Robot Partners:
Collaborative Perceptual Robotic Systems

Working paper

Cooperative Distributed Vision Project Workshop

James J. Little

Laboratory for Computational Intelligence
Department of Computer Science
University of British Columbia
Vancouver, British Columbia, CANADA V6T 1Z4
little@cs.ubc.ca 604-822-4830 604-822-5485 FAX

Abstract

Over time we have developed and used a variety of real-time visually guided robotic systems, including remote-brained soccer players, visually guided mobile robots, and visual tracking systems. As part of this history, we have moved from specialized high-performance systems, to a current system using standard processors, but maintaining a similar level of performance.

Our work has been supported by the the Networks of Centres of Excellence Institute for Robotics and Intelligent Systems (IRIS 2) in the Integrated Systems Theme, as Project IS-6, Constraint-Based Visual Robotic Systems.

The next stage of our activity involves collaboration between robot and human, in teleoperation and telepresence, as well as collaboration between robots. This necessitates communication among robots, as well as between robots and humans.

The Robot Partners project is a proposal to the next phase of the IRIS project, IRIS 3. We first review it, then its current instantiation, Constraint-Based Visual Robotic systems.[1]

We review our current project and the series of systems it has realized, and then describe the next generation of robots and their intended use as robot partners. As a first step toward scene understanding systems, we propose to build geometric scene models that will facilitate operation in dynamic unstructured environments and analyze them to provide reference systems for communication and action. By examining a series of scenarios using our mobile robots and eye-head, we can see the directions our research must take to enable them to see and act cooperatively.

1 Introduction

Increasingly machines play the role of partners with humans in activities ranging from information processing to communication, teleoperation, and robot control. Such partnerships include information agents, telerobotic systems which have manipulators and sensors, and team activities such as scouting, surveying, and assembly. Collaborative robots have applications in medical robotics, teleconferencing, entertainment, construction, manufacturing, and resource industries. The design and implementation of perceptual collaborative robotic systems is the focus of the Robot Partners project, which includes David

[1] This research was supported by the Natural Sciences and Engineering Research Council of Canada and the Networks of Centres of Excellence Institute for Robotics and Intelligent Systems, Project IS-6.

Lowe, Alan Mackworth, Dinesh Pai, Jim Little, (all of UBC), and James Clark from McGill University. The project is a proposal for the IRIS 3 Network of Centres of Excellence in Canada.

Our challenge is to design and implement collaborative robotic systems. A robotic system consists of a robot plus its environment. A robot is multi-level, with descriptions at a variety of levels. The environment may consist of physical objects, processes, and other agents, either people or robots. A collaborative robotic system is one in which a robot works to achieve common goals with at least one other agent. Collaboration requires shared goals, communication about the state of local internal and external environments, and coordinated action to achieve the shared goals. Telerobotics is an instance of communication of requirements or goals; it is a remote collaboration, where one partner has more complete access to the environment. It demonstrates one of the key advantages of collaboration: each agent contributes its niche strength to the team.

The state of the art in building collaborative systems is at an early stage. We need tools to program collections of agents, construct, analyze and verify controllers, interact with sensors, organize and model collaboration, and reason about the system.

The practical applications of this technology include cheaper, more reliable, more intelligent, more flexible and more aware robots for tasks such as surveillance, cleaning, delivery, assembly and teleoperation. Unsupervised, mobile robots can be used in surveying, mining (carrying ore along tunnels), logging (silviculture, tree harvesting), surveillance and inspection, cleaning, delivery and materials handling (hospital meals, warehousing and inventory control), assembly (flexible assembly cells with multiple loosely coordinated activities, construction), and office assistants. Using teleoperation, combining supervision with flexible autonomous agency guided by vision, it can be used mining, retrieving rock from dangerous locations, and hazardous materials handling.

Robots can be also be seen as the latest development in high-bandwidth computer I/O peripherals that require new design methods for reliable, embedded, perceptually-aware, hybrid systems.

2 Objectives of the Project

The Robot Partners project will develop a new approach to the specification, design and implementation of collaborative robotic systems, which are a class of hybrid systems. Using collaborative activity as the target application will lead to a new framework for constraint-based design of embedded intelligent systems.

We will develop techniques to understand and integrate the effects of control actions, including exogenous events, to characterize system behaviours in terms of primitives, to reason about continuous time, concurrency, interrupt mechanisms for responding to exogenous events attention, resource allocation, and coordination among concurrent activities.

The intelligent tools forthcoming will include:

- synthesis of multi-robot distributed controllers
- real-time systems for vision, with multiple coordinated sensors
- a programming system for control with event-based interaction with perception
- teleoperation based on both specific and generic local models acquired online during teleoperation, and utilized during teleoperation
- models of collaboration
- means for human visual communication with robots

We will build systems to construct useful models of the world being perceived by the robot. Using these models, we will simplify the human interfaces to the world model so that an untrained user can specify actions at a higher level, interacting with the model, without having to provide detailed control. This will enable the robot to do standard tasks autonomously and to enforce safety.

Among the systems and techniques we hope to realize are:

- Develop a real-time vision system to determine robot motion by tracking features in the environment.
- Develop integrated control and perception components for sensing in an agent.
- Show simple real-time gestural control
- Coordinate tracking and navigation through real-time control
- Complete development of a computer vision system for telerobotics that will allow recognition and tracking of general objects.
- Integrate vision system with telerobotic system.
- Develop partially autonomous embedded systems, with complex sensors.
- Extend design theory for multi-robot teams with communication.
- Demonstrate robot teams collaborating in complex tasks, with human interaction, supervision, intervention and guidance.
- Direct team of robots through visual control

2.1 Approaches to be Used

We plan to design and implement collaborative perceptual systems, including facilities for control, communication, and coordination. We will develop design principles, models of collaboration, constraint-based tools, real-time sensors linked with control programs, and visual communication.

Collaborative Telerobotics Our work on model-based telerobotics consists of manipulation of simple blocks with vision for object recognition. This system was successfully demonstrated at the 1996 IRIS/PRECARN meeting in Montreal, with the operator site located there, the remote site in Vancouver, and communication over the Internet. Conference attendees were able to manipulate the blocks with no prior training. Local models of the scene are used to recognized the scene objects: their pose is transmitted to the remote user who manipulates a virtual model—the manipulation generates an assembly plan that is then carried out back at the robot. We plan to extend this work to make telerobotic systems true collaborative systems.

Enhanced human-machine collaboration in teleoperation will require improved capabilities for object recognition and tracking, and new methods for acquiring object models from multiple images. These models of object appearance can be automatically formed from matching key features in multiple images. New methods for indexing and object matching will be developed to make the object recognition more accurate and robust.

Teams of Autonomous Robots A second visually-guided mobile robot, José, will collaborate with our stereoscopically-guided robot, Spinoza, both directly and indirectly over long distance via radio modems. They will jointly observe the environment and manage a common model of it for partially autonomous activity. Operating in unstructured environments requires comprehensive real-time sensing, with which the perceptual system will build its online model of the environment. The system will integrate information over time and from multiple views, and maintain its location in the model.

Our goal is to have continuously operating robotic systems whose components can be replaced during operation. The Java language will permit us to send new methods to an interpreter during operation and update the controller. As the cost of computers plummets and their capabilities soar, it will be possible to equip robotic agents with multiple sensors that can complement and support each other. To support tight integration of control and perception, we will develop event-based component systems: perception services both tightly and loosely coupled with robot controllers. We will extend existing control languages by developing vision services that can be handled seamlessly as part of the control activity, and coordinated to monitor and respond to relevant events.

Visual Communication for Collaboration Communication from a supervisor can be achieved by gestural input, such as hand or other body motions. These commands can interrupt actions of the robot, and redirect it to new locations or new actions, allowing both human and machine access to shared visual space. Our simple system uses colour tracking to identify human skin and then processes hand gestures by geometric analysis. We will use motion cues to identify the supervisor and segment the gesture, tracking with an active camera on the robot to keep the supervisor, in view. Models both of the environment and supervisory actions will make tracking robust, and enable a rich vocabulary of interactions.

3 History

The Constraint-based Visual Robotic Systems project, under the Integrated Systems theme in the IRIS 2 program, studies the development of algorithms, design methods, and technologies for real-time robotics in dynamic environments. It includes Jim Little, David Lowe, Alan Mackworth, Dinesh Pai, and Bob Woodham. The technical staff are Rod Barman and Stewart Kingdon. The graduate students involved are Cullen Jennings, Don Murray, Jochen Lang, Ping Shi, and Vladimir Tucakov. John Lloyd and Jiping Lu are postdoctoral fellows associated with the project.

Balancing the real-time constraints of a robot in a dynamic environment challenges the limits of both technology and our scientific understanding of embedded systems. Dynamic environments mean unpredictable, asynchronous events, requiring low latency in response, while processing visual information means high data-rate communications and significant computation.

The scientific goals of the project include determining the power of vision as a sensing mode for a robot, finding how to integrate ongoing sensing, identifying how much knowledge and reasoning is needed for action with plans: in general to delve into the interaction of processing/perception/action in a dynamic environment.

Other goals include learning about capabilities of situated/embedded vision, integration of technology, exploration of dynamic environments, development of a flexible computing environment that can handle multiple visual tasks, processing modes and multiple cameras, and realization of visually guided robots to operate outdoors.

As a platform for research, the robot provides an environment in which autonomous activities, with reactive behaviours, can be programmed in the lower levels, and control information can direct, as from humans, the goals and pursuits of the lowest levels.

3.1 Tasks for mobile robots

At the highest level, the robot's tasks include mapping, navigation, exploration, tracking, and manipulation. More concretely, we expect the robot, either under program control, or full or limited teleoperation, to act as a remote physical agent, to perform a range of tasks, including finding lab members, identifying whether equipment is busy, to check the status of the lab, to guide tours, to find things, to fetch coffee. Its range will be the entire building, and perhaps beyond. Untethered operation is thus critical.

From these tasks, we can identify a set of capabilities to demonstrate complex visual processing for control of a mobile robot. Its vision requirements include:

- blob detection—isolation of coloured objects for recognition and tracking
- stereo—passive distance estimation
- pointable camera—for tracking
- optical flow—computing perceived motion for segmentation, recognition, and obstacle detection
- tracking—visual servoing
- integration of multiple modes—cooperation among sensing systems (cameras) and vision processes

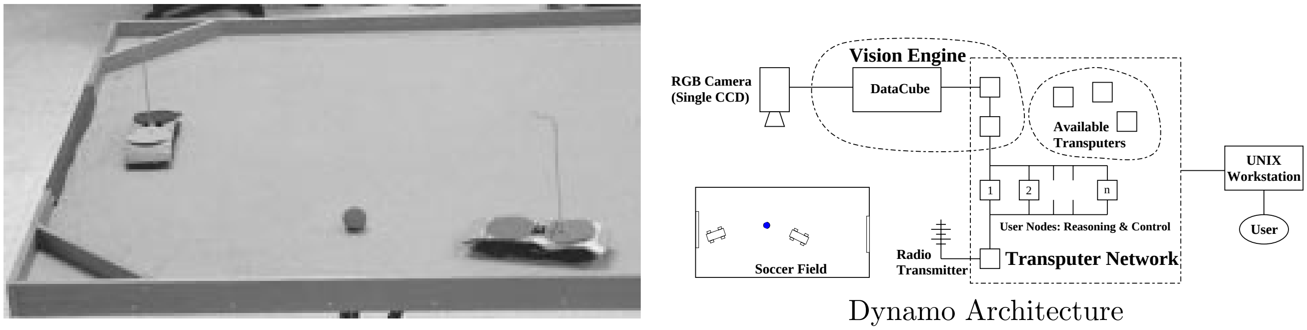
Intended extensions of its capabilities include addition of a manipulator, such as an extendable finger or hand. Previous versions have included speech output to communicate the state of the robot, such as "I'm trying to find the ball!". Current telerobot inputs are clicks and pushes of buttons on various interfaces, but gesture and speech are among planned inputs.

How do we realize a system that meets our constraints and has the capabilities we need? One method of specifying robotic systems has been the "reactive" situated approach that exploits regularities of the task and environment of the robot[HB88, And88]. Typically these systems have simplified the sensing capabilities of the robot so as to meet the physical and cost limitations, suited to the task and environment. Horswill[HY94] has implemented a more general, but inexpensive processor, with limited capabilities. Others[DFR96] move much of the signal and image processing offboard.

Over the years we have realized a series of perceptual robotic systems. The technologies have moved in several directions: from offboard computation to onboard, from specialized architectures to generic processors, and from niche applications to general tasks. The remainder of this sections describes some of the history and the current systems.

3.2 Dynamo project

The Dynamo testbed is a collection of independently controlled mobile robot vehicles that play soccer [BKL+93]. The system demonstrates offboard vision processing and distributed processing. The vision component was originally prototyped on the Datacube MaxVideo200 in the UBC Vision Engine. Currently the system is realized as a simple custom hardware to process RGB signals, followed by run-length encoding and centroid calculation on Transputers. A single off-board camera sensor (the "eye in the sky")



Dynamites

Dynamo Architecture

Figure 1: Spinoza: the stereoscopic vision robot

communicates its signals to the centralized sensor processor. The sensor process provides positional information to the control processes for each competing soccer player at 60Hz (once per image field) with a lag of at most 5 ms after the end of field. The structure of the full system is shown in Figure 1.

The Dynamo system has been used to explore novel reactive strategies for control[Sah94] as well as the testbed for ideas on control, specification, and reasoning about real-time systems[ZM92]. The "remote brain" idea, offboard visual processing, was used for soccer players because of size/weight limitations, and for our initial work with Spinoza: ROLL, (Real-time Onboard Localization with Landmarks) identifies its position in real-time, using passive visual localization of a single landmark[Lit94].

The UBC Vision Engine[LBKL91] exemplifies distributed, heterogeneous systems that have multi-rate processes. The Engine is general-purpose and supports all levels of vision. The system consists of multiple architectures: pipelined (a Datacube MaxVideo200) image processor and a MIMD multicomputer (20 T800 2MB Transputers, connected via a crossbar), connected by a bidirectional video-rate interface. The Transputer system controls an eye/head platform for vergence, pan and tilt.

3.3 Motion tracker

Our motion tracker[LK93] has several processes running at different rates in near real-time. The tracker follows a moving object, with no knowledge of its target, based on dense optical flow input.

The tracker identifies the largest moving object as the target. Since the tracker gathers visual data during head movements and identifies the target solely on motion and stereo cues, it must compensate for the apparent motion caused by head movement. It estimates the image motion caused by camera motion and *cancels* the apparent motion in the accumulated optical flow. It is then simple to find the target since the moving object "pops out" relative to a static background.

The tracker has multiple stages: correlation motion on the Datacube, optical flow accumulation, target selection and eye-head control. All but the first stage are implemented on Transputers. A *smart buffer* process[LK93] accepts the optical flow data stream, computed on the Datacube using the

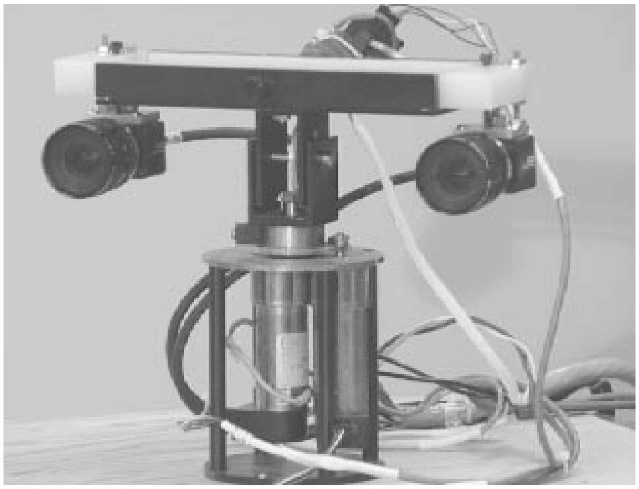


Figure 2: Eye-head

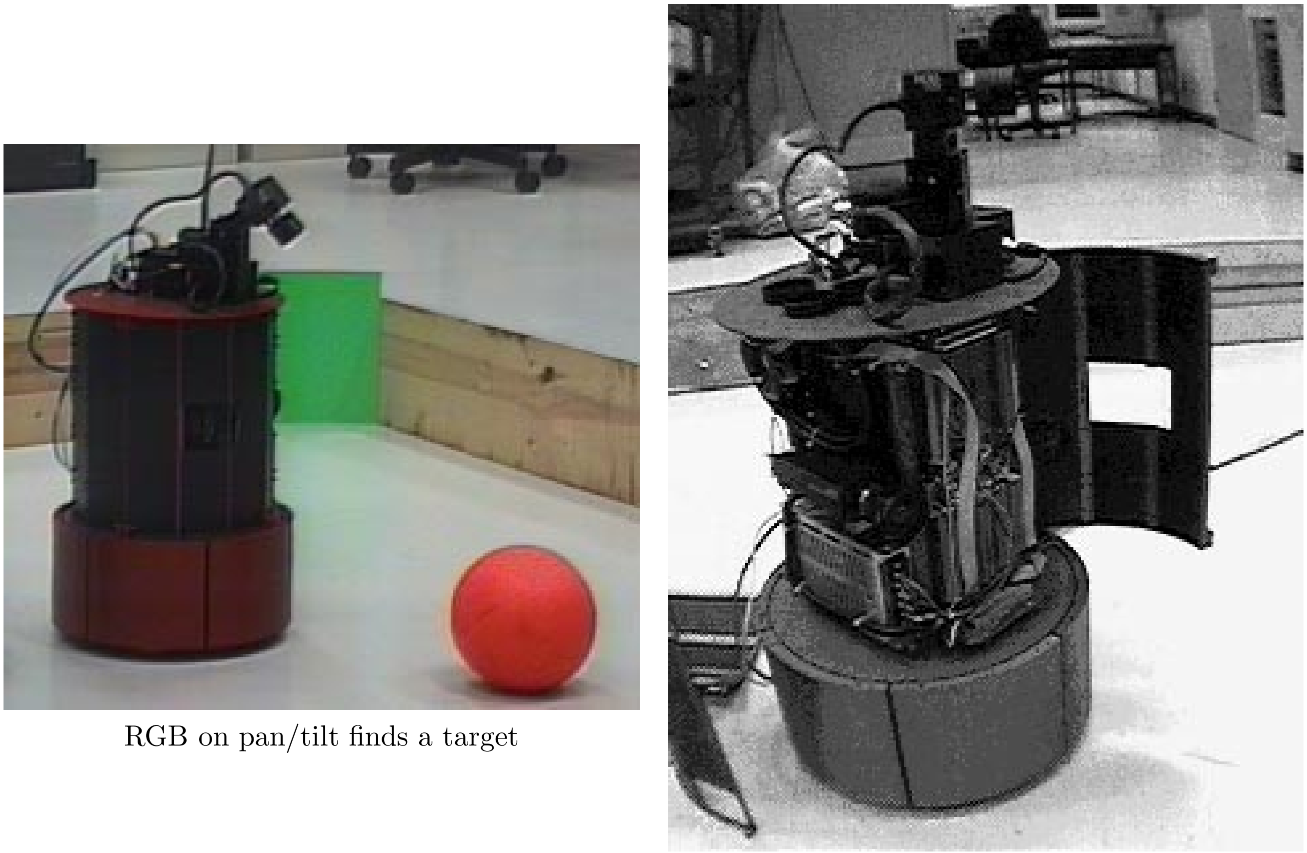
algorithm[BLP89]. Because the processing is on the slow Transputer system, target localization (connected components) takes 300 to 500ms. An important constraint for compensating for egomotion is that the eye-head complete its commanded movement before the target localization system requests optical flow data from the smart buffer. Incorrect cancellation occurs when the motion is not complete. Thus, the stage synchronize using message-passing, when the head controller completes the motion. Currently the system is able to track a person moving at a normal walking pace 2 meters from the cameras.

An implementation of this system on a Pentium system could accept the motion data from the Datacube at frame rates and localize the target with only a single frame delay.

3.4 Mobile Robots: Spinoza

There have been several stages in Spinoza's evolution. Initially the robot demonstrated capabilities for low-level vision in support of real-time robotics, such as obstacle avoidance, using coarse stereo, and remote control to specify goals for the robot and display images on a workstation. The ROLL system, for example, demonstrated single blob tracking for self-localization.

Next the system demonstrated flexible use of multiple cameras, vision modules to support mobile robot by using a pointable camera for tracking. Currently, Spinoza demonstrates high-level vision modules supported by fast real-time low-level vision. A user can specify a high-level plan (series of points to visit), or point at an map location and ask the robot to move to that location. The robot extends its a priori map with local snapshots using short/long term memory, using stereo. It can also objects to specified locations by pushing. We aim to produce a system that can mediate various goals, recognize locations, pursue moving objects ("follow that person"), recognize gestures and individuals, and engage in cooperative and competitive behaviours (such as playing soccer).



RGB on pan/tilt finds a target

Onboard processing

Figure 3: Spinoza: the stereoscopic vision robot

Spinoza[TSM$^+$97] is a self-contained robot, with host support, consisting of an RWI B-12 base with an RGB camera on a Directed Perception pan-tilt platform mounted on top and trinocular monochrome stereo cameras in the body. Its requirements include fast vision processing, flexible controllers, high bandwidth communication, and support for high-level processes. Its computing system (Figure 3 and Figure 4) has three Transputers plus two TI TMS320C40 digital signal processors, one of which is enhanced with lookup tables and a cascaded pair of A110 Inmos convolution modules that each perform a 3x6 convolution at 10 Mpixels per second. Since the bulk of early vision (image processing) computation is filtering, these elements relieve the DSP of that burden. One of the C40s grabs colour (RGB) and stereo camera information and passes the data to the second for stereo processing. The Transputers are chosen for their ease of use in real-time applications, communications and distributed processing. A high-speed radio modem, operating at 2.5 watts at 4.4 GHz, sends 1.6Mb/s or 200KB/s, which is 12.5 Hz of 16KB images (128x128) for offboard video. Currently the system gets 20KB/s (because of protocol overheads) across the radio modems. Spinoza's host workstation is a Sparcstation 20.

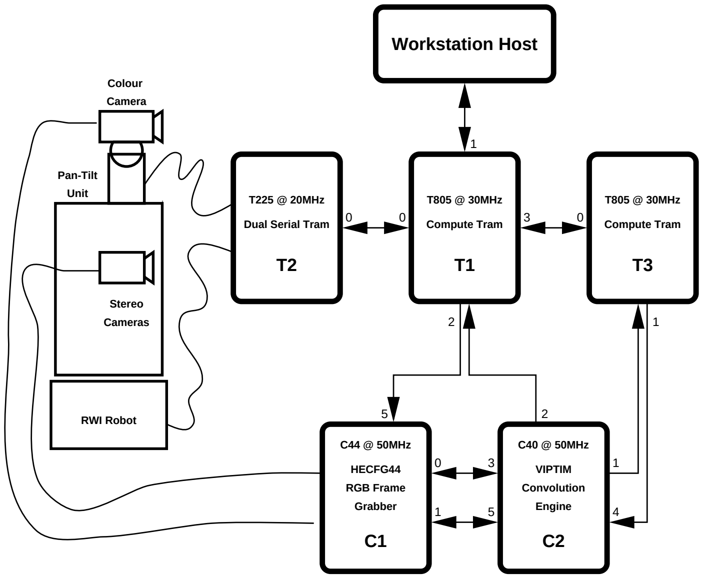


Figure 4: Spinoza Hardware

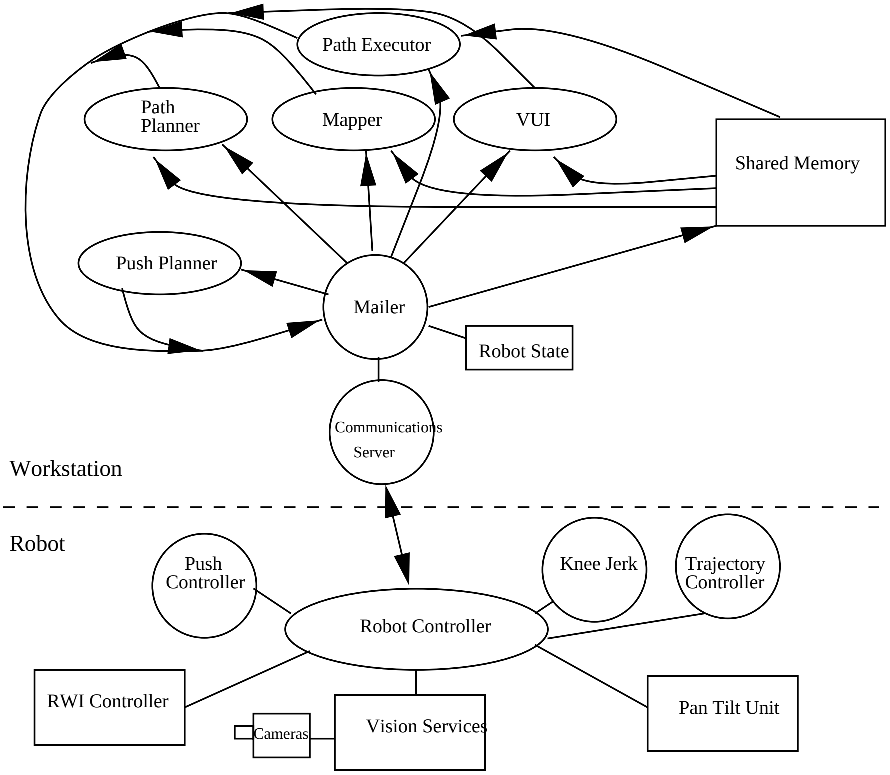


Figure 5: Spinoza Architecture

3.5 System Architecture

There are a variety of modules that implement the levels of robot architecture, as depicted in Figure 5. Spinoza can be teleoperated to move through an unknown environment, acquiring a map of the environment indicating the presence and absence of obstacles. At any time it can be directed, by means of a visual user interface (VUI), to move to a point on its map of the environment, plan a path to the goal point, and avoid both obstacles in the map and any unknown obstacle detection by its sensors. It can also push a box with a coloured marker on it around in the environment, without bumping into the known obstacles. Should an unknown obstacle loom, a proximity reflex based on stereo halts its movement and a path is replanned.

The communications module (Mailer) provides an abstraction of the robot and an interface to communication to the mobile robot itself. Teleoperation means that high-bandwidth communication must be supported between the robot, where the lowest levels exist, and wherever the high levels are implemented, which communicate with the user. The information managed in this fashion include: radial range maps, colour blobs, control mode (i.e., point and click, explore, etc.), robot state: batteries, odometry, etc., path plans, and goals posted by various task modules. Messages can request information with a range of time specifications: next, last, or update—their flow through the modules serve as control and information flows.

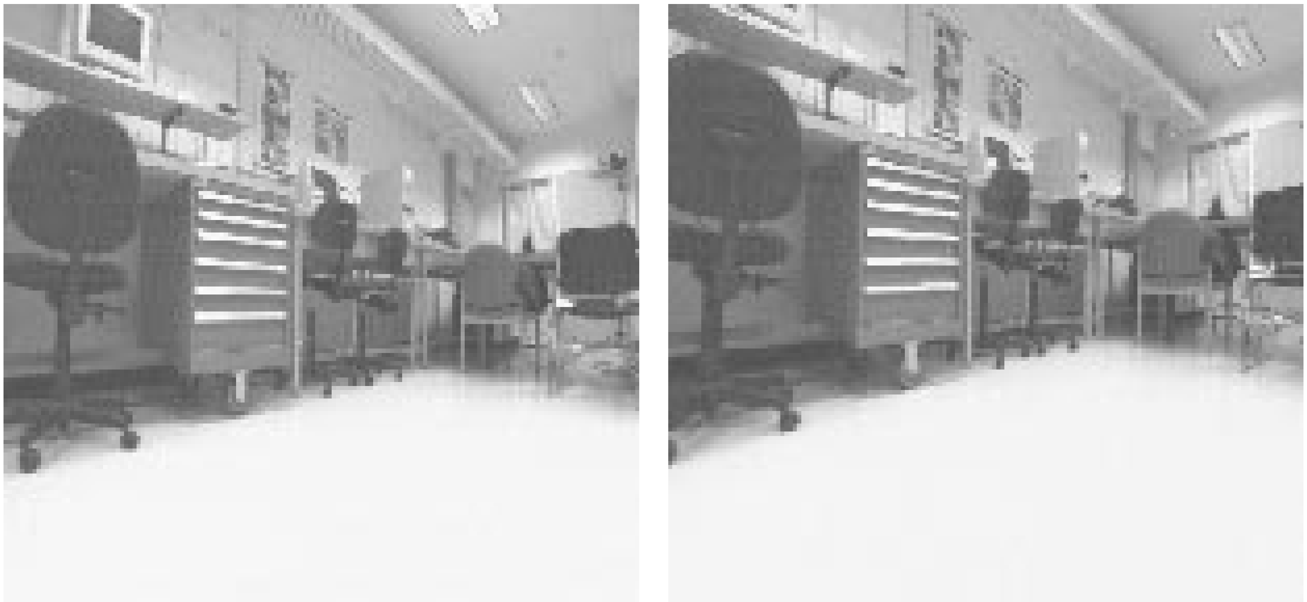
Task modules are programmed at a relatively high level. A Vision User Interface (VUI) provides a graphical user interface to task modules as well as to the interface to the robot abstraction, and can access the state of the robot including its battery charge, odometry, images as currently seen.

3.6 Stereo Processing

The C40 image grabber corrects for warping of images due to epipolar geometry not aligned with rows of images (for trinocular stereo). Cameras are calibrated[LT88] and the mapping is computed offline. Images are first smoothed, then downsampled in the correct coordinates via a large table. The right image from the cameras is shown both distorted and corrected in Figure 6.

The reliability of stereo data is paramount in obstacle avoidance—stereo is computed in trinocular format, requiring slightly more computing, but with a useful increase in reliability [HAL88]. Dense stereo [BLP89, OK93] permits obstacle avoidance without segmentation or interpretation. The system processes 128x128 pixel images at 20 disparities at 3 Hz. This replaces stereo previously implemented on the Datacube system (Vision Engine) which could operate at 15Hz.

The stereo-based mapping and navigation system is describe in [MJ97]; figures from it are used in the



Right image, distorted

Right image, corrected

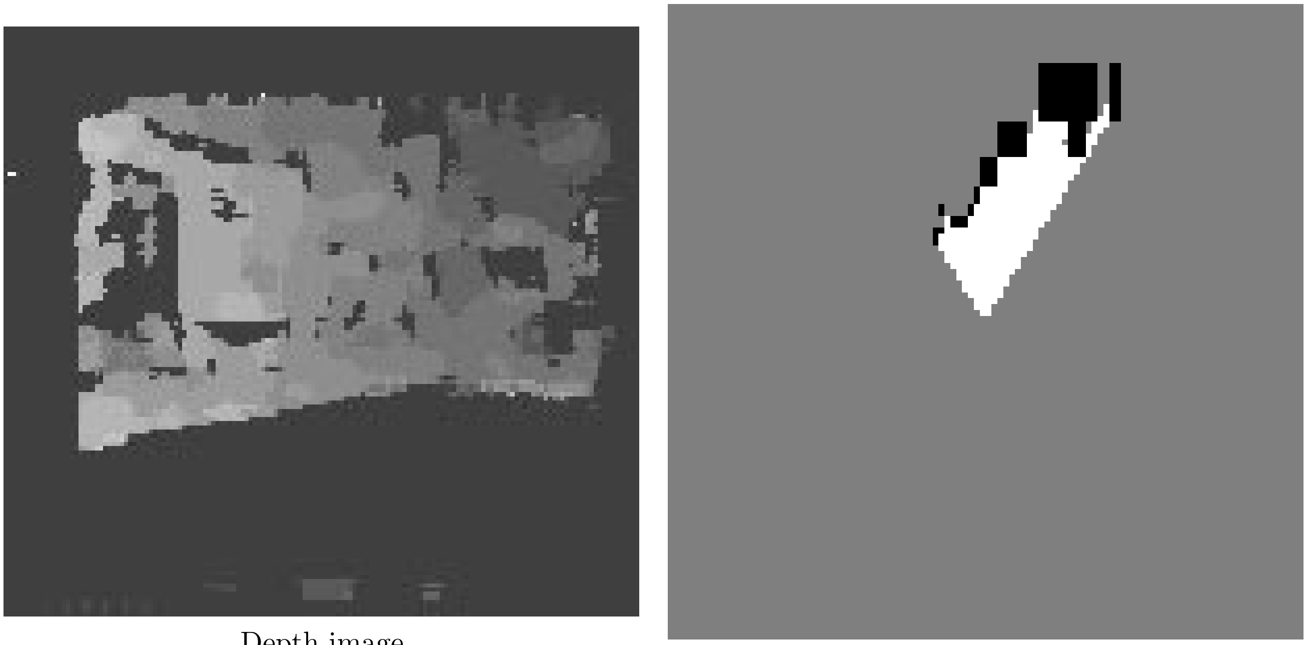
Figure 6: Images taken with a fisheye lens and then corrected.

following discussion, with permission. The depth image computed from the scene in Figure 6 is shown in Figure 7. Points where the normalized correlation value is low are eliminated (declared as invalid).

3.7 Mapper

The executive receives simplified range images: each distance array is a 2D map of the distance to an object in world coordinates. The generated map is a top-view 2D map aligned with the floor, similar to a floor plan. For such a map, stereo data can be reduced by projecting all obstacles through a vertical column in 3-space, reducing the 2D map to a 1D "radial" range map recording the distance, for each direction represented by a column, of the nearest point, regardless of vertical displacement. Each range estimate carries with it an estimate of positional uncertainty; the directional estimate has low uncertainty but error in range is proportional to the range (Figure 7).

The host application aggregates these directional range maps into a 2D map of occupancy[Elf89]. Such a map is represented by a tesselation of the mapped space into a grid. The value of each grid is related to the probability that this space is occupied by any part of an obstacle. Initially the "mapper" creates a map with all values at 50% probability, indicating that the entire space is unknown. As new radial range maps arrive, the mapper updates the occupancy grid so that each cell contains a probability that the cell is occupied by an object. Every point between the current position of the robot and the nearest point in a given direction is cleared. The probability at the cell at the given range is updated, combining its previous probability with the uncertainty of the range estimate. Cells beyond the object detected are unaffected. A sample occupancy grid map is shown in Figure 8.



Depth image

Planar Map

Figure 7: The depth image from trinocular stereo and its planar radial map.

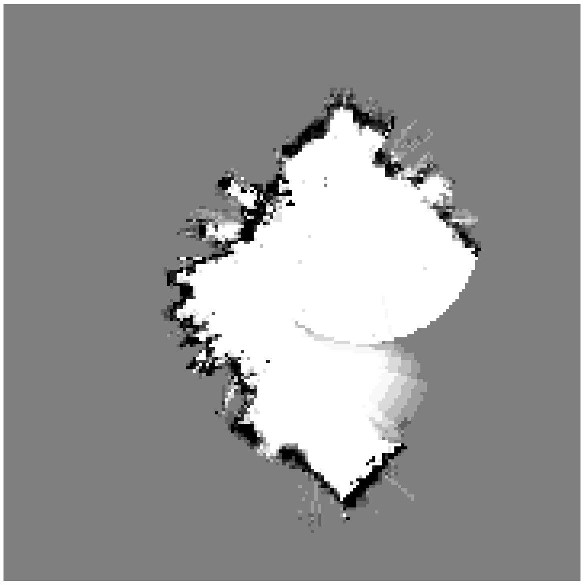
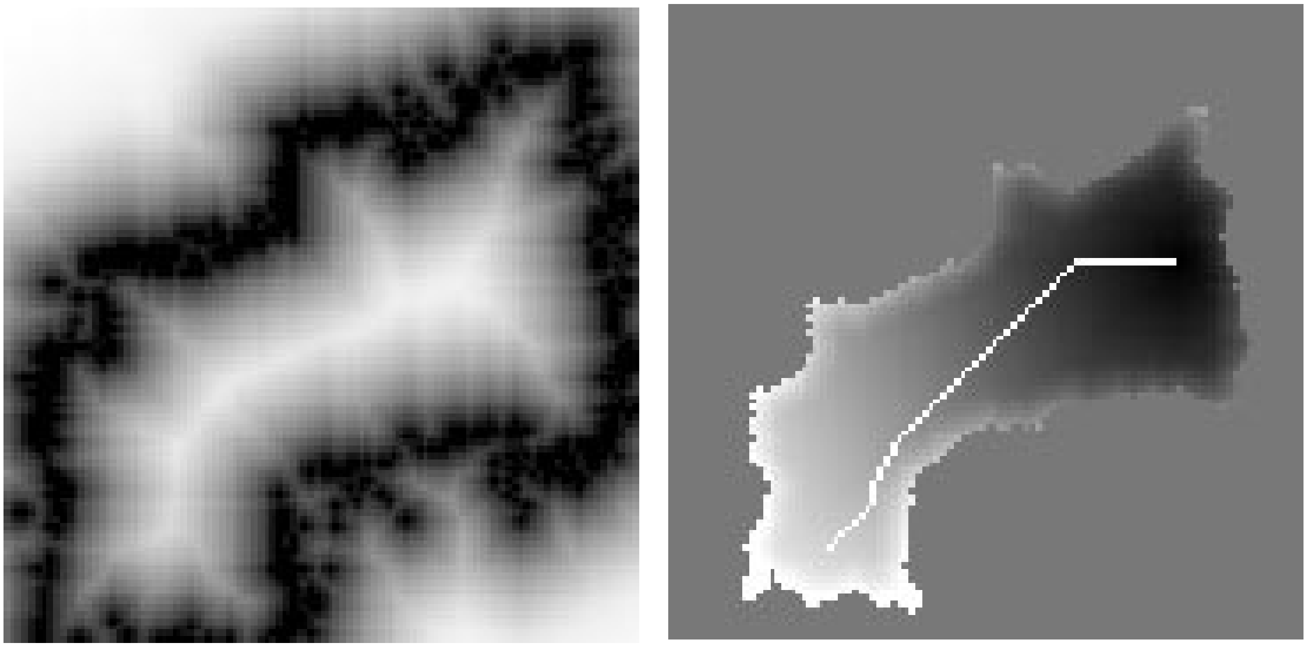


Figure 8: Occupancy Grid Map



Distance from obstacles

Final cost function with path

Figure 9: Potential map and cost function with path.

The path planner produces paths from the environment map. A simple wavefront expansion[Kha86] creates potential field between the goal and initial position, augmented by repulsive forces around obstacles. A cost function computes the minimum cost path between current position and the goal position.

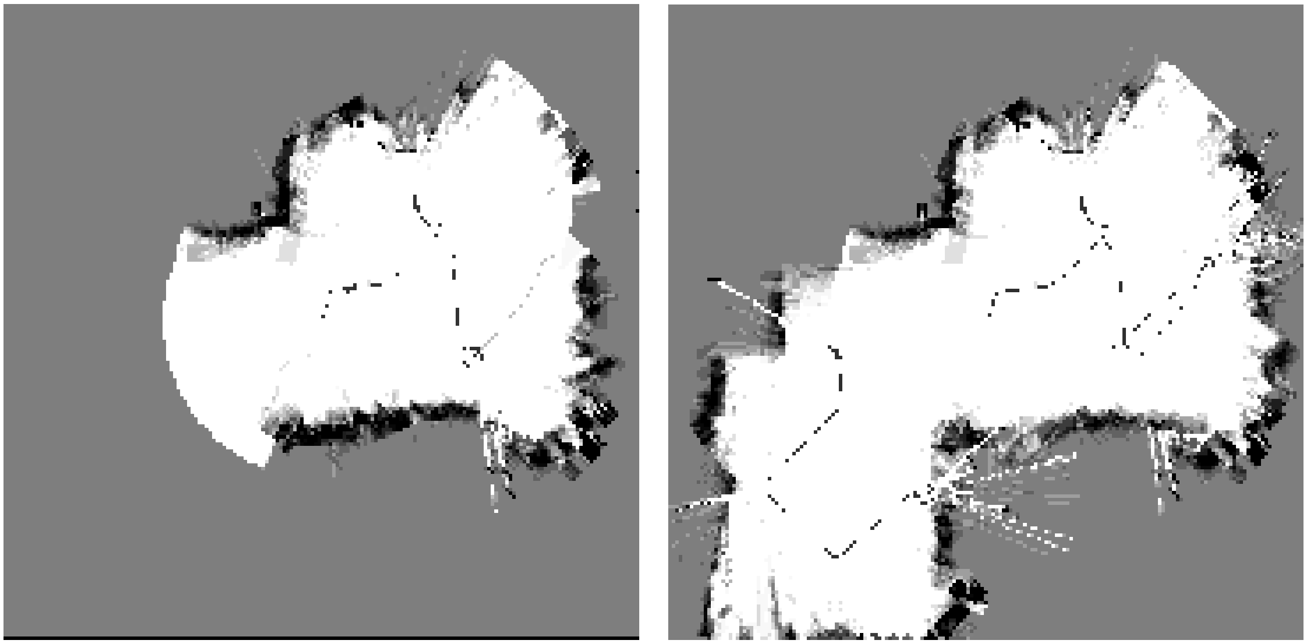
Spinoza can explore the environment: the robot aims to reduce all unknown areas in its map either to clear or blocked (obstacle). To produce exploratory behavior, the system assigns an attractive potential field to all unknown areas, while maintaining the obstacle repulsion fields. Figure 10(left) shows exploration in progress: the dark line indicates Spinoza's path. The right half shows the completed exploration, with no unknown destinations.

3.7.1 Robot Control Loops

The radial range map being sent from the vision services to the Robot controller may contain depth values that indicate that an object is "too close". This prompts the controller to stop forward progress of the robot. This tight loop between the RWI controller and the vision services must be implemented on the robot where delays are not incurred. As well, the trajectory controller divides such a path from the path planner into a sequence of closely spaced commands that smoothly combine rotation and forward movement.

3.8 Current Robot: José

We have moved to a new Pentium B-14 RWI robot, José Narvaez. The stereo system we had running on the C40s now runs on a 166MHz Pentium MMX processor, producing 160x120 stereo images at 32



Exploration in progress

End of Exploration

Figure 10: Exploration tracks.

disparities at 5Hz. The trinocular cameras use wide-angle lenses so the system must smooth, and resample the images before performing stereo computation. This technology has been commercialized by Point Grey Systems (http://www.ptgrey.com).

On the Pentium-based robot, the entire system, including mapping and high-level functionality, can reside on the robot itself. Only communication with the user (VUI), for teleoperation, resides offboard (see Figure 5).

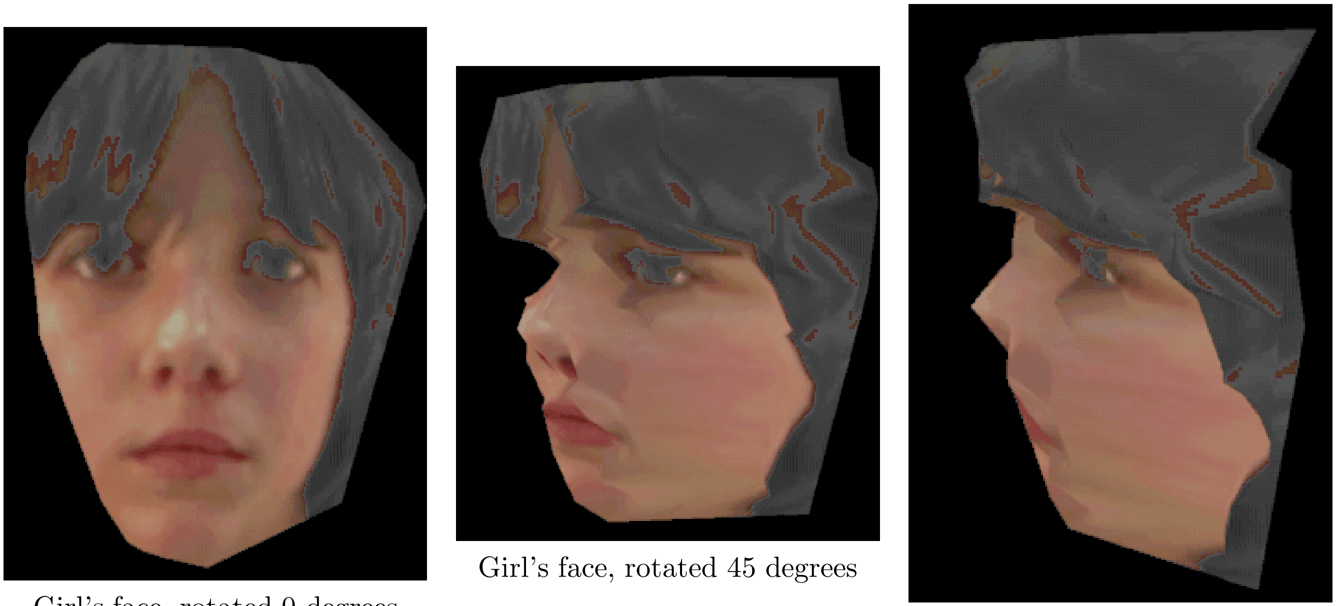
4 Scenarios for Robot Partners

We have a collection of stereoscopically guided mobile robots, sharing a common high-level interface, as well as pointable cameras. There a many possible future avenues for development of collaborative activities for these sensors/actors. This section explores several. Underlying these is always the issues of simultaneous activity in an unstructure dynamic environment.

There are a wide range of roles of pairs of viewers, including director (as in movies), coach's assistant (from high in the stands), tv commentator, lookout, spy, navigator, scout (reconnaissance), guard (eyes on the back of the head) or watcher (surveillance). Many of these appear in the scenarios to follow.

4.1 3D maps

Our current visually guided robot, José, maps its environment using real-time stereo, and integrates in a simple manner the recent stereo data with previous data in an occupancy grid.



Girl's face, rotated 0 degrees

Girl's face, rotated 45 degrees

Girl's face, rotated 90 degrees

Figure 11: VRML models, textured with captured image, derived from stereo.

We plan to extend its model from a 2D model to a 3D model to use the stereo data fully. Moreover, the wealth of data provided by multiple stereo views, under different lighting conditions and from differing viewpoints permit fusion over several views. Our current stereo algorithm has been used[JM97a] to produce range data that is then approximated by an incremental Delaunay triangulation algorithm, from which we create VRML models of faces, as shown in Figure 4.1, where the model has been rotated by 0, 45 and 90 degrees. (These can be seen at `http://www.cs.ubc.ca/spider/jennings`.)

4.2 Localization

As part of navigating through this environment, the robot must localize itself. A map can be constructed for the environment and provided with landmarks, easily observable fixed points that can be used for localization, as we have done with coloured blobs in specific configurations (personal communication, Steenburgh). This however is tedious for a large environment, especially when a robot or robots explore previously unknown sites.

It is better to allow the robot to acquire either visual or geometric information in the environment from which it can localize itself[OHD97]. Because our robots have the advantage of stereo vision, they operate with geometric information directly, and not simple images, whose appearance can be reconstructed from differing viewpoints only with difficulty and under restrictive assumptions. Locally salient geometric locations, such as corners, or door joists, or pillar/floor junctions can be identified directly from the world map and used as "discovered" landmarks. We plan to learn the useful, stable, observable points (surface patches) from our 3D world maps, and use them as landmarks.

4.3 Motion and Stereo

Scene dynamics mean that a simple scheme for maintaining an environment map will not suffice. As part of building and maintaining a world map, a motion sensor can be an important aid in sorting the critical changes in the environment from the incidental small variations. Most sites are a combination of fixed elements: walls, floors, ceilings, and movable items, with varying degrees of movability.

A mobile robot collaborating with eye-head in mapping could be aided by a record, for example, of the activity during times when the robot is not active. The motion history would be a collection of trajectories of the objects over the period during which the mobile robot was not in the viewspace of the eye-head. We can compute such trajectories using the optical flow seen by two orthogonal motion sensors (with more or less orthogonal view lines), under several assumptions: no transparency, simultaneous viewing.

Two robots have the advantage of simultaneous viewing of a phenomenon, whereas in a dynamic situation, a single robot must also try to account for physical movement and change over time, as well as localization.

4.4 Communication between visual sensors

To pursue goals in a purposive fashion, the robots must share goals, either explicitly or implicitly (reactive designs). When they communicate, what is to be the language? As well they need to have an understanding of the actions and their expected results. But all this is based on a shared model of the world. Within the context of our project, we will focus on the shared geometric model. Certain aspects of our work move beyond this to recognized known objects, specifically for collaborative telerobotics.

In order to collaborate with another robot, a robot must share a coordinate system, either topologically described, or metrically oriented. How can two robots communicate about space? What is the relative geometry of two robots?

Landmarks can provide the skeleton of a topological basis for coordinate system in which several robots or sensors can operate. When two sensors observe a sufficient set of landmarks, they can derive, using their own interior camera orientation, their exterior orientation. This permits them to describe their observations, in a common coordinate system.

To pursue cooperative goal-driven activities, robots must also communicate about the location and identity of their focus of activity. A large set of the literature on the philosophy of agents has been devoted to this problem, specifically the approaches championed in [AR96] under the guise of the *deictic* interpretation of agency. Every action is undertaken in a frame of reference centred on the object. For example, while pushing a box, the sensing and acting occurs in the robot reference system, but the action is performed relative to the box. that of the box.

Robots can communicate by referring to shared items in their world models: the map must then either be shared explicitly or implicitly. Alternatively, they can communicate by referring to observable items in the environment. Such communication could take the form of passing descriptions, or even strategies: programs by which a point in the world can be identified.

Colour histograms have been proposed[SB91] as likely easily identifiable locations. However, many lab environments lack these over large areas. We can imagine systems in which one robot might identify a location to another by passing to the other the local subimage around its point of attention. The other could reconstruct the image patch as seen from its point of view, using view interpolation[SD95]. In our robots, this is greatly simplified since they have access to surface structure as well as appearance, using stereo. The robots can then predict the local surface appearance, given its position and orientation from stereo.

Moreover, the robots can share an understanding of the focus of activity—this is the core of the deictic approach. The "object at which I am looking" or "the door toward which I am moving" can be identified by an outside observer provided it can follow my gaze or identify my trajectory. Thus collaboration on tasks can be aided by "reciprocal viewing". This allows robots to "look there", where they might direct each other. Of course this can be engineered by giving the robots laser pointers or some such device. However, if a robot is visually marked with an easily identified local reference frame (its own), a collaborating sensor can determine its gaze direction. Robots can share a reference without a common, agreed-upon external coordinate system, by using each other's physical markings to determine the relative orientation.

Sharing references is enhanced by sharing semantic models, where a robot can identify the "chair over there" rather than sharing a geometric description and a direction, but this requires a rather richer set of capabilities than we envision in the short term. However, the appearance of the chair from differing views may be simpler to access.

4.5 Centralized versus Distributed Models

An important issue is the rate at which the model of the world is updated, and whether it is centralized or not. Maintaining a large database of the world in which a robotic system (perhaps realized as many robots) puts a large burden on the robots, reminiscent of the problems of the sense-model-plan-act paradigm as criticized by Brooks[Bro87]. A collection of robots might better be served by current views of the world, rapidly communicated and anchored onto a sparse set of landmarks.

The limitations of such approaches will be discovered during their use, but some are apparent: each viewer has a limited field of view, and perhaps only intermittent ability to acquire information, perhaps only at a fixed slow sampling rate. In robot soccer, the limited field of view of an onboard sensor predominates, while an observer from above is not impeded by occlusions.

4.6 Visual Understanding of Human Action

Unlike robots, humans cannot communicate with robots digitally. We can use, however, vision sensors to acquire "gestures" that humans can easily learn so that they can communicate with robots, without the aid of devices. Interpreting observed activity is the root of gestural communication: when the set of actions are limited to a set of "signs" these form the gestures. Building systems that can respond to signs is of general interest. Usually such systems constrain the signer to a small region before the camera. This task becomes quite challenging when the signs can be generated under fairly unrestricted conditions, because it requires tracking and localizing the signer during a robot's movement[KSPF96, JM97b]. Our work identifies the signer's hands using hue segmentation (Figure 12), and then computes points of locally maximal convexity, and radiates lines from them (Figure 13) to find fingers, and finally recognizes configurations of fingers as signs. The gestures signify simple actions: stop what you are doing, go over there, follow me, pick that up. With this kind of communication, we should be able to construct gestures to communicate more complicated ideas like: go over there, pick that up and follow me, or move all those objects from A to B.

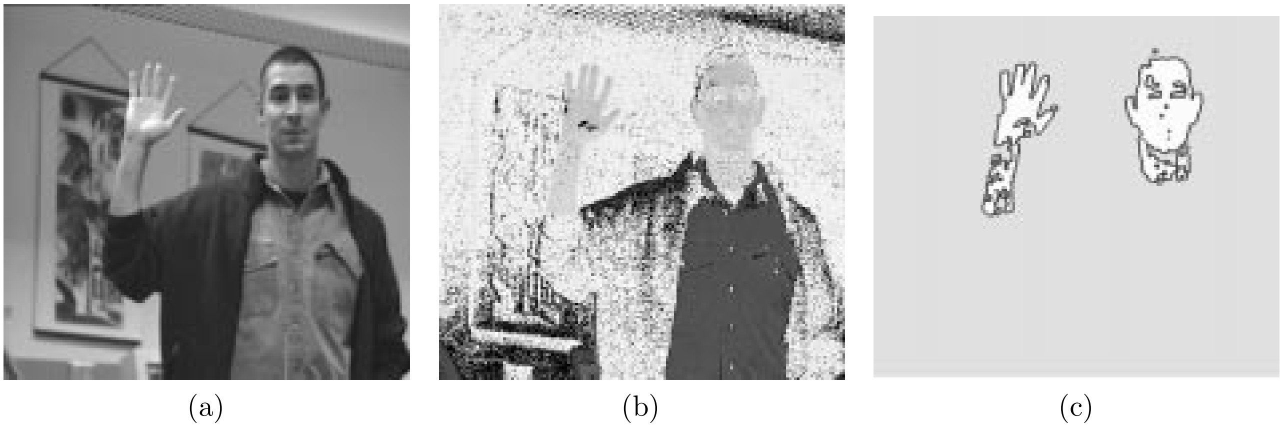
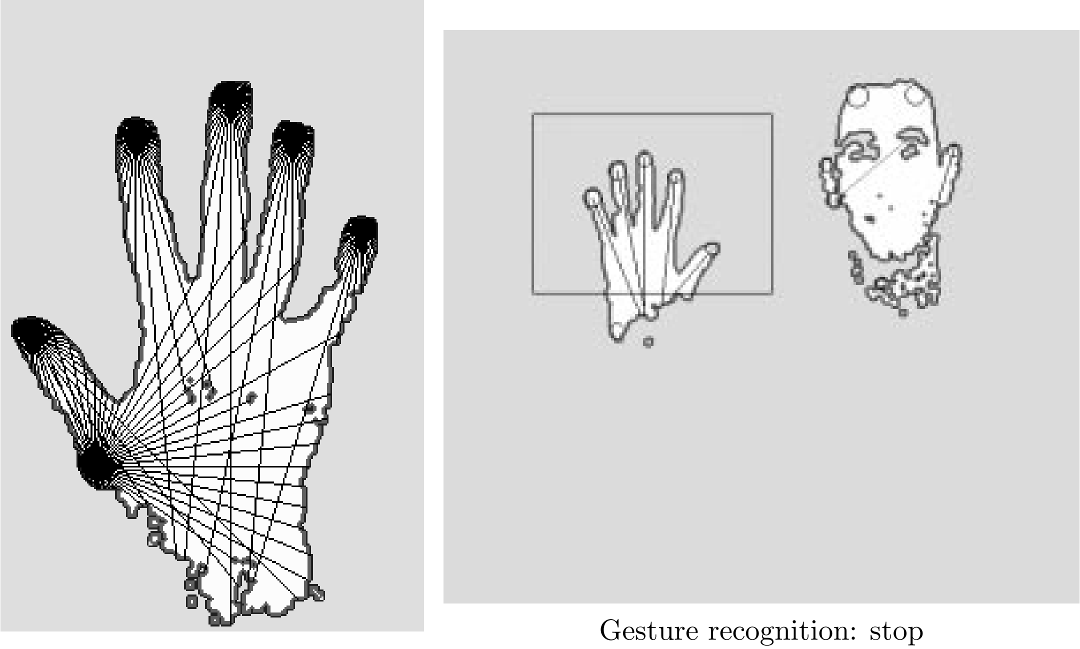

(a) (b) (c)

Figure 12: Input, hue, and segmented images

However, there is a larger possibility of interpreting human actions by observing simply their motion. Davis and Bobick[DB97], among others, have been developing methods to characterize actions by the observed motion, while we[LB95, BL97] have been using the observed motion to recognize particular individuals by their gait. Figure 14 shows a walking person whose gait our system recognizes, using variations in the *shape of motion*, a geometric description (centroids and moments, as shown in Figure 15)



Radiating lines from convex points

Gesture recognition: stop

Figure 13: Gesture recognition.

of the distribution of optical flow. Identification of moving figures can be accomplished by shape, color, or size, but this work identifies the way in which people move. (The entire system can be seen at http://vision.ucsd.edu/ jeffboyd/gait/gait.html.)

Robots must interpret the meaning of actions to interact with humans, but recognizing the intention of motion also aids interpretation of video sequences, to support, for example, the recognition of the strategy of an opponent (for a player) or a team (for a coach).

5 Discussion

Our experience with the series of visual platforms, together with our goals of realizing collaborative perceptual robotic systems, dovetails nicely with the Cooperative Distributed Vision Project. From the Dynamo Project to our Robot Partners project, we have realized a set of real-time robotic systems, both for pointable cameras and mobile robots. By maintaining a dense, detailed 3D environment map, accurate navigation is provided. Moreover, real-time stereo sensors allow safe operation in cluttered unstructured environments. José, our Pentium-based visually guided RWI robot with trinocular stereo has operated for many hours, safely, at crowded conference demonstrations.

As a first step toward scene understanding systems, we propose to build geometric scene models that will facilitate operation in dynamic unstructured environments and analyze them to provide reference systems for communication and action. By examining a series of scenarios using our mobile robots and eye-head, we can see the directions our research must take to enable them to see and act cooperatively.

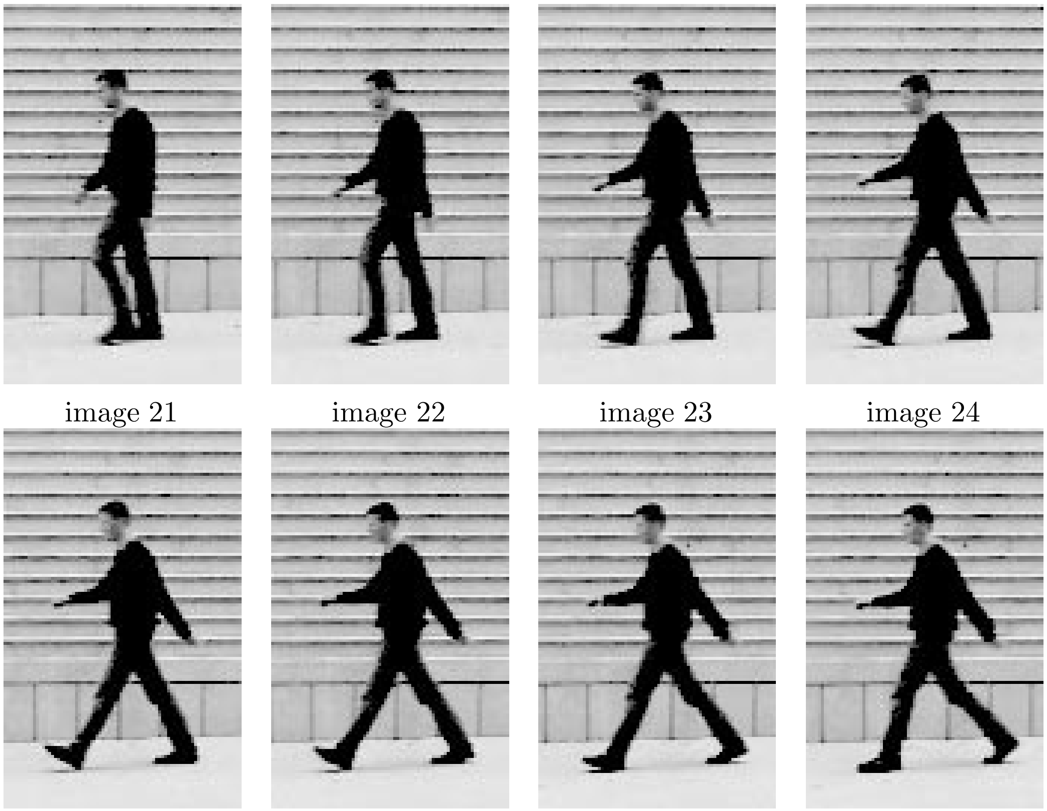


image 21 image 22 image 23 image 24

image 25 image 26 image 27 image 28

Figure 14: Subimages 21 through 28 of an image sequence.

References

[And88] R. L. Andersson. *A Robot Ping-Pong Player: Experiment in Real-Time Intelligent Control.* MIT Press, Cambridge, MA, 1988.

[AR96] Philip E. Agre and Stanley J. Rosenschein, editors. *Computational Theories of Interaction and Agency.* MIT Press, 1996.

[BKL+93] R. Barman, S. Kingdon, J.J. Little, A.K. Mackworth, D.K. Pai, M. Sahota, H. Wilkinson, and Y. Zhang. DYNAMO: real-time experiments with multiple mobile robots. In *Intelligent Vehicles Symposium*, Tokyo, July 1993.

[BL97] Jeffrey E. Boyd and Jim Little. Global vs. segmented interpretation of motion: Multiple light displays. In *IEEE Nonrigid and Articulated Motion Workshop*, pages 18–25, 1997.

[BLP89] H. Bulthoff, J. J. Little, and T. Poggio. A parallel algorithm for real-time computation of optical flow. *Nature*, 337:549–553, February 1989.

[Bro87] Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2:14–23, 1987.

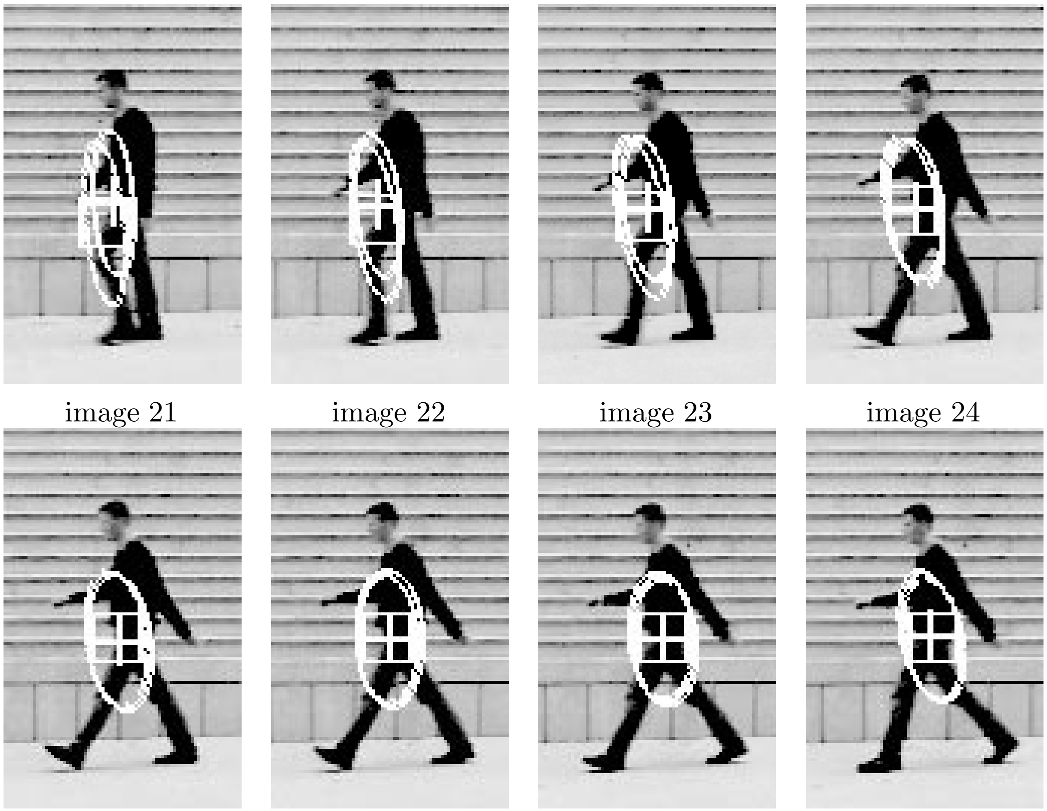


image 21 image 22 image 23 image 24

image 25 image 26 image 27 image 28

Figure 15: The centroid and moment ellipse for the moving points (box and solid lines) and for flow distribution (cross and dashed lines) for images in Figure 14.

[DB97] James W. Davis and Aaron F. Bobick. The representation and recognition of human movement using temporal templates. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1997*, pages 928–934, June 1997.

[DFR96] Gregory Dudek, Paul Freedman, and Ioannis M. Rekleitis. Just-in-time sensing: efficiently combining sonar and laser range data for exploring unknown worlds. In *Proc. IEEE Conf. on Robotics and Automation, 1996*, pages 667–672, April 1996.

[Elf89] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer*, 22(6):46–67, June 1989.

[HAL88] C. Hansen, N. Ayache, and F. Lustman. Towards real-time trinocular stereo. In *Proc. 2nd International Conference on Computer Vision*, 1988.

[HB88] I. D. Horswill and R. A. Brooks. Situated vision in a dynamic world: Chasing objects. In *AAAI-88*, pages 796–800, St. Paul, MN, 1988.

[HY94] Ian Horswill and Masaki Yamamoto. A $1000 active stereo vision system. In *Proc. Workshop on Visual Behaviors*, pages 107–111, 1994.

[JM97a] Cullen Jennings and Don Murray. Constructing TIN representations of stereo data. Technical Report, UBC, September 1997.

[JM97b] Cullen Jennings and Don Murray. Gesture recognition for robot control. Technical Report, UBC, September 1997.

[Kha86] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1):90–99, 1986.

[KSPF96] Roger E. Kahn, Michael J. Swain, Peter N. Prokopowicz, and R. James Firby. Gesture recognition using perseus architecture. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1996*, pages 734–741, 1996.

[LB95] James J. Little and Jeffrey E. Boyd. Describing motion for recognition. In *IEEE Symposium on Computer Vision*, pages 235–240, November 1995.

[LBKL91] J. J. Little, R. A. Barman, S. J. Kingdon, and J. Lu. Computational architectures for responsive vision: the vision engine. In *Proceedings of CAMP-91, Computer Architectures for Machine Perception*, pages 233–240, December 1991.

[Lit94] James J. Little. Vision servers and their clients. In *Proc. 12th International Conference on Pattern Recognition*, pages 295–299, October 1994.

[LK93] James J. Little and Johnny Kam. A smart buffer for tracking using motion data. In *Proc. Workshop on Computer Architectures for Machine Perception*, pages 257–266, December 1993.

[LT88] R. K. Lenz and R. Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-d machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:713–720, 1988.

[MJ97] Don Murray and Cullen Jennings. Stereo vision based mapping for a mobile robot. In *Proc. IEEE Conf. on Robotics and Automation, 1997*, May 1997.

[OHD97] S. Oore, G.E. Hinton, and G. Dudek. A mobile robot that learns its place. *Neural Computation*, 9(3):683–699, April 1997.

[OK93] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.

[Sah94] Michael K. Sahota. Reactive deliberation: An architecture for real-time intelligent control in dynamic environments. In *Proc. 12th National Conference on Artificial Intelligence*, pages 1303–1308, 1994.

[SB91] Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, November 1991.

[SD95] Steven M. Seitz and Charles R. Dyer. Physically-valid view synthesis by image interpolation. In *IEEE Workshop on Representation of Visual Scenes*, 1995.

[TSM+97] Vladimir Tucakov, Michael Sahota, Don Murray, Alan Mackworth, Jim Little, Stewart Kingdon, Cullen Jennings, and Rod Barman. Spinoza: A stereoscopic visually guided mobile robot. In *Proceedings of the Thirteenth Annual Hawaii International Co nference of System Sciences*, pages 188–197, January 1997.

[ZM92] Y. Zhang and A. K. Mackworth. Modeling behavioral dynamics in discrete robotic systems with logical concurrent objects. In S. G. Tzafestas and J. C. Gentina, editors, *Robotics and Flexible Manufacturing Systems*, pages 187–196. Elsevier Science Publishers B.V., 1992.