Multimodal User Supervised Interface and Intelligent Control (MUSIIC) for Assistive Robots

Zunaid Kazi, Marcos Salganicoff, Matthew Beitler,
Shoupu Chen, Daniel Chester and Richard Foulds
Applied Science and Engineering Laboratories
Alfred I. duPont Institute/University of Delaware
Wilmington, Delaware USA

Abstract

This paper reports on a method and system which integrates human-computer interaction with reactive planning to operate a telerobot for use as an assistive device. The system is intended to meet the needs of individuals with physical disabilities and operate in an unstructured environment, rather than in a structured workcell. This allows the user considerable freedom and flexibility in terms of control and operating ease. We describe a novel approach for an intelligent assistive telerobotic system for such an environment: speech-deictic gesture control integrated with a knowledge-driven reactive planner and a stereo-vision system which builds a superquadric shape representation of the scene.

1.0 Introduction

The rehabilitation robotics research literature describes many demonstrations of the use of robotic devices by individuals with disabilities [13, 17]. Unfortunately, many of the existing interface strategies, while important steps forward, have not met all of the desires of the user community. community. Prototype interfaces have taken two approaches to achieving effective use by individuals with disabilities. Many use commands which are issued by the user and activate the robot to perform pre-programmed tasks [32]. Similar types of pre-programmed commands were employed in vocational workstations [14,16,18,33,34].

In contrast to the command oriented rehabilitation robots, there have been a number of projects in which the user directly controls all the movements of the manipulator much like a prosthesis. [23,24,36]. This approach offers tremendous flexibility since there are no restrictions for a preset number of commands, a structured environment, or machine *knowledge* of the objects in the world, but is likely to be too demanding of many prospective users. Direct control of robots present many problems, including the requirement of good motor dexterity on the part of the operator and many other real-time perceptual and motor requirements which maybe difficult for many users to satisfy.

At the other extreme are completely autonomous systems that perform with effectively no user supervision, the long elusive goal of AI, robotics and machine vision communities. Unfortunately, this goal seems far away from the state of the art at this point, although many important incremental advances have been forthcoming in the past decades. Furthermore, absolute automation poses a set of problems stemming from incomplete *a priori* knowledge about the environment, hazards, strategies of exploration, insufficient sensory information, inherent inaccuracy in the robotic devices and the mode of operation [31].

A system with some built-in intelligence is needed to lighten the cognitive and physical load on prospective users. We describe a system that can exploit the low-level machine perceptual and motor skills and excellent AI planning tools currently achievable, while allowing the user to concentrate on handling the problems they are best suited for, namely high-level problem solving, object recognition, error handling and error recovery. By doing so, the cognitive loading of the system on the user is decreased, the system becomes more flexible, pleasant to use and less fatiguing. The resulting system is ultimately a more effective assistant.

Our approach is based on the assumption that the user's world is unstructured, but that objects within that world are reasonably predictable. We reflect this arrangement by providing a means of determining the three-dimensional shape and pose of objects and surfaces which are in the immediate environment, and an object-oriented knowledge base and planning system which superimposes information about common objects in the three-dimensional world. A third aspect involves the user interface which interprets the deictic gesture and speech inputs with the objective of identifying the portion of contour that is of interest to the user.

2.0 User Requirements

During the development of this project, several planning sessions were conducted with panels of individuals with disabilities. These sessions included a review of existing rehabilitation robotic devices (supported by video presentations of various systems) and discussion of the strengths and weaknesses of these approaches. The participants strongly supported the concept of a rehabilitation robot, but felt the existing interface strategies were ineffective in offering the full potential of the device to a person with a disability. The panel strongly suggested that an effective rehabilitation robot system should:

- operate in an unstructured environment
- require low mental load
- provide maximum speed of operation
- offer opportunities for use in a variety of environments (as opposed to a fixed workstation)
- be "natural" to operate (i.e use user functions which are easy and intuitive)
- provide maximum use of the range and other capabilities of the robot

These informal concerns matched a more quantitative study done by Batavia and Hammer, in which a panel of experts with mobility-related disabilities ranked 15 criteria of importance in assistive technology [4]. The results for robotic devices indicate that effectiveness is the highest priority, while operability is ranked second. These two criteria are defined by Batavia and Hammer as:

- **effectiveness**- the extent to which the functioning of the device improves the consumer's living situation, as perceived by the consumer, including whether it enhances functional capability and/or independence.
- **operability**- the extent to which the device is easy to operate and responds adequately to the consumers's operative commands.

The panel discussions and further research prompted the investigation into the development of a reactive, intelligent, "instructible" [11] telerobot controlled by means of a hybrid interface strategy where the user is part of the planning and control loop. This novel method of interface to a rehabilitation robot is necessary because in a dynamic and unstructured environment, tasks that need to be performed are sufficiently nonrepetitive and unpredictable, making human intervention necessary.

However, the design of the instructible aspect of the assistive robot system requires careful design; simple command based interfaces may be inadequate. The limitations of a command-based interface were discussed Michalowski et al [27]. While modern speech recognizers provide access to large numbers of stored commands, these investigators present the case that effective command of a robot will require use of more commands than is reasonable for the user to remember. As the number of possible commands grows, the human/machine interface becomes increasingly unmanageable. They propose greatly expanding the capability of the robot to not only recognize spoken words, but also understand spoken English sentences.

In a continuation of this work, Crangle et al [10, 11] provided an example where the user spoke the sentence, "*Move the red book from the table to the shelf.*" The proposed system would recognize the spoken sentence and understand the meaning of the sentence. The system would have a knowledge of the immediate world so that the robot knew the locations of the table and shelf, as well as the placement of the book on the table. While the use of such *natural language* interfaces is extremely interesting, and would offer great benefit, the limitations are many. The requirement that the world be entirely structured so that the robot knows precisely where every item is, is likely to be too demanding, and there are many unsolved issues in natural language processing. In addition, the inclusion of a vision system to accommodate a less structured environment will require the ability to perform object recognition.

A different approach to command-based robot operation was proposed by Harwin et al [20]. A vision system viewed the robot's workspace and was programmed to recognize bar codes that were printed on each object. By reading the barcodes and calculating the size and orientation of the barcode, the robot knew the location and orientation of every item. This was successful within a limited and structured environment. This system did not easily lend itself to a variety of locations and was not able to accommodate the needs of individuals with disabilities in unstructured environments. It did, however, demonstrate the dramatic reduction in *machine intelligence* that came by eliminating the need for the robot to perform object recognition and language understanding.

2.1 Multimodal Interfacing

Researchers have proposed a number of systems which investigate alternate modes of human-computer interaction in addition to speech and vision based ones. Work has been carried out in using gestures and hand pointing as a mode of man-machine interface. In some systems, researchers have required the users to use hand gloves [9, 35], while others require calibration for each individuals hand shapes and gestures [15]. Cipolla et al. [9] report in a preliminary work on gesture-based interface for robot control. Their system requires no physical contact with the operator, but uses un-calibrated stereo vision with active contours to track the position and pointing direction of a hand. Based on a ground plane constraint, their system is then capable of finding the indicated position in the robot's workspace. Pook describes a deictic gesture based tele-assistance system for direct control of a telerobot, although the system lacks a perceptual component [37]. Funda et al. [39] describe a teleprogramming approach which extracts user intentions from interaction with a virtual model of a remote environment, but their system requires an *a priori* 3-D model of the remote scene.

Some researches have also attempted to extend this concept by using multiple modes of man-machine interfacing. The concept of multimodal interfacing has been discussed extensively by Richard Bolt of the MIT Media Laboratory [5]. Bolt introduced the expression "*put that there*" in describing his work in optimizing the interface between a user and a large 2-D graphical display. Cannon at Stanford extended this concept to three dimensional robot operation [7]. Cannon's system has worked quite well in laboratory trials. However, it presents problems when being considered as an interface for rehabilitation robotics. The requirement that the user control two video cameras acting as a manually operated range-finder makes this less than desirable for an individual with disabilities.

This research extends the combined deictic gesture and spoken word of Bolt to true 3-D environments manipulated by a robot. It details an intuitive and efficient interface between the user and the manipulator as well as a reactive planning mechanism. We describe a new hybrid interface strategy combines command and control approaches to provide for user control of the robot through the use of multiple modes of interface in conjunction with sophisticated capabilities of the machine. Users of our system use deictic gestures (pointing, achieved by a head mounted laser pointer) to indicate locations, and spoken commands to identify objects and specific actions. The combination of spoken language along with deictic gestures performs a critical disambiguation function. It binds the spoken words in terms of nouns and actions to a locus in the physical workspace. The spoken input is used to supplant the need for a general purpose object recognition module in the system. Instead, 3-D shape information is augmented by the users spoken word which may also invoke the appropriate inheritance of object properties using the adopted hierarchical object-oriented representation scheme.

The use of multiple modes of control and command allows the user to operate the robot in a manner which more closely matches the user's needs. This multimodal approach is based on the assumption that the user's world is unstructured, but the properties and behaviors of objects within that world are reasonably predictable. Our work reflects this arrangement by providing a means of determining the three-dimensional shape and pose of objects and surfaces which are in the immediate environment, and an object-oriented knowledge base and planning system which superimposes information about common objects on the three-dimensional world.

We describe an architecture for task planning which incorporates a novel reactive planning mechanism where the user is an integral component of the planning mechanism. The planning mechanism is based on an object oriented knowledge base incorporating in it the relaxed assumptions about the world that are essential for the mechanism to be practical in the real world and facilitating human-computer interaction as a means of providing reactive and re-planning capabilities.
Reactivity is achieved in two ways. An autonomous runtime reactivity is obtained through sensor fusion. Sensory information from the vision system, force sensors, etc. will be used by the planner to obtain information for not only task planning but also to react to environment changes. Both sensing uncertainty and computational complexity prevents having a totally sensor based reactive

planning system, and hence user input is necessary for imparting the necessary reactivity.

Our hierarchical human-machine interface and object oriented representation allows the user to interact with the planning system at any level of the planning hierarchy, from low level motion and grasp planning to high-level task planning of complex tasks such as feeding. The generic plans and specialized plans are supplemented by user interaction whenever incomplete information precludes the development of correct plans by taking over control of the planning mechanism or providing information to the knowledge bases to facilitate the development of a plan capable of handling a new or uncertain situation. Furthermore, incomplete sensory information may be supplemented by user input, enabling the planner to develop plans from its plan library without the need for extensive user intervention.

Given this underlying architecture, the system first determines what the user wants, and then makes plans to accomplish the task. As a consequence of insufficient information, uncertainty, advent of new information, or failure of a plan, the system engages in a dialogue with the user which enables the planner to revise its plans and actions.

3.0 The Planner

3.1 The Architecture

The basic architecture in brief is composed of a knowledge base of two parts: A knowledge base of objects (Object Base) and a knowledge base of actions (Plan Base). In addition there is a world data base where the workspace information is stored. The planner uses the two knowledge bases and user /sensor provided feedback, to construct robot plans. For the whole system to work coherently we also require a domain theory that contains information regarding both temporal and spatial relationships between objects.

3.2 The Object Base

Objects are represented in an increasingly specialized sequence of objects in an inheritance hierarchy. At the top level, we start with a generic abstract object and at the bottom we end up with specific objects whose attributes are fully specified. From the abstract top level objects, we derive objects with intermediate levels of specializations; the choice of these intermediate classes of objects is dependent on the kind of general objects that the system might encounter and the set of tasks that the system might be called on to perform on these objects.

Each object, depending on the degree of generalization, has a set of attributes that will assist the planner in developing correct plans. An initial investigation into the kind of tasks the robot might be called on to undertake prompts us to visualize a set of attributes which include, shape, size, dimensions, weight, approach point, grasp points, constraints and plan fragments. The constraints and plan fragments attributes need to be described in a little more detail to explain the working of our model.

Constraints—Constraints may be placed on objects which further constrain approaching, grasping, and moving primitive actions. For example, we may place a constraint on a cup such that the cup is moved in a specific orientation in order to prevent spillage. These constraints are dependent on which action is being invoked upon the object. For example, in the case of the cup, the constraint about the fixed orientation must be over-ridden if the action involves pouring something out of the cup. Thus the representations of constraints in this field is further qualified by which actions these constraints are applicable to.

Plan Fragments—Another needed component would be plan fragments that are going to be incorporated into plans formed by the planner. Certain tasks may be specific to an object, and those plan fragments may be associated with the object in question in order to facilitate correct planning.

3.3 World Data Base

In addition to the knowledge base of objects, the system also maintains a data base of objects that it sees in the domain, called the *Domain Base*. The objects in the domain contain additional attributes which get instantiated after objects have been identified by the system. Currently, the attributes considered necessary are location and orientation, and attachments to other objects and the workspace.

3.4 Object hierarchy illustration

A very simple example of the object hierarchy is shown below. Prior to interaction with the user, the system sets up the world data-base as a collection of *blobs* of different sizes and shapes, with only the position with respect to the world origin being known. The *blob* world image is obtained from the vision system and size and location parameters are instantiated in the world data base from the information obtained by the vision system. We do not do any object recognition. Based on the premise that the user is in the loop, the user points to a *blob* and identifies it to the system. For example, she may point to a specific blob and inform the system that this is a *cup*. The system then updates the attribute slots of the *blob* with attributes that it obtains from the knowledge base. The user may also identify the blob as a specific object, such as *my-cup*; in such a case, the system is aware of a specific object in the knowledge base which is known as *my-cup* and the blob in its domain-base is replaced by the exact *my-cup* that the system knows, and the attributes of *my-cup* in the domain-base are set up from the knowledge base and information derived from the snapshot of the world. It is entirely possible that the user may not have identified any specific blob, and the system then is only aware of the general shape, and the blob is identified at a certain degree of generalization, such as *cylindrical* which is provided by the vision and shape fitting system.

3.5 Plan Base

The planner is based on a modified STRIPS-like planning mechanism [29, 30]. The main difference between conventional STRIPS-like planning and our proposed system is that we take full advantage of the underlying object oriented representation of the domain objects which drives the planning mechanism. Plans in this model are considered as general templates of actions, where plan parameters are instantiated from the object knowledge base during the planning process. For example, the constraint slot for a Move action might contain the slot Object-constraints. This implies that this slot parameter is going to be filled up from the constraint field of the object on which the action is being invoked. In the case of the cup example previously illustrated, the constraint that the cup must be maintained in a certain orientation is used to instantiate the constraint slot of the Move action. The constraints instantiated from the object in question are added to the set of constraints already present. Sometimes, some of the constraints obtained from the objects themselves may be in direct contradiction to constraints already present in the action being invoked. When that happens, the constraints obtained from the object override default constraints in the action body. All plan slots may be instantiated from information obtained from objects on which they are invoked in a similar manner.

3.5.1 Handling Exceptions

Another way in which the object oriented paradigm has extended the classical STRIPS planning mechanism is illustrated below. As mentioned previously, the body of an action may contain further subactions into which the actions may be decomposed. This facilitates hierarchical planning, one of the essential features of a planning system. Certain tasks which can be generally handled for most objects may not be applicable to certain objects in the real world.

Suppose we have an appliance that is used often in the domain of the user. The instrument has a peculiar shape and must be picked from a specific point. To approach the grasp-point, it may not be possible to just simply specify a certain approach point and assume that the robotic arm will then be able to pick up that appliance. The approach path may be convoluted and hence there must be some way to specify such an atypical case in our planning system. This is done by the use of the plan-fragment associated with an object. In a manner similar to the way action slots are filled depending on the object on which the actions are invoked, subtask slots are also filled, if so specified, from the object's plan-fragment.

Thus we see that this integration of knowledge base planning with an object oriented approach allows us to use general plans whenever we can. Additionally, this method will allow us to develop plans for specific objects peculiar to the domain without the need to perform computationally expensive operations. Moreover, each action has a generalized version and specialized versions that are invoked according to knowledge about the object. This allows us to abstract out the general features of an action and invoke them on objects about which the knowledge base might not have any information. It also allows us to view an action as a single action that is applicable to many kinds of objects instead of as a set of actions, each applicable to only one kind of objects as is done in other STRIPS-like systems.

3.6 A Simple illustration of the approach

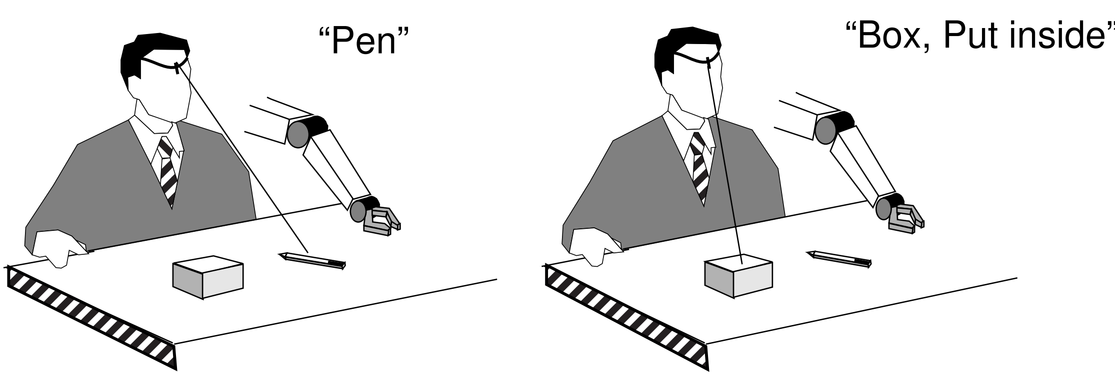


Figure 1. A Simple Illustration

The user approaches a table on which there are a *pen* and a *box*, both of which are in the knowledge base. The user points to the pen, and says, *pen*. From the knowledge base the system knows how to approach the bottle. The user points to the box and says *box*, *put inside*, indicating that the object is a box, and the final location of the pen is inside the box. Based on knowledge-base and sensory informations, the pen is moved to its desired destination.

4.0 Vision Processing

For our multimodal system, the vision requirement is to provide the knowledge-based planning system with the parameterized shape and pose information of the objects in the immediate environment. This information can then be used to fill slots in the object oriented representation and support both the system planning and simulation activities. The vision processing proceeds in three phases, extraction of highly precise 3-D point information using a calibrated line-based stereo matching algorithm, segmentation of the entire point sets into object-based sets, and non-linear minimization to fit the parameterized shapes to respective objects in the scene. A feature-based matching algorithm is practically suitable for this application. To reduce the false extraction rate a high intensity structured-light source with non-parallel stripes is employed in this design. The distorted light patterns in the images can be easily extracted and processed. To recover the 3D contour of the objects the vision system needs to find the correspondence of the distorted patterns in two images. In this paper the straight line pattern is selected as it naturally incorporates the figural continuity constraint. A line-segment pair-match scheme is developed based on the geometric characteristics of the features obtained from the images.

Images are taken by two CCD cameras through SGI's Galileo Video board and sent to Noesis' VISILOG image processing system run on the SGI Indigo XS24. The light source is generated by a slide projector in a form of light-stripes or grid. The existing stereo vision techniques are classified into several categories: full scale nonlinear optimization method, two plane method, and linear least-squares method [19]. The linear least-squares method is adopted in this project.

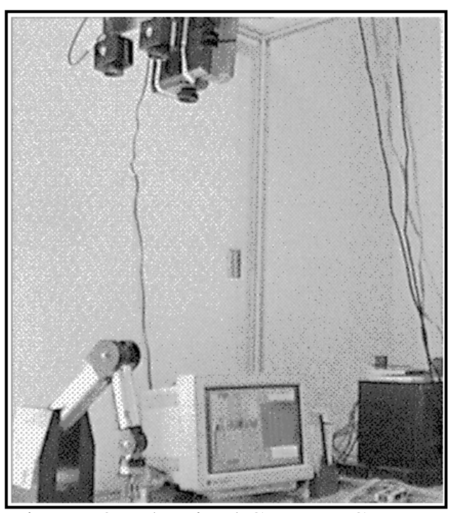


Figure 2: Physical System Setup

4.1 Calibration

Before the vision system can be used to extract points from the stereo image pairs a precise calibration must be achieved to ensure that the disparity measurements resulting from the edge matching process can be triangulated to yield the true three-dimensional depth.

It is convenient to use a two-coordinate system to describe the geometrical relationship between a three-dimensional object point and its projected image point. The mapping from the global coordinates of the k^{th} point, $\boldsymbol{u}^k = [u_X^k, u_Y^k, u_Z^k, 1]$, an augmented vector in 3D space, to the perspective coordinates of the n^{th} camera, ${}^k\underset{\sim}{v}^n = \left[{}^k\underset{\sim}{v}_x^n \; {}^k\underset{\sim}{v}_y^n \; {}^k s^n\right]$ can be implemented through the perspective transformation

$$ {}^k\underset{\sim}{\boldsymbol{v}}^n = \boldsymbol{A}^n \boldsymbol{u}^k \tag{1} $$

where $k \in Z = [1, 2, \ldots, K]$; $\boldsymbol{A}^n \in \boldsymbol{R}^{3 \times 4}$ is the perspective transformation matrix for the n^{th} camera and $n = [1, 2, \ldots, N]$. The image coordinates ${}^k\mathrm{v}^n = \left[{}^k\mathrm{v}_x^n \; {}^k\mathrm{v}_y^n\right]$ can be computed by ${}^k\mathrm{v}_x^n = {}^k\underset{\sim}{\mathrm{v}}_x^n / {}^k s^n$ and ${}^k\mathrm{v}_y^n = {}^k\underset{\sim}{\mathrm{v}}_y^n / {}^k s^n$. The unknown parameters A_{ij}^n, ($i = 1, 2, 3$; $j = 1, 2, 3, 4$), can be estimated by using least-squares optimization method with $K > 6$ non-coplanar locations, further details are given in [Chen94]. Once the perspective transformation matrices are available, the computation of the 3D location is straightforward as long as the corresponding image vectors $v^n, n \in Z = [1, 2, \ldots, N]$ are found for the same object vector u in the world space. The next section addresses the issue of matching corresponding vectors in different images.

4.2 Line-segment pair matching process

The objective of stereo vision is to recover 3D information about the object in the work environment using images taken from different viewpoints. The most essential and most difficult procedures in the stereo vision is image-matching. The purpose of the match process is to find the correspondence among the features extracted from two images. The difficulty of image match problem stems from the facts such as image variations due to different perspective projections and the source of lighting *etc*. Researchers in this field have been developing various algorithms in the past two decades. Basically, these algorithms can be classified into two major categories: area-based (intensity level as the feature) and feature-based (semantic features with specific spatial geometry) techniques [2]. Early representative works can be found in [3,26,28] in which the relaxation labeling technique is used to solve the stereo image matching problem. More recent developments incorporating structural information between image entities in addition to entity properties solve the correspondence problem [6, 21, 25]. It should be noted that there is no unified approach to the stereo correspondence problem; it is very much application dependent.

4.3 Line segment pair matching scheme

The *epipolar constraints* and the *x-disparity constraints* can be used to develop a simple scheme for line segment pair matching in two images. First, denote the line segment sets in two images by $\boldsymbol{L}^l = \{L_s^l\}$ and $\boldsymbol{L}^r = \{L_t^r\}$ where $s = [1, 2, \ldots, S]$ and $t = [1, 2, \ldots, T]$ are the number of line segments in the left and right image. The pair search process takes place between $\{L_s^l\}, \forall s$ and $\{L_t^r\}, \forall t$. There is total $S \times T$ pairs under comparison.

For each potential match pair three penalty measures are defined. The x_position_penalty is defined as $P_x(s, t) = \left|{}^s v_x^l - {}^t v_x^r\right|, \forall s, \forall t$ where ${}^s v_x^l$ is the lower end x coordinate of the s^{th} line segment in the left image and ${}^t v_x^r$ is the lower end x coordinate of the t^{th} line segment in the right image. The x_position_penalty is normalized such that $P_x(s,t)_{max} = 1$.The y_position_penalty is defined analogously as $P_y(s,t) = \left| {}^s v_y^l - {}^t v_y^r \right|, \forall s, \forall t$ where ${}^s v_y^l$ is the higher end y coordinate of the s^{th} line segment in the left image and ${}^t v_y^r$ is the higher end y coordinate of the t^{th} line segment in the right image. The y_position_penalty is also normalized such that $P_y(s,t)_{max} = 1$. Finally, the length_penalty is defined as $P_\rho(s,t) = \left| {}^s\rho^l - {}^t\rho^r \right|, \forall s, \forall t$ where ${}^s\rho^l$ is the length of the s^{th} line segment in the left image and ${}^t\rho^r$ is the length of the t^{th} line segment in the right image. Again, the length_penalty is normalized such that $P_\rho(s,t)_{max} = 1$. Finally, a total_penalty function for each pair is defined as

$$P(s,t) = \alpha P_x(s,t) + \beta P_y(s,t) + \gamma P_\rho(s,t), \forall s, \forall t. \quad (2)$$

The weights, α, β, γ are selected according to the importance of each penalty function in the search process. Unfortunately, there is no general rule for picking up the values for these coefficients. (In the experiment of this study $\alpha = 0.3$, $\beta = 0.3$, and $\gamma = 0.4$)
In addition to three penalty measures defined above, a slope measure for each line segment is also introduced. A line segment can be represented by the model

$$L = v_x(v_y; \sigma, \lambda) = \sigma + \lambda v_y \quad (3)$$

The parameters σ and λ are estimated through the linear regression test. In this paper, only λ is used for line-pair search. The value of λ can be positive, negative or zero depending on the original non-parallel stripes designed. This combined penalty function along with the line orientation constraint for disambiguation is then used by a fast and robust iterative search technique to match the line segments by minimization. Details of the search process can be found in [8].

4.4 Experimental result

In order to simplify the pair match process the light stripes are arranged such that there is enough distance between any two lines in the images. The experiment is designed to obtain the height information of two objects in the world space. Figure 3 shows one of the images captured. The lines are first processed so that noises are removed and the width of each line is eroded to one pixel. The resultant is then fed to the line segment pair search process described previously.

Figure 3: Captured Image

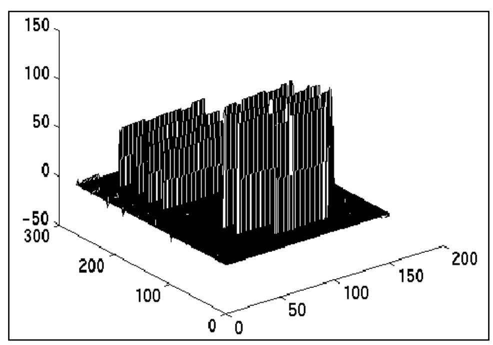


Figure 4: Recovered Shapes

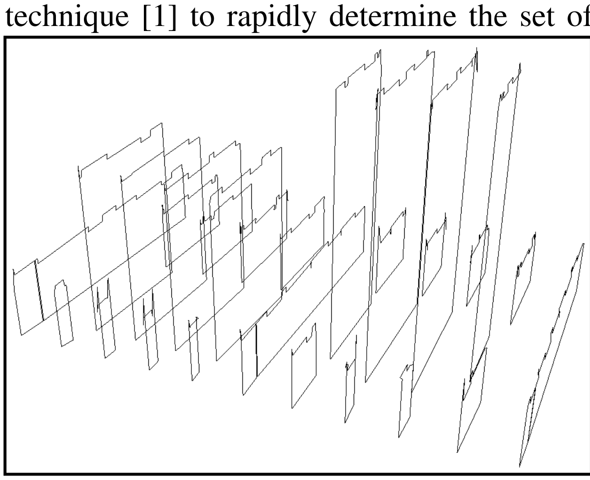
There are 28 line segments in each image. After the search process the final results shows a 100% match for this experiment. By applying the least-squares 3D restoration method to the line pairs the recovered shapes are shown in Figure 4. In general, the method performs quite well.

4.5 Segmentation and Shape Fitting

The purpose of the shape extraction system is to provide a mechanism for deriving a set of shapes from a large number of point-wise measurements on the surfaces of the different objects in the scene derived by the stereo matching algorithm. Numerous representations are currently used for shape representations in both the CAD and vision communities, such as spline surfaces, generalized cones and superquadrics. Superquadrics are a superset of the class of ellipsoids which can represent and approximate many shapes from spheres to cubes and cylinders that occur in man-made environments (in fact, superquadrics were originated by the Danish Designer Peit Hein.) [38] Superquadrics provide two major advantages: a well developed mathematical foundation for their recovery from sets of range points [1], and a concise shape description appropriate for planning, graphical display, and manipulation activities that occur in planner and graphical simulation world. For example, descriptions of objects in the planner's representation are in terms of shape primitives such as cylinders at given x-y-z locations in the environments and with given dimension and the graphical environment can generate polygonal mesh approximations from such shape descriptions as well.

The shape extraction process consists of thresholding, segmentation and shape fitting of each respective point group. Since the height of the surface of support of the objects can be known *a-priori*, a threshold height may be set for the purpose of foreground background segmentation. Once the thresholding is complete, a point-set clustering is performed to the single set of point that have been labelled as foreground points since there may be multiple objects in the scene. A nearest-neighbor metric is used to bottom up cluster the point-set into subsets of connected-components according to a scaled Euclidean distance metric. The scaling allows for selectable merging distance thresholds in each of the orthogonal directions. Each resulting connected component point-subset then corresponds to an object in the scene.

Once the individual point sets have been clustered, then the shape fitting process may be run on each individual point-set. The shape fitting process computes the shape parameters which control the shape and size and location of each superquadric shape. We use a non-linear minimization technique [1] to rapidly determine the set of shape parameters that best-fit the raw 3-D points
ibe th
ch tha
entati
ation



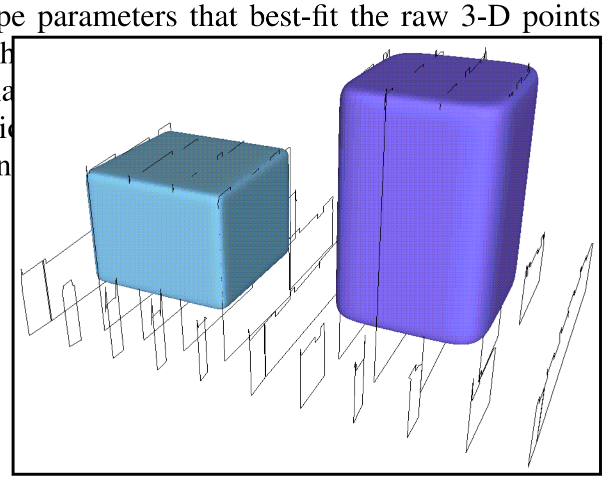


Figure 5: Point Segmentation and Resulting Object Shape Fitting

Figure 6: Recovered 3-D Points from Structured Light Stereo Line Matching

5.0 A Simulation Environment

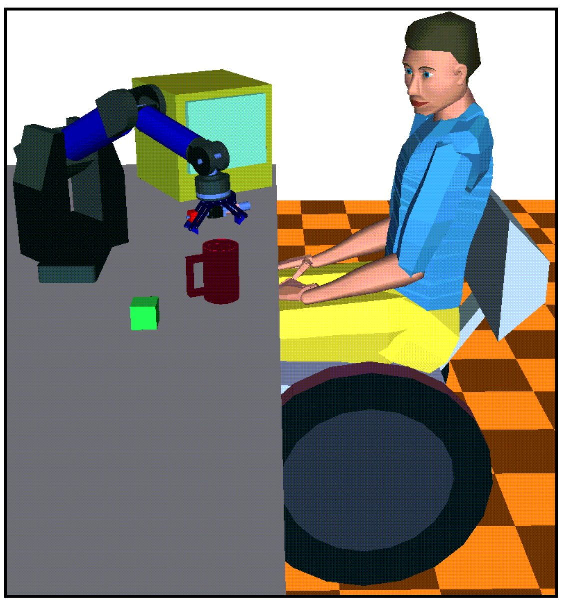


Figure 7: The Simulated Multimodal User Control Environment

While the combination of different modalities makes our interface method more adaptable and natural than other telemanipulation methods, understanding the interaction and the embodied meaning of the numerous modal inputs is a challenging and incomplete research area. We are developing a simulation environment (Figure 6) that will allow us to investigate the modalities of the human-computer interaction in a low risk fashion. Designing a multimodal control system which can properly respond to directives from a user depends heavily on understanding the user's perception of the depth, distance, orientation and configuration of objects in their operating domain. It is important to note that our system does not require the user to provide information about the depth, distance, orientation and configuration of objects, but a mutual understanding of the user's perception of these features, and the machine vision, perception and planning systems intentions is necessary to insure that tasks are carried out as the user intended. An important objective of the simulated multimodal environment is to allow our research team to rapidly implement and experiment with different methods of interpreting the discourse and gesture information. The simulated environment will also allow us to experiment with different techniques for combining the results of each analysis to extract their joint meaning. The facets of multimodal control which we hope to better understand through the simulated multimodal environment are:

- User perception of object location and orientation
- User methods of interacting with objects and the robot
- Proper interpretation of the user's speech and gestural inputs
- Feedback about multimodal control system's interpretation of the user's intentions
- Determine a level of automation which allows the user the best control and flexibility
- User's insights into the system's possibly incomplete or erroneous scene representation
- Plan preview and error replay

An additional issue which the simulated environment will help to address is user safety. When the user commands the multimodal control system they are expecting the system to complete that task without injuring them or damaging the objects it is manipulating. To provide feedback to the user about the plans of the system, the simulated environment will be incorporated into the multimodal control system during its actual operation. The simulated environment will inform the user of the system's plans and interpretations of the world by showing them a preview of what the system intends to do. Using the simulated environment to show a preview is very important because when the user entrusts the multimodal control system with a task, they are "trusting" that the task will be performed correctly. Every time the user issues a command they are also taking a "risk" that the system can do the job correctly [12]. Providing the user with a visual preview of the intention of the multimodal control system will effectively strengthen the "trust" between the user and the multimodal control system.

6.0 Conclusion

As mentioned in the introduction, human intervention as well as an intelligent planning mechanism are essential features of a telerobotic assistive system. We have described a new model of robot planning system that will be practical in the real world where we need to relax some strong assumptions about the domain made in classical planning systems. This is achieved by integrating a novel gesture-speech driven human interface to a reactive planning mechanism. The desired features of the planning mechanism are further enhanced by multimodal-fusion and by an underlying planning mechanism that is based on integrating knowledge based planning with the object oriented programming paradigm.

We believe that this novel approach of gesture-speech based human-machine interfacing enables our system to make realistic plans in a domain where we have to deal with incomplete knowledge and uncertain situations. The flexibility of our system to work in real-world environments imparts to our system both effectiveness and operability, which were identified by a panel of experts with mobility-related disabilities as the top two desired criteria for an assistive device [4].

Acknowledgments

Work on this project is supported by the Rehabilitation Engineering Research Center on Rehabilitation Robotics, National Institute on Disabilities and Rehabilitation Research Grant #H133E30013, Rehabilitation Services Administration Grant #H129E20006 and Nemours Research Programs.

References

[1] Bajcsy, R., & Solina, F. (1987). Three Dimensional Object Representation Revisited. In?, 231-240

[2] Barnard DT, Fischler MA. Computational stereo. *ACM Computing Survey* 1982;14(4):553–572.

[3] Barnard ST, Thompson WB. Disparity analysis of images. *IEEE Trans. on Pattern Anal. Mach. Intell.* 1980;2(4):333–340.

[4] Batavia, A. & Hammer, G. (1989). Consumer Criteria for Evaluating Assistive Devices: Implications for Technology Transfer. In *Proceedings of the 12th annual Conference on Rehabilitation technology* (pp. 194–195). Washington: RESNA Press.

[5] Bolt, R. A. (1980). "Put-That-There": Voice and Gesture at the Graphics Interface. *Computer Graphics*, 14(3), 262–270.

[6] Boyer KL, Kak AC. Structural stereopsis for 3d vision. *IEEE Trans. on Pattern Anal. Mach. Intell.* 1988;10(2):144–166.

[7] Cannon, D. (1992). Point and Direct Telerobotics: Object Level Strategic Supervisory Control in *Unstructured Interactive Human-Machine System*. Unpublished Doctoral Dissertation, Stanford University, Department of Mechanical Engineering.

[8] Chen S, Kazi Z, Foulds R, Chester D. (1994), Multimodal direction of a robot by individuals with a significant disability. In: *Proceedings of the 4th International Conference on Rehabilitation Robotics*. Wilmington, DE, USA.

[9] Cipolla, R., Okamoto Y., & Kuno Y. (1992), Qualitative visual interpretation of hand gestures using motion parallax. Proc. *IAPR Workshop on Machine Vision Applications*, (pp 477-482).

[10] Crangle, C., Liang, L., Suppes, P., & Barlow, M. (1988). Using English to Instruct a Robotic Aid: An Experiment in an Office-Like Environment. In *Proc. of the International Conf. of the Association for the Advancement of Rehabilitation Technology* (pp. 466–467). Montreal

[11] Crangle, C., & Suppes, P. (1994). Language and Learning for Robots. CSLI Publications, Stanford, Ca

[12] Foner, L. N., (1993). What's an Agent Anyway? A Sociological Case Study. E15-305, MIT Media Laboratory, 20 Ames Street, Cambridge, MA 02139

[13] Foulds, R. A. (1986). *Interactive Robotics Aids–One Option for Independent Living: an International Perspective*, volume Mono graph 37. World Rehabilitation Fund.

[14] Fu, C. (1986). An Independent Vocational Workstation for a Quadriplegic. In R. Foulds (Ed.), *Interactive Robotic Aids-One Option for Independent Living: An International Perspective*, volume Monograph 37 (pp. 42). World Rehabilitation Fund.

[15] Fukimoto, M., Mase, K., & Suenga, Y. (1992. Realtime detection of pointing action of a glove free interface. Proc. IAPR Workshop on Machine Vision Applications, (pp 473-476).

[16] Gilbert, M. & Foulds, R. A. (1987). Robotics at the Tuffs New England Medical Center. In *Proceedings of*

the 10th Annual Conference on Rehabilitation Engineering (pp. 778–780). San Jose.

[17] Gilbert, M. & Trefsger, J. (1990). In *Proceedings of the 1990 International Conference on Rehabilitation Robotics* Wilmington, DE.

[18] Hammel, J., Van der loose, M., & Perkash, I. (1991). Evaluation of DeVAR-IV with a Quadriplegic Employee. In *Annual Report of the Rehabilitation Research and Development Center* (pp. 99–100). Palo Alto, CA: Palo Alto VA Medical Center.

[19] Hall EL, Tio J. Measuring curved surfaces for robot vision. *Computer* December 1982;42–45.

[20] Harwin, W., Ginige, A., & Jackson, R. (1986). A Potential Application in Early Education and A Possible Role for a Vision System in A Workstation Based Robotics Aid for Physically Disabled Persons. In R. Foulds (Ed.), *Interactive robotic aids-one option for independent living: An international perspective*, volume Monograph 37 (pp. 18–23). World Rehabilitation Fund.

[21] Horaud R, Skordas T. Stereo correspondence through feature grouping and maximal cliques. *IEEE Trans. on Pattern anal. Mach. Intell.* 1989;11(11):1168–1180.

[22] Kak, A. C., Boyer, K. L., Chern, C. H., Safranek, R. J., & Yang, H. S. (1986). A Knowledge-Based Robotics Assembly Cell. *Proc. of IEEE Int'l Conf. on Robotics and Automation*.

[23] Kwee, H. (1986). Spartacus and Manus: Telethesis Developments in France and the Netherlands. In R. Foulds (Ed.), *Interactive Robotic Aids-One Option for Independent Living: An International Perspective*, volume Monograph 37 (pp. 7–17). World Rehabilitation Fund.

[24] Kwee, H., Thonninsen, M., Cremers, G., Duimel, J., & Westgeest, R. (1992). Configuring the Manus System. In *Proc. of RESNA International'92* (pp. 584–587). Washington, DC: RESNA Press.

[25] Matsuyama T, Arita H, Nagao M. Structural matching of line drawing using the geometric relationship between line segments. *Computer Vision Image Proc.* 1984;27:177–194.

[26] Medioni G, Nevatia R. Matching images using linear features. *IEEE Trans. on Pattern Analy. Mach. Intell.* 1984;6(6):675–685.

[27] Michalowski, S., Crangle, C., & Liang, L. (1987). Experimental Study of a Natural Language Interface to an Instructible Robotic Aid for the Severely Disabled. In *Proc. of the 10th Annual Conf. on Rehabilitation Technology* (pp. 466–467). Washington, DC: RESNA Press.

[28] Price K. Hierarchical matching using relaxation. *Computer Vision Graphics Image Proc.* 1986;34:66–75.

[29] Sacerdoti, E. D. (1975), The Non-Linear Nature of Plans, In *Proceedings of IJCAI-75*.

[30] Sacerdoti, E. D. (1977), A Structure for Plans and Behavior, American Elsevier, NY

[31] Sheridan, T. B. (1992), Telerobotics, Automation, and Human Supervisory Control. The MIT Press, Cambridge, MA

[32] APL/JHU robotic arm workstation. In R. Foulds (Ed.), *Interactive Robotic Aids-One Option for Independent Living: An International Perspective*, volume Monograph 37 (pp. 51). World Rehabilitation Fund.

[33] Van der loos, M., Hammel, J., Lees, D., Chang, D., Perkash, I., & Leifer, L. (1990). Voice Controlled Robot Systems as a Quadriplegic Programmer's Assistant. In *Proceedings of the 13th Annual RESNA Conf.* Washington, DC.

[34] Van der loos, M., Hammel, J., Lees, D., Chang, D., & Schwant, D. (1991). Design of a Vocational Assistant Robot Workstation. In *Annual Report of the Rehabilitation Research and Development Center* (pp. 97–98).

Palo Alto, CA: Palo Alto VA Medical Center.

[35] Wiemer, D., & Ganapathy, S. G. (1989), A synthetic visual environment with hand gesturing and voice input. Proc CHI'89, (pp 235-240).

[36] Zeelenberg, A. P. (1986). Domestic Use of a Training Robot-Manipulator by Children with Muscular Dystrophy. In R. Foulds (Ed.), *Interactive Robotic Aids-One Option for Independent Living: An International Perspective*, volume Monograph 37 (pp. 29–33). World Rehabilitation Fund.

[37] Pook, P. (1994), Teleassistance: Contextual guidance for autonomous manipulation, *Proceedings of the National Conference on Artificial Intelligence,* v 2 1994. AAAI, Menlo Park, CA, USA. (pp 1291-1296.)

[38] Barr, A. (1981). Superquadrics and angle-preserving transformations. *IEEE Computer Graphics and Applications,* 1(1), (pp. 11-23).

[39] Funda, J., Lindsay, T.S., Paul, R. P., (1992, Teleprogramming: Towards delay-invariant remote manipulation. *Presence*, 1(1), (pp 29-44).