Emerald Sensor Review

SIMPLE EXPERT SYSTEMS TO IMPROVE AN ULTRASONIC SENSOR-SYSTEM FOR A TELE-OPERATED MOBILE-ROBOT

Journal:	*Sensor Review*
Manuscript ID:	SR-10-647.R1
Manuscript Type:	Original Manuscript
Keywords:	Ultrasonic < Sensor Review, Sensors < Sensor Review, Teleoperation < Industrial Robotics, Sensor Review, Collision Avoidance < Industrial Robotics, Military < Manufacturing < Industries

Title Page

SIMPLE EXPERT SYSTEMS
TO
IMPROVE AN ULTRASONIC SENSOR-SYSTEM
FOR A
TELE-OPERATED MOBILE-ROBOT.

ABSTRACT

Time taken to complete a tele-operated task with a mobile-robot partly depends on how a human operator interacts with the mobile-robot. Current tele-operated systems tend to rely heavily on visual feedback and experienced operators and this paper investigates how to make their tasks easier using an expert system to interpret joystick and sensor data. Simple expert systems improve that interaction for a tele-operated mobile-robot using ultrasonic sensors. Systems identify potentially hazardous situations and recommend safe courses of action. Results are presented from a series of timed tasks completed by tele-operators using a joystick to control a mobile-robot via an umbilical cable and watching the robot while operating it or sitting at a computer and viewing the area ahead of the robot. Tele-operators completed tests both with and without sensors and using the recently published systems to compare results. The new systems described here consistently performed tasks more quickly than some recently published systems. The paper also suggests that the amount of sensor support should be varied depending on circumstances.

Keywords: *tele-operation, mobile-robot, expert system, sensor, ultra-sonic*.

SIMPLE EXPERT SYSTEMS TO IMPROVE AN ULTRASONIC SENSOR-SYSTEM FOR A TELE-OPERATED MOBILE-ROBOT.

1. Introduction

This paper describes simple expert systems[1] for a tele-operated mobile-robot[2] using ultrasonic sensors[3] to identify potentially hazardous situations and to recommend safe courses of action. Current tele-operated systems[4] tend to rely heavily on visual feedback and experienced operators and this paper investigates how to make their tasks easier by using an expert system to interpret joystick and sensor data.

The way in which a human operator interacts with a mobile-robot can affect efficiency[2,5], and time-critical operations in emergencies require especially efficient human-machine interaction. This paper investigates improvements to that interaction using new algorithms for mixing data from ultrasonic sensor systems with joysticks controlling tele-operated mobile-robots.

Potential applications for mobile-robots range from maintenance[6], cleaning[7], service[8,9], nuclear[10-12], search and rescue[13-15], security[16], rescue[13,15,17,18], automated mapping and surveillance[19], space exploration[20], inspection[6,11,12,21-25], mine clearance[26,27] and underground exploration[28]. Many types of unmanned vehicles do these jobs, including boats[29,30], aircraft[30-32] and surface vehicles[32-35].

Although wheeled vehicles find it difficult to move freely over terrain such as swamps or wasteland and walking robots are improving[5,11,12,36-38], wheeled mechanisms are still the main mechanisms for moving over ground[2,15,17,39-42].

Tele-operated robotic systems put an operator at a distance to increase operator safety[2,14,17,34,43] and a robot can be remotely operated from outside a hazardous environment. Tele-operation has been well used for maintenance in nuclear industries and the real challenge in unstructured and difficult environments such as hazardous areas is primarily for mobile-robots[9]. This research is timely because these sorts of systems can be used in radioactive environments and many nuclear power plants are coming to the end of their service[10,44]; nuclear facilities are aging and many will close within two decades.

Systems have tended to be open loop. Users have indicated a direction and the mobile-robot then moved in the required direction. Common disturbances include differences in mobile-robot wheels or tractors or their different reaction to surfaces and surface or gradient[5,45-47]. Users have been left to react to disturbances and correct trajectories. Intelligent systems required sensors and control algorithms that could process information from tele-operators and sensors and use this information to assist them.

Tele-operated mobile-robots are generally guided using manual controls, often joysticks[5,34,46,48] although other input devices are available, such as switches[40], or pointers[49-51] or custom built, such as Virtual Reality interfaces[52].

Much research has aimed to improve tele-operation and robotics in hazardous or unpleasant environments[2,9-12,26-29,53], or in places where conventional techniques required cost intensive supporting infrastructures[54-56]. Conventional robots are limited when a task requires a level of perception and decision making which cannot be met in a cost effective or robust way. For many applications, mobile-robots may not need autonomous control with models of the environment. Instead a human operator may help a mobile-robot to explore environments.

Mobile-robots and advanced Automated Guided Vehicles (AGVs) need to navigate around obstructions or take different routes to avoid other AGVs[57,58]. AGVs have used various sensors to achieve local avoidance: light and laser[59], ultrasonic[2-4,14,17,34] and infra-red[60,61]. Positioning has used odometry, fibre optic gyro, tilt sensor and acoustic[62,63]. GPS[64] is a de facto positioning system but GPS does not operate indoors or in electromagnetically shielded areas. Assisted GPS[65] may be applicable indoors, but requires network assistance and special hardware. Vision integration opens up a range of new possibilities[8,66-72], but these systems require data processing and have been relatively expensive and complicated, although costs are reducing and computer power is increasing[73]. Vision systems are more robust and affordable than ever and tele-operated mobile-robots have relied on them for detection but then for guidance by a human being, sometimes using some force feedback[43,74].

A human operator was the most accurate source of data but that source could be impaired by distance, poor vision or restricted view (even with a more local camera). Although the technology is limited by the shapes of surfaces and the density or consistency of material, ultrasonic ranging was selected to assist because it was simple and robust[2,14,34]. Ultrasonics could detect clear objects, for example windows, and target colour and/or reflectivity did not affect the sensors, which operated reliably in high-glare environments. A human user then guided the tele-operated mobile-robot[4,17,34].

Some tele-operating systems are described by Shao[75]. AGVs are described by Rocha[76]. They are included here for reference and wider reading. Current challenges being faced in tele-operation are described by Chen[77].

2. Tele-operated mobile robot system

A recently published ultra-sonic sensor system for a mobile-robot was described by Sanders[2]. That used 40 KHz ultrasonic transmitter and receiver pairs mounted in front of a mobile-robot[14]. The system transmitted a 1ms pulse of ultrasonic energy and the pulse was reflected from objects in its path. Some reflected energy returned. Distance from sensors to object was then calculated from time taken for the pulse to return. With suitable processing the ultrasonic image was converted to a simple representation of the environment and objects in the mobile-robot path were detected.

In new work described in this paper, the tele-operated mobile-robot base shown in Figure 1 was initially controlled through a joystick. A controller interpreted joystick control signals and provided power for the motors. The mobile-robot was electrically powered with a front wheel drive chassis and fibreglass body. The base was a heavy steel plate chassis to provide stability and rigidity. Two driven wheels were at the front and two trailing casters at the back. A camera could be mounted between the driving wheels and ultrasonic sensor pairs could be mounted over each driving wheel and in the middle at the front.

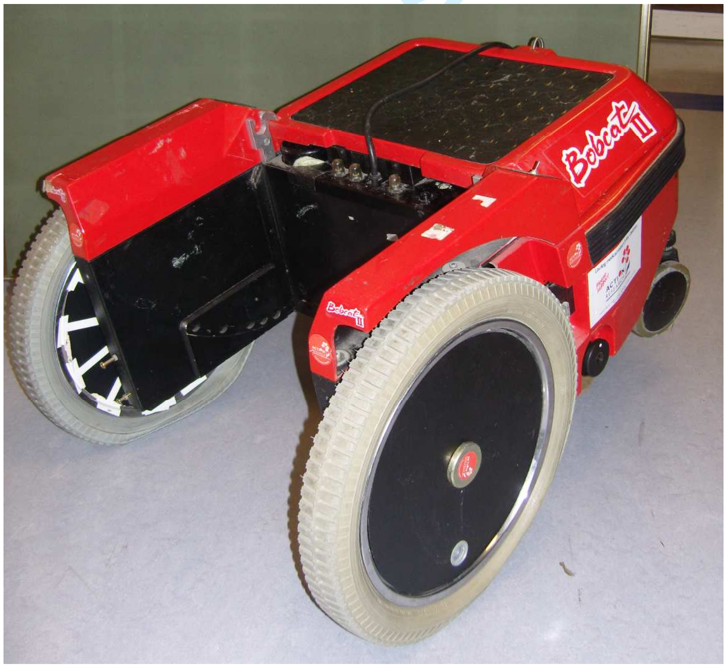

Figure 1 *Bobcat II Base Unit.*

Trailing casters supported the rear and driving wheels were powered by two 12V DC motors through a worm drive right angle reduction gearbox. Correction was applied by means of differential motor drive[5]. Altering the differential of rotational speed of the driving wheels affected steering. The mobile-robot consisted of a power source, drive motors, an input device and a controller. Power, communications, joystick, interfaces and potentiometric and input devices are described in [2].

The direct link between the mobile-robot and joystick was severed and a computer processed control information. Three modes of operation were then possible:

- Joystick data could be processed and sent to the controller without modification.
- Sensors were activated and interrogated by the computer and the computer was programmed to modify the mobile-robot path.
- Sensors were activated and interrogated by the computer and the computer was programmed to modify the mobile-robot path using new algorithms described in this paper.

Hierarchical code structure is shown in fig 2 and was similar to levels described by Tewkesbury[78-80]. Algorithms apply the following rules:

- User remained in overall control.
- Systems only modify trajectories when necessary.
- Movements were smooth and controlled.
-

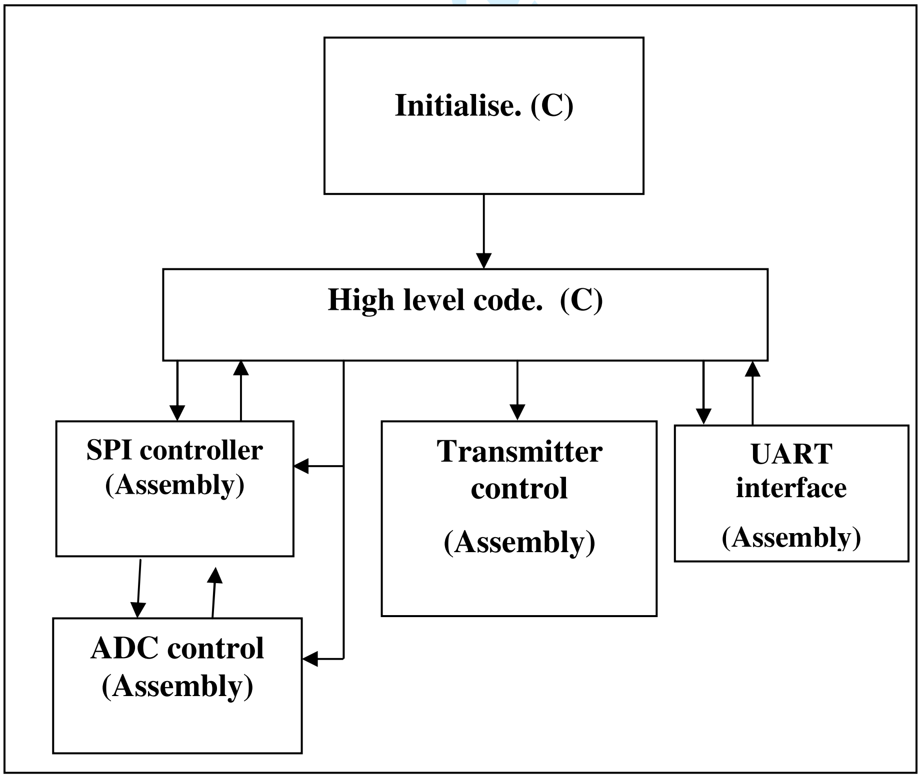

Figure 2. *Hierarchical structure of the initial code showing the C Modules and Assembly Code*

An imaginary potential field was generated around objects in response to sensor information[81,82] to assist users if the mobile-robot was approaching an object and could collide.

The sensors: Ultrasonic transmitters required a pulse of 3ms duration to reach maximum output power. A long ultrasonic pulse contained more energy and detected objects at greater ranges. If speed of sound in air is assumed to be 330m/s… physical length of a 3ms pulse of sound is 0.99m. Allowing for the pulse to leave the transmitter, bounce back from an object and return to the receiver, then minimum range for a 3ms pulse would be 0.5m. Because closer ranges were required, shorter pulse lengths were needed. Pulse lengths of 10us, 100us, 500us and 1ms were examined. A range finder was created to automatically switch between pulse lengths as the range changed. If no object was detected, the range finder hunted by systematically increasing pulse length to increase range.

Work on adaptive range finding is ongoing but other simpler implementations were used in research described in this paper because multiple targets tended to appear and disappear as the mobile-robot moved around and it was difficult for the adaptive range finder to lock on to a target.

3. Histogrammic mapping.

Despite the advantages, ultrasonic sensors tended to be noisy and return misreads. A method for filtering out misreads was selected to improve sensor reliability that was based on Histogramic In-Motion Mapping. Volumes in front of each sensor were divided into a simple grid of three volumes: near, middle and far (as shown in figure 3).

They were stored as an array in micro-controller memory. When a range was returned, it was classified as near, middle or far. Different numbers of sensors were mounted so that their beams swept the area in front of the mobile-robot. The arrangement for three sensors is shown in figure 4.

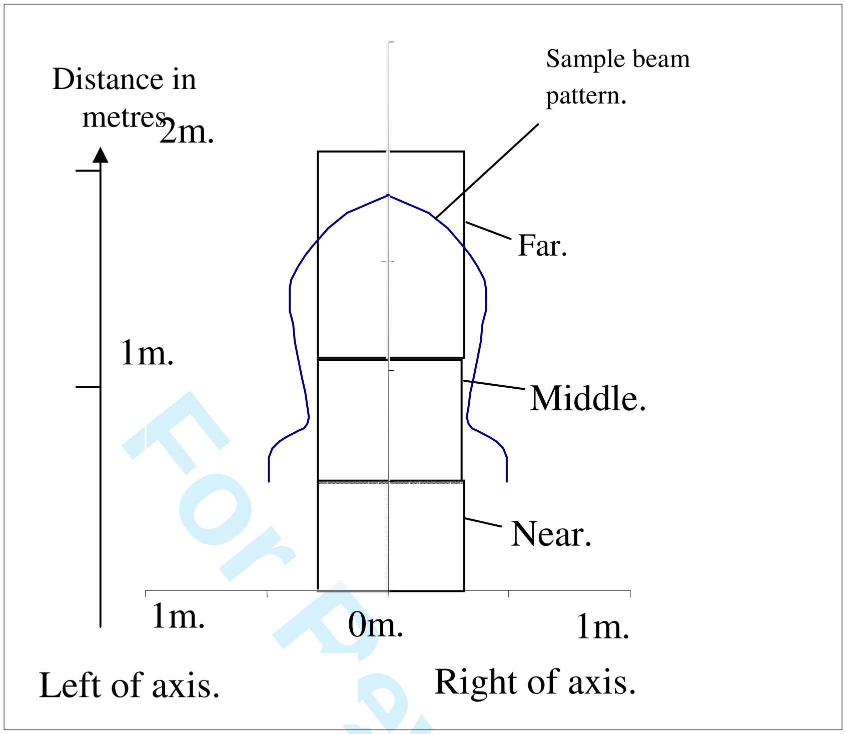

Figure 3. *A simple representation of the envelope of a ultrasonic single sensor.*

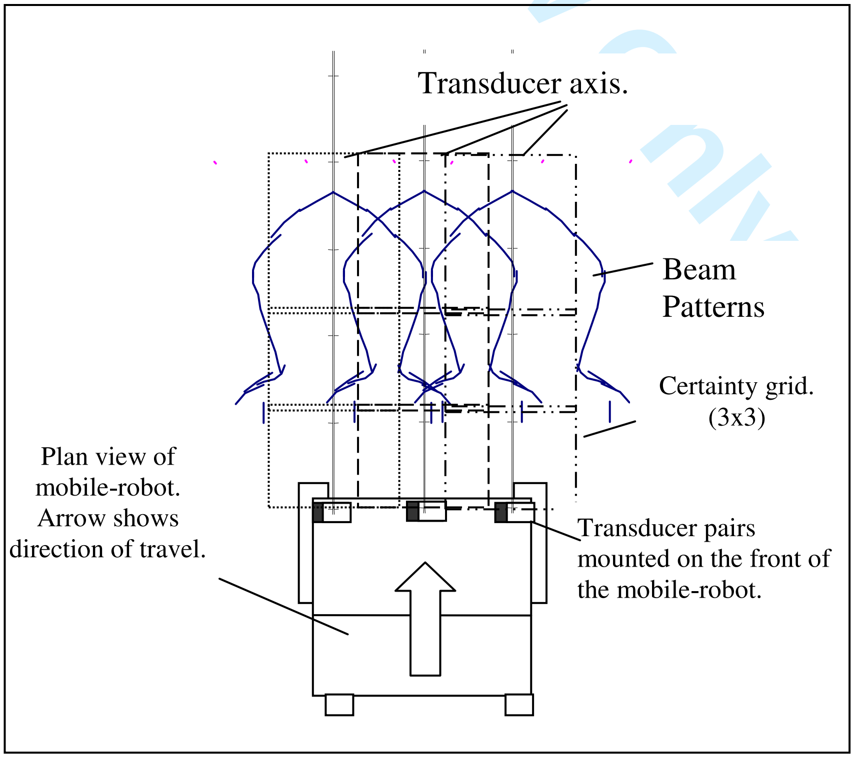

Figure 4. *Representation for a three-sensor array.*

Array elements representing the area in which an object was detected were incremented by a higher number, for example, three. Other array elements were decremented by a lower number, for example, one. The arrays typically had a maximum value of 15 and a minimum value of zero. Figure 5 shows an example of the simple three-element histogrammic representation of the environment and the position of an object in the third element causing that element to ramp up. An object occupying a grid element would cause that element to quickly ramp in value to the maximum. Random misreads in the other elements incremented that element temporarily, but the value of the false reads were decremented each time the system updated. If the object moved to a different element, the new element quickly ramped up to its maximum value and the old element ramped down to the noise level. A reliable range could be acquired within half a second. Figure 6 shows the structure of the histogrammic object detection process.

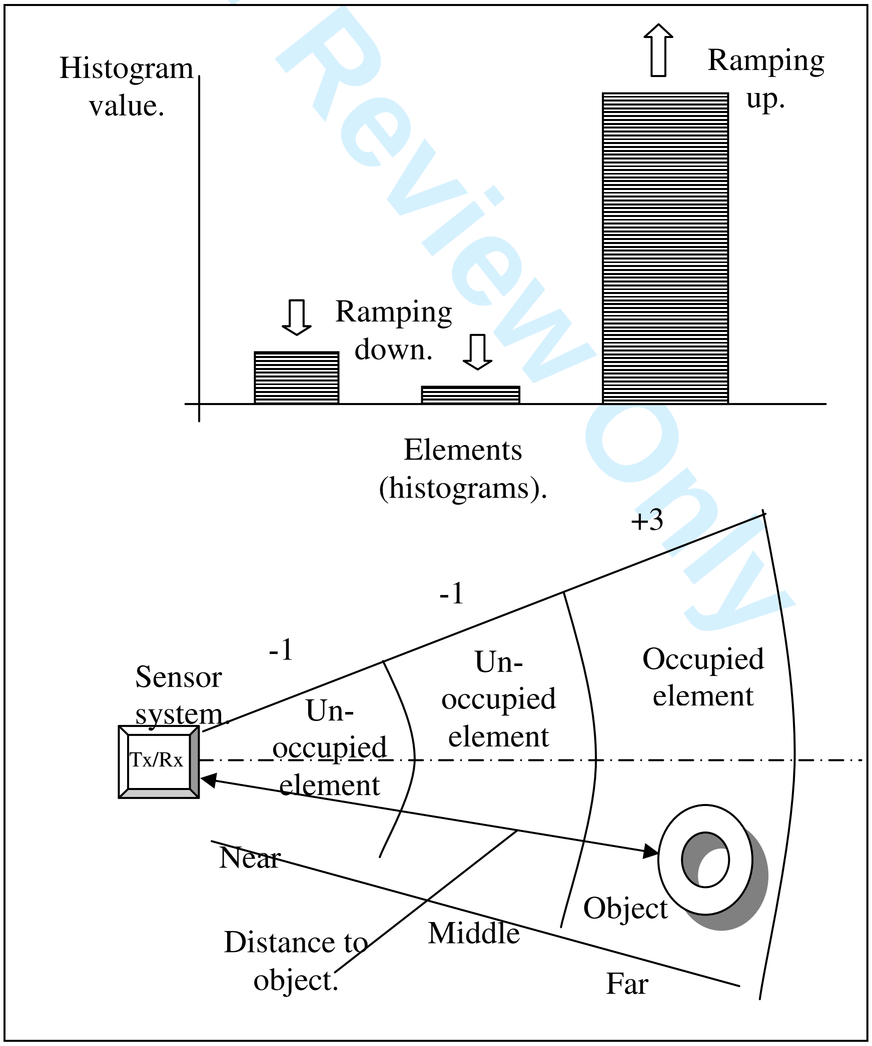

Figure 5. *A representation of the sensor histograms.*

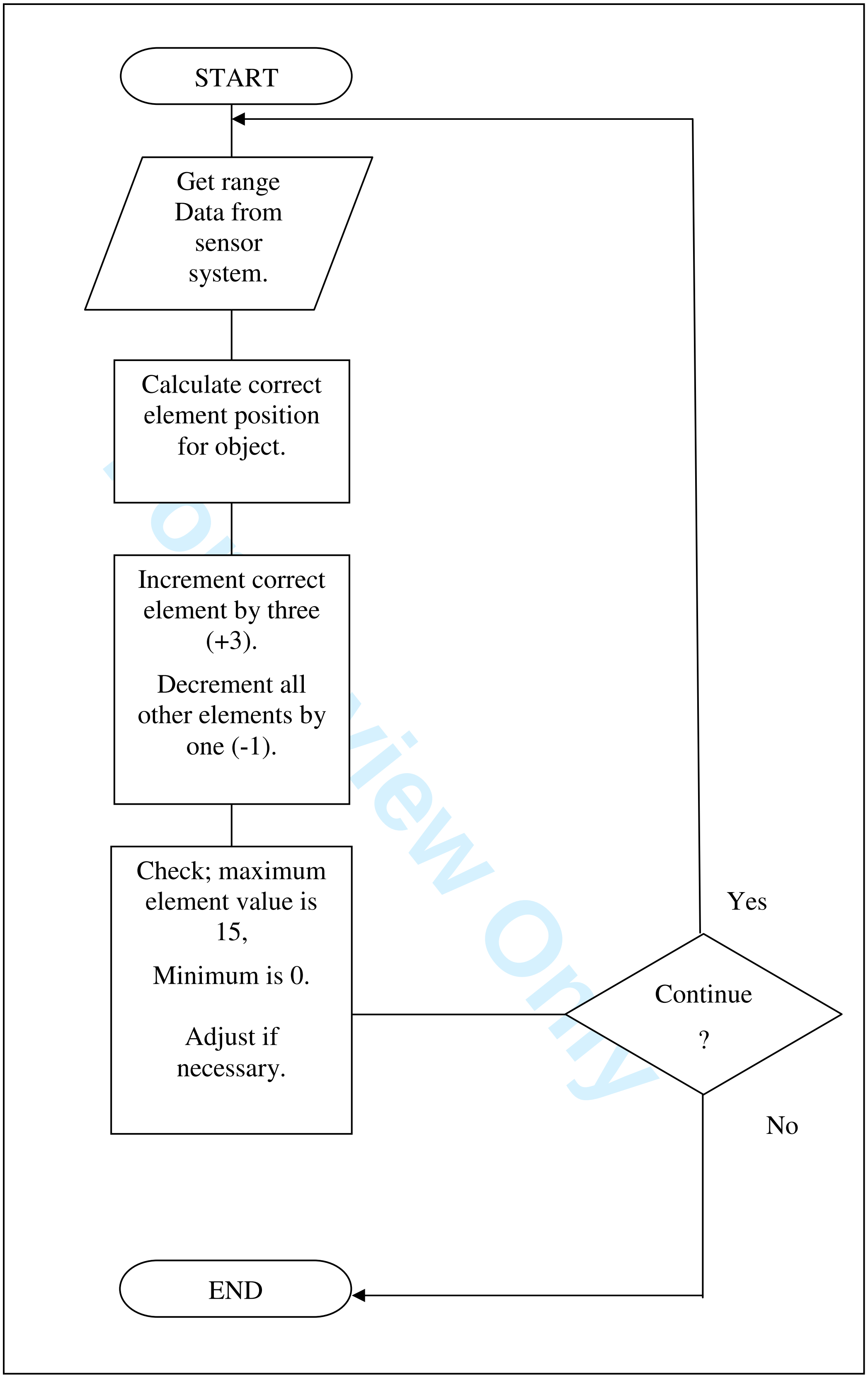

Figure 6. *The structure of the histogrammic object detection process.*

4. Algorithms to interpret the joystick.

A standard Penny and Giles Potentiometric joystick was fitted that contained two potentiometers to provide two channels of joystick voltages. Joystick position could be read by an analogue to digital converter (ADC) as a set of Cartesian co-ordinates. Cartesian co-ordinates were not a convenient way to express joystick position; co-ordinates did not contain information on joystick signal direction or magnitude. To interpret joystick data in a more convenient manner, Cartesian co-ordinates were converted to polar co-ordinates using trigonometrical functions and Pythagarus' theorem. Joystick data was now in the form: $|J|\angle\theta$. Where $|J|$ was magnitude (or how far the joystick had been pushed) and $\angle\theta$ was the angle of the joystick. Standard mathematical functions were used from C libraries for arctan and square root functions for the Cartesian to polar conversion.

Joystick output was integrated to provide a level of confidence in user intentions. Magnitude could be integrated simply as it was a scalar quantity. The angle of the joystick introduced a directional element which could not be integrated. The joystick angular position was quantified so that intended direction could be estimated. This allowed algorithms to measure the length of time that a joystick had been held in a consistent direction and helped the new systems to identify the wishes of the tele-operator. Joystick angles were defined as:

Spin left	**1.54 – 2.36 radians**
Spin right	**5.50 – 6.28 radians**
Turn left	**0.89 – 1.54 radians**
Turn right	**0.00 – 0.69 radians**
Forward	**0.69 – 0.89 radians**
Reverse	**2.36 – 5.50 radians**
Stop	**magnitude<16**

Sectors are shown on figure 7. Joystick angle and magnitude were calculated using:

```
argument = JS0/JS1;            //opposite over adjacent (ATAN)
angle = atan(argument);        // joystick direction in radians
```

Magnitude was calculated using: **magnitude = sqrt((JS0*JS0)+(JS1*JS1));** where JS0 and JS1 were the Cartesian co-ordinates with the origin centred on the joystick stop position. Magnitude and angle were then used to calculate the sector that the joystick was occupying. The position and confidence of the joystick could be expressed as an array. Each joystick sector contained two array values:

- "Angle Confidence" (0 to 15) indicated certainty that a joystick was being held in a sector.
- "Magnitude" indicated joystick position with regards to demanded mobile-robot speed.

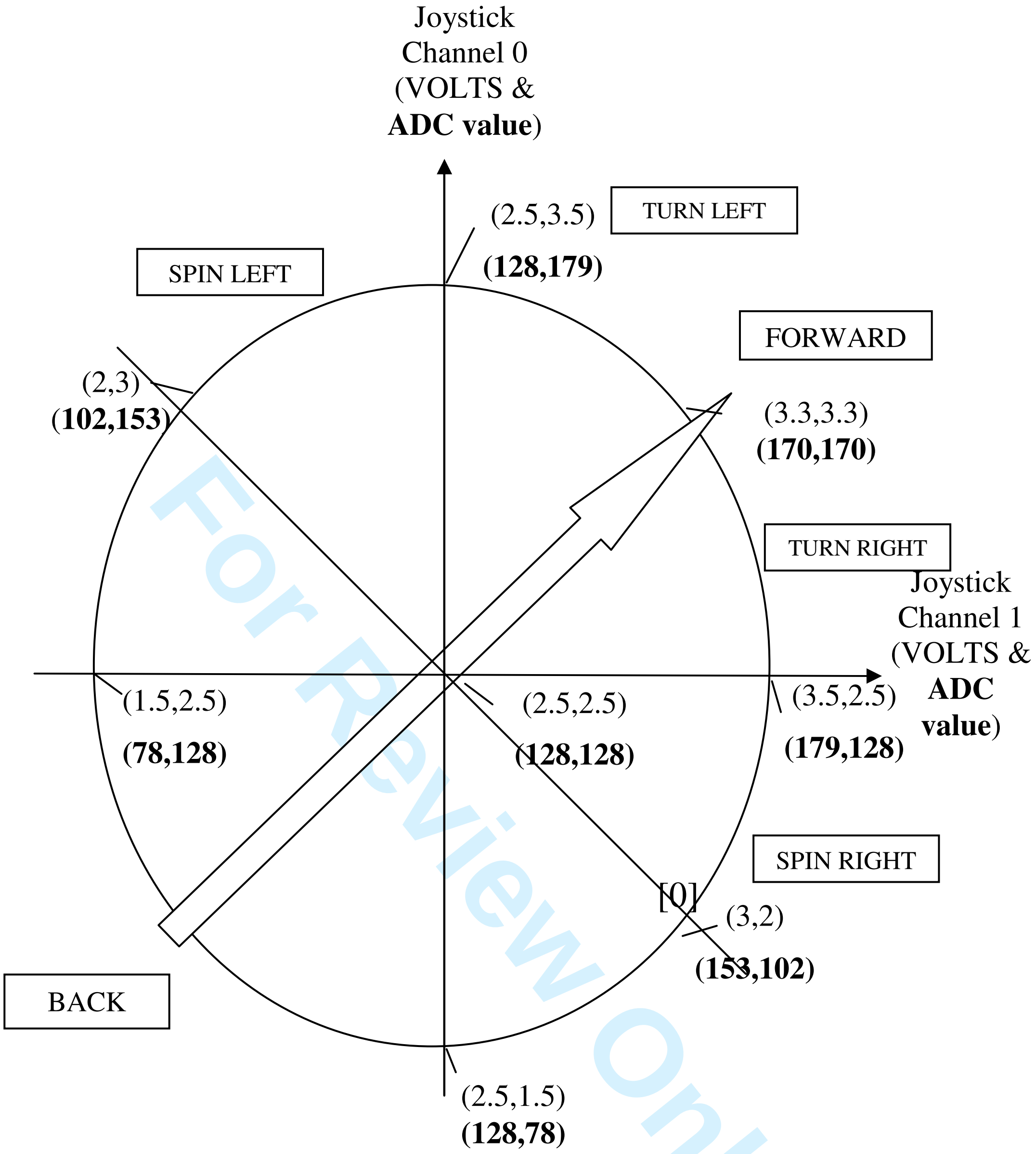

Figure 7. *Map of joystick voltage and ADC values.*

An histogrammic representation was then used as a pseudo-integrator. If the joystick was held in a position, the array element relating to that position was incremented to raise its overall value. All other array elements could then be decremented to reduce their effect. The array element with the highest value was used as the latest and most confident joystick position. The joystick occupying a joystick array element would cause that element to quickly ramp in value to maximum. Random joystick action in the other elements incremented them temporarily, but values of false reads were decremented each time the system updated. If the joystick moved to a different element, the new element quickly ramped up to maximum and the old element ramped down to the noise level or zero.

Joystick position was represented as a histogram where the highest histogram element represented the most likely direction for the user to be indicating as the desired direction. An example histogram for the joystick is shown in figure 8.

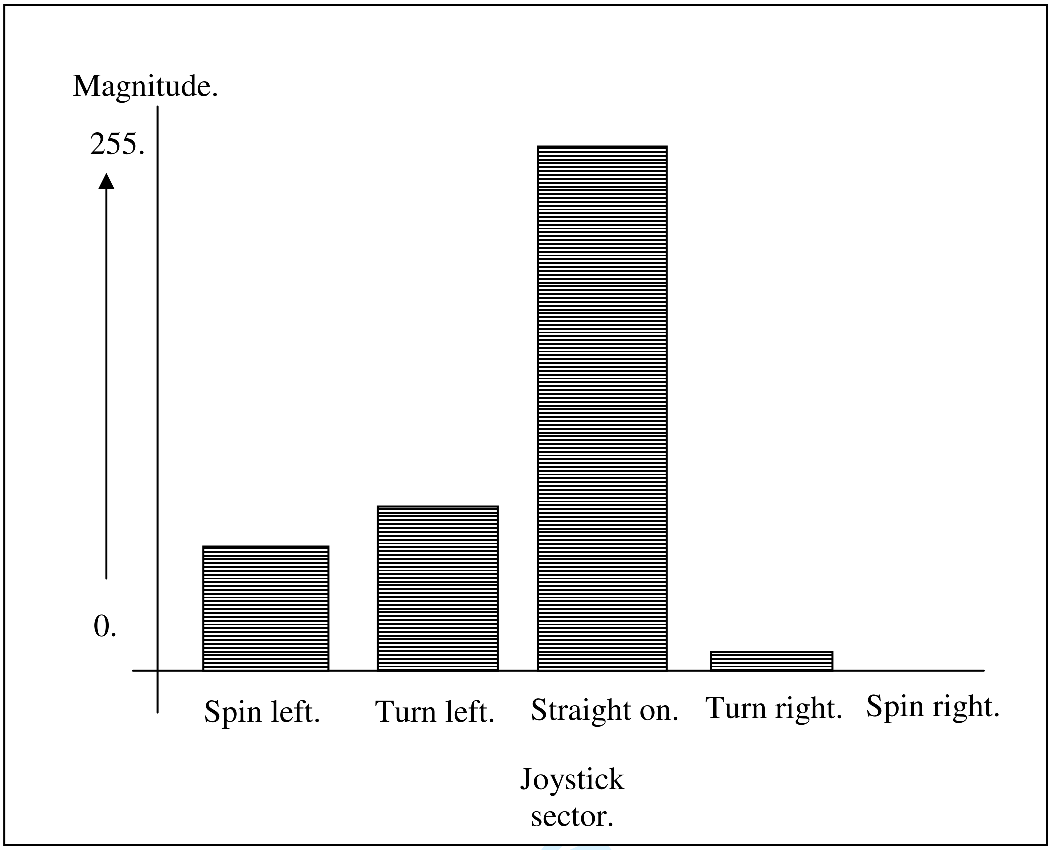

Figure 8. *A representation of the joystick using histograms.*

A module called JSArray tested joystick position and angle, and indicated which sector the joystick was occupying. The appropriate element of the “angle confidence” (Aconf) was then increased by magnitude 40. All Aconf elements were then decreased in magnitude by 30 to decay the un-occupied elements. The occupied element was therefore subject to an increase of 10 magnitude and all other elements were subject to a decrease in magnitude of 30. This allowed the histogram elements to decay rapidly and build in value more slowly. A joystick array element was able to increase to its maximum value of 225 in a minimum time of 0.5 seconds (approximately) and decay to zero in approximately 170ms. The ramping and delay weighting factors were determined experimentally by driving the mobile-robot with several different weighting factors in operation. The delay induced in the response of the mobile-robot by the weighting factors could be set to an individual user or task.

5. An initial prototype expert system

Expert knowledge[83] was acquired from human "experts"; in this case a human mobile-robot driver and an engineer. Knowledge[84,85] required to drive a powered mobile-robot to most people was intuitive. A little time was usually needed to familiarise oneself with the response of the system to the joystick. Some people were more naturally dextrous and could learn to drive in less time than others. When familiarisation was completed and system response is known, a user could drive effectively. Rules were intended as generative rules of behaviour; given some set of inputs then rules determined what the output should be.

It was important that the system operated in real time[86,87] in order to assist a tele-operator. There were two real time inputs to the system; the input device (joystick) and sensors. A user indicated a speed and direction and the sensor system gathered information about the environment. A module called Sensor Expert analysed sensor information and made a recommendation for a path to prevent collisions. Data often conflicted. Another expert, called the Fuzzy Mixer considered both inputs and was responsible for final motor controller outputs. Joystick Monitor was responsible for interpreting the wishes of the user. Variables such as joystick position and consistency were examined by Joystick Monitor to assess the desired mobile-robot trajectory. The first prototype consisted of:

(a) Fuzzy Mixer.
(b) Joystick Monitor.
(c) Sensor Expert.
(d) Doorway.

(a) Fuzzy Mixer apportioned control effort between joystick and sensor systems. It matched joystick and sensor system recommendations, examined conflicts and kept controller voltage within parameters. It received information (or advice) from Sensor Expert, Joystick Monitor and Doorway. Proximity Stop was a failsafe anti-collision function that stopped the mobile-robot from crashing. For safety, Fuzzy Mixer could override any input with "Proximity Stop". Fuzzy Mixer took joystick confidence values and sensor information and mixed them. Low joystick confidence meant the system needed to avoid obstacles[8,82,88,89] and drive safely in the direction set by the joystick. High confidence in the joystick meant it accurately reflected user wishes and the sensor system had less influence. For example, if the joystick was directed to make the mobile-robot hit a wall, then the joystick effect was initially reduced but if the joystick was held in the same position, then joystick confidence increased and the mobile-robot eventually moved to the wall. At a pre set distance, Proximity Stop activated and the mobile-robot stopped close to the wall.

(b) **Joystick Monitor** checked for changes in joystick position and consistency. A steady joystick position indicated a desire to go there. A joystick moving randomly indicated an unsure or out of control driver and the system relied more on sensors to steer.

(c) **Sensor Expert** applied knowledge of sensor combinations. Sensor Expert created a sensor grid and made recommendations on courses of action to take the mobile-robot away from an object or to prevent the mobile-robot from hitting it. Sensor Expert did not consider the wishes of the user.

(d) **Doorway** extracted information from data supplied by Sensor Expert. It was an object avoidance program that avoided objects through a "distance function" algorithm. Doorway was over-ridden or allowed to affect the trajectory by Fuzzy Mixer. Distance to an object measured by the sensors determined how the mobile-robot should react.

Joystick information was combined with sensor information so that:

$$\underline{\mathbf{Output}}\mathbf{(left)} = \underline{\mathbf{Input(left)}} - \underline{\mathbf{F(right)}}$$
$$\underline{\mathbf{Output}}\mathbf{(right)} = \underline{\mathbf{Input(right)}} - \underline{\mathbf{F(left)}}$$

where output *was the resultant mobile-robot controller voltage,* Input *was the joystick voltage, and* F *was the distance function value generated by the sensor system*. They were vector quantities, having two values, one for each wheel (left and right). This was presented to the mobile-robot controller driving the wheels.

"Doorway" was effective at turning the mobile-robot away from the nearest object, slowing the mobile-robot down smoothly as it became closer to objects and centralising the mobile-robot between two objects (such as door frames).

5.1. Algorithms to mix data from the joystick and sensors.

Fuzzy Mixer controlled the relationship between the joystick and sensor system and apportioned control to the joystick or sensor system depending on environmental conditions or the wishes of the tele-operator. Instantaneous relationships could be:

- all joystick, no sensors,
- all sensors, no joystick
- or somewhere in between.

Fuzzy Mixer constantly assessed inputs. Algorithms apportioned control between inputs:

TargetLeft = (((JS0*Aconf[Joysticksector])+((TargetLeft-125)*(255-Aconf[Joysticksector])))/255)+125;

TargetRight = (((JS1*Aconf[Joysticksector])+((TargetRight-125)*(255-Aconf[Joysticksector])))/255)+125;

Where;
TargetLeft/Right = Desired controller voltages.
JS0/1 = Actual joystick values.
Aconf[] = Joystick confidence value.

Algorithms used distance functions to create target values for left and right controller voltages. Distance functions were:

TargetLeft = 2.5*result[1] + 110;

TargetRight = 2.5*result[0] + 110;

Where: result[] = instantaneous range from the sensors.

The result[] was scaled (*2.5) and a constant (110) added. This converted the sensor data to a form compatible with the Target (ADC) data. To recognise the position of the joystick in order to make an assessment of the wishes of the user, the joystick map was divided into sectors: Forward, Turn right, Turn left, Spin right, Spin left, Stop and Back as shown in figure 7.

Factors to increase joystick confidence (**Aconf**[]): Joystick agrees with sensor system,
Joystick held in a steady position (consistent)
Joystick position increased against sensor action.

Factors to decrease joystick confidence: Joystick – sensor conflict,
Joystick not held steady.

Sensor Expert applied a set of algorithms to information generated from sensors. There were seven possible actions:

"Nothing"	meaning carry on under user control,
"Stop"	collision is imminent, stop immediately,
"Slow"	approaching a dangerous situation, slow down,
"Turn left"	a gentle turn left away from an object,
"Spin left"	sharp left turn away,
"Turn right"	a gentle turn right away from an object,
"Spin right"	sharp right turn away.

Sensor information was used to create a Sensor Byte; constructed by considering a list of all possible combinations of sensor array configurations. A Sensor Expert Rule Set[90-92] was then extracted from the mapping. A two to eight bit Sensor Byte was created from the sensor arrays. Each sensor array had two bits to represent the position (or not) of an object within the array. Each pair of bits was expressed as zero to three (2 bit binary):

0 no detection for this array,
1 detection in “far” array element,
2 detection in “middle” element,
3 detection in “near” element.

These numerical operators were used to search Sensor Byte for object configurations so that Sensor Expert could recommend action. Sensor Expert algorithms were based on recognition of patterns in Sensor Byte and are shown in the flow diagram figure 9. The two bit numerical operators were examined in isolation from each other and simple algorithms were developed. The algorithms detected numerical patterns in the Sensor byte that indicated a course of action to be recommended. The recommendations and algorithms are listed in Table 1.

Test	***Action***	***Remarks***
Sensor byte = 0	Nothing	Sensors have not detected anything.
“Centre” = 3	Stop	Object close and in the centre of wheelchair path.
“Right” = 3 ***AND*** “Left” ≠ 3	Spin Left	Object close and near the wheelchair’s right hand front corner.
“Left = 3 ***AND*** “Right ≠ 3	Spin Right	Object close and near the wheelchair’s left hand front corner.
Sensor byte = 1 ***OR*** 2 ***OR*** 6 ***OR*** 18 ***OR*** 22	Turn Left	Object near the wheelchair’s right hand side in the wheelchair’s path.
Sensor byte = 16 ***OR*** 32 ***OR*** 33 ***OR*** 36 ***OR*** 37	Turn Right	Object near the wheelchair’s left hand side in the wheelchair’s path.
ELSE	Slow	Wheelchair’s path blocked but will not collide yet.

Table 1. *The sensor expert algorithm.*

5.2 Testing the initial prototype systems.

The initial prototype system was downloaded to the hardware mounted on the mobile-robot. Systems were tested by driving the mobile-robot in an unstructured but uncluttered environment. Because the early prototype code was inefficient, the response of the mobile-robot system was comparatively slow and inflexible. Figure 10 shows the mobile-robot during a doorway passage test; the mobile-robot had stopped due to a “local minima”.

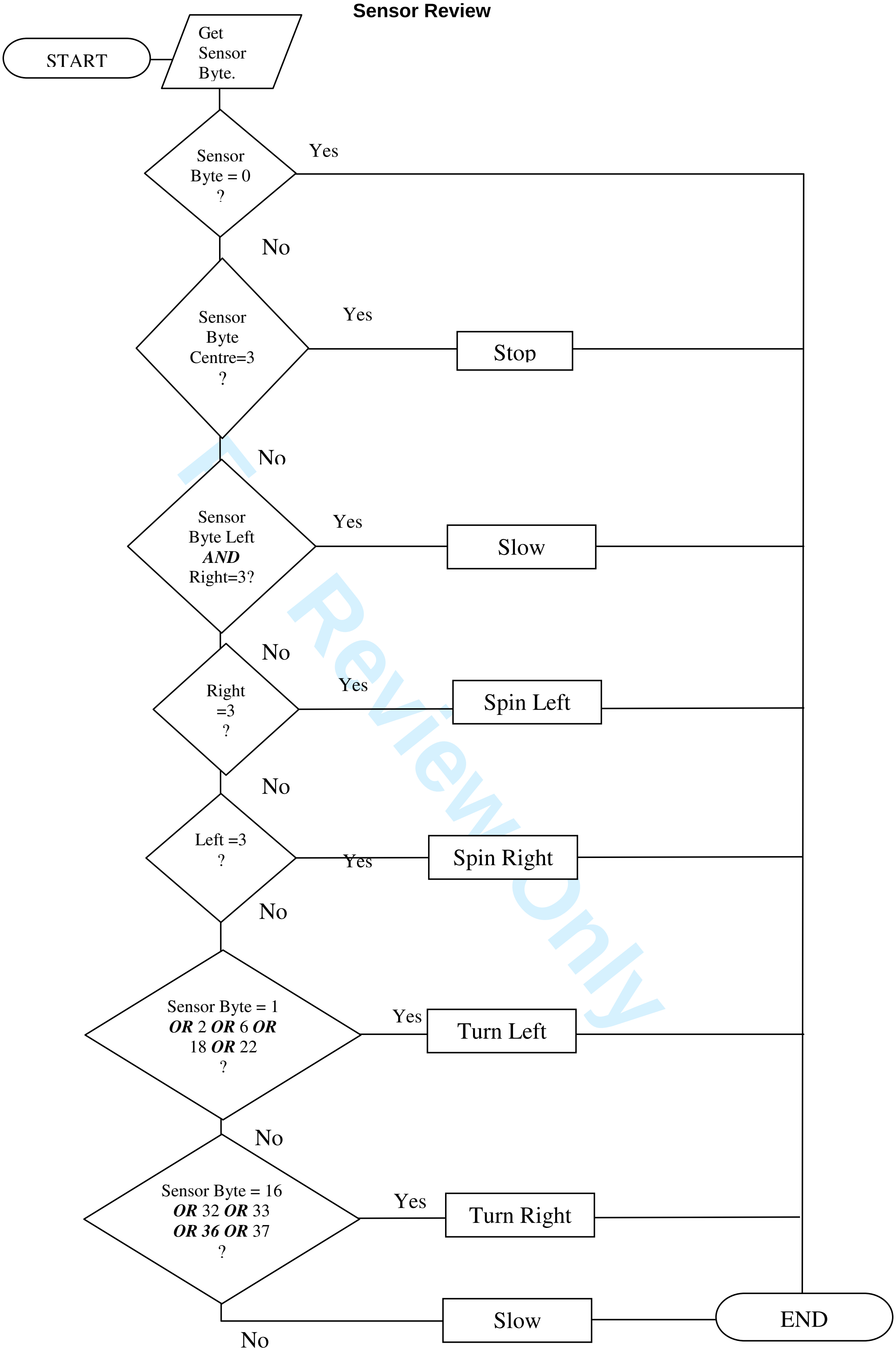

Figure 9. *The sensor expert algorithm flow diagram.*

Figure 10. *The mobile-robot attempting an unassisted doorway passage.*

Distance functions prevented the mobile-robot from passing through the doorway as the sides reached the minimum allowable distance from an object. Distance function algorithms were adjusted to reduce their effect and allow the mobile-robot to move close to (and to touch) an object. This allowed the mobile-robot to move through the doorway. However, system response was reduced which did not allow the mobile-robot to turn away from obstructions. Figure 11 shows the mobile-robot and its path as it attempted to move away from a wall. White tape indicates the mobile-robot path; system response was not sufficient to turn the mobile-robot in time and a collision occurred.

Figure 11. *Video clip of the mobile-robot during an unsuccessful collision avoidance manouvre.*

The prototype expert system structure made response difficult to predict. Many variables acted upon the output and some variables were in opposition. Parts of the system were redundant and were removed to create a new and more efficient system. A more structured approach was taken with a new system; system response was speeded up and control algorithms were simplified.

6. A new mobile-robot expert system.

A new system was created that used the same methods as the prototype but was created to simplify some processes and speed up operation to make the systems more efficient. A simplified Blackboard framework was used as the program structure which was similar to the structure of Hearsay II Blackboard[78]. The program was easier to control in this structure as the main modules communicated with a blackboard (MainCode) and passed important data to the blackboard. Code was written in C (C) or Assembly Code (Assembly) modules:

MainCode (C) controlled program flow and scheduled major events.

GetRange (C) was called when the system required range data from the sensors.

BuildArray (C) created an array of sensor data gathered by GetRange. The array was an expression of the range data as histograms.

Transmit (Assembly) fired a single ultrasonic transmitter. The transmitter channel was defined as a variable passed with the call statement. The length of the transmitter pulse was regulated by an interrupt set by a timer. The timer was activated at the beginning of the pulse and when it timed out, an interrupt was generated. The interrupt handling routine turned the transmitter off.

Joystick (Assembly) read the status of the joystick outputs and the inputs to the mobile-robot controller. The module used an ADC to read the controller and joystick channels.

Checkswitch (Assembly) checked the status of an external toggle switch that prevented the main program from looping. PWM outputs were turned off and the mobile-robot reverted to manual control.

Sensorexpert (C) decoded sensorbyte value to create a course of action for the mobile-robot to take. Boolean tests were performed on sensorbyte to interpret the probable best course of action. Patterns existed in the sensorbyte data and once a pattern had been detected, a recommendation was made.

Forward (C) caused the mobile-robot to drive forwards.

Spinleft (C) caused the mobile-robot to spin left.

Spinright (C) caused the mobile-robot to spin right.

Turnleft (C) caused the mobile-robot to turn left.

Turnright (C) caused the mobile-robot to turn right.

STOP (C) caused the mobile-robot to stop.

Normal (C) drove the mobile-robot without interference from the system.

AdjustControllerVoltage (C) adjusted controller voltages to required voltage.

JSArray (C) created an array from the joystick data. The sampled joystick channels were converted to Polar form. The Polar data was used to convert the numerical values of joystick position to an expression of joystick position by sector.

Conflict (C) considered the sensor recommendation and the joystick position. If the joystick and the sensors agreed on the direction that the mobile-robot should be travelling, there was no requirement for the system to act. If there was a conflict between the sensors and the joystick, the trajectory of the mobile-robot was modified.

GetADCvalue (Assembly) activated the ADC. The value of the ADC channel was passed back to the calling function upon termination of the module.

Code was written in a mixture of high and low level languages and compiled to a single machine level file. This file was loaded into non-volatile memory in a micro-controller. Using an integrated programming environment with access to high level editing and de-bugging tools assisted in the creation of the systems. A modular structure was adopted to simplify program construction and minimise duplication of code. Final structure was similar to a Blackboard type framework. However the similarities were limited by the size of micro-controller memory of the on-board real time systems which ruled out the creation of complicated structures and large amounts of code.

The new re-written algorithms speeded up system response and made the systems more predictable. If the joystick and the sensor expert were indicating “forward”, the system set the trajectory as straight-ahead. The sensor system was still interrogated to determine the distance that the mobile-robot was from the nearest object. The speed of the mobile-robot was reduced as the mobile-robot became close to an object.

SpinLeft or SpinRight turned the mobile-robot. Although controller voltage settings were set to the spin values, the system tended to apply the spin settings for the minimum time required to turn the mobile-robot. The mobile-robot rarely performed a “spin” manoeuvre in this mode as the system settings had returned to a “forward“ mode. The application of a spin manoeuvre for a limited time simulated a user moving the joystick completely to one side to execute a turn. Observing users driving a mobile-robot and their use of a joystick, it appeared common for the joystick to be moved in exaggerated movements (even to perform gentle manoeuvres).

When a joystick was in a “Turn” position, different algorithms were applied to the system, for example an algorithm that prevented the mobile-robot from driving quickly into an obstruction during a TurnRight manouvre.

7. Testing the new system.

The new system was downloaded into the mobile-robot-mounted hardware and tested by driving the mobile-robot in an unstructured but uncluttered environment. System response was fast enough for the mobile-robot to navigate itself along a corridor and align itself with a doorway with the joystick held in a forward position. The mobile-robot path indicated that Sensor Expert was recommending suitable trajectory changes. Figure 12 shows the mobile-robot during a doorway passage test. The white tape trailing the mobile-robot shows the trajectory of the mobile-robot when the joystick was held in the forward position but the systems automatically aligned with the doorway and avoided obstacles and the door posts.

Figure 12. *Video clip showing the mobile-robot negotiating a doorway.*

Figure 13 shows the mobile-robot and its path as it navigated along a corridor with the joystick held in the forward position. Algorithms were effective in suggesting a path that avoided objects and walls.

When operating a joystick controlled vehicle, users tended to use large deflections of the joystick to manoeuvre the vehicle. Controller dynamics and mobile-robot physical dynamics made large deflections of the joystick suitable for accurate control. Small deflections caused sluggish reactions or the inputs were ignored. Large changes in the controller input voltages caused smooth changes to be made to the mobile-robot trajectory.

Human tele-operators are highly sophisticated and capable and the intention was not to replace them but to consider ways of assisting them. Addition of the sensor system assisted mobile-robot tele-operators with navigation. Mobile-robot systems were tested in a laboratory and then in a variety of environments.

Figure 13. *Video clip showing the mobile-robot navigating along a corridor.*

Test runs were limited to 30 metres by the lengths of the umbilical cables used. Cables were up to 15 meters long and that allowed a distance of 15 metres out and back. Tele-operated users quickly learned how the mobile-robot responded and learned to apply control signals early and to estimate stopping distance. A set of tests were conducted to compare the speed of human tele-operation with computer assisted operation using a recently published system[2,14] and using the new systems.

Tests observed system operation under joint computer and human control to measure time taken by:

- human tele-operators by themselves,
- and then again with the assistance of the initial systems,
- and then again with the assistance of the new systems described in this paper.

Figure 14 shows a tele-operator navigating through one of the complicated corridors (with some obstacles) and using the ultrasonic sensor system to assist in steering.

Figure 14 *Tele-operator navigating through one of the complicated corridors using the ultrasonic sensor system to assist in steering.*

Figure 15 shows the scene from a camera mounted on the front of the robot as it moved though a complicated corridor. The mobile-robot is being controlled via an umbilical cable.

Figure 15 *View from a camera mounted on the robot connected via an umbilical cable and moving though a complicated corridor while being assisted by the sensor system.*

A tele-operator in a laboratory was guiding the mobile-robot assisted by the sensor system on the mobile-robot. A researcher with a laboratory digital clock can be seen at the end of the course and another researcher was following the mobile-robot with a stop watch.

The tests were to:

- Gauge the reaction of users to the system and capture potential improvements to the operation and interfacing.
- Observe the operation of the system under joint computer and human control.
- Measure the improvement (if any) of the assistive systems.
- Measure the time taken by human tele-operators by themselves and then again with the assistance of sensor systems and the expert systems.

For each mobile-robot system, tests took place without the sensor system or any automatic assistance. Then tests were repeated with the sensor system engaged and then with assistance provided by the expert system. For each test, an obstacle course was set up in an environment. This meant that the mobile-robot had to deal with the following environments:

LABORATORY

- Just two obstacles and a constant open floor space with vertical walls around the edges.

EMPTY CORRIDORS

- Flat surfaces and sloping surfaces.

- Corridor restricted with vertical walls and doorways.

- Three obstacles offset in a staggered formation.

COMPLICATED CORRIDORS

- Flat surfaces and sloping surfaces.

- Corridor restricted with vertical walls and doorways but with items on the walls (for example radiators and door surrounds).

- Doorways to pass through.

- Obstacles offset in a staggered formation.

ENVIRONMENTS OUTSIDE

- Complex environment with different flat and sloping surfaces.

- Bounded by different vertical and sloping edges.

- People walking through and around the environment.

- Objects in the environment as well as obstacles placed in the environment.

All tele-operators were volunteers and came mainly from staff and students at the University of Portsmouth, and many were undergraduates or research students. Only four of the 21 participants were not students.

Participants received a clear explanation of the study (including risks and benefits of participation) and the University of Portsmouth Ethics Committee approved all experimental procedures. There were 16 males and 5 females. The 21 participants were aged 18-51 years (Mean age: 21).

Tele-operators were human beings and as such they were variable in their performance and so where possible, experiments were repeated several times. For each of the series of tests, the tele-operators were allowed to repeat tests (with or without computers assisting them) as many times as they liked, or hours available allowed. That allowed them to learn the systems and to perform at their best in the time available. Testing was regarded as fun by participants and was popular. Competition was encouraged and people tried to beat their best in each test and tried to beat others at the same tests.

In several cases, some people only managed to complete the tele-operation test or only managed to complete the computer assisted test etc and their results were discarded so that comparisons were only made between the same tele-operators.

The tests used the umbilical cable and compared the speed of human tele-operation with computer assisted operation in a series of standard environments. If a fastest time was achieved by any participant in one set of the tests then they made at least one attempt again at the other test to check that the result was not just due to learning the operation of the systems. If they managed a fastest time at the other test then they made at least one attempt at the original test.

Tests began at a pre-determined and constant start-position (and from a standing start) and timings across the finish lines were measured with both a stopwatch and laboratory clock (an average was taken between the two if there was any discrepancy). Only successful attempts were recorded. That is, any attempt that resulted in a collision was discarded. If too few sets of results were recorded or if there were no pairs of results then results for that environment were discarded. The method of measurement was reproducible.

8. Results.

The mobile-robot successfully negotiated obstacles in various set courses during testing. Assistive computer systems allowed automatic recovery from potential collisions. Some chaotic factors existed. For example, trailing casters could throw the mobile-robot off-line. Variation in floor surface, slope or wheel position could affect results. Delays between sensor systems providing feedback information and controllers passing results of that feedback information to mobile-robot motors could also cause variations.

Typical results is shown in figure 16. New systems were compared to some recently published tele-operator systems[2,6] and to a tele-operator controlling the robot without the aid of any sensors. The vertical scale shows the average best time in seconds to complete various courses for tele-operators without any sensors to assist (*left of each block in figure 16*), using a recently published sensor system[2,6] (*titled Basic computer assistance in the centre of each block*) and the improved system described in this paper (*right of each block*). The horizontal shows the names of the different courses used for testing, which became progressively more complicated as they move from left to right in the figure.

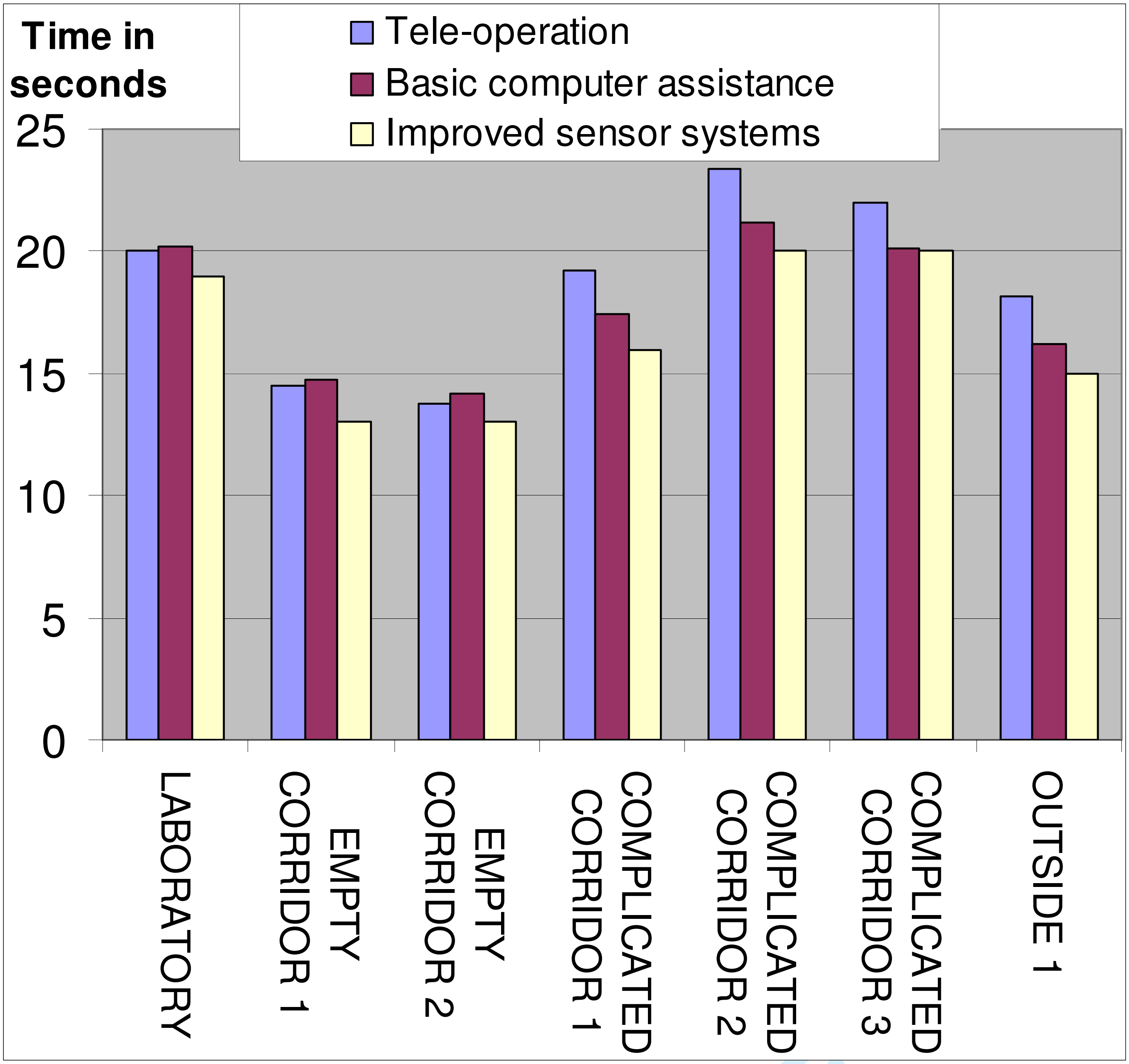

Figure 16 *Results from tests when the tele-operator was watching the mobile-robot. The vertical scale shows the average best time in seconds to complete various courses for tele-operators without any sensors to assist (left of each block), using a recently published sensor system (titled Basic computer assistance in the centre of each block) and using the improved system described in this paper (right of each block). The horizontal axis shows the names of the different courses used for testing.*

Tests were completed when the tele-operator was watching the mobile robot (figure 14) rather than observing the view ahead on a computer screen (figure 15). The new system can be seen performing faster (on average) than a recently published system in every test.

In addition, in simple environments (laboratory and empty corridor), tele-operators completed tasks more quickly without any aid from computer and sensor systems. In more complicated environments (complicated corridor and outside), tele-operators completed tasks more quickly with the aid of computer and sensor systems. The form of results was repeated when a camera was mounted onto the mobile-robot. As the environments became more complicated (or the gaps were made smaller) then the human operators found it more difficult to judge the width of the gaps or the successful trajectory of the mobile-robot to pass through those gaps. The human tele-operators often had to slow the robot or stop the mobile-robot and reverse it to avoid collision. When environments became more complicated, then human tele-operators consistently performed better with assistance from sensors and computer systems. Items on walls (for example radiators and door surrounds) sometimes slowed mobile-robots as sensors detected them, whereas human tele-operators often ignored them. Overall the assisted tasks were performed more quickly.

Different surfaces, slopes and boundaries tended to turn robots and sensors became most useful in steering in those cases.

The new automated systems managed to consistently correct the trajectory of the mobile-robot to a repeatable standard and out performed a recently published system.

9. Discussion and conclusions

The Student's t-test was used to compare the means of the samples shown in figure 16. From each sample, the average (mean) $\bar{x}$ was calculated with a measure of dispersion (range of variation) of data around the sample mean (variance S^2) and thence the standard deviation (S). Having obtained those values, they were then used to estimate population mean μ and variance σ^2. Each individual set of tests were not necessarily statistically significant so that caution was required before generalising the results.

Because pairs of tests and results took place, then it was possible to use a paired-samples statistical test. Results were arranged into two sets of replicate data; pairs of results with and without sensor assistance. The paired samples test was used because people (tele-operators) were inherently variable. Pairing removed much of that random variability. When results were analysed using a paired-samples statistical test then results were statistically significant. The paired-samples statistical test shows the use without a sensor system and with a sensor system to be significantly different at $p < 0.05$ (95% probability that this result would not occur by chance alone) and the new systems described in this paper were significantly better than recently published systems at $p < 0.05$ (95% probability that this result would not occur by chance alone).

The new system performed every test faster on average than the recently published systems.

More effective control of the mobile-robot could be achieved if more information about the environment was available, especially in tight spaces. Infra-red could be a simple and suitable medium for a short-range sensor system. With more information available for analysis, the central processor could have tighter control of robot movements.

Systems on the mobile-robot could also be used to monitor the user in terms of driving skill. For example, the tele-operator could be assessed by the number of near misses or collisions occurring over a period of time and the way in which the joystick was used could be monitored.

More control of the power outputs to the motors would be useful. The system needs to take more direct control of the output for fine manoeuvring. A new controller would be needed in order that the algorithms could be closely integrated with the control algorithms required for normal operation of the mobile-robot.

The position of the joystick was the only indication of the intentions of the tele-operator. Any extension of this work should further analyse user intent from actions exerted on any input device. Joysticks could be replaced by haptic devices so that tele-operators could feel a back-force generated by the signal from the sensor sub-system. That way distance feedback could be provided through the joystick.

A spring loaded system could be included for the umbilical cable so that the cable could be retracted when the mobile root was returning to the start.

References

1. DA Sanders and AD Hudson, “A specific blackboard expert system to simulate and automate the design of high recirculation airlift reactors” Mathematics and computers in simulation 53 (1-2), 41-65 (2000).

2. D Sanders “Comparing speed to complete progressively more difficult mobile robot paths between human tele-operators and humans with sensor-systems to assist” Assembly Automation 29 (3), 230-248 (2009).

3. MHF Rahiman, Z Zakaria and RA Rahim et al “Ultrasonic tomography imaging simulation of two-phase homogeneous flow” Sensor Review 29 (3), 266-276 (2009).

4. IJ Stott, DA Sanders and MJ Goodwin “A software algorithm for the intelligent mixing of inputs to a tele-operated vehicle” Journal of systems architecture 43 (1-5), 67-72 (1997).

5. D Sanders “Controlling the direction of “walkie” type forklifts and pallet jacks on sloping ground” Assembly Automation 28 (4), 317-324 (2008).

6. Sanders, DA; Graham-Jones, J; Gegov, A “Improving ability of tele-operators to complete progressively more difficult mobile robot paths using simple expert systems and ultrasonic sensors”. Ind Rob – An Int Jrnl Vol 37 (5), 431-440 (2010).

7. T Akinfiev, M Armada and S Nabulsi “Climbing cleaning robot for vertical surfaces” Ind Rob: An int’ jnl, 36 Issue: 4 Pages: 352-357 (2009).

8. SM Ha, JB Sheng and K Takase “Obstacle Recognition for a Service Mobile Robot Based on RFID with Multi-Antenna and Stereo Vision” IEEE Int Conference on Information & Automation 1-4, 125-130 (2008).

9. BL Luk, DS Cooke, S Galt, AA Collie and S Chen S “Intelligent legged climbing service robot for remote maintenance applications in hazardous environments”, Jrnl Rob & Autonomous Syst (53/2), 142-52 (2005).

10. D Sands “Cost effective robotics in the nuclear industry” Ind Rob: An Int’ Jnl. Vol 33, No 3, 170 - 173 . ISSN: 0143-991X (2006).

11. BL Luk, AA Collie, DS Cooke and S Chen “Walking and climbing service robots for safety inspection of nuclear reactor pressure vessels” Measurement & Control 39 (2), 43-47 (2006).

12. BL Luk, KP Liu, AA Collie, DS Cooke and S Chen “Tele-operated climbing and mobile service robots for remote inspection and maintenance in nuclear industry” Ind Rob: An Int’ Jnl. Vol 33, No 3, 194 – 204. ISSN: 0143-991X (2006).

13. R Molfino, M Zoppi and L Rimassa “Rescue robot module with sliding membrane locomotion” Ind Rob: An int’ jnl, 35 (3), pp 211–216 (2007).

14. DA Sanders “Comparing ability to complete simple tele-operated rescue or maintenance mobile robot tasks with and without a sensor system” Sensor Review, Volume 30, Issue 1, 40-50 (2010).

15. Z Wang and Gu Hong “A review of locomotion mechanisms of urban search and rescue robot Ind Rob: An int’ jnl, 34 (5), 400–411 (2007).

16. DA Carnegie and JC Cordes “The mechanical design and construction of a mobile outdoor multi-terrain robot” Proc’ IMechE Part B – Jnl Eng Man, 218 (11), 1563-1575 (2004).

17. DA Sanders “Analysis of the effects of time delay on the tele-operation of a mobile robot in various modes of operation”. Industrial Robot: An International Journal, Vol. 36 (6), 570 – 584 (2009).

18. L Rimassa, M Zoppi and R Molfino, “A modular serpentine rescue robot with climbing ability” Ind Rob: An int’ jnl, 36 Issue: 4, 370-376 (2009).

19. A Milella, G Cicirelli and A Distante “RFID-assisted mobile robot system for mapping and surveillance of indoor environments” Ind Rob: An int’ jnl, 35 (2), 143–152 (2008).

20 Anon “Mars rovers just keep on going” Ind Rob: An int’ jnl, 35 Issue: 1, 80-80 (2008).

21. C Hillenbrand, D Schmidt and K Berns “CROMSCI: development of a climbing robot with negative pressure adhesion for inspections” Ind Rob: An int’ jnl, 35 Issue: 3, 228-237 (2008).

22. DA Sanders, G Lambert, J Graham-Jones, GE Tewkesbury, S Onuh, D Ndzi and C Ross. “A robotic welding system using image processing techniques and a CAD model to provide information to a multi-intelligent decision module” Assembly Automation, Vol. 30 (4), 323 - 332 (2010).

23. TP Sattar, HL Rodriguez and B Bridge “Climbing ring robot for inspection of offshore wind turbines” Ind Rob: An int’ jnl, 36 (4), 326-330 (2009).

24. DA Sanders “Recognizing shipbuilding parts using artificial neural networks and Fourier descriptors” Proceedings of the Institution of Mechanical Engineers Part B 223 (3), 337-342 (2009).

25. C Distante, G Indiveri and G Reina “An application of mobile robotics for olfactory monitoring of hazardous industrial sites” Ind Rob: An int’ jnl, 36 (1), 51-59 (2009).

26. JA Cobano, R Ponticelli and RG de Santos “Mobile robotic system for detection and location of antipersonnel land mines: field tests” Ind Rob: An int’ jnl, 35 (6), 520-527 (2008).

27. EF Kececi “Completely mechanical quick changeable joints for multipurpose explosive ordnance disposal robots” Robotica 27, 555-565 (2009).

28. H Omori, T Nakamura and T Yada “An underground explorer robot based on peristaltic crawling of earthworms” Ind Rob: An int’ jnl, 36 (4), 358-364 (2009).

29. YC Sun and CC Cheah “Adaptive control schemes for autonomous underwater vehicle” Robotica 27, 119-129 (2009).

30. E Honary, F McQuade, R Ward et al “Robotic experiments with cooperative Aerobots and underwater swarms” Robotica 27, 37-49 (2009).

31. C Lok “Flying robot - Thirty-gram aircraft steers itself” Technoogy review 108 (9), 83-84 (2005).

32. R Bloss “By air, land and sea, the unmanned vehicles are coming” Ind Rob 34 (1) 12–16 (2007).

33. R Bloss “Latest unmanned vehicle show features both innovative new vehicles and miniaturization” Ind Rob: An int’ jnl, 36 (1), 13-18 (2009).

34. DA Sanders and IJ Stott “Analysis of failure rates with a tele-operated mobile robot between a human tele-operator and a human with a sensor system to assist” Robotica ROB-REG-08-0199

35. I Waheed and R Fotouhi “Trajectory and temporal planning of a wheeled mobile robot on an uneven surface” Robotica 27, 481-498 (2009).

36. G Dip, V Prahlad and PD Kien “Genetic algorithm-based optimal bipedal walking gait synthesis considering tradeoff between stability margin and speed” Robotica 27, 355-365 (2009).

37. N Liu, JF Li and TS Wang “The effects of parameter variation on the gaits of passive walking models: simulations and experiments” Robotica 27, 511-528 (2009).

38. KJ Waldron “The state of the art of climbing and walking robots” Ind Rob: An int’ jnl, 36 (4), 313-313 (2009)

39. RS Inoue, AAG Siqueira and MH Terra “Experimental results on the nonlinear H-infinity control via quasi-LPV representation and game theory for wheeled mobile robots. Robotica 27, 547-553 (2009).

40. DA Sanders, M Langner and GE Tewkesbury “Improving wheelchair-driving using a sensor system to control wheelchair-veer and variable-switches as an alternative to digital-switches or joysticks” Industrial Robot: An International Journal, Vol. 37 (2), 157 - 167 (2010).

41. T Nakamura and K Satoh “Development of an omni-directional mobile robot using traveling waves based on snail locomotion”. Industrial Robot: An International Journal, 35 (3), 206–210 (2008).

42. JM Toibero, F Roberti and R Carelli “Stable contour-following control of wheeled mobile robots” Robotica 27, 1-12 (2009).

43. AI Cahyadi and Y Yamamoto "Teleoperated 3-DOF micromanipulation system with force feedback capability: design and experiments". Industrial Robot: An International Journal, 35 (4), 337-346 (2008).

44. M Harada, T Nagatomo, S Yanagihara and M Tachibana "A study on co-operative motion planning of a dual manipulator system for measuring radioactivity". Industrial Robot: An International Journal, 35 (6), pp 541–548 (2008).

45. S Erwin-Wright, D Sanders and S Chen "Predicting terrain contours using a feed-forward neural network" Engineering Applications of AI 16 (5-6), 465-472 (2003).

46. I Stott and D Sanders "New powered wheelchair systems for the rehabilitation of some severely disabled users" Int Jrnl of Rehabilitation Research 23 (3), 149-153 (2000).

47 D Sanders and I Stott "A new prototype intelligent mobility system to assist powered wheelchair users" Ind Rob: An int' jnl, 26 Issue: 6, 466-475 (1999)

48. DA Sanders and A Baldwin A "X-by-wire technology" Total Vehicle Technology: Challenging current thinking, pp 3-12 (2001).

49. DA Sanders and GE Tewkesbury "A pointer device for TFT display screens that determines position by detecting colours on the display using a colour sensor and an ANN" Displays 30 (2), 84-96 (2009).

50. DA Sanders, SD Urwin-Wright, GE Tewkesbury et al "Pointer device for thin-film transistor and cathode ray tube computer screens" Electronics Letters 41 (16), 894-896 (2005).

51. S Chester, GE Tewkesbury and D Sanders et al "New electronic multi-media assessment system" Web Information Systems and Technologies 1, 414-420 (2007).

52. I Stott and D Sanders "The use of virtual reality to train powered wheelchair users and test new wheelchair systems" Int Jrnl of Rehabilitation Research 23 (4), 321-326 (2000).

53. S Urwin-Wright, D Sanders and S Chen "Terrain prediction for an eight-legged robot" Jrnl of robotic systems 19 (2), 91-98 (2002).

54. A Eisinberg, A Menciassi, P Dario, J Seyfried, R Estana and W Woern "Teleoperated assembly of a micro-lens system by means of a micro-manipulation workstation" Assembly Automation, 27 (2), pp 123–133 (2007).

55. H Zhang, J Zhang, W Wang, R Liu R and G Zong "A series of pneumatic glass-wall cleaning robots for high-rise buildings" Ind Rob: An Int' Jnl. Vol 34, No 2 Page: 150 – 160 (2007).

56. DA Sanders, G Lambert and L Pevy "Pre-locating corners in images in order to improve the extraction of Fourier descriptors and subsequent recognition of shipbuilding parts" IMechE Part B, Volume 223 (9), 1217-1223 (2009).

57. MJ Goodwin, DA Sanders and GA Poland et al "Navigational assistance for disabled wheelchair-users" Jrnl of Systems Architecture 43 (1-5), 73-79 (1997).

58. DA Sanders "The modification of pre-planned manipulator paths to improve the gross motions associated with the pick-and-place task Robotica 13 (Part 1), 77-85 (1995).

59. J Larsson, M Broxvall and A Saffiotti "Laser-based corridor detection for reactive Navigation" Ind Rob: An int' jnl, 35 (1), pp 69–79 (2008).

60. S Lee "Use of infrared landmark zones for mobile robot localization. Ind Rob 35 (2), pp 153–159 (2008).

61. S Lee "Use of infrared light reflecting landmarks for localization" Ind Rob 36 (2), 138-145 (2009).

62. JX Yu, ZX Cai, and ZH Duan “Dead reckoning of mobile robot in complex terrain based on proprioceptive sensors”. 7th Int’ Conf’ on Machine Learning & Cybernetics, Peoples R China, Vols 1-7, pp 1930-1935 (2008).

63. O Horn and M Kreutner “Smart wheelchair perception using odometry, ultrasound sensors, and camera” Robotica 27, 303-310 (2009).

64. V Milanes, JE Naranjo and C Gonzalez et al “Autonomous vehicle based in cooperative GPS and inertial systems” Robotica 26, 627-633 (2008).

65 D Lim, S Lee, and D Cho “Design of an assisted GPS receiver and its performance analysis”, IEEE Int Symposium on Circuits and Systems, pp. 1742-1745 (2007).

66. R Bloss “Vision and robotics team up at the 2007 Show”. Ind Rob: An int’ jnl 35 (1), 19-26 (2008).

67. DA Sanders, YC Tan, I Rogers and GE Tewkesbury “An expert system for automatic design-for-assembly” Assembly Automation, Vol. 29 (4), 378 - 388 (2009).

68. D Sanders “Environmental sensors and networks of sensors” Sensor Review 28 (4) 273-274 (2008).

69. D Hopper “The long perspective for robotic vision” Assembly Automation 29 (2) 122-126 (2009).

70. D Sanders “Perception in robotics” Ind Rob: An int’ jnl 26 (2), 90-92 (1999).

71. D Sanders “System Specification 2”. Microprocessing and microprogramming 38 (1-5), 833-833 (1993)

72. DA Sanders, P Harris and E Mazharsolook “Image modellling in real-time using spheres and simple polyhedra “4th Int Conf on Image processing and its applications 354, 433-436 (1992).

73. D Sanders “Progress in machine intelligence” Ind Rob –an int’ jnl: 35 (6), 485-487 (2008).

74. DA Sanders “Force sensing” Ind Rob –an int’ jnl: 34 (4), pp 268-268 (2007).

75. H Shao and K Nonami “Bilateral control of tele-hand system with neuro-fuzzy scheme” Ind Rob: An Int’ Jnl. Vol 33, No 3, pp 216 – 227. ISSN: 0143-991X (2006).

76. R Rocha, A Cunha, J Varandas and J Dias “Towards a new mobility concept for cities: architecture and programming of semi-autonomous electric vehicles” Ind Rob: An Int’ Jnl. Vol 34, No 2, pp 142 – 149 (2007).

77. JYC Chen, EC Haas and MJ Barnes “Human Performance Issues and User Interface Design for Teleoperated Robots” IEEE Trans on Systems, Man & Cybernetics C: Apps & Rev 37, 1231-1245 (2007).

78. DA Sanders, AD Hudson and GE Tewkesbury “ Automating the design of high-recirculation airlift reactors using a blackboard framework” Expert systems with applications 18 (3), 231-245 (2000).

79. GE Tewkesbury and DA Sanders “A new simulation based robot command library applied to three robots” Jnl of Robotic Systems 16 (8), pp 461-469 (1999).

80. GE Tewkesbury and DA Sanders “A new robot command library which includes simulation” Jnl of Robotic Systems 26 (1), pp 39-48 (1999).

81. YC Chang &Y Yamamoto “On-line path planning strategy integrated with collision and dead-lock avoidance schemes for wheeled mobile robot in indoor environments” Ind Rob 35 (5), pp 421–434 (2008).

82. O Khatib "Real-time obstacle avoidance for manipulators and mobile robots", Proc of IEEE Int. Conf. on Robotics and Automation, Vol. 5 No.1, pp.90-98 (1986).

83. AD Hudson, DA Sanders and H Golding et al “Aspects of an expert design system for the wastewater treatment industry” Jrnl of systems architecture 43 (1-5), 59-65 (1997).

84. DA Sanders “Introducing AI into MEMS can lead us to brain-computer interfaces and super-human intelligence” (invited viewpoint review paper) Assembly Automation Vol. 29 (4), 309-312 (2009).

85. D Sanders “Ambient-intelligence, rapid-prototyping and where real people might fit into factories of the future” Assembly Automation 29 (3), 205-208 (2009).

86. DA Sanders, BP Haynes and GE Tewkesbury et al “The addition of neural networks to the inner feedback path in order to improve on the use of pre-trained feed forward estimators” Mathematics and computers in simulation 41 (5-6), 461-472 (1996).

87. DA Sanders “Real time geometric modelling using models in an actuator space and cartesain space” Jrnl of robotic systems 12 (1), 19-28 (1995).

88. F Fahimi, C Nataraj and H Ashrafiuon “Real-time obstacle avoidance for multiple mobile robots” Robotica 27, 189-198 (2009).

89. G Tewkesbury and D Sanders “The automatic programming of production machinery for de-flashing plastic parts” Advances in manufacturing technology VIII, 279-283 (1994).

90. J Bergasa –suso, D Sanders and G Tewkesbury “Intelligent browser-based systems to assist Internet users” IEEE Transactions (E) Volume: 48 (4), 580-585 (2005).

91. S Chester, G Tewkesbury and D Sanders et al “New electronic multi-media assessment system” WEBIST 2006: Proc’ of 2nd International Conference on Web Information Systems and Technologies, 320-324 (2006).

92. GE Tewkesbury and DA Sanders “The use of distributed intelligence within advanced production machinery for design applications” Total vehicle technology: Challenging current thinking 255-262 (2001).