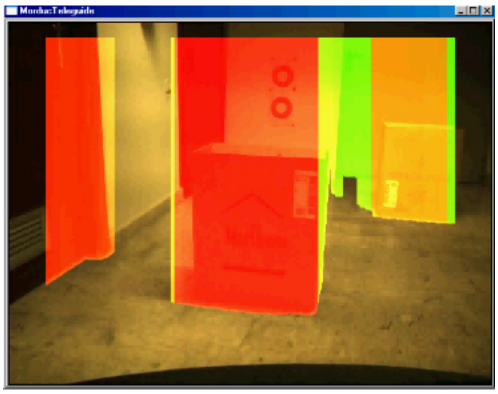

Figure 5.9: Colors Mode

Chapter 6

Conclusions

6.1 Conclusions and future work

In this project, a new method for fusion laser and visual data to add information about distance of objects has been proposed. Our work is especially designed for 3MORDUC robot teleguide. The application enables a driver to remote teleguide Morduc, watching images from stereo cameras in which distance information over objects is shown. Nevertheless, if we consider only fusion algorithm, this can be adopted to any system composed by a single camera and laser, each one with its referement system. In fact, the initial calibration process allow user to align laser map with real vision data watched by robot system. When manual calibration is done, teleguide can start.
From obtained results, we believe that fusion of data provided an appropriate software tool for mobile robots to perform any kind of navigation task for collision avoidance. It is our intention to further investigate new features of software, such as: improvements to edge detection, a more accuracy laser map and the applicability of stereo techinques.

Bibliography

[1] *"Wikipedia, The Free Encyclopedia"*
http://en.wikipedia.org/

[2] *"School of Electronic, Communication and Electrical Engineering"*
http://www.herts.ac.uk/courses/schools-of-study/electronic-communication-and-electrical-engineering/

[3] *"DIEES - Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi"*
http://www.diees.unict.it/

[4] *"Service Robots Group - Universita' di Catania"*
http://www.robotic.diees.unict.it/

[5] *"A Sensor Fusion Based User Interface for Vehicle Teleoperation."*
of Roger Meier, Terrence Fong, Charles Thorpe, and Charles Baur.
Institut de Systms Robotiques, L'Ecole Polytechnique Fdrale de Lausanne CH-1015 Lausanne EPFL, Switzerland and The Robotics Institute of Carnegie Mellon University Pittsburgh, Pennsylvania 15213 USA.

[6] *"Colored 2D Maps for Robot Navigation with 3D Sensor Data."*
of Oliver Wulf, Christian Brenneke, Bernardo Wagner.
Institute for Systems Engineering, University of Hannover.

[7] *"Fusion of laser and visual data for robot motion planning and collision avoidance."*
of Haris Baltzakis, Antonis Argyros, Panos Trahanias.
Institute of Computer Science, Foundation for Research and Technology Hellas (FORTH), P.O. Box 1385, Heraklion, 711 10 Crete, Greece.

[8] *"Depth enhanced mobile robot teleguide using laser and video sensors"*
of Carmelo Arena Lucio.
Thesis of the Corso di laurea in Ingegneria dellAutomazione e Controllo

dei Sistemi Complessi Dipartimento di Ingegneria Elettrica Elettronica e dei Sistemi - Catania.

[9] *"3D Stereo Visualization advantages for the mobile robot teleguide"*
of S. Livatino and G. Muscato (supervisors), F. Privitera (student).
Report Medialogy Semester 9, December 2004.

[10] *"Probabilistic robotics"*
of S. Thrun, W. Burgard, D. Fox.
Ed. The MIT Press, 2005.

[11] *"Developing and Experimenting Online Mobile RobotTeleguide"*
of S. Livatino (supervisor) and D. Di Mauro(student).
Report Medialogy Semester 9, June 2007.

[12] *"Depth-Enhanced Mobile Robot Teleguide based on Video Images"*
of S. Livatino, G. Muscato and S. Sessa (supervisors), C. Arena, A. Pennisi and D. Di Mauro (students).
Report Medialogy Semester 9, December 2007.

[13] *"Mobile Robot tele-guide based on laser sensors"*
of S. Livatino, G. Muscato, D. Longo and S. Sessa (supervisors), V. Neri (student).
2008

[14] *OpencvWiki*
http://opencv.willowgarage.com/wiki/

[15] *STH-MDCS2-VAR/-C Stereo Head - Users Manual*
November, 2004 - Videre Design

[16] *LMS200/211/221/291 Laser Measurement Systems - SICK*

[17] {http://www.pages.drexel.edu/˜weg22/edge.html}

[18] {http://www.pages.drexel.edu/˜weg22/can_tut.html}

[19] {http://www.se.rit.edu/˜jrv/research/ar/}