RoboCupRescue 2006 - Robot League Team <MRL (IRAN)>

Ali M. Shahri, M. Norouzi, S. Khalilzade, A. Fasih, A. H. Mashat, A. Paikan,

S. M. Mavaei, A. Sangari, M. Jadaliha, A. H. Mandegar, J. Chegini, M. Yaghoubi, M. Razaghi

Mechatronics Research Laboratory
Azad University of Qazvin
Qazvin, IRAN
shahri@iust.ac.ir
http://robocup.mechatronics.ws/rescue_robot.htm

Abstract. In this paper the MRL rescue robot team and its robots are explained. We have designed and built three new robots including two autonomous indoor robots and one out-door (rough terrain) robot for different situations/arenas. Our main goal of this activity is to achieve a practical rescue robot for real situation such as earthquake which is quite common in our country. We have also arranged to initiate some research programs on autonomous mobile robot such as; simultaneous localization and mapping, navigation strategies, collision avoidance algorithms, sensor fusions, robot cooperation and search algorithms. All of our research works are carried out at the Mechatronics Research Laboratory.

Introduction

Rescue operation in a disaster situation is quite important and should be fast enough to save victims life. So implementing high technologies such as robotics could be quite helpful for search and rescue operations. There are so many earthquakes every year in many countries such as Japan, USA, Turkey and Iran. Robocup real rescue competition has provided quite stimulating situation for University educator to involve in such a humanitarian activity.

Our team members in Mechatronics Research Laboratory are planning not only to take part in the competition but also to get enough knowledge to achieve a practical robot to help search and rescue operations in a disaster situation.

In this paper the MRL rescue robot team and its robots are explained. We have designed and built three new robots including two autonomous indoor robots and one out-door (rough terrain) robot for different situations/arenas. Our main goal of this activity is to achieve a practical rescue robot for real situation such as earthquake which is quite common in our country. We have also arranged to initiate some research programs on autonomous mobile robot such as; simultaneous localization and mapping, navigation strategies, collision avoidance algorithms, sensor fusions, robot cooperation and search algorithms. All of our research works are carried out at the Mechatronics Research Laboratory.

It should be mentioned that our previous robots such as NAJI-II which took 1st place in German open 2005 are still active and running.

Obviously, based on the environmental situation a special robot with proper abilities is required. In other words, there could be no unique robotics solution for searching and rescuing program in a disaster situation. As a result we have designed different robots with different maneuverability. For example NAJI-II and NAJI-V with a high power and flexible mechanism which overcome hard obstacles are also capable of supporting a powerful manipulator for handling objects. Fig.1 illustrates NAJI-II in red arena in German open-2005.

Fig. 1. NAJI-II Climbing a 45 Slope and Step Field in German Open-2005

NAJI-V is a new design and modified version of NAJI-II which is more powerful and flexible while it is lighter and smaller. Fig, 2. illustrates a 3D Model of NAJI-V. Our new design is inspired from our acquired experience on NAJI-II in German open-2005 and Robucup2005 in Osaka. There are so many rough and hard terrains in a disaster situation which the rescue robot should be fast enough and low weigh to pass and explore environment quickly while it is stable.

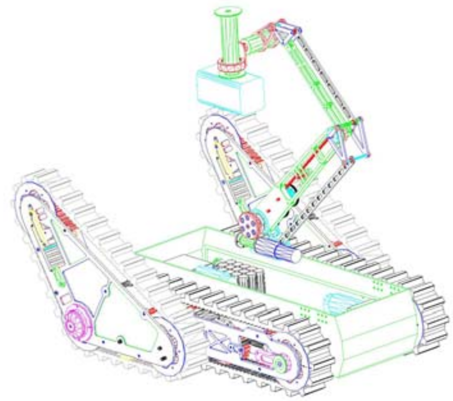

Fig. 2. 3D Model of NAJI-V

NAJI-I and NAJI-III are good examples of such a robot while NAJI-III with a novel mechanical design is quite faster, flexible and more stable. NAJI-IV which is facilitated with most required sensors is an autonomous mobile robot to carry out different research programs which is also suitable for the yellow arena.
All our robots are powered by Embedded PC based on RT/Linux which are able to process data and control the robot in Real-time.

1. Team Members and Their Contributions

• A. M. Shari	Team leader
• M. Norouzi	Team organizer
• A. Fasih	Hardware/Software
• A. Paikan	Software developer / Operator
• A. H. Mashat	Electronics design / Instrumentation
• S. M. Mavaei	Electronics design / Instrumentation
• J. Chegini	Electronics design
• A. Sangari	Localization and mapping Algorithms / Operator
• M. Jadalihi	Localization and mapping Algorithms
• A. H. Mandegar	Communication
• S. Mokaram	Software developer / Operator
• M. Yaghoubi	Software developer
• M. Razaghi	Instrumentation
• S. Khalilzade	Mechanical design
• M. Fasih	Mechanical Technician
• Azad University of Qazvin	Sponsor

2. Operator Station Set-up and Break-Down (10 minutes)

As it is compulsory to set-up and break down in less than 10 minutes, we are going to consider several things for fast handling and operation. We are going to design a Mobile Control Pack (MPC) including; notebook, joystick, access point, antenna, I/O Extension board and case with appropriate connectors. The pack which is still is under construction is shown in Fig. 3.

Our operators are practicing more and more in a real situation to do everything as fast as possible. A good joystick with several control keys and handles is selected for easier and faster operation. It should be noted that we are educating two operators to replace any of them in case of happening any problem.

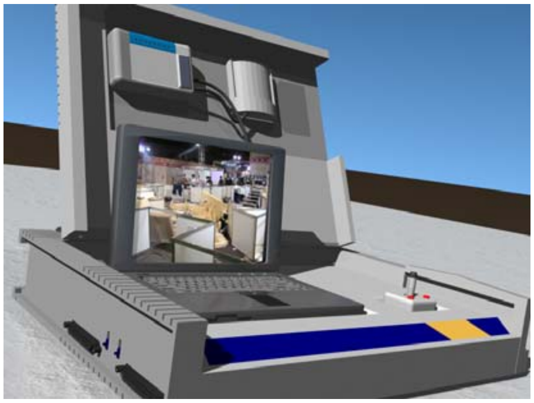

Fig. 3. Control Pack

Our operators are equipped with a communication headset to contact their team members in case of requiring anything or organizing the set-up and break down process just on time. (This headset is just for the set-up and break-down tasks).

3. Communications

As it is suggested, we are going to use W-LAN 802.11A (5GHz).

Our communication system is designed somehow to be able to switch between W-LAN and radio modem. In case of missing the network connectivity or PC failure it is possible to command the robot through radio modem which is interfaced to the embedded microcontroller system.

Rescue Robot League		
MRL (IRAN)		
Frequency	**Channel/Band**	**Power (mW)**
5.0 GHz - 802.11a		100
2.4 GHz - Bluetooth	spread-spectrum	
1.2 GHz	1-4	1000

4. Control Method and Human-Robot Interface

According to previous real rescue leagues, it seems that there is a trend to make the robot as intelligent and autonomous as possible. Now it is a rule that all robots in yellow arena should be fully autonomous. Accordingly, it is our plan to manage and prepare for an intelligent robot capable of navigating, mapping and searching for victims automatically. Therefore, we have generally designed two types of robots; semi-autonomous (NAJI-II & NAJI-V) and fully autonomous (NAJI-IV) which are presented in the following sections. Concerning system overview both types are mostly similar. In other words the difference between two types is the inference of human operator in semi-autonomous robot to command and navigate the robot and search for victims. Even our intelligent collision avoidance algorithm has more priority than the command of the operator.

4.1 Software Overview

Our software controller is developed to be executed on Real-time Linux. Moreover, it is equipped with intermediate software layer to communicate with RT-HAL in lower level and Wireless LAN in higher level. The robot sends the generated path, Log file and sensors data to central computer in the operator station. Goals of our software are to develop source code capable of executing in the optimized real time Linux and communicate with RT-HAL and Wireless LAN.

4.1.1 Real Time Hardware Abstracted Layer (RT-HAL)

It should be noted that for executing a high level control algorithm in a robot and consequently decreasing the system dependency on intermediate devices, employing an abstracts layer is unavoidable. To approach this goal the Real Time Hardware Abstracted Layer (RT-HAL) is designed in modular form on Linux Kernel. RT-HAL in order to directly access to the lower layer (hardware) and upper layer (high level controlling process). This process can be broken down into a number of important sub tasks which are:

- Collecting the sensors data
- Sensor fusion and data filtering
- Low Level controlling such as PID controller
- Self localization and obstacle detection
- A communication interface to high level process

Sensors data will be collected by associated data acquisition module and passed to sensor fusion module to be fused and acquire a better perception of environment. This data will be analyzed by Self Localization and Obstacle Detection (SL/OD) to obtain the robot and obstacles pose. Sensors data will be sent to sensor fusion module for error correction and sensor fusion algorithm. Output data such as robot rotation ($\Delta\theta$), movement (Δd) and obstacles distance will be sent to SL/OD and PID Controller module to further process. For example, the encoder's pulses which are used to determine the angle of rotation (θ_e) will be fused with digital compass data (θ_c), and consequently more accurate angle of rotation will be achieved (Fig. 4).

.

$$\theta_f = \theta_e + \frac{\sigma_e^2}{\sigma_c^2 + \sigma_e^2}(\theta_c - \theta_e)$$

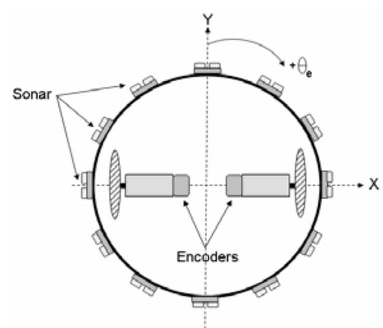

Fig. 4. Encoder and sonar position in NAJI-IV

Where

θ_f is heading angle estimation after sensor fusion.

θ_e and θ_c are heading angle of encoders and compass.

σ_e^2 and σ_c^2 are variance of encoders and compass estimation.

Moreover, this module has an interface which uses the FIFO to transmit these data to higher layer. SL/OD module periodically, places data into the FIFO and the high level application obtain its required data from FIFO. RT-HAL by implementing the driver as a device files, prepares a base for high level application to send their controlling command to actuators. Consequently RT-HAL decreases the high level application independency and prepares the standard API to communicate with the hardware which leads to code the application in any programming languages. Figure 5 illustrates the RT-HAL layers.

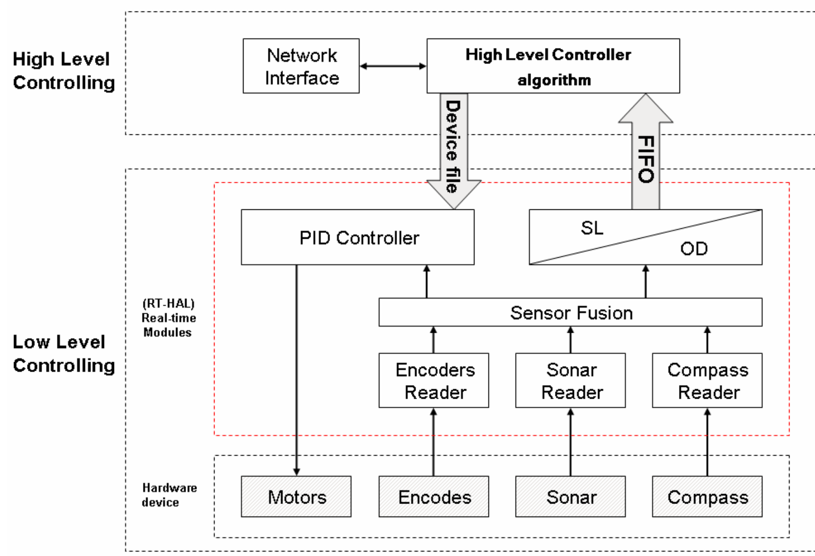

Fig. 5. RT-HAL layer overview

4.2 Hardware Overview

Our robots which are based on differential drive system are equipped with sonar sensor, Laser Scanner, CO_2 sensor, thermopile array detecting infra-red sensor, IMU, digital compass, optical encoders and digital/analog cameras. The computational system is PC-104 with VIA 800 MHz processor, 512 MB Compact Flash Memory, 128 MB RAM, IO Card with 32 digital IO ports and PCMCIA Wireless LAN 802.11a/g. The operating system is RT-Linux which optimized and equipped with Hardware Abstracted Layer for best performance.

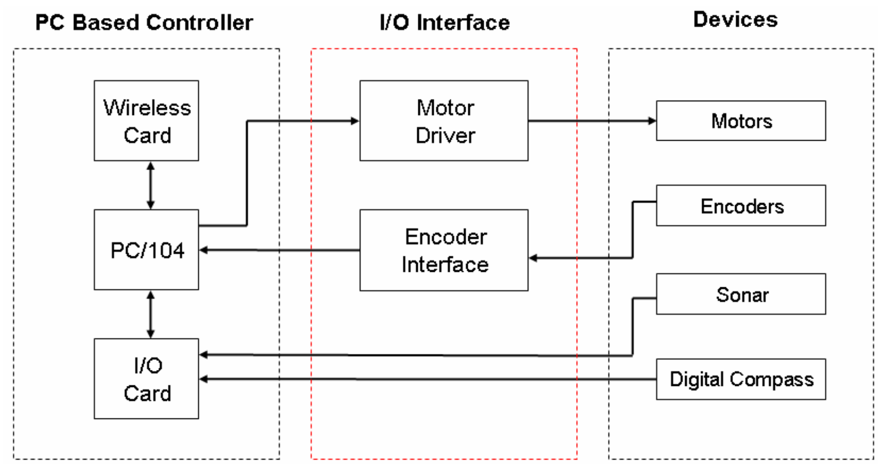

Fig. 6. Hardware Overview

4.2.1. Robot System Overview

Our robot system overview is designed based on three levels High, Abstraction and Low level top to bottom including the following steps orderly;

- Localization: Given sensors and a map, where am I?
- Vision: If my sensors are eyes, what do I do?
- Mapping: Given sensors, how do I create a useful map?
- Searching Algorithms: Given an unknowable world but a known goal and local sensing, how can I get there from here?
- Kinematics: if I move this motor somehow, what happens in other coordinate systems?
- Motor Modeling: what voltage should I set now?
- Control (PID): what voltage should I set over time?

The control scheme of our autonomous robot is partially autonomous. It means that the cameras images are sent to the computer are processed by operator to navigate the robot. All other sensors information are also sent to the operator to investigate the arena and detect all possible victims. To avoid colliding the robot with the environment obstacles or victims an ultrasonic obstacle avoidance algorithm has more priority than the command of the operator.
Although the map generation is autonomous, when a victim is located, operator has to define the victim conditions based on the sensors data. In order to save time, a proper

GUI is designed with several push bottom keys to define the victim's condition just by clicking the mouse button.

In case of loosing the control of the robot by the operator, a program is designed to return the robot back to the starting point autonomously using the map stored in the robot. All the sensors data are collected in a data bank to be used even off-line after the operation.

It should be mentioned that the applied servomechanism is digitally controlled by a PID control algorithm implemented on an AVR microcontroller system. An optical shaft-encoder is used to feedback the position and speed of the robot. All embedded microcontroller system designs for servomechanisms, instrumentations and drivers are modular for easy repair and maintenance.

5. Map generation/printing

Map generation method in our semi-autonomous robot (NAJI-II) is based on the operator assessment in conjunction with the collected data and a GUI program, which enables operator to locate and register different object such as victims, obstacles, walls and doors. However, NAJI-IV generates 2D and 3D maps automatically. As the robot proceeds to search and identify the field, the map of the environment will be developed which might be printed at the end of the robot mission. In our program, two maps will be generated. First, a 3D-map of the robot path with the location of all the victims will be generated autonomously. In this method, environmental conditions will be recorded by URG-X003 laser scanner and a set of 12 SRF08 sonar sensors which are distributed around the robot. It should be noted that each of these two types of sensors are necessary to fulfill the deficiency of the others.

In NAJI-II, while the robot is navigating the arena, the victims' condition will be recorded by the operator decision in the victims' data bank. At the end of the operation the victims' conditions will be printed by selecting each victim individually as the second map. Fig 7 illustrates a sample page of the NAJI Rescue Robot GUI.

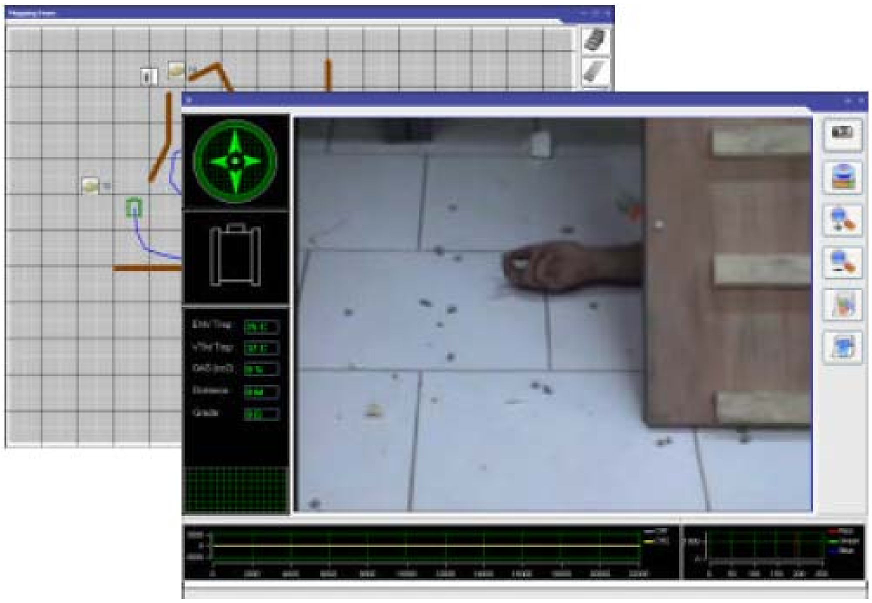

Fig. 7. A sample page of NAJI Rescue Robot GUI

6.1. 2D/3D Map Generation Using Laser Scanner

Our goal is to generate a 2D/3D map from the environment which the autonomous mobile robot has navigated. A laser scanner from HOKUYO Automatic Co. [12] is used. URG-X003 is a laser sensor for area scanning. The sensor's light source is infrared laser of wavelength 785nm. Scan area is 240° semicircle with pitch angle 0.36° which is good enough for indoor applications. Sensor outputs the distance measured at every point. The sensor's maximum measurement is up to 4000 mm with the accuracy of 1 mm.

At 4000mm, the laser beam diameter is 40mm. The principle of the distance measurement is based on calculation of the phase difference, due to which it is possible to obtain stable measurement with minimum influence from object's color and surface gloss. URG-X003 is designed under JISC8201-5-2 and IEC60947-5-2 standards for the industrial applications such as obstacle detection. Fig. 8 illustrates the coverage area by URG-X003.

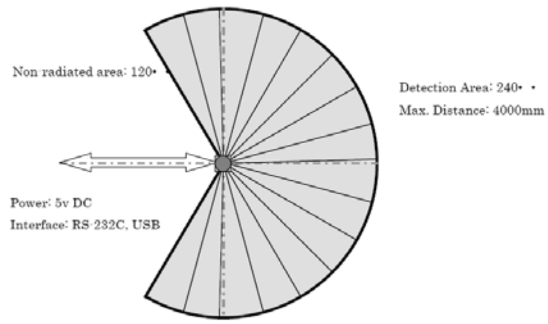

Fig. 8. Coverage area by URG-X003

The laser scanner system is located in front of the robot which is able to scan 240° semicircle ahead of the robot. In order to filter the thermal noise of the laser, a simple statistical method based on variance is applied. The odometry, digital compass and laser scanner data are fused to determine the location and heading angle of the robot with 0.5 degree resolution. According to this data the environment map will be generated.

By sampling the laser data and applying it to the rotation matrix which varies according to the angle of the robot heading, it is possible to have an absolute scan which is not dependant to the robot heading angle. It means that if the robot rotates around its center of gravity a few rounds while scanning the environment, the data calculated from rotation matrix for each scan are mostly equivalent. So this is the first and important step to generate 2D map.

Now, it is possible to merge the scanned data using different methods such as SLAM [7], Bayesian and Kalman filter to have 2D map. Using a tilt sensor (ADXL202) it is possible to distinguish between motion of the robot on a slope or a flat area which is important not to make mistake in generating 2D map.

In order to generate a 3D map we have used two different structures for NAJI-II and NAJI-V. First we have designed a servomechanism to rotate the laser scanner vertically. For our fully autonomous robot (NAJI-IV), we have attached the laser scanner to the back of the robot while the laser beam is toward the ground. By this way, when the robot is moving forward and laser scanner is scanning at the same time, we may have a 3D map.

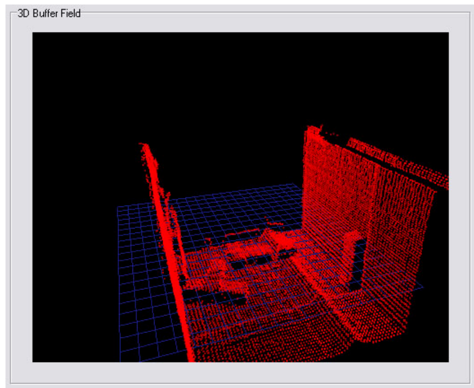

Fig. 9. Generated 3D Map on NAJI-IV

We have used a simple calculation of the neighbor points gradient as a segmentation method to distinguish between walls, ground surface and objects. Fig. 9 illustrates a sample of generated map.

6. Sensors for Navigation and Localization

In order to navigate the semi-autonomous robot in an unknown environment, a set of two cameras are used to help the operator to command/guide the robot and also a set of ultrasonic sensors to detect unknown obstacles which can be used in an automatic collision avoidance algorithm. As we have used digital compass and optical encoder for localization, these sensors can also be used for navigation as well. We have also used tilt senor (ADXL-202) to measure tilt angle when the robot is navigation of a slope. The localization is carried out by different methods to increase the precision and certainty. Conventional Dead Reckoning in conjunction with a digital compass decrease the error associated with this method.

Our Autonomous robots are equipped with the following sensors for localization and navigation;

- Optical Shaft Encoder,
- Ultrasonic Sensors (SRF08) with range of 6 meters to locate obstacles, [13]
- Laser Scanner, [12]
- Accelerometer ADXL-202 as tilt sensor,
- Digital Compass to measure the robot heading angle, [13]
- Inertial Measurement Unit [15] to fuse its data to reduce odometry error and
- A set of Ultrasonic sensors for co-operation of two robots in order to search and navigate an environment with minimum error. [13]

6.1. Robots Co-operation in Search Algorithm

We have initiated a research program to use co-operation of two fully autonomous robots. This will help us to reduce odometery errors and explore an unknown area to map and search for victims. In this project, we are using ultrasonic sensors to detect the distance and the angle differences of robots headings. Fig. 10 shows are proposed setup including transmitting and receiving sensors in two robots. It should be mentioned that each robot acts in two different mode. When a robot is assigned to be transmitter, the other is receiver. In order to synchronize the Tx and Rx ultrasonics sensors, a Bluetooth radio module is used.

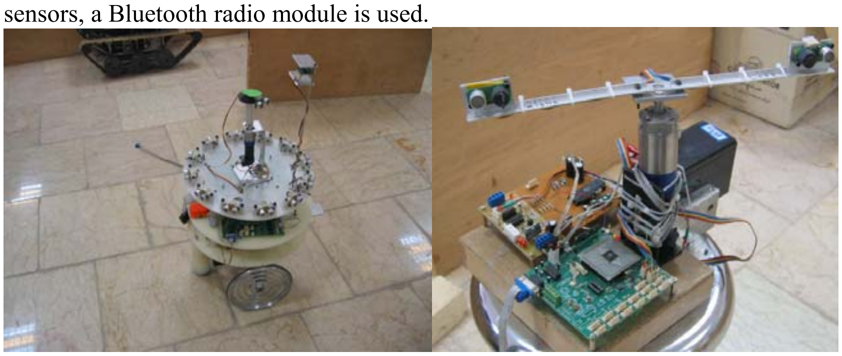

Fig. 10. Robot Co-Operation using Ultrasonic Senosrs

The idea of distance measuring between two robots is based on the Triangulation method which is shown in Fig. 11. In this method both Tx and Rx sensors which are motorized try to track each other until become perpendicular.

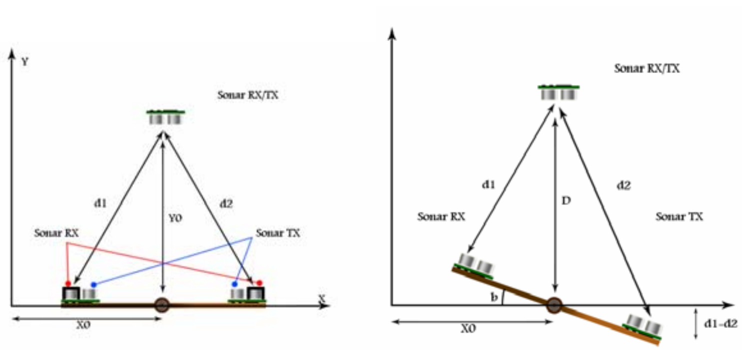

Fig. 11. Ultrasonic Tracking System

The Co-operation between these two robots makes the searching process quite fast and accurate compare to having a single robot.

6.2. SLAM based on Fusion of Sonar and Laser Scanner

Autonomous mobile robots require two types of algorithms for exploration and mapping, in an unknown environment; [7]

- Localization and mapping algorithm that extracts features of object from row sensors data to map the environment and identify pose of robot in that map.
- Exploration algorithms which guide robots to uncovered parts of map. [8]

.

6.2.1 Localization and Mapping [5]

Localization and mapping in cluttered and unstructured environment require fusion of row sensors data to reduce the uncertainty of each sensor in order to fulfill individual sensors properties. Several approaches are applied in sensor fusion, including conventional methods based on statistics theorem such as; Bayesian estimation, dempster-shafer inference, linear and extended Kalman filter, as well as intelligent algorithms such as; fuzzy logic and genetics algorithms.

Moreover, localization and mapping may need to be performed in two different methods;

1. Concurrent Localization and Mapping
   Firstly, robot maps the existing around objects, and then moves to reach its intermediate targets (set point). Consequently, it estimates its pose and modifies the generated map iteratively.
2. Simultaneous Localization And Mapping (SLAM) [7]
   SLAM is a technique used by robots to build up a map within an unknown environment while at the same time keeping track of its current position. This is not as straight forward as it might sound due to inherent uncertainties in discerning the robots relative movement from its various sensors. In this method, map is presented by a set of geometrical objects while pose estimation of the robot and objects are updated in a state space matrix iteratively. SLAM realization could be made by Monte-Carlo, Markov or Extended Kalman Filter (EKF). [8]
   As we have implemented our SLAM algorithm by EKF, this method is described below (Fig. 12.);
   In this method, each estimation has an associated covariance matrix which expresses the correlation between variables in state space matrix. This covariance matrix also describes the estimation uncertainty. So we have to calculate covariance matrix in each iteration.
   Firstly, dead reckoning method based on shaft encoder data, calculates rotational speed of robot's wheel. In observation step, the robot acquires sonar's and laser scanner dataset, then environmental objects geometrical parameters could be determined by separate feature extraction algorithms for sonar and laser scanner data. Laser scanner feature extraction could be performed by Hough transform [9] to fit lines to the laser scanner return points. Because of wide angular range, feature extraction for sonar's data is more complicated. The method that we have used in our robot is the same as Leonard's research work which is described in "Direct Sonar Sensing". [6]
   Using two sensor types for mapping in an unknown environment has an essential benefit, which returns in physical principals of sensors. For example laser scanner couldn't detect dark object perfectly, and mostly it returns distance of those

object with an offset. In contrast, sonar could detect dark object as well as bright ones, but sonar has a wide angular range that limits detection angular resolution. Beside ultrasonic sonar sensing is slower than laser and other optical sensing system. With respect to these reasons, we have used sonar and laser scanner which have their own individual feature extraction procedures. After calculating objects parameters, these parameters must be compared to find out which parameters belong to the same object and which ones are different. If sonar and laser data detect same object with minor differences in parameters, mostly, laser's data are more suitable than sonar's. Generally it could be presumed that laser's data are more precise than sonar's.

But while differences between extracted parameters are bigger than a threshold, we assume that these parameters belong to different objects and should run data association procedure.

Data association is a matching procedure between sensors measurement and predictions from previous estimation of object locations existing in map, to assign one prediction to each sensor measurement. If a sensor measurement does not match to any predictions, it means that this object is observed for the first time as a new object, and it should be added to the previous existing objects in map. Realization of data association in our robot is performed by Nearest Neighbor Standard Filter (NNSF). [10]

Based on innovations and measurement covariance matrixes, Kalman filter gain could be find simply and finally, the post prior state estimation is updated and this iteration step is completed.

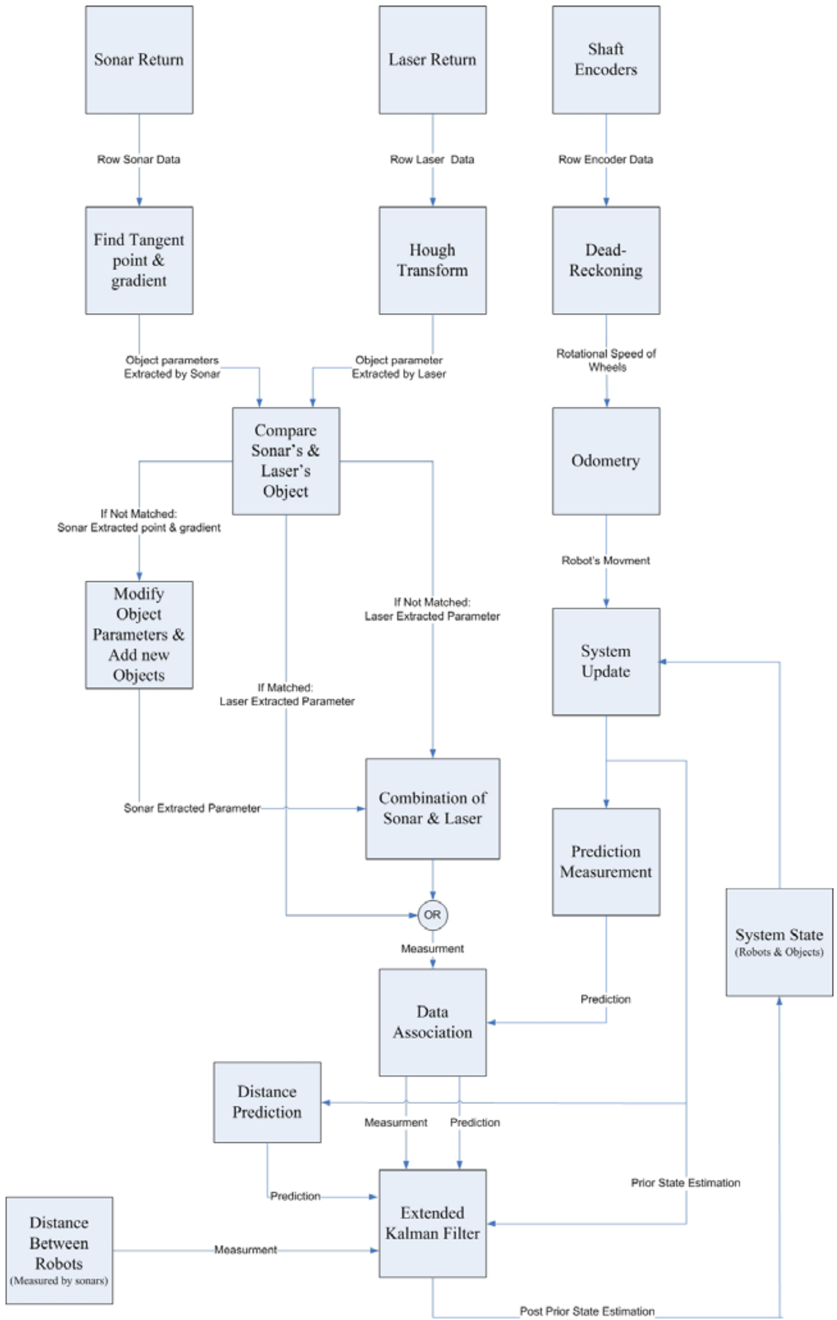

Fig. 12. SLAM Algorithm Flowchart

6.3. Fuse of Inertial Navigation System and Odometery

In our fully autonomous robot we are working on a research project to fuse the odometery data and an Inertial Measurement Unit data to reduce summing odometery errors. The applied IMU is from Xsense Company which is shown in Fig. 13.

Typical performance characteristics of MTi and MTx orientation output.

Dynamic Range:	all angles in 3D
Angular Resolution:	0.05° RMS [02]
Static Accuracy (roll/pitch):	<0.5°
Static Accuracy (heading) [03]:	<1.0°
Dynamic Accuracy:	2° RMS [04]
Update Rate:	user settable, max 120 Hz [0]

Fig. 13. MTi Sensor from Xsense Company

As the robot velocity is quite slow and sometimes it is completely stationary, we have experienced a few problems regarding the biased data of accelerometers inside the IMU. Besides we have to integrate twice from the accelerometer to find the position of the robot, bias voltage would be so disturbing. To solve this problem we have proposed a fuzzy algorithm to reset the summing errors related to the biased output of the IMU [3]. Fig. 14 shows the algorithm.

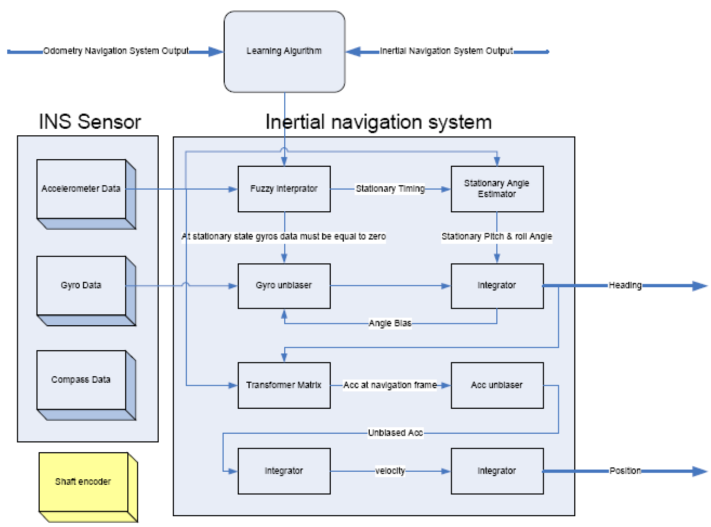

Fig. 14. Fuzzy Algorithm to Fuse IMU and Odometery Data

The main idea of this navigation algorithm is based on use of "tightly coupled" configuration of odometry and INS data by a fuzzy decision maker algorithm [1], [2], [4].

7. Sensors for Victim Identification

For victim identification we have used a set of different sensors which are explained below;

- TPA81Thermopile Array to measure Victim's temperature [13]

The TPA81 which is shown in Fig. 15 is a thermopile array detecting infra-red in the 2um-22um range. This is the wavelength of radiant heat. These Pyro-electric sensors can only detect a change in heat levels though - hence they are movement detectors. Although useful in robotics, their applications are limited as they are unable to detect and measure the temperature of a static heat source. The TPA81 can also control a servo to pan the module and build up a thermal image. The TPA81 can detect a candle flame at a range 2 meters (6ft) and is unaffected by ambient light.

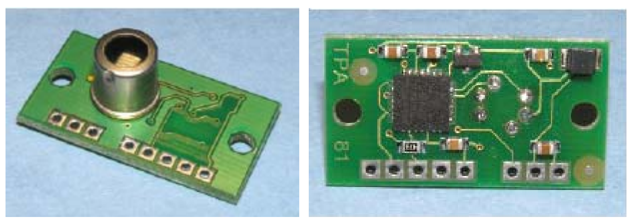

Fig.15. TPA81 Thermopile Array

- Infrared Gas Sensors (Carbon Dioxide CO_2 Sensor) [11]

The sensor that we have used to sense the exhale CO_2 from victims is Cascard II module (Fig. 16) from Edinburgh Instrument Co. The Gascard range of OEM infrared gas sensors offer excellent quality detection and measurement of many gases, using pumped or diffusion aspiration technology to provide rapid instrument response characteristics. The Gascard II range provides high accuracy detection and measurement of CO2, CH4 (methane), C6H14 (hexane), and other hydrocarbon gases, with available measurement ranges varying from 0-2000ppm to 0-100% by volume.

MODEL	Gas	Accuracy*	Stability	Repeatability @ zero	Repeatability @ span
Gascard II (0-3000ppm)	CO_2	+/-2% of range	+/- 2% of range over 12 months	+/- 0.3%	+/- 1.5%

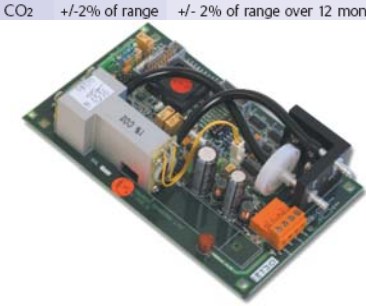

Fig. 16. Carbon Dioxide Module

We have also used a set of Cameras to identify the victim's tag, motion and its gesture. A Microphone is also used to detect victim's sound.

8. Robot Locomotion

We have designed two different types of robot for different arena (situation). But as a general rule all robots are based on differential servomechanism which is explained briefly. We have used two identical dc servomotors to drive each robot. The robot may travel forward or backward by driving two motors in CW or ACW direction and it may rotate around its center of gravity by driving motors in reverse directions.

It should be mentioned that all the mechanical design process are carried out by our team members using AutoCAD2002 and 3DWORKINGMODEL. All the robot parts are built by CNC machines.

The servomechanism control system is based on an embedded microcontroller and H-bridge driver using PWM method. The velocity and position of the servomechanism are controlled using PID control algorithm based on optical encoder feedback.

NAJI-V:

This robot is designed for outdoor and harsh environment. This robot is a new version of NAJI_II. Fig 17 illustrates the 3D views of the robot.

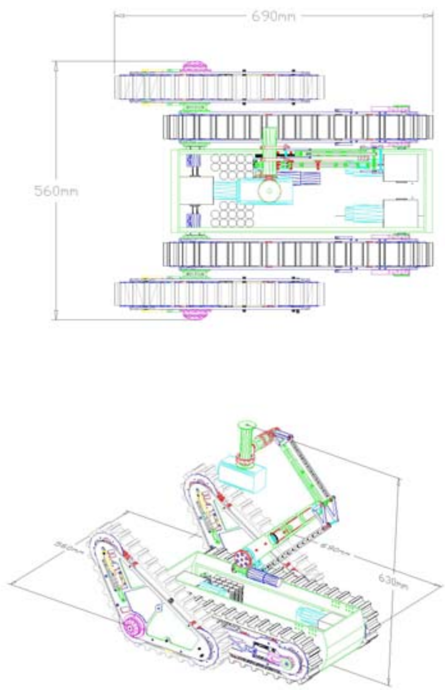

Fig 17.: NAJI-V Dimensions.

It should be noted that at the time of preparing this TDP, the robot is designed and all individual parts are manufactured but not assembled yet. We are waiting to receive the motors from FAULHABER Co. Different parts of the robot are shown in Fig. 18.

Fig. 18. NAJI_V Manufactured Parts

Table 1: NAJI-V Specifications

Weight	15	Kg
Carriage Load	20	Kg
Max Velocity	20	Cm/Sec
Length	690	mm
Width	560	mm
Height	322	mm
Locomotion	2 DC servomotor	150 W

NAJI-IV:

This simple robot is just designed for indoor and flat surface. The control and navigating of the robot is quite easy. Dead-reckoning technique with digital compass and IMU, provides a very good and accurate localization method. The systematic mechanical errors compare to other NAJI robots is quite less.

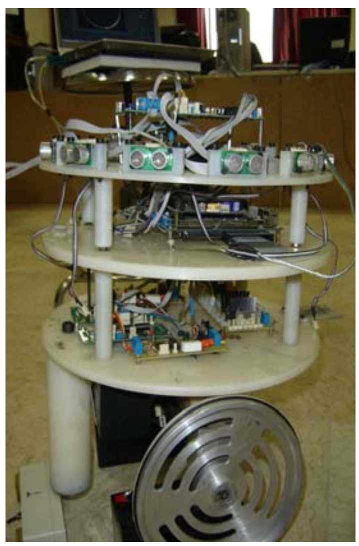

Fig 19. Photo of the NAJI-IV

Table 2: NAJI-IV Specifications

Weight	5	Kg
Carriage Load	7	Kg
Max Velocity	0.5	m/Sec
Length	378	mm
Width	378	mm
Height	180	mm
Locomotion	2 DC servomotor	120 W

9. Team Training for Operation (Human Factors)

As mentioned before we have managed to train two operators. We have also prepared a combination of the Yellow and Red arena next to the Mechatronics Research Laboratory to test our robots. So operators can improve their skill to command the rescue robot professionally.

10. Possibility for Practical Application to Real Disaster Site

As we experience so many earthquakes in our country, our main goal is not only to take part in Robocup competition but we are aiming to design and build real rescue robot for real situation. For example our first robot NAJI-I which was in Padova was tested in Bam earthquake to realize real problems and finding proper solutions.

11. System Cost

Most of the mechanical parts of the robots are designed and built by our team members. Depends on the type of the motor we have bought our DC servomotors from Faulhaber. Ultrasonic sensors and digital compass are bought from Devantech Robot-Electronics. Other sensors, Wireless-LAN card and electronics parts are bought from local shop. The cost for each robot differs depending on the size and complexity of the robot but approximately each robot cost us about 3000-7000 $US.

Table 3. The sensor's cost:

Sensor's Type	Price Per Piece	Distributor
Co2 Detector	600 US$	www.edinst.com
Laser Scanner	180000 yen	www.hokuyo-aut.jp
Temperature sensor	43.02 £	www.robot-electronics.co.uk
Ultrasonic	25.50£	www.robot-electronics.co.uk
Camera 260x	329 US$	www.pinecomputer.com
IMU	2100 US$	www.xsense.com
Motors	*Table 4	www.minimotor.ch

Table 4. Motors Price List

Pos.	Product	Type	Num	
New	**DC-Micromotor**	**2642 W 012 CR**	**5**	
		Encoder HEDS5540 A14	**8**	
		Planetary Gearhead 26/1 159:1	**5**	
New	**DC-Micromotor**	**3257 G 024 CR**	**4**	
		Encoder HEDS5540 A14	**4**	
		Planetary Gearhead 38/1 66:1	**4**	
Previous offer				
1	**DC-Micromotor**	**3863 A 024C**	**4**	**CHF 646.00/unit**
		Encoder HEDS5540 C12	**4**	
		Planetary Gearhead 38/1 3,71:1	**4**	
2	**DC-Micromotor**	**3863 H 024C**	**2**	**CHF 828.00/unit**
		Encoder HEDS5540 C12	**2**	
		Planetary Gearhead 44/1 23:1	**2**	
3	**DC-Micromotor**	**2642 W 024 CR**	**2**	**CHF 517.00/unit**
		Encoder HEDS5540 C14	**2**	
		Planetary Gearhead 30/1 43:1		
4	**DC-Micromotor-Enc.**	**2224 U 024 SR IE2-128**	**2**	**CHF 302.00/unit**
		Planetary Gearhead 23/1 415:1	**2**	
5	**DC-Micromotor-Enc.**	**2224 U 024 SR IE2-128**	**2**	**CHF 286.00/unit**
		Planetary Gearhead 23/1 246:1		

References:

[1] Lauro Ojeda, Giulio Reina, and Johann Borenstein, "Experimental Results from FLEXnav: An Expert Rule-based Dead-reckoning System for Mars Rovers" University of Michigan Advanced Technologies Lab

[2] Xiaoying Kong, "Inertial Navigation System Algorithms for Low Cost IMU" Department of Mechanical and Mechatronic Engineering, The University of Sydney, August 27, 2000

[3] Krantz, D. and Gini, M., 1996, "Non-Uniform Dead-Reckoning Position Estimate Updates." *Proc. of the 1996 IEEE Int. Conference on Robotics and Automation Minneapolis*, MN, April 22-25, pp. 2061-2066.

[4] KEVIN J. WALCHKO, “LOW COST INERTIAL NAVIGATION: LEARNING TO INTEGRATE NOISE AND FIND YOUR WAY”, A THESIS PRESENTED TO THE GRADUATE SCHOOL OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE UNIVERSITY OF FLORIDA 2002

[5] J. Andrade-Cetto, A. Sanfeliu, "Concurrent Map Building and Localization on Indoor Dynamic Environments", Institut de Rob`otica i Inform`atica Industrial, Spain.

[6] J. J. Leonard, "Directed Sonar Sensing for Mobile Robot Navigation", PhD Thesis, Cambridge, U.S.A, 2002.

[7] P. Holmberg, "Sensor Fusion with Coordinated Mobile Robots", Master Thesis Linköping University, Sweden, 2003.

[8] Ioannis Rekleitis, "Cooperative Localization and Multi-robot Exploration", Master Thesis, McGill University, 2003.

[9] V. Nguyen, A. Martinelli, N. Tomatis, R. Siegwart, "A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics", Autonomous Systems Laboratory Ecole Polytechnique F´ed´erale de Lausanne (EPFL), Switzerland.

[10] I. Hwang , H. Balakrishnan , K. Roy , J. Shin , L. G. C. Tomlin, "Multiple Target Tracking and Identity Management", Stanford University.

[11] www.edinst.com
[12] www.hokuyo-aut.jp
[13] www.robot-electronics.co.uk
[14] www.pinecomputer.com
[15] www.xsense.com
[16] www.minimotor.ch