Modular Design and Structure for a Mobile Sensory Platform

Ayssam Elkady, Vipul Babariya, Jovin Joy and Tarek Sobh
School of Engineering
University of Bridgeport

***Abstract*—A mobile manipulator is a manipulator mounted on a mobile platform with no support from the ground. We are already in the process of building a platform (RISCbot II) which consists of a comprehensive sensor suite and significant end-effector capabilities for manipulation. In order to reduce the uncertainty in localization, sensor fusion is used to create an efficient and effective user interface to facilitate teleoperation by enhancing the quality of information that is provided to the teleoperator.**

This paper presents the modular design process of the RISCbot II mobile manipulator. In the design process, the overall design of the system is discussed and then the control process of the robot is presented. Furthermore, the tasks that the RISCbot II can perform such as teleoperation, navigation, obstacle avoidance, manipulation, face detection and recognition, and map building are described.

I. Introduction

A mobile manipulator offers the dual advantage of mobility offered by the platform, and dexterity offered by the manipulator. The mobile platform extends the workspace of the manipulator. We are developing and constructing a mobile manipulation platform called RISCbot II. The RISCbot II mobile manipulator platform is shown in figure 1. Mobile manipulators are potentially useful in dangerous and unpredictable environments such as construction sites, space, underwater, service environments, and in nuclear power stations.

Fig. 1. The RISCbot II mobile manipulator platform.

Sensor fusion has been an active area of research in the field of computer vision and mobile robotics. Sensor fusion is the combination of sensory data from different sources, resulting in better and more reliable information of the environment than data derived from any individual sensor. Sensor fusion algorithms are useful in low-cost mobile robot applications where acceptable performance and reliability is desired, given a limited set of inexpensive sensors like ultrasonic and infrared sensors. Depending on the modalities of the sensors, sensor fusion can be categorized into two classes (as described in [1]): sensor fusion using complimentary sensors, and sensor fusion using competing sensors. Complementary sensors consist of sensors with different modalities, such as a combination of a laser sensor and a digital camera. In contrast, competing sensors are composed of sensors suit which have the same modality, such as two digital cameras which provide photographic images of the same building from two different viewpoints. Sensor fusion has some critical problems, including the synchronization of sensors. Different sensors have different resolutions and frame rates, so the sensors need to be synchronized before their results can be merged by fusing the data from multiple sensors, and presenting the result in a way that enables the tele-operator to perceive the current situation quickly. Sensor fusion is used to reduce the workload for the operator to enable him or her to concentrate on the task itself. Sensor fusion is commonly used to reduce uncertainty in localization, obstacle avoidance, and map building. Furthermore, sensor fusion may be used to improve teleoperation by creating user interfaces which efficiently facilitate understanding of remote environments and improve situational awareness.

In this paper we describe our mobile manipulation platform, and introduce the tasks that the RISCbot II is performing them in, specifically teleoperation, navigation, obstacle avoidance, manipulation, face detection, and face recognition.

K. Elleithy et al. (eds.), *Technological Developments in Networking, Education and Automation*,
DOI 10.1007/978-90-481-9151-2_76, © Springer Science+Business Media B.V. 2010

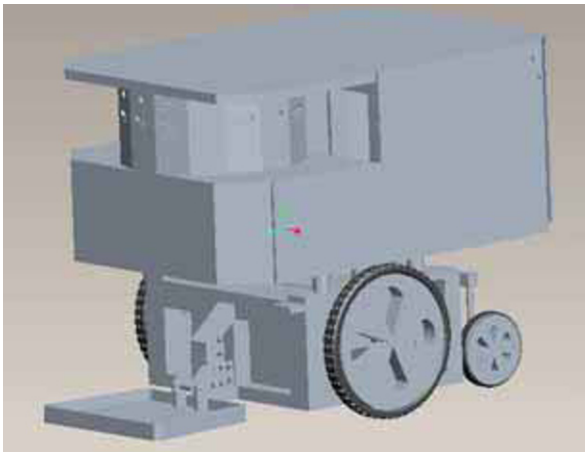

Fig. 2. The Prototype Structure.

II. DESIGN STRUCTURE OF THE RISCBOT II

In the final design as shown in figure 2, the three layer design was maintained. The base comprises of plywood holding a square core iron structure to provide support for the manipulator. Cylindrical structure to mount sensors is fixed on the central layer. The topmost layer is used to mount a stereo camera and a manipulator. The manipulator is mounted in such a way as to transfer its weight to the square core iron from the base.

Furthermore, there is a 2" diameter hole being drilled in the bottom layer to pass the wirings from the battery. The joystick of the wheelchair is mounted on the core iron structure vertically. There are seven sensors mounted on the back face of the bottom plywood and the center of all sensors is kept in one line. Different views of the structure are shown in figure 3. Plywood was chosen in the final design for building the layered structure due to its cost effectiveness and availability.

A. Individual Parts Description

A summary of individual parts used in the final prototype is described in table I.

TABLE I
INDIVIDUAL PARTS USED IN THE FINAL PROTOTYPE.

Part	Material Type	Quantity
Top Plywood Sheet	Plywood	1
Middle Plywood Sheet	Plywood	1
Bottom Plywood Sheet	Plywood	1
Central Support Structure	Core Iron	1
Support Part	Core Iron	4
Cylindrical Part	Wood	1
Backward Sensor Part	Wood	3

1) Bottom Plywood sheet: The bottom plywood sheet used has a dimension of $32" \times 22" \times 0.75"$ as shown in figure 4. In order to mount all components, thickness of the plywood sheet was set at 0.75". Front pocket is used to mount the laptop is shown in figure 5. This plywood sheet is directly mounted on six bosses extending from the wheelchair frame. Two I-support and Central support structure are mounted on top of the sheet.

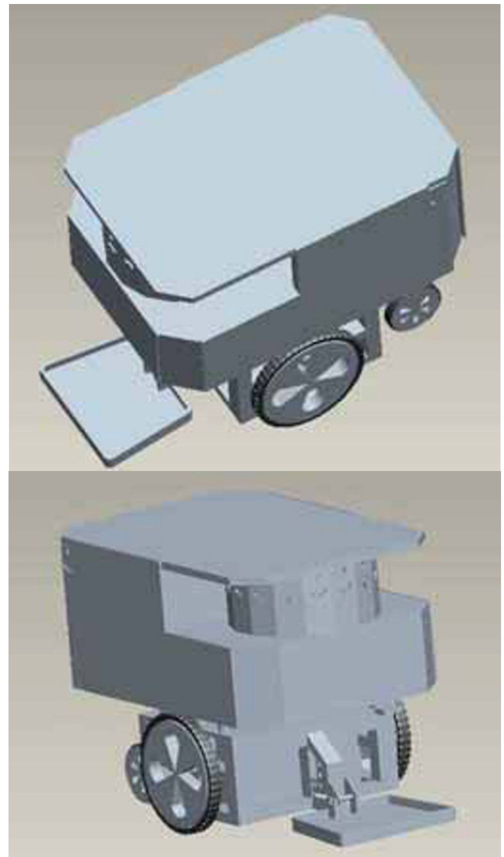

Fig. 3. Different views of the prototype structure.

Fig. 4. Bottom plywood sheet.

2) Middle Plywood sheet: This plywood sheet shown in figure 6 is mounted on the square brackets of I-support and central support structure extending out of the bottom. Its dimensions are $16" \times 22" \times 0.75"$. This sheet is used to support the cylindrical sensor mount.

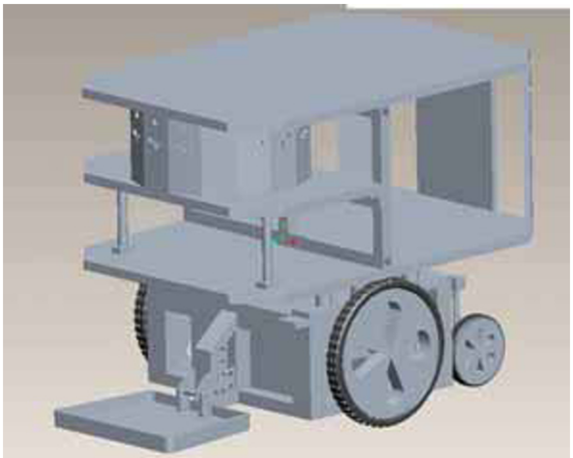

Fig. 5. The prototype showing structural support.

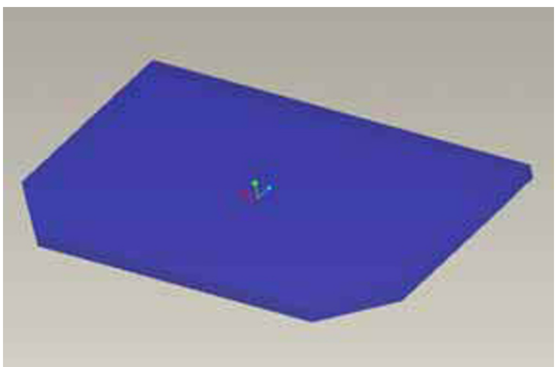

Fig. 6. Middle Plywood sheet.

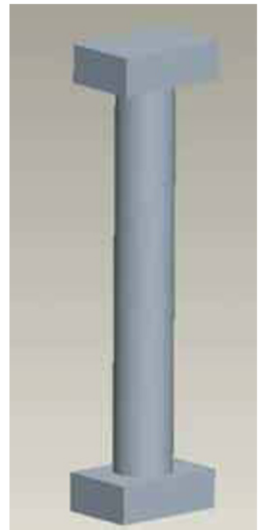

Fig. 7. I-Support Part.

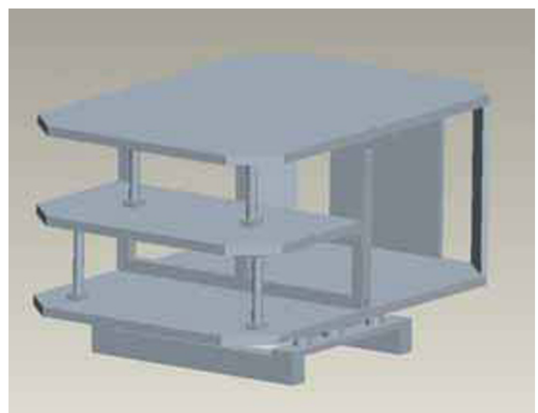

Fig. 8. The prototype showing structural support without Cylindrical Sensor Mount.

3) I-Support Part: It is manufactured from core iron of 0.5" diameter with square brackets welded on each end as shown in figure 7. Four of these were used in the design; two of them are used for supporting the middle plywood sheet and the others are mounted on the middle plywood and is located in the Cylindrical Sensor mount to support the top plywood as shown in figure 8.

4) Cylindrical Sensor Mount: This structure is assembled from 9 rectangular wooden plates arranged around a incomplete circle (1.25π) of radius 9.5". Each rectangular plate is 2.75" thick and mounts 2 type of sensors, namely infrared and ultrasonic, as shown in figure 9.

5) Central Support Structure: This structure is a rectangular hollow frame made from core iron of 1" thickness. In the middle, an angle bracket is welded to support the middle plywood sheet. This support structure rests on the bottom plywood sheet and it provides support for the top plywood and manipulator.

6) Top Plywood sheet: The top plywood sheet, shown in figure 11 is mounted on top of the central support structure. It is also supported by two I-support extended from the middle plywood, cylindrical sensor mount, and the rear sensor rack. This sheet supports the manipulator and stereo camera.

B. Analysis Of the final Structure

The analysis of the structure was done on ANSYS Workbench 11.0. Static structural analysis was done to check stress, strain, and deformation of the structure. The procedure used is described below:

1) Import the final prototype model designed using Pro-Engineer software into ANSYS.
2) Defining the material properties of each individual part

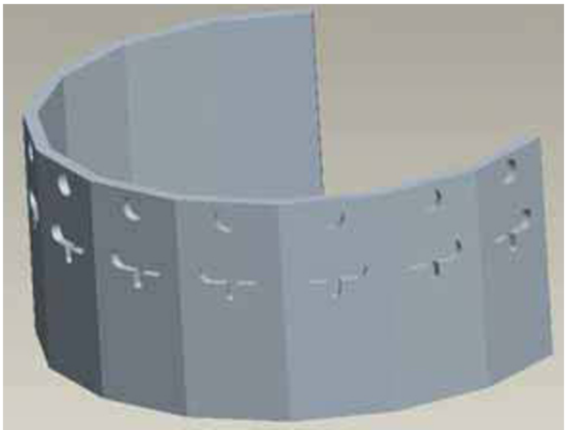

Fig. 9. Cylindrical Sensor Mount with rectangular plates on perimeter.

Fig. 10. Central Support Structure.

Fig. 11. Top Plywood sheet.

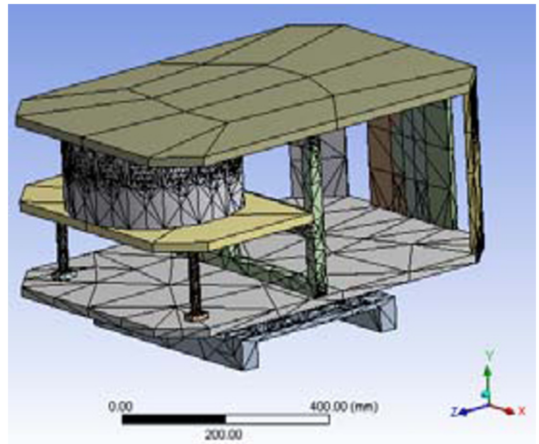

Fig. 12. Structure Meshing

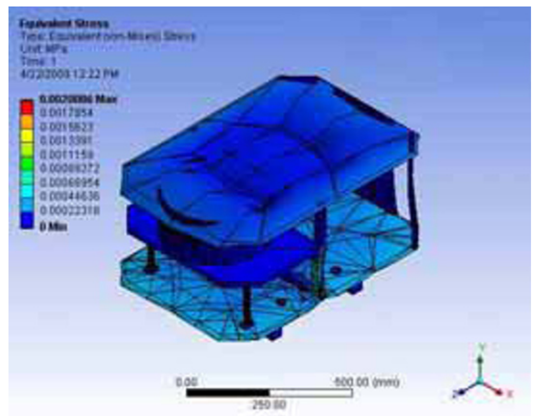

Fig. 13. Equivalent stress distribution throughout structure.

3) Assign meshes to the structure (figure 12)
4) Choose various support conditions and load application on the structure
5) Assign all the forces that affect the structure

The stress distribution in the structure is shown figure 13 and the maximum stress of 2.0086 kPa is measured on the edges of plywood. Strain generated in all the assembled components can be observed in figure 14. Total deformation found throughout the structure from the simulation was uniform as shown in figure 15. The maximum equivalent Elastic Strain and Total deformation found is negligible.

III. Applications

A. Face Detection

Efficiency of any detection process can be improved dramatically if the detection feature encodes some information about the class that is to be detected. Our primary method

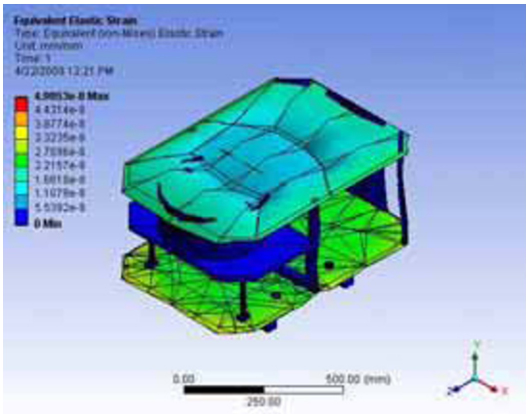

Fig. 14. Equivalent elastic Stain simulation result.

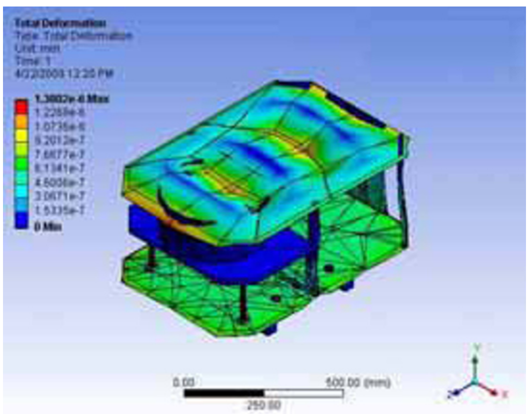

Fig. 15. Total deformation.

of detection is based on Haar-Link Features (as described in [2]) and OpenCV computer vision library. Haar encodes the existence of oriented contrasts between regions in images. A set of features can be used to encode the contrast exhibited by human faces and their special relationships.

Our detection program is derived primarily from OpenCV and uses classifiers (cascade of boosted classifiers working with haar-like features) that are trained with a few hundred sample views of human faces. Classifiers used in our project were obtained from OpenCV library. The classifier is built using positive and negative examples of the same size. After a classifier is trained it can be used to search across an image at locations. The output of the classifier is 1 if it finds an object (face) in a region. The classifier is resized, rather than the image, to detect objects (faces) of different sizes. Therefore, to find faces of unknown sizes, the scan procedure is done several times at different scales.

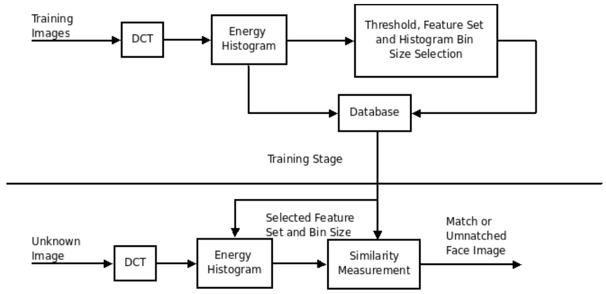

Fig. 16. Overview of Face Recognition Module

B. Face Recognition

Face Recognition is a task that identifies or verifies a person from a digital image or a video source. The primary method of face detection uses a overlapping energy histogram as described in Face Recognition by Overlapping Energy Histograms [3]. For the sake of efficiency, OpenCV library was used extensively while developing the code.

1) General Approach: Face Recognition algorithm is a two step process with a training stage and search stage. The general approach is shown in figure 16. The training stage is done initially to create a database which is not computed again as it is a time consuming process. The search phase uses the database created from the previous stage to look up unknown images. Overlapping energy Histogram comprises of taking DCT coefficients of every 8x8 pixel values with a 75 percent overlap as shown in figure 17 (i.e Taking DCT for every 8x8 block after every 2 pixels). A sample of which is shown in Figure 18. Once the image has been transformed into a histogram Euclidean Distances as show in Eqn 1 is computed, where Ω is the feature vectors calculated via the histogram. Further Euclidean Distance has been used in many face recognition techniques [3]. This distance is used to calcualte the Database Threshold(θ), which is used for identifying and rejecting images at the search stage

$$\epsilon_n = \|\Omega - \Omega_n\|^2 \tag{1}$$

C. Feature Extraction

Once the image with a face is transformed using overlapping DCT procedure values of Feature Set F2,

$$\mathrm{F2} = [\mathrm{DC}; \mathrm{AC}_{01}; \mathrm{AC}_{10}; \mathrm{AC}_{11}]$$

,is put into a histogram with a bin size of 60. The real challenge has been in selecting the appropriate appropriate bin sizes for calculating threshold values. A general deviation for the algorithm was using a predetermined bin size of 60 which proved to be optimally good, both computationally and for detection. This was determined by running a series of test on known images at the time of development.

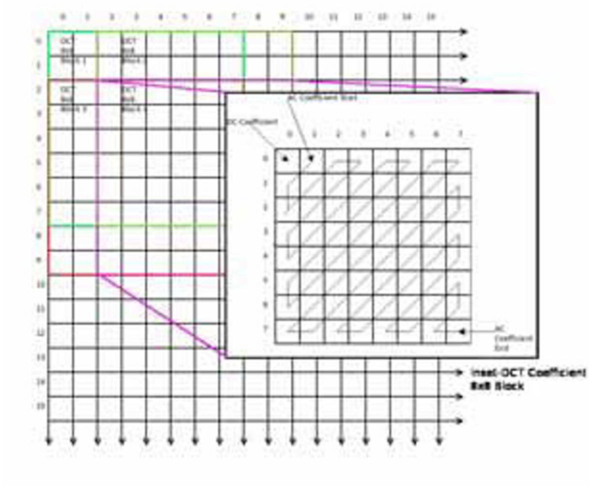

Fig. 17. Energy Histogram With 75 Percent Overlap

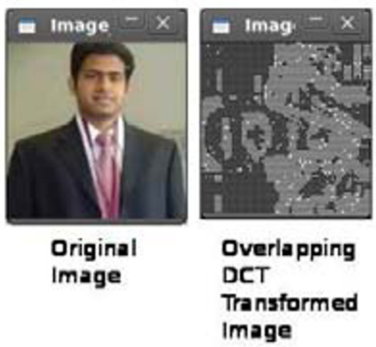

Fig. 18. Image and Corresponding Overlapping DCT Representation

1) Threshold Selection: Threshold for the image database is calculated by intra and inter class information gained from the training dataset. Intra class(D) information is obtained via computing the Euclidean distance between the images of an individual and inter class(P) information is obtained via computing the distance between images of an individual with others in the database. Once this is done for all the images in the database the Threshold(θ) is defined by

$$\theta = \frac{D_{max} + P_{min}}{2} \tag{2}$$

2) Image Search: Once the required feature from the image to be recognized is extracted, it's features are cross computed with each individual in the database to see if it falls withing the range of the database threshold. If more than one individual matches with the image that is searched for, the individual with the closest threshold value is usually the correct person.

Fig. 19. Face Detection Module Detecting Multiple People in a Single Frame

Fig. 20. Face Recognition Module Output

3) Face Detection/Recognition Testing: Face Detection and Recognition modules has to be tested simultaneously with the input of Detection Program fed into the recognition program. The Face Detection program on an average detects up to 5 faces real time(30 Frames/Second) running on a Dual Core Intel Processor, therefor bringing the total to 150 Images/Second. This is a 10 fold increase from the original Viola-Jones implementation. Figure 19 shows a sample output from the detection program. The main concern regarding the number of images that can be detected per frame is the computational requirement and the need to maintain real time performance.

Recognition module on the other hand can take each of the detected faces and search the database to find possible matches. The initialization process of the Face Recognition database was found to be a processor hog, hence plans to recompute the database values at run time had to be abandoned. Another bottleneck noticed was the total memory requirements for the database, which increased due to storing the feature vectors in uncompressed formats in system memory. Output of face recognition module can be seen in Figure 20.

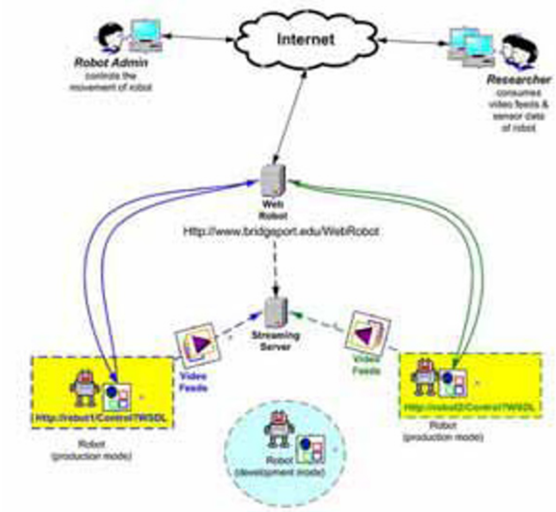

Fig. 21. RISCbot II website Architecture.

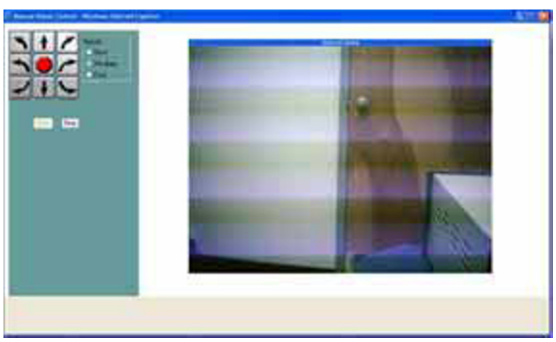

Fig. 22. Web-based manual control.

D. Teleoperation

We have implemented a web-based application in order to control the RISCbot II. The framework is implemented with ASP .NET 2.0 in $C\#$. The Web Form communicates with the physical robot using Web Services which are also implemented in $C\#$. In this application, a graphical user interface is provided to RISCbot's teleoperator to allow him to interact with the robot. There are two modes to control the robot; autonomous mode and manual mode. Since the RISCbot II could carry out a number of tasks, this web-based console allows the user to link the video/sensor data from the robot with various computer vision applications. The architecture of the web-based application for RISCbot II is shown in figure 21 and the manual control mode is shown in figure 22.

E. Manipulability

Studying the performance characteristics of the robot such as dexterity, manipulability, and accuracy is very important to the design and analysis of a robot manipulator. The manipulability is the ability to move in arbitrary directions. The manipulability index is considered as a quantitative and performance measure of the ability for realizing some tasks. This measure should be taken into consideration in the design phase of a serial robot and also in the design of control algorithms.

In [4], we presented a new method for measuring the manipulability index, and then justified this concept by visualizing the bands of this index resulting from our experiments implemented on different manipulators, such as the Puma 560 manipulator, a six DOF manipulator, and the Mitsubishi Movemaster manipulator. In fixed-base serial manipulators, manipulability depends on link lengths, joint types, joint motion limits and the structure of the manipulator. In mobile manipulators, the manipulability depends on the kinematics, geometric design, the payload, and the mass and mass distribution of the mobile platform. Thus, the manipulability measure in mobile manipulators is very complicated due to the coupling between the kinematic parameters and the dynamics effect. Furthermore, we use the proposed method for measuring the manipulability index in serial manipulators to generalize the standard definition of the manipulability index in the case of mobile manipulators.

F. Navigation and Obstacle Avoidance

A prerequisite task for an autonomous mobile robot is the ability to detect and avoid obstacles given real-time sensor readings. Given partial knowledge about the robot's environment and a goal position or a series of positions, navigation encompasses the ability of the robot to act based on its knowledge and sensor values so as to reach its goal positions as efficiently and reliably as possible. The techniques used in the detection of obstacles may vary according to the nature of the obstacle. The resulting robot motion is a function of both the robot's sensor readings and its goal position. The obstacle avoidance application focuses on changing the robot's trajectory as informed by sensors during robot motion. The obstacle avoidance algorithms that are commonly used can be summarized as the following [5]: the bug algorithm, tangent Bug, Artificial Potential Fields, and Vector Field Histogram.

G. Path planning and map building

Given a map and a goal location, path planning involves identifying a trajectory that will bring the robot from the initial location to the goal location. During execution, the robot must react to unforeseen obstacles and still reach its goal. In the navigation problem, the requirement is to know the positions of the mobile robot and a map of the environment (or an estimated map). The related problem is when both the position of the mobile robot and the map are not known. In this scenario, The robot starts in an unknown location in an unknown environment and proceeds to gradually build the map of the existing environment. In this case, the position of the robot and the map estimation are highly correlated. This problem is known as *Simultaneous Localization and*

Map Building (SLAM) ([6] and [7]). SLAM is the process of concurrently building a feature based map of the environment and using this map to get an estimation of the location of the mobile platform.

In [8], the recent Radio Frequency Identification (RFID) was used to improve the localization of mobile robots. This research studied the problem of localizing RFID tags with a mobile robot that is equipped with a pair of RFID antennas. Furthermore, a probabilistic measurement model for RFID readers was presented in order to accurately localize RFID tags in the environment.

IV. Implementation and Results

We are developing and constructing the mobile manipulator platform called RISCbot II(as shown in figure 1). The RISCbot II mobile manipulator has been designed to support our research in algorithms and control for an autonomous mobile manipulator. The objective is to build a hardware platform with redundant kinematic degrees of freedom, a comprehensive sensor suite, and significant end-effector capabilities for manipulation. The RISCbot II platform differs from any related robotic platforms because its mobile platform is a wheelchair base. Thus, the RISCbot II has the advantages of the wheelchair such as high payload, high speed motor package (the top speed of the wheelchair is 6 mph), Active-Trac and rear caster suspension for outstanding outdoor performance, and adjustable front anti-tips to meet terrain challenges.

In order to use the wheelchair as a mobile platform , a reverse engineering process has been used to understand the communication between the joystick of the wheelchair and the motor controller. This process was done by intercepting the continuous stream of voltages generated by the joystick after opening the joystick module and reading the signals within joystick wires that send the signals to the wheelchair controller.

We used different types of sensors so that the RISCbot II can perceive its environment with better accuracy. Our robot hosts an array of 13 $LV-MaxSonar^{\circledR}-EZ0^{TM}$ ultrasonic sensors. The working envelope of the 13 sonars is shown in figure 23 The sensors are suitable for obstacle avoidance applications but their wide beams are unable to distinguish features within the beam angle, making sonars a poor choice of sensor for fine feature extraction within indoor environments. This resolution problem is magnified for objects further away from the robot (i.e., objects appearing at the wide end of the beam). Lastly, our robot is also equipped with an array of 11 Sharp GP20A21YK infrared proximity sensors above the sonar ring. The sonar and infrared sensors were mounted together so that their beams are oriented in the same direction. The configuration of sonar and infrared sensors is shown in figure 24. These sensors allow the RISCbot II to obtain a set of observations and provide these observations to the controller and higher decision making mechanisms. The controller acts upon the received observations to cause the robot to turn in the correct direction. The Integration of these modules together constitutes an intelligent mobile robot.

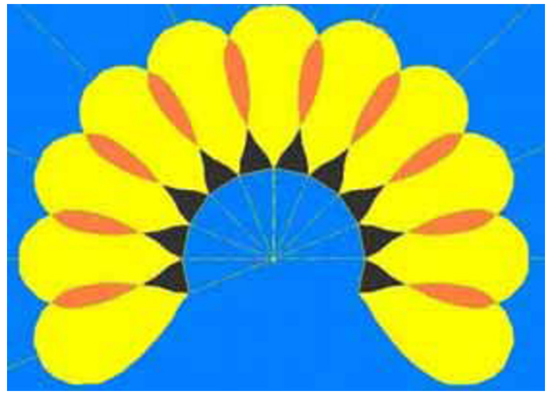

Fig. 23. Working Envelope of EZ0 sonar Sensors.

Fig. 24. A close-up view of the sonar and infrared sensors array.

A main drawback of these sensors is that they can only accurately measure obstacle distances within a range of 0.1m to 0.8 m. Another drawback of these sensors is that they are susceptible to inaccuracies due to outdoor light interference as well as an obstacle's color or reflectivity characteristics which can be seriously affected by windows and metallic surfaces.

Note that since our sonar and infrared sensors are in fixed positions, our experiments concentrated on performing data fusion on data obtained from a particular fixed height in the environment. In this project, sonar and infrared sensors are used together in a complementary fashion, where the advantages of one compensate for the disadvantages of the other.

As shown in figure 25, the RISCbot II software which is written in Visual $C\#$ and runs on a laptop reads the values of all sensors at a rate of 10 HZ gathered in the data acquisition. The RISCbot II software maps the sensory inputs to a series of actions which are used to achieve the required task. Based on the used algorithm, the RISCbot II software responds to the sensor data by generating streams of voltages corresponding to the joystick signals to the wheelchair controller. These voltages control the direction and the speed of the wheelchair causing

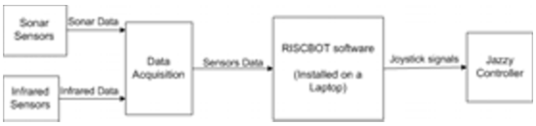

Fig. 25. The components of the RISCbot II system.

the RISCbot II to turn in the desired direction.

The experimental result indicates that the RISCbot II can detect unknown obstacles, and avoid collisions while simultaneously steering from the initial position towards the target position.

V. Conclusions and Future Work

In this paper, the mobile manipulation platform RISCbot II has been presented. The RISCbot II platform differs from any other robotic platform because its mo-bile platform is a wheelchair base. Thus, the RISCbot II has the advantages of the wheelchair. Furthermore, the RISCbot II consists of a comprehensive sensor suite, and significant end-effector capabilities for manipulation. In addition, we have used infrared and sonar sensors to monitor if any type of obstruction is in the path of the robot. This research aspires to find real-time collision-free trajectories for mobile manipulation platforms in an unknown static or dynamic environment containing some obstacles, between a start and a goal configuration.

Path planning for mobile robots is one of the key issues in robotics research that helps a mobile robot find a collision-free path from the beginning to the target position in the presence of obstacles. Furthermore, it deals with the uncertainties in sensor data.

The objective for this project is to implement a general purpose mobile manipulator that can be used for various applications such as teleoperation, navigation, obstacle avoidance, manipulation, 3-D reconstruction, map building, face detection, and face recognition. There are great benefits in using a mobile manipulator in dangerous, inaccessible and toxic environments. In teleoperation, a human operator controls the RISCbot II from a distance. The teleoperator has some type of display and control mechanisms, and the RISCbot II has sensors which gather all the information about the remote environment, an end-effector, and mobility.

In our anticipated future work, there will be an ongoing effort for the development of multiple mobile manipulation systems and platforms which interact with each other to perform more complex tasks exhibiting intelligent behaviors utilizing the proposed manipulability measure.

References

[1] A. Yilmaz, "Sensor fusion in computer vision," in *Urban Remote Sensing Joint Event*, April 2007, pp. 1–5.

[2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2001*, vol. 1, 2001, pp. I–511–I–518.

[3] R. Tjahyadi, W. Liu, S. An, and S. Venkatesh, "Face recognition via the overlapping energy histogram," in *IJCAI*, 2007. [Online]. Available: http://www.ijcai.org/papers07/Papers/IJCAI07-465.pdf

[4] T. S. M. Mohammed, A. ElKady, "New concept in optimizing manipulability index of serial manipulators, using svd method," in *Industrial Electronics, Technology and Automation (IETA 07)*, Dec 2007.

[5] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, 1st ed., ser. Intelligent Robotics and Autonomous Agents series. The MIT Press, April 2004.

[6] S. C. H. D.-W. M.W.M.G. Dissanayake, P.Newman and M. Csorba, "A solution to the simultaneous localization and map building (slam)problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229 – 241, Jun 2001.

[7] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 242 – 257, Jun 2001.

[8] D. F. K. F. Dirk Hähnel, Wolfram Burgard and M. Philipose, "Mapping and localization with rfid technology," in *IEEE International Conference on Robotics and Automation (ICRA 2004)*, vol. 1, New Orleans. LA, April 2004, pp. 1015 – 1020.