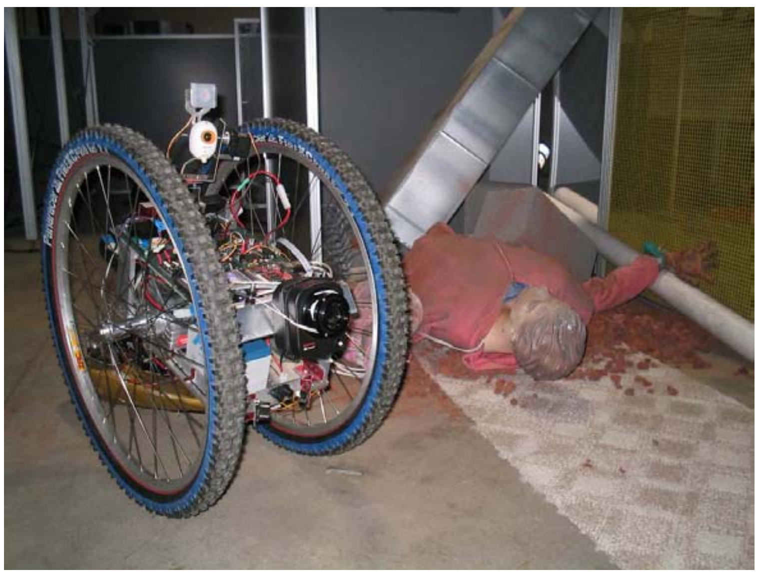
INSTITUT DE PRODUCTION ROBOTIQUE (IPR)
LSRO2 – VRAI-Group
Diploma Work 2003/2004
Supervisors: Reymond Clavel, Illah Nourbakhsh
Assistants: Dr Charles Baur, Terry Fong

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE



Human Detection for Robotic Urban Search and Rescue

Steve Burion
Microtechnique
February 2004

Table of contents

1 INTRODUCTION

1.1 What is USAR?

There is many different kind of catastrophe in natural and man-made disaster: earthquake, flooding, hurricane and they cause different disaster area like collapsed building, landslide or crater. During these emergency situations, and specially in urban disaster, many different people are deployed (policeman, fire fighters and medical assistance). They need to cooperate to save lives, protect structural infrastructure, and evacuate victims to safety.
In these situations, human rescuers must make quick decisions under stress, and try to get victims to safety often at their own risk. They must gather determine the location and status of victims and the stability of the structures as quickly as possible so that medics and firefighters can enter the disaster area and save victims.
All of these tasks are performed mostly by human and trained dogs, often in very dangerous and risky situations. This is why since some years, mobile robots have been proposed to help them and to perform tasks that neither humans dogs nor existing tools can do. For this project, we will focused only on robots which will work in a disaster environment of man made structure, like collapsed buildings. They are called Urban Search And Rescue (USAR) robots.

There are several teams working on USAR robotics. Currently, Carnegie Mellon University is being founded by the National Science Foundation to investigate the use of semi-autonomous robots for urban search and rescue. These robots will assist firemen, police, and disaster agencies with reconnaissance, site evaluation, and human detection. The goal of this research is to develop mobile robot hardware (mechatronics and sensors) and software systems (user interfaces and navigation, planning and coordination module) to support these tasks. Compare to the other projects, these robots should have sufficient autonomy to maximize limited capabilities and attention of the human operator.

1.2 Objectives of the project

The robot built by Carnegie Mellon researchers on the USAR team is capable of navigating the difficult terrain of a disaster site but lacks sensors for victim detection. The contribution of this work is to provide a sensor suite for human detection in the urban disaster environment. The philosophy of the USAR project at Carnegie Mellon is that the robot team should be low cost, semi-autonomous, heterogeneous, and work together under a human coordinator.
In line with this philosophy, one of the most challenging parts of this project is to find a lightweight and low cost solution that can fit on the existing USAR robot.
Conditions in a disaster area are extreme with many unknown parameters. Victims may be covered in debris, trapped in voids, or entombed, making it difficult to find them and determine their state of health. This is why it will be important to choose a set of different sensors which are complementary and able to operate in these conditions.

This project consists of three main parts. The first step will be to determine the state of the art in USAR robotics, with special emphasis on sensors for victim detection.

Next, a set of appropriate and complementary sensors will be selected in accordance with chosen criteria, mainly that the sensors be low-cost and lightweight.
The selected sensors will be integrated with the USAR robot. This involved developing hardware and low level data acquisition software solutions. Tests will be used to determine the robustness, limitations, and accuracy of each sensor and this data will be used to develop a comprehensive system that fuses the information from all the sensors to determine the location and probability of human presence.
Finally, a graphical user interface will be developed to provide useful information back to the human operator while allowing the user the power to interact with individual sensors.

2 STATE OF THE ART IN USAR

Currently, search and rescue robotics is a large and active field in both academia and industry. Robots are ideal for when it is dangerous to send in human rescue workers including environments resulting for earthquakes, fires, avalanches, nuclear catastrophes, mine fields, floods, etc.
USAR (Urban Search and Rescue) efforts at Carnegie Mellon University focus on urban disaster sites. That is why this chapter will be only on the state of the art for robots working in that kind of environment.

2.1 University Research

Many universities are purchasing urban search and rescue robotics, especially in the USA, Japan and Europe. Three of the most advanced research teams are presented below.

- CRASAR (Centre for Robot-Assisted Search and Rescue): University of South Florida
  That is maybe the most advanced project for a search and rescue robot. According to [19] and [20], the aim of this robot is to help the first-aid workers by giving them a picture of a place that they can not reach. So they can see the environment, see if there are victims, or something else. This robot was used for first time in real conditions on 11th September 2001 in the World Trade Center disaster. A good overview of the results about using this robot in these conditions can be found in [4].

Figure 1 : Packbot robot after the World Trade Center Crash

  This robot use different sensor like millimeter wave radar for measuring distance, a camera for vision and a forward-looking infrared camera (FLIR) for the human heat detection. Another new sensor has just been implemented on this robot recently. This is a SpO2 sensor to measure the oxygen quantity in blood. Then the user will be able to know if the victim is still alive. This robot is totally operated with a human. It sends its information to the user to allow him to take decision and to drive the robot in an interesting place.

- Utility Vehicle for Search UVS: Kobe University, Japan
  Researchers from Japan's Kobe University [21] have several homogeneous small robots that can link together to form a large robot in order to climb large obstacles. Most of their research appears to be focused on large scale coordination efforts such as disaster relief after the Hanshin-Awaji Earthquake that hit Kobe City in 1995. They have also developed

a simulator for RoboCup-Rescue Simulation league emphasizes coordination rather than victim detection and issues individual robots must solve.

Figure 2 : Utility Vehicle for Search (Japan)

- Kohga: University of Tokyo
  According to [22], they are developing snake robots for exploration of small spaces in disaster sites. They are designed so that they can be dismantled into many parts for transportation to the site though their mobility in a disaster area is somewhat limited. The snake robots are equipped only with a camera and microphone and do not seek to detect victims autonomously.

Figure 3 : Different robot project in the University of Tokyo

2.2 Industry Efforts

Although many companies market mobile robots, sensors, and electronics useful for search and rescue work, only few companies have commercially available robots that designed for USAR. NASA and the United States Department of Defense also have ongoing efforts with various universities to develop intelligent robots with a variety of sensors. The most advanced projects in industry are listed below.

- iRobot [24]
  Financed by the US government, they develop some robots which replace the human in several interventions, when it is too dangerous or to small to enter. They have many different projects. Some robots are used exploration in urban or outside places. The

Packbot robot for example has an aluminium body and it has different sensors like, cameras, microphones, laser range finders, sonars and IR sensors.

Figure 4 : Robots from iRobot

Another project is the Deployer, which has a team of little robots that it can place where it wants. This robot has the calculation power of a big robot and the flexibility and mobility of little ones.
In the future, these robots will help victims and give them the first aid rescue, like give morphine using an auto-injector or information via a bilateral radio.

- Inuktun [25]
  Their robots are generally used in wet environment or flood, but they can have some applications in urban search and rescue as well. They are specialized in moving on little places like a tube. These robots are equipped with cameras. One of them can modify its shape (see Figure 5) to pass through complex environments or to climb some obstacles.

Figure 5 : Robots from Inuktun

- Nasa
  In the NASA's Jet Propulsion laboratory, they have done some research for an urban robot with different sensors mounted on it (stereo camera, IR distance sensor, GPS). More information can be found in [7] and [23].

2.3 Robot competition

As explained in [30], the goal of the urban search and rescue robot competitions is to increase awareness of the challenges involved in search and rescue applications, provide objective evaluation of robotic implementations in representative environments, and promote collaboration between researchers. Robots show their capabilities in mobility, sensory perception, planning, mapping, and practical operator interfaces, while searching for simulated victims in unstructured environments.

A (Czech Rpublic Team) B (Iran Team) C (Iran team 2)

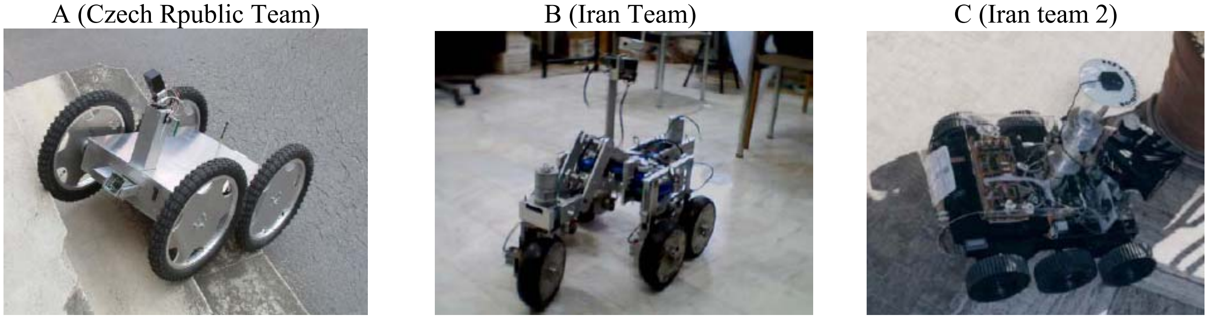


Figure 6 : The three winning robots at the USAR Robocup competition 2003

In Figure 6, there are the three winning robots of the 2003 USAR Robocup edition. In addition of localization sensors (like GPS), the robots were equipped with sensors for victim detection. All three robots used vision but in different configurations. The team (c) used omni directional vision with a spherical mirror and the other a pan tilt head. Microphone, non-touch infrared thermometer and ultrasonic transceiver were the other sensors connected on these robots. Team (c) used also visual motion detection to identify victims too. More information about these robots can be found in [30].

3 CHOICE OF SENSORS

The aim of this project is to detect a victim in an urban disaster environment. This is a very difficult task especially in the unstructured environment of a collapsed building. The physical parameters of a victim that we can detect using different kinds of sensors are:

- voice
- temperature
- clothing texture
- motion
- scent
- skin color
- shape

3.1 Sensors currently available

There are many sensors commercially available for the human detection and all have their advantages and disadvantages. In the following part the most common sensors available for human detection are described. The most important condition for this project is to find a low-cost solution to put on the USAR robot.

3.1.1 Vision

Vision is the most used sense for detection of human presence. It has made its proof with humans, so it is one of the most effective. There exist different kinds of vision sensors:

- **Linear camera:** is the cheapest vision sensor, but it is not very effective to detect a human presence. We need more than one line of pixels to detect a human.
- **Color camera:** exists in many different versions, like low cost USB cameras with CMOS sensors (for example a webcam used on a personal computer) or more expensive cameras with CCD sensors and good optics used in professional systems. It is very sensitive to lighting conditions, especially in outdoor environments and it is very effective for the discrimination between human and non human presence. However there is a significant disadvantage: the image processing is computationally expensive. Currently, there is a lot of research about people detection in a picture.
- **Stereo vision:** uses two color cameras to have supplementary information. The difference between the two images gives the depth information. This is called the disparity and it can be use to compute the distance of the object from the camera. This is rather expensive equipment (usually two CCD cameras). It has the same properties as a single camera and the same advantages to detect a human but it is more computationally expensive because it provides supplementary information.
- **Infrared camera**: may be the best solution to make the discrimination between human and non human presence. This is the most commonly sensor used for this application. This sensor gives a picture of the environment heat which is very useful in human detection. Although infrared camera is the most expensive vision sensor, they seem to be essential to a robust and efficient solution for human finding. The most interesting products can be found in different web sites listed in [27].

3.1.2 Heat sensor

- **Pyroelectric**: As explained in [8] and [28] Pyroelectric sensors are designed specifically for human detection. This sensor is made of a crystalline material that generates a surface electric charge when exposed to heat in the form of infrared radiation. It is calibrated to be sensitive to human heat wavelength (8 - 14 µm). These sensors are very sensitive, cheap and robust. They are composed of two infrared sensors, so they detect humans only if the human or the sensor is moving. They are generally used in alarm systems or remote light switches. They have a better field of view than thermopile.
- **Thermopiles** are like contact-less thermometers and it return the average temperature in the field of view. Some of thermopile are fully integrated and can give a precise value of the temperature. Therefore it can be used for human detection.

3.1.3 Microphone

Sound in the audible spectrum is another human characteristic that we can detect and measure. In a disaster area, it is very difficult to filter a human sound like a shout. However, to find a survivor, the rescue people sometimes stop all activity to listen to a shouting person. It is possible to hear some people in this condition. Microphones are also a low cost sensor but not very easy to interface to process its data.

3.1.4 Laser rangefinder

Laser rangefinders have a very high resolution, but are quite expensive. Some of them allow three dimensional scans, but produce too much information to be processed on a small computer. It gives the distance between an obstacle and the robot but it can not make the distinction between a human or non-human presence.

3.1.5 Ultrasonic sensors

This is a common sensor used in mobile robotics because of its low price and ease of use. It is used for basic target classification in term of surface, or basic shapes but has many disadvantages. It is sensitive to air condition, and there are some problems with the echo location when the place or the target has a complex shape. With this sensor only, it is almost impossible to make the distinction between human or non-human presence.
Now, it is more and more replaced by the radar, which is more robust and depends less on the environment conditions.

3.1.6 Radar

Millimeter-wave radar is especially efficient for long distance motion measurement. They are working on very high frequency (5 – 24 GHz) and they can operate through smoke, dust, fog or rain. As mentioned in [34], some of this device are small and low power radar, and can detect motion up to 6 m. The main disadvantage of these sensors is the price. They are quite expensive.

3.1.7 CO_2 sensors

These sensor allows to detect the carbon dioxide emission, and then the breathing cycle of a victim. It is possible to determine if he is still alive. According to [10], the response time of a CO_2 sensor is very slow. The sensor has to be very close to the victim to have useful data because it is very directional. Indeed, it depends much on the air conditions like humidity, temperature, wind, and dust. So it is quite difficult to use it in a disaster area. Moreover, it is not easy to buy only the sensor. It is often sold with outside monitoring equipment (screen, case) because it is usually used in the medical field. Finally, it is a quite expensive device and bulky with the monitoring as explained in [10] and [29].

3.1.8 SpO_2 sensor

Like the CO_2 sensor, SpO_2 is used to determine if somebody is still alive. It needs direct contact with the person. SpO_2 sensing of blood oxygen content requires only the pressing of a small, cool, red light on any bare inch of skin. It is not a common sensor in robotics. It is only used in the CRASAR project [20]. This is not a commercially available sensor.

3.1.9 Summary of the available sensors

	Technology	Feature detected	External size	Cost	human/non human distinction	strengths	Weakness
Linear camera	CCD/CMOS EM 0.4 – 1.1 µm	vision	-	-	-	price	low resolution
USB camera	CCD/CMOS EM 0.4 – 1.1 µm	vision	+	+	++	cost/ performance	resolution
Stereo vision	CCD/CMOS EM 0.4 – 1.1 µm	vision/ distance	++	++	++	Vision + distance info.	computation ally expensive
Infrared camera	CCD/CMOS EM 7 – 14 µm	heat	++	+++	+++	human distinction	price
Pyroelectric	crystalline sensor EM 7 – 14 µm	body heat	-	-	++	price, human distinction	only motion detection
Thermopile	thermocouple EM 5.5 – 13 µm -25°C – 100 °C	heat	-	-	+	price	only average temperature
Microphone	membrane SW 100Hz – 16 kHz	Sound	-	-	+	price	noise sensitivity
Laser rangefinder	time of flight/triangul. EM 620 - 820 nm	distance	++	+++	-	precision of measure	price
Ultrasonic sensor	membrane SW 130 – 290 kHz	distance	-	-	-	price	echo sensitivity
Radar	time of flight EM 5 – 24 GHz	distance	+	+++	-	precision with big range	price
CO_2 sensor	Electro-chemical	gas	++	++	++	human distinction	too directional
SpO2	light absorption (650nm and 805nm)	blood oxygen/pul se rate	-	N/A	+++	human distinction	not available for robotics

EM = electromagnetic waves; SW = sound waves

Table 1: summary table of all the listed sensors

The Table 1 provides a good overview of the available sensors mentioned in the precedent chapter. The important characteristic for our application are listed for each sensor with qualitative criteria for size, cost and human/non human distinction. That can help to make a good choice for sensor selection.

3.2 Solution chosen for a human detection set of sensors

With the results of the research obtained above, it can be seen that one sensor is not enough to detect the presence of a victim in a disaster area. We need a set of several sensors which measure different physical human characteristics that can be put on the USAR robot from Carnegie Mellon. Here are the most important criteria for sensor selection for our application:

- Low cost
- Small size
- Low weight
- Simplicity (easy to interface)
- Robustness

Interesting advices about sensors and their applications in USAR are described in [6]. They explain that video camera are essential for USAR robot since they permit the workers to navigate and see the site via teleoperation, but for victim identification digital thermal camera appear much better. Motion detection is also a good solution for victim detection. Better view of the environment should be seen with Omni-cam or fish eye camera. For distance measurement Laser range finder may produce better result than sonars because of the high density of sharp edges and inconsistent material properties. Microphone and speaker is also important to interact with the victims. Finally, gas sensors should be useful to know if it is safe for rescue workers to enter in the disaster area but this sensor is not very efficient for victim detection.

3.2.1 Camera

To follow the advice in [6], we need a vision sensor to have a robust solution. This is a powerful sensor because it gives a lot of information and is useful both in human detection and navigation. We decide to use a USB camera because of low cost, small size and ease of use. Moreover it already exists some software to acquire the image.
In addition to the criteria listed above, it is necessary that the camera has a Linux driver available for sound and image. Thus we chose the Philips ToUCam Pro webcam (model PCVC740).

We have chosen a webcam with a built in microphone so that we can detect sound, which is an important parameter in USAR. These are two powerful sensors, but they give a lot of information to compute making challenge to find the right equilibrium between speed of computation and quality of detection.

Figure 7 : Webcam Philips ToUCam Pro

- Sensor : CCD 640x480 pixels
- Interfacing : USB
- Frame rate : up to 30 fps
- Field of view : 33° x 25°
- Weight : 110g
- External dimension : 84 x 67 x47 mm
- Microphone integrated

3.2.2 Pyroelectric infrared sensor

This is another low-cost sensor which is complementary to the vision and sound chosen before. As explained in section 3.1.2, it is efficient detector for human presence. It is a very cheap and commonly used device in robotics, because the interface with the robot is very easy. With a Fresnel lens it can detect a person several meters away and it is not dependent on external light. Many pyroelectric sensors which are available have the same characteristics. All of them need an electronic circuit to amplify and filter the signal. A common application is to have a comparator to have a binary output. A very small device with all the electronic parts built in was found by Murata. The model is IMD-B101-01 and has the following properties (see also datasheet in Appendix A):

Figure 8 : Murata pyroelectric sensor IMD-B101-01

- External dimension : 20 x 13 x 8 mm
- supply voltage : 2.6 – 5.5 V
- Current consumption : 45 µA (ready period) 85 µA (active period)
- Output: analog or digital
- Wavelength Range: 5 – 14 µm
- Detection length : 1 m (without lens), 5 m (with Fresnel lens)
- Field of view : 104° x 30° with Fresnel lens (not specified without lens)

The advantages of this device are its whole built-in electronic package, small size and the ease of interface with its digital output.

3.2.3 IR camera

An IR camera is a powerful sensor to detect human presence in different kinds of environmental conditions. Then, a lot of searches were done to find a product not too expensive and that could be use on a robot.

We believe the best camera for this application is the IR camera 2000b by Raytheon Infrared. This is one of the cheapest with interesting characteristics and small size (see the specifications below and the datasheet in Appendix A). This camera is a longwave (7 – 14 µm) infrared thermal imaging video. It uses a new kind of sensor, based on the focal plane array (FPA) technology. There is a plane of infrared detectors. More technical information about IR imager can be found in Appendix D .

The field of view and the focusing distance are two important criteria to choose the product. For the USAR application it is important to have a large field of view in order to see an entire human body close to the robot. According to Figure 9, the horizontal field of view to see a 1.80 m entire body at 3 m is:

$$\alpha = 2 \;\; Arctg\left(\frac{h/2}{d}\right) = 33.4°$$

1.80 m is the height of adult people. It is important that the robot can see the entire body not too far to have a good overview of the victim's state and because in a disaster place, the area is often small to navigate. The focusing distance has also to be small to have a good image quality of closer objects.

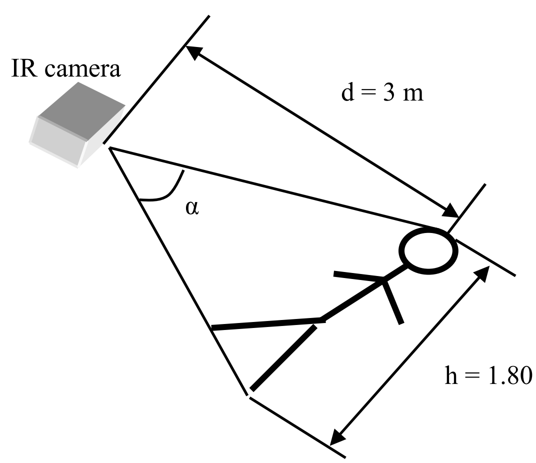


Figure 9: How to chose the FOV of the IR camera

The Raython Infrared IR camera 2000b has all these properties. The main product specifications are mentioned below:

Figure 10 : Raytheon Infrared IR camera 2000b

- Size : 9.4 x 12.5 x 10.2 mm
- Weight : 0.73 kg
- Operating voltage : 9 – 28 VDC
- Video output : analog (NTSC) digital (16 bit parallel)
- Spectral Range : 7 – 14 µm
- Resolution : 320 x 240 pixels
- Optical : 18 mm lens
- Depth Field : 3 m to infinity
- Field of View : 46° H x 35° V

4 ROBOT CONFIGURATION

All the sensors selected below are implemented on the USAR robot of Carnegie Mellon. This robot, with two bicycles wheels (see Figure 11), use a differential drive by varying the speeds between the left and right wheels. The solution provides to the robot a great mobility. These two big wheels allow it to climb small obstacles. With its two high power motor wheel controlled by a PID, it can climb ramps too. It has also a pan tilt head for the camera.

Figure 11 : USAR robot of CMU

The controller board of the robot is a Stayton board made by Intel. This family board offers several benefits for embedded systems including robotics. Specifically, it is designed to optimize low power consumption and high performance processing for a wide range of wireless networking applications. It currently runs version 2.4.19 of Linux. The main characteristics of this board are listed below (other technical data can be found in [31]):

- 400 MHz PXA250 XScale processor (new chip family based on the ARM architecture)
- 64 MB SDRAM
- 32 MB Flash EPROM
- USB host and slave interfaces
- 2 PCMCIA slots
- Serial port

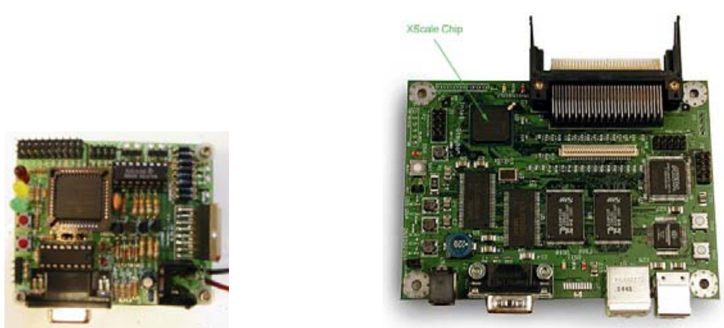


Figure 12 : Cerebellum (left) and Stayton board (right)

Another controller is used for the sensors and the servo motors. This is the Cererbellum. It is a fast, low-cost microcontroller for mobile robotics and embedded applications. Cerebellum is a PIC (16F877 from Microchip) based board with integrated programming and motor driver circuits. It has analog and digital I/O ports, a serial port and an I^2C port. As shown in Figure 13, the servo motors for the pan-tilt head, the pyroelectric sensor and the Sharp proximity sensors are connected to the Cerebellum.
The stayton board is working as a server. It controls the cerebellum through the serial port and manages the wireless TCP/IP connection with an external computer. The Philips webcam is connected to the Stayton too with the USB port. All these connections are summarized in Figure 13 below.

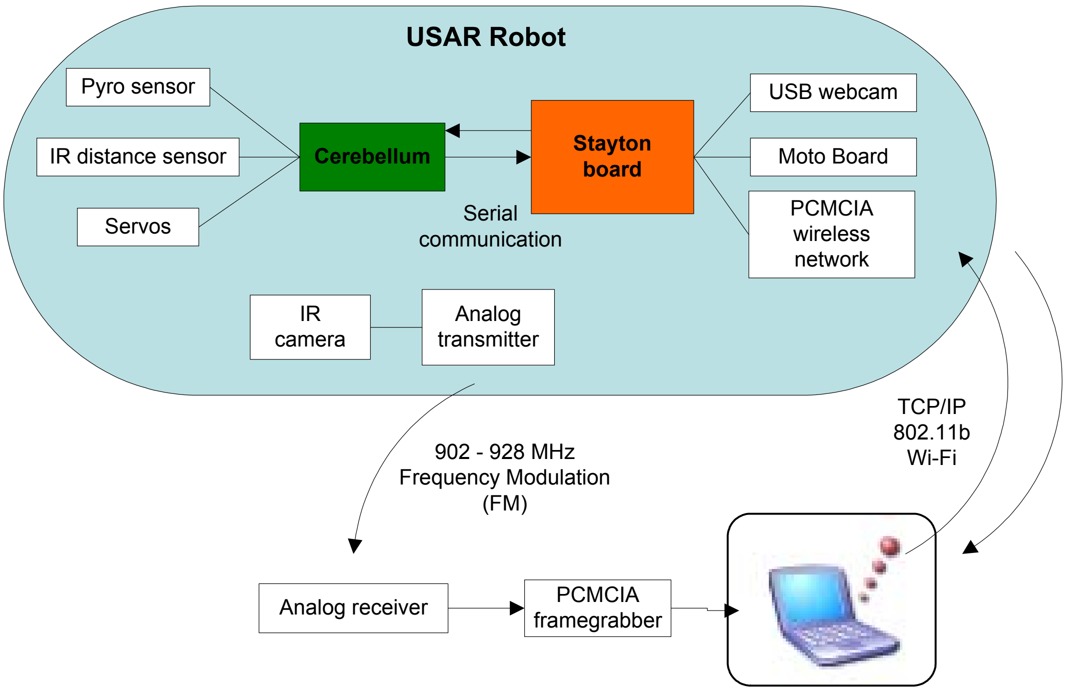


Figure 13 : USAR robot's configuration and connections

Figure 14: hardware for USAR robot teleoperation

5 Sensors implementation and tests

5.1 Pyroelectric sensor

The pyroelectric sensor is connected to the digital port of the Cerebellum and the digital port value is read to know if the sensor has detected something (state 1) or not (state 0). The sensor can only be read once a second (bandwidth = 1Hz) to let the amount of charge on the sensitive crystalline material decrease (see datasheet in Appendix A).

5.1.1 Results

Different kinds of experiments are done to determine the sensitivity of this sensor.
First the sensor is placed on the robot in a fixed position and the robot remained in place. A human stand in different places in the field of view of the sensor and gestures.

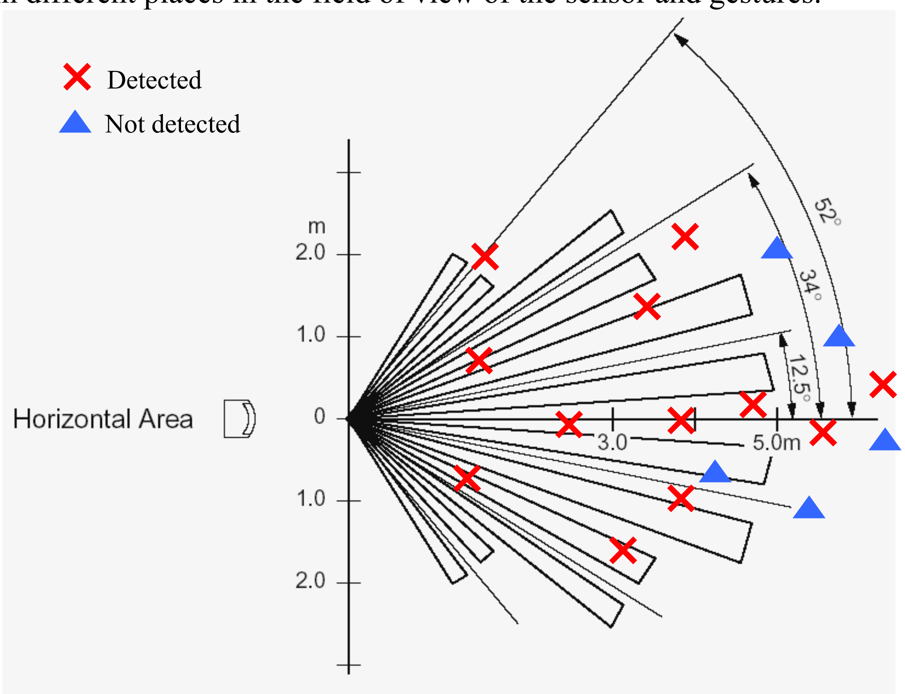


Figure 15 : pyroelectric sensor field of view with the Fresnel lens

According to Figure 15, the human is nearly always detected in the sensor field of view. Sometimes the detection distance is longer than the specifications because the sensor detects movement at more than 5 meters.

For the second test, the sensor is placed on a pan tilt head and scans the horizon. Another lens with a narrower field of view is used. The sensor stops every 10° over 180° to take a measurement (see Figure 16 below). With its Fresnel lens the sensor detects the human before it is in front of it, because of its field of view. Depending where is standing the human, the detection range become wider.

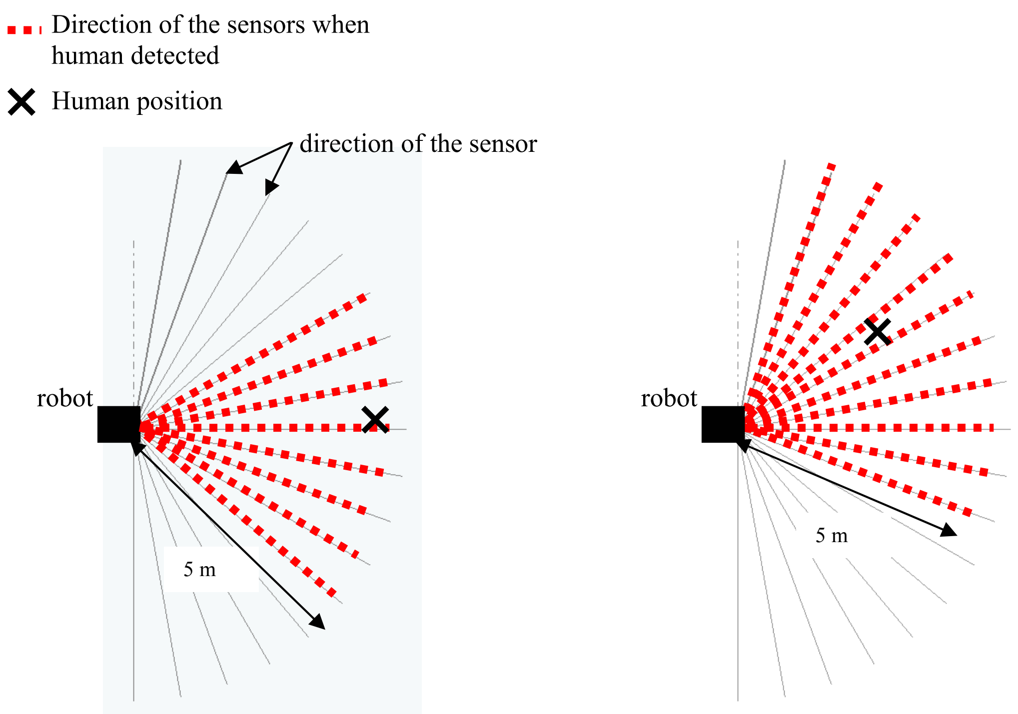


Figure 16: Results of the pyro scan in presence of one people

The third experiment is the same than the one above but without a human in the sensor field of view. In 60 readings 28 readings detected a human. Thus the false positive result are quite high, almost the half. Actually, these kinds of measures are strongly dependent of the environmental condition (heat reflexion, artificial heat …) because this sensor is supposed to work in a static position, not moving.

- Results table (number of measurements):

	Human	No human
Human detected	20 (TP)	28 (FP)
Nothing detected	1 (FN)	32 (TN)

Table 2: Pyro sensor results of measurments

TP = true positive FP = false positive
FN = false negative TN = true negative

If these results are expressed in a probabilistic way and if H means "there is a human", NH "there is no human", D "human detected" and ND "nothing detected", the following results can be established:

$$P(D|H) = \frac{TP}{TP + FN} = 0.952 \qquad P(ND|NH) = \frac{TN}{TN + FP} = 0.533$$

	Human	No human
Human detected	**95.2 %**	46.7 %
Nothing detected	4.8 %	**53.3 %**

Table 3: pyro sensor results in percentage

$$\text{Accuracy: } A = \frac{TP + TN}{TP + TN + FP + FN} = 0.64 \qquad \text{(ideally = 1)}$$

As the sensor had many false positives, it is not very reliable and its accuracy value is not very high. But the Table 3, shows that if the sensor does not see anything, we can almost be sure that there is no human presence. The opposite way is not true.
Moreover, the accuracy value will be useful to determine the confidence of this sensor as we will see later in section 6.3.

5.1.2 Limitations

When pyroelectric sensor is moving, it is not very accurate. It is made to be use in a static position. It seems to be also quite dependant of the environmental condition, but it is difficult to say exactly what, because the output is only binary. We do not have good information to understand the false positive value. Thus, the results of this sensor have to be used cautiously.

5.2 USB camera

5.2.1 Software

The pictures are captured with the Philips ToUcam USB camera. The software to acquire data from the device and convert it to a picture is Vidcat. This is a free software for Linux written in C code. The data is captured in YUV format and converted in RGB. The picture is then compressed in JPEG format to be sent through the TCP/IP protocol to an external computer for visualization. As the Stayton board has limited computing power, only basic image processing is done directly on the robot. The image processing is in RGB format, just before the JPEG compression. This picture format is used because it is easier if we have to work with color and easier to display final result with color at the end of the process. The picture is converted in RGB also because the current JPEG compression function in Vidcat needs this format as input.
The resolution chosen is 320x240 pixels. It is a good compromise between time processing and quality of displayed image for the user. It takes about 50 ms to acquire the picture and between 130 and 250 ms to send it through the network.

5.2.2 Processing

There are different ways to process pictures to extract information about human presence. Color detection, motion detection and human body modeling (see [9]) are the most common ways. The first image processing used on the robot which does not require large amounts of computation is motion detection. Two images are subtracted with a threshold to filter noise and avoid little light changes between the two images. To characterize the motion, the percentage *p* of the "moving" pixels in the image compared to its size is calculated; the mean value *m* of the difference of each pixel between the two images is also calculated (see Appendix B for the flow chart of the motion function).

$$\text{percentage of changing pixel} : p = \frac{N}{h * w}$$

with : N = number of changing pixels between the two images
h = image height
w = image width

$$\text{mean value} : m = \frac{\sum_{i=0}^{N} \left| x_{i,2} - x_{i,1} \right|}{N}$$

with : $x_{i,1}$ = value of pixel i on the first image
$x_{i,2}$ = value of pixel i on the second image
N = number of changed pixels

5.2.3 Calibration

There are several changing parameters for motion detection :

- Distance between camera and the moving object
- Kind of motion
- Delay between two pictures
- Difference between the value of the pixel in each pictures

The calibration is done to set some parameters before measuring the motion. First, we have to choose when we consider that a pixel has changed. i.e. if x_1 and x_2 are the pixel value of the first and second image respectively,

$$|x_2 - x_1| > \text{threshold}$$

Moreover, time t between the two pictures can be set. If the motion is slow the time will be high and vice versa. Finally, depending of the kind of motion that has to be detected, the number of changing pixel *N* can be change to know when we consider that is a motion or not. It is actually a second filter. This value can also change with the distance between the moving object and the robot. For the same motion, the closer is motion from the robot, the more pixels will change. Then for calibration the moving object has to be at the same distance.
We suppose that victim who is detected move only arms, or finger but not the entire body. Looking at the example below (see section 5.2.4), we can define the range that the value of the changing pixel N should take.
Further calibration depends on whether or not the motion to be detected is known. If the motion is known, some measurements of *N* are taken during the calibration to know how many pixels are changing in the picture. The function f_m (see Figure 17) is built around this known value *N* to accept some fluctuation around that value. These fluctuations can appear from a difference in the distance between the camera and the moved object or even if it is at the same distance, the motion is not always exactly the same.
If the motion is not known, to accept more different motion, a more tolerant function is built to have the confidence level p_m of the motion (see Figure 17).

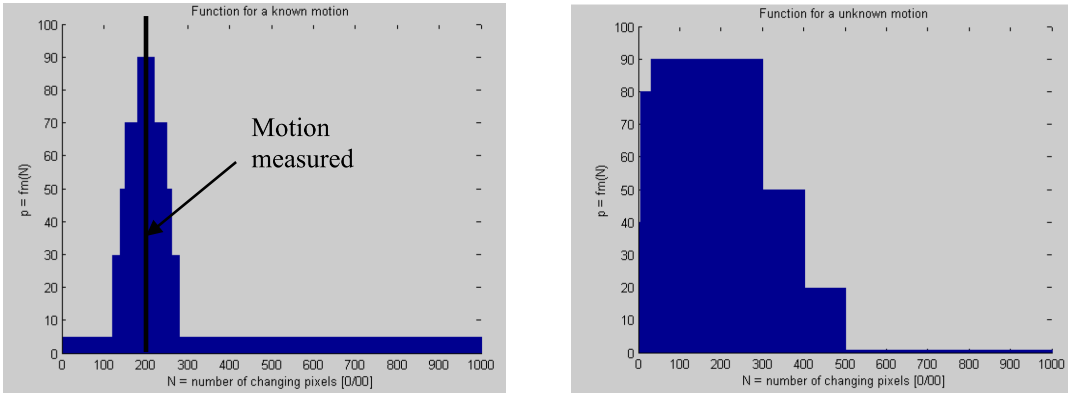
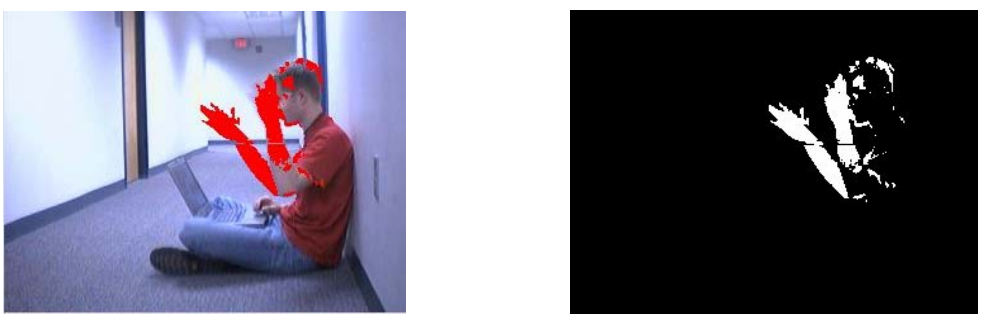


Figure 17: Confidence of motion detected as a function of the number of changing pixels N

The graphs above are built using empirical data, after having taken many measurements. The confidence level that there is actually motion in the scene, p_m, depends on the number of pixels that are different between the two images.

5.2.4 Results

The results of this motion detection algorithm are shown in Figure 17. The image on the left is the original image with the motion superimposed in red. The binary image on the right is the result of the subtraction of the two images.
The percentage of changing pixels and the distance between the robot and the human are below each set of pictures.



At 3m: changing pixels: N = 3.2%

At 4m: changing pixels: N = 1.3 %


At 5 m: changing pixels: N = 0.8 %

At 2 m: N = 18.7 %

at 5m: N = 6.3%

at 8 m: N = 1.4%

at 2m: N = 2.1 %

At 2 m N = 33.8 %: changing light condition = noise

Figure 18: Results of motion detection

For the urban search and rescue domain, the goal is to detect moving hands and arms with the supposition that fully mobile people do not need to be rescued. These examples illustrate that motion of a limb is relatively easy to detect, assuming the victim is within a known distance range, because motion of the entire body (*N* is ten times higher than limb motion at the same distance) or changes due to changing light conditions result in pixel changes an order of magnitude higher than motion caused by a single limb.
The further away the motion is from the camera, the less the pixels change, then lower is *N*. Even if the function f_m can tolerate some variation in distance, the confidence will decrease quickly if the distance between motion and robot is varying too much. That is why to have better result we have to do the assumption that the distance is a known parameter. This assumption could be enforced with range sensors.

5.2.5 Limitations

The largest limitation to this motion detection algorithm is that it assumes that the camera is stationary. This means that the algorithm will not work while the robot is in motion. The camera is mounted on a pan tilt head on the robot. Each turn of the head results in small oscillations which interfere with the algorithm. Even if these oscillations were reduced by using a very short step for the servo motor, there must be a delay of several milliseconds before taking the pictures.

The second essential limitation is light condition. If there is too much difference in light condition or varying illumination between two pictures the results is completely false. To ensure the light conditions are consistent across pictures, the time between pictures must be reasonably short which means that some slow motion may not be detected. And if the time between the images is long, it inhibits the motion of the robot, and it might miss cyclic motion. One way to deal with this might be to use more than 2 images and compare sequential images but also the first to last image. Of course this has the drawback of requiring a lot of memory and processing power.
Furthermore in USAR situation, there are much more external conditions that can alter the results like dust or smoke in the air. So this kind of process can not be used in every situation.
This motion detection algorithm is also insufficient by itself to determine whether or not the motion was generated by a human. Even when the type of motion is known, there are many external parameters that cannot be modeled.

5.3 Microphone

5.3.1 Software

As explained in 3.1.3, the sound data is acquired using the built in microphone of the Philips ToUcam Pro USB camera. Free software, VSR, which is running under Linux and use the OSS library sound, records sound from the camera to a wave file. This software, written in C, stores data from the device in a shared memory buffer, while two other threads convert the data to wav format and write it to the disk.

5.3.2 Processing

The most common way to do sound processing is to work under the frequency domain using the Fourier transform. This method is computationally expensive and requires powerful hardware for use in real-time. According to [11], a sampling rate of 8000 Hz for 30 ms (240 samples) is necessary to classify the human voice. A Fast Fourier Transform (FFT) with a 200 MHz processor and a buffer of 100 samples takes about 150 ms, as explained in [11]. Even if our processor is twice as fast, this is not possible in real-time. Reducing the sample number also reduces the quality of the Fourier transform. It is better to do more simple sound processing in the time domain, which is fast enough to be used in real-time. Since humans are very good at sound processing, the best solution may be to transmit the audio to a computer so a human can determine whether or not the sound is human.
In a disaster area, rescue people sometimes stop all activity to listen for a shouting victim. Basic voice recognition involves looking for a high amplitude noise during this time. This approach supposes that the voice of a shouting person is louder than environmental noise. However, the threshold will be able to be set when the environmental condition are known.

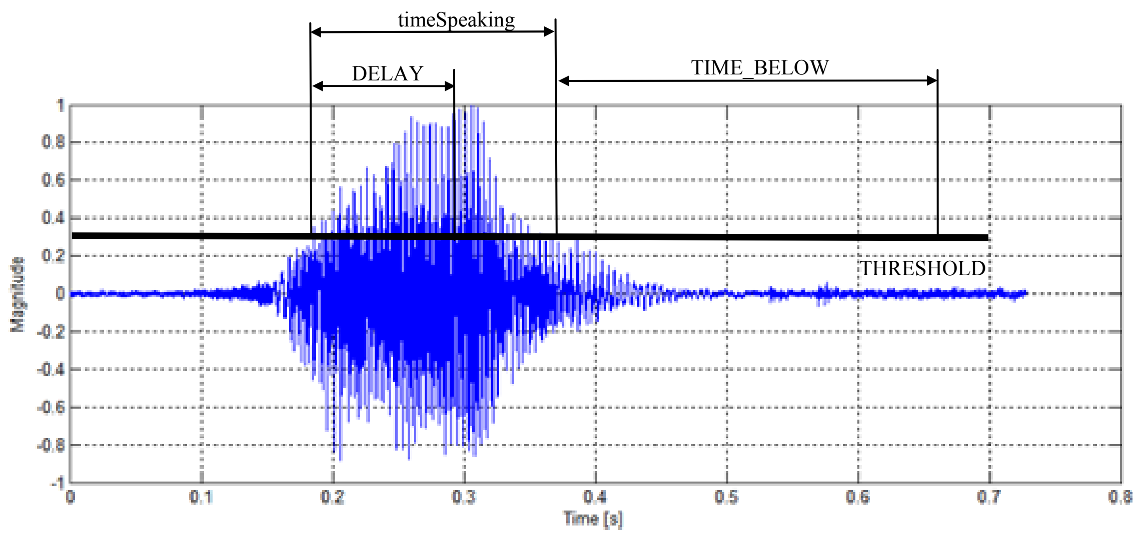


Figure 19 : Man shouting "help" in a quite environment

The software VSR was modified to do the processing in real time and not to write the data to a wave file to gain time. The data are extracted at the end of the writer thread. The function `read16bit` is called and after the processing is done with the function `voiceDet`. This function start a timer when the amplitude is higher than the threshold and calculate how long it is above the threshold to know the time of speaking (see Figure 19 and Appendix B). A filter suppresses the very short high amplitude noise. When a voice is detected, the algorithm extracts the duration and average amplitude above zero. This basic information is sufficient to roughly characterize sound.

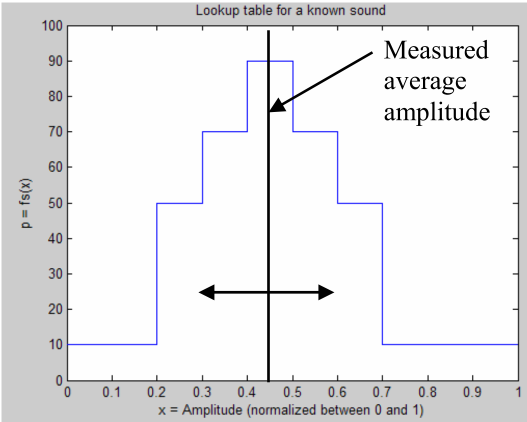
5.3.3 Calibration

To determine the threshold and duration that characterize human noise, the system must be calibrated. This is accomplished by recording the ambient noise for several seconds and setting the threshold slightly above the upper limit of the amplitude during this time. Further calibration depends on whether or not the sound to be detected is known.

If the sound is known, it can be recorded after the threshold calibration. The duration of shout is also known and it can be trigged to suppress all noise that is shorter.

It is supposed that a shout must be longer than 0.5 sec to avoid detecting short noise. Then the *DELAY* value must be bigger than 22'000 (= 44100 Hz* 0.5s).

After the recording, the average amplitude is known too and the lookup table in Figure 20 can be used to determine the confidence level, $p_{s,}$ that a human voice was detected.



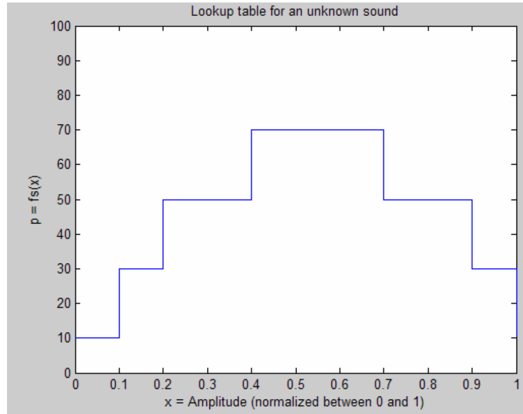
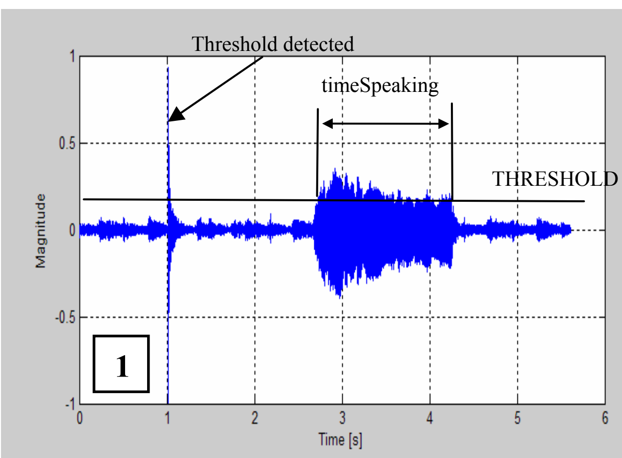
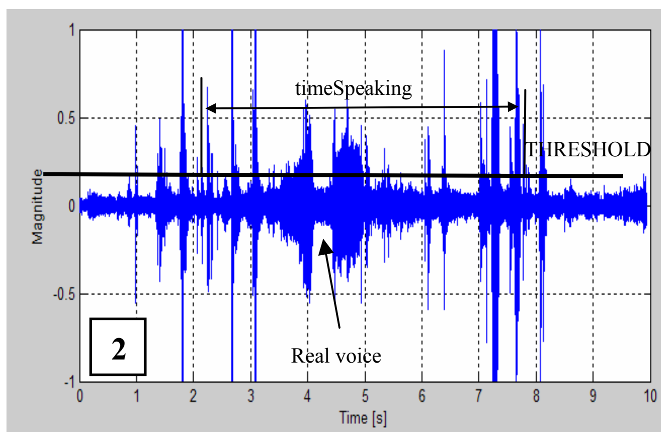


Figure 20 : Lookup table which give the confidence level depending on the amplitude

If the sound to detect is not known, the global confidence value for microphone will be lower (see section 6.3). The lookup table for the confidence is more tolerant (see Figure 20). These two lookup table are established in an empirical way after taking a lot of measurements.

5.3.4 Results

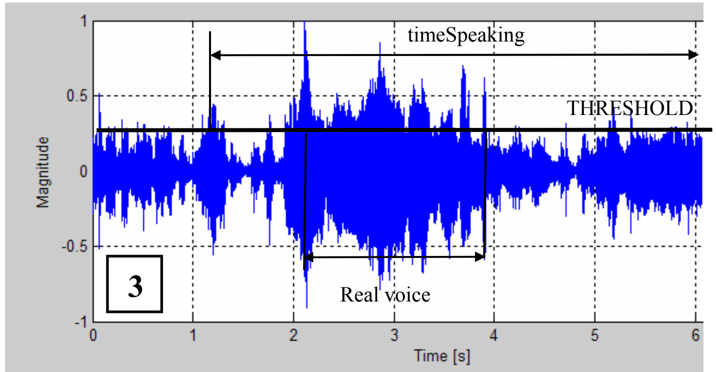




THRESHOLD = 0.12
timeSpeaking : t = 1.5 s
meanValue of speaking : x = 0.21

THRESHOLD = 0.18
timeSpeaking : t = 5.8 s
meanValue of speaking : x = 0.20

Figure 21: Examples of sound acquisition



THRESHOLD = 0.28
timeSpeaking : t = 5.1 s
meanValue of speaking : x = 0.29

Figure 22: Examples of sound acquisition

The results of the voice detection algorithm are shown in Figure 22 and Figure 22. In these examples the amplitude is normalized between 0 and 1. However in VSR software it is coded in 16 bit, so the highest value is 2^{16} = 65'536.
Many of the limitations of the algorithm are also illustrated by these results. In the first example the detection works well because the ambient noise is low and the human shout is well differentiated from it. High amplitude is also present before but the algorithm does not take it like human shout because it is too short. However, in the two last cases (2 and 3 Figure 22) the detection fails. Ambient noise is too loud, thus the algorithm takes a bigger value for *timeSpeaking* and the *meanValue* than the real voice. This method is not very efficient when there is much ambient noise. Then calibration is very important to determine if sound detection will work well in the environment where the robot will drive.

5.3.5 Limitations

We can see easily that is not the most powerful way to detect human voice. The algorithm presented does not detect human voices but rather long, high noises. It is not useful in a noisy environment, but in a quiet environment, it can be very powerful. Calibration is, therefore, essential.
Another limitation is the microphone used is not very directional. Even if it correctly detects a human voice, it is very difficult to determine the direction of the source. For better results, two microphones could be used, but in a disaster area, there is a lot of noise reflection on the walls. Even with multiple microphones, locating the source of the sound would be very difficult. As explained later in 8.2, another function could also be added to detect regular sound banging.

5.4 IR camera

5.4.1 Software

The tLib library [17], from the VRAI-Group at EPFL, is used for the processing of the infrared images. This image processing library was written specifically for real time object tracking and contains a lot of functionality. It is very efficient, easy to use, and portable. It can take multiple sources as input. However, it requires significant computing power and needs to be run on an external computer rather than on the Stayton board on the robot. The analog transmitter sends the output of the infrared camera to an external computer with a frame grabber (see Figure 13). The infrared camera is used to detect humans in the environment by looking for hot spots that correspond to skin and by looking for motion of that spots.

5.4.2 Processing

The image provided by the infrared camera is in grayscale. White corresponds to warm object and black to cold object. Figure 25 shows that, clothing on a human body has different levels of gray, but, most importantly, human skin is uniform bright white. Some objects in the environment can be seen as well and objects that came into contact with a human remain warm for some time.

- Human extraction

The goal of human extraction is to determine the position of the human in the image. The skin is the brightest part of the picture, i.e. the hottest. Some basic image processing can be applied to locate the human (see also Appendix B):

- Threshold to select only a part of the grey (Figure 23 – 2)
- Convert to a binary picture (Figure 23 – 3)
- Opening[1] twice to suppress the white noise (Figure 23 – 4)
- Extract with a minimum size
- Select largest blob as location of head (Figure 23 – 5)

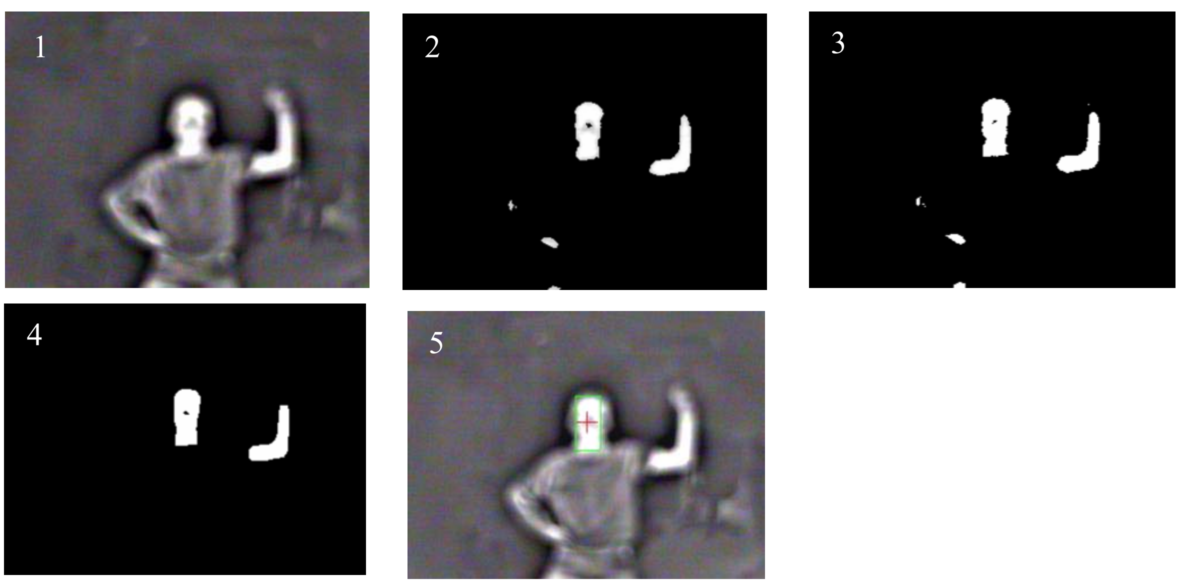


Figure 23 : Every step for the processing of the infrared picture

[1] Opening function: consist of two morphological functions: erode and then dilate with a mask

- Motion detection

As Figure 25 illustrates, human skin is not always the only hot part in a picture. To improve detection, motion detection algorithms can be applied to the IR images. If a hot spot is moving, it is very probable that it represents a human.

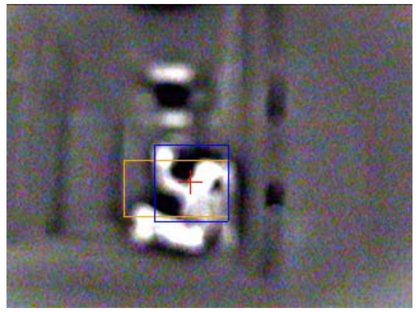


Figure 24: Result of IR motion detection

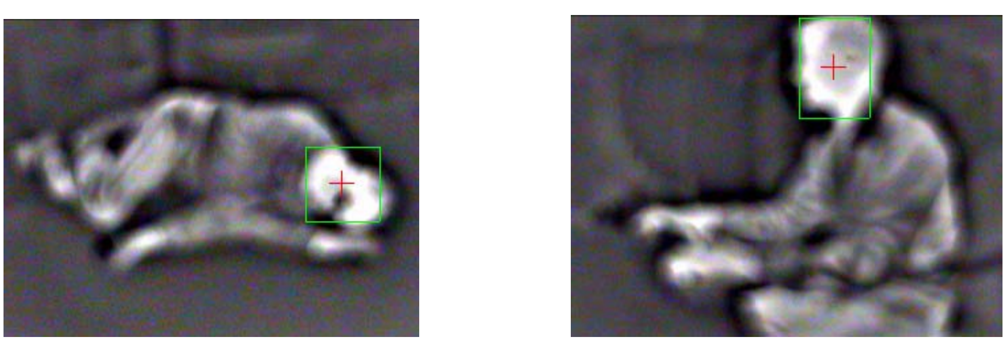
An example is shown in Figure 24, where two rectangles are displayed to show the two successive positions of the hottest blob.
At the end of the process, the size of the blob is known, his position in the picture and the vector of motion if there is one.

5.4.3 Calibration

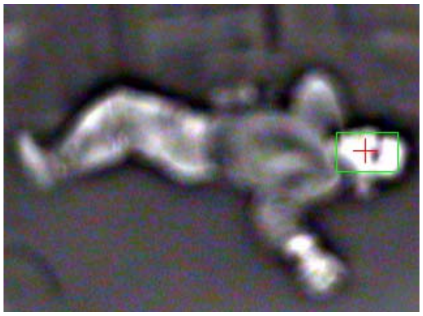
In analog mode, the gain and the level of the infrared can be set with a potentiometer. These adjustments can be made directly on the picture too, as the gain corresponds to the contrast of the picture and level corresponds to brightness. Depending on the environment, the threshold value to select the grey level can be changed to improve detection.
To determine the confidence value p_{IR} depending on the blob size, the same method than in section 5.2.3 is used. A similar function than in Figure 17 is build around a given value of blobSize.

5.4.4 Results

Figure 25 shows examples of human detection in infrared pictures. Tracking the human can be performed in real time at a rate of about 20 fps. Usually, the infrared camera detects humans well. Occasionally, another object in the room is larger than the human face and this object is tracked instead.

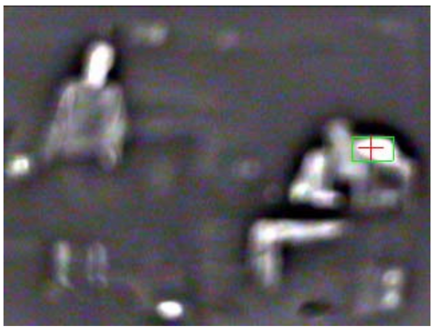


The human head is detected

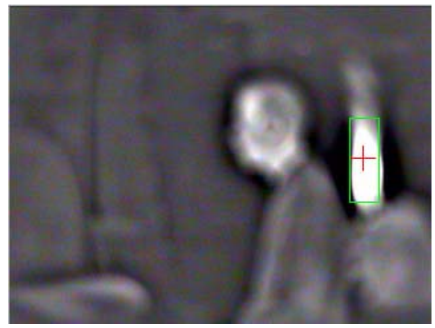


The human head is detected

No human in picture, nothing detected



Computer screen is selected instead of the human head



If there is fire, it is brighter than the human face

Figure 25: Results of human detection in an IR picture

5.4.5 Limitations

This IR camera is one of the most efficient sensors that we can use to detect people (see Figure 25). It is completely independent of the light condition. The only limitations of this sensor are the other heat source in the environment if they are warmer than the human body. In a disaster area there are different heat sources such as, pipes, light sources and fire. In the presence of these elements, the detection of the largest hottest part in the picture is insufficient. Shape of the warm element should improve the results in these cases

5.5 Limitations of each sensor: summary

The Table 4 below shows all external events or objects which are not human but which can occur in a disaster area and the effect they have on each sensor.

	Pyro sensor	Sound detection	Motion detection	IR pictures
Varying illumination	+	-	+++	-
Sound noise	-	+++	-	-
Moving object	+	-	+++	+
Warm object	++	-	-	++
Dust, smoke	+	-	+++	++
Fire	+++	-	++	++
Water/liquids	+	++	++	-
Obstruction debris	-	+	-	-
Mirror glasses	+	+	+++	-
Window glasses	++	+	-	++

Table 4: External influences on each sensor

- : no influence + : influence ++ : strong influence

This table is very qualitative, but we observe that every sensor is not influenced on the same way by the same events. For example if there is variation in illumination, motion detection will be influenced a lot but it will not affect the infrared picture. Or if there is a glass window and a human on the other side, the pyro sensor and the infrared camera will not detected him because glasses is cutting the infrared wavelength, but it will be possible to detect his motion with the USB camera.
Globally, these sensors are complementary and as we will see after in section 6, the fusion of their data will be useful to minimize the false positive or false negative results of each of them.

	Pyro sensor	Sound detection	Motion detection	IR pictures
Range [m]	0 – 5	1 – 10	1 - 10	1 - 15
time of processing [ms]	1000	19’000	160	42
Accuracy	0.64	+	++	+++
Precision	+	++	++	++

Table 5: sensors comparison

The Table 5 shows global information about the performance of each sensor we use. The range gives an approximate range distance where the detection method of this sensor is working. It supposes that the environment is free between the human and the robot. For sound detection it is hard to define this range exactly because it is strongly dependant of sound reflection i.e. what kind of wall structure there is around the victims and the robot.
Time of processing gives an approximate value of the execution time of the function which is used to read the data from the sensor. For pyro sensor it is dependant of the time of discharge of the sensitive crystalline material of the sensor. For motion detection, the processing time does not include waiting time between the acquisitions of the two pictures. For IR picture it is also the time of motion detection algorithm (see Appendix B) and does not include the waiting time between acquisition of the two pictures.
The sound detection is very long because for the moment it does not run in parallel with other application. Moreover the time to open audio threads and close them at the end of the process is long. In this value is included the five seconds of sound recording. This value is not very representative.
For accuracy and precision, qualitative description is preferred. It is hard to do exactly the same human motion many times and measure it with a video sensor. It is the same for sound detection. The accuracy of pyro sensor is defined in section 5.1.1. Then, qualitative accuracy descriptions of the other sensor are compared to the pyroelectric.
These two tables can help to define confidence value for each sensor depending if they have good characteristic in define conditions.

6 DATA FUSION

6.1 Theory

Data fusion is the process of combining information from different sources to provide a robust and complete description of an environment or an interest feature. The fusion of the data may be complicated due to the fact that each sensor has its own performance characteristics and its own level of precision, and because different sensors may detect different physical phenomena. Table 4 shows that every sensor has different advantages and disadvantages depending of the environmental conditions. Reliable results can not be found with only one kind of sensor. Multiple sensors are used to improve the robustness of the final result. Fusing the data of each sensor yields results that are less dependant on the weaknesses of any single sensor.
There is different way to apply sensors fusion according to [2]:

- Probabilistic method (Bayes therorem, Log Likelihood, mutual information, Fuzzy logic...)
- Multi sensors estimation (different form of Kalman filter)

6.2 Solution selected

Almost all sensor fusion approaches use the probability density function of each sensor. In our case we have different sensors with different kinds of data (binary or discrete). We do not have an exact model so we will use a fusion method with confidence values. Each sensor is assigned a confidence marker indicating the certainty of the measurement value for this sensor.
For each sensor there is a lookup table or function f_i (see Figure 17 and Figure 20) which give a probability p as a function of the measurement, x. For each sensor i we have:

$$p_i = f_i (x_i)$$

where x_i is the measure from the sensor i, f_i is the function which give a probability p_i between 0 and 100 that the sensor has detected a human or not. The higher p is, more probable is the presence of a human.
Then, each sensor i has its own confidence c_i between 0 and 100. The higher confidence c_i is, more reliable is the result of the sensor.
So if there are n sensors, the final probability of a human presence is:

$$p_f = \frac{c_1 f_1(x_1) + c_2 f_2(x_2) + \ \dots \ + c_n f_n(x_n)}{c_1 \max(f_1(x_1)) + c_2 \max(f_2(x_2)) + \dots + c_n \max(f_n(x_n))} = \frac{\sum_{i=0}^{i=n} c_i \, p_i}{\sum_{i=0}^{i=n} c_i \max(p_i)}$$

6.3 Applications

For our application, for motion detection, sound, and IR pictures, we have the confidence value c_m, c_s, c_{IR} respectively, and $f_m(x_m)$, $f_s(x_s)$, $f_{IR}(x_{IR})$. These functions are given in Figure 17 and Figure 20.
For the pyro sensor, x_p is binary (0 or 1).

Then: $f_p(x_p) = \begin{cases} 4.8 = P(ND|H) & if\ x_p = 0 \\ 53.3 = P(ND|NH) & if\ x_p = 1 \end{cases}$ (see 5.1.1)

So with the sensors on the robot the probability a human is detected is:

$$p_f = \frac{c_m\, p_m + c_s\, p_s + c_p\, p_p + c_{IR}\, p_{IR}}{c_m \max(p_m) + c_s \max(p_s) + c_p \max(p_p) + c_{IR} \max(p_{IR})}$$

Next the confidence value of each sensor has to be determined. Confidence values must be determined empirically because we lack a probability density function for these sensors. With some observations we know that some sensors are more accurate than others because they are less dependant on environmental condition. First like the accuracy value of the pyro was calculated in 5.1.1, it is a good reference for the confidence of this sensor. Then if we select the range of the confidence values between 0 and 100:

$$c_p = 64$$

After some experience, in an environment with reasonably little noise and with constant light intensity, the other confidence values are selected as follows:

$$c_m = 70 \qquad c_s = 60 \qquad c_{IR} = 90$$

These values are selected with comparison between false results of pyro sensor and false results of every other sensor. The IR processing has given the best result i.e. the less false result; after is the motion detection and then the sound.
These values do not have to be constant. They can change with the environmental conditions and are setup during calibration. If there is a lot of ambient noise, the sound confidence will be decreased. Similarly for the motion detection, if there is much change in the light condition the motion confidence will be decrease.

6.4 Results

Results of different measure in various environments are shown in this chapter, to see how the sensors are reacting to various situations and what the result of data fusion is.
For every experiment, the robot is doing a 180° turn around itself in seven steps. At every step, data from every sensor are taken. Motion is calculated in the normal view; in IR view, the size of the blob and its motion are computed; state of the pyroelectric sensor is read. At the end of the panorama, sound is recording and voice detection is performed. Then the probability to have a human is computed for every step with all the data of the sensor stored in memory. At the end,

the user can see three pictures that show the direction where there is the highest probability to have a human (see also section 7.1). On these pictures the motion is displayed in red to know which part in the image is moving. A green rectangle is also displayed to locate the warmest object in the normal view.

For every experiment, the results are shown with the panorama of the USB camera on the first row and the panorama with the IR camera on the second row. The result of each sensor is written below every picture to know their value at each step. Two graphs display the probability to have a human at each step of the panorama; the graph on the right one is before voice detection and the left one just after voice detection. Then the pictures given the three directions where there is the highest probability to have a human are shown for every experiments.

6.4.1 Experiment 1: in a room

The first experiment is done in a room with two people. One is not moving and the other is moving his arm. In this room there are many different objects and some of them are warm like computer or battery charger. The motion is calibrated for an arm moving at about four meters.

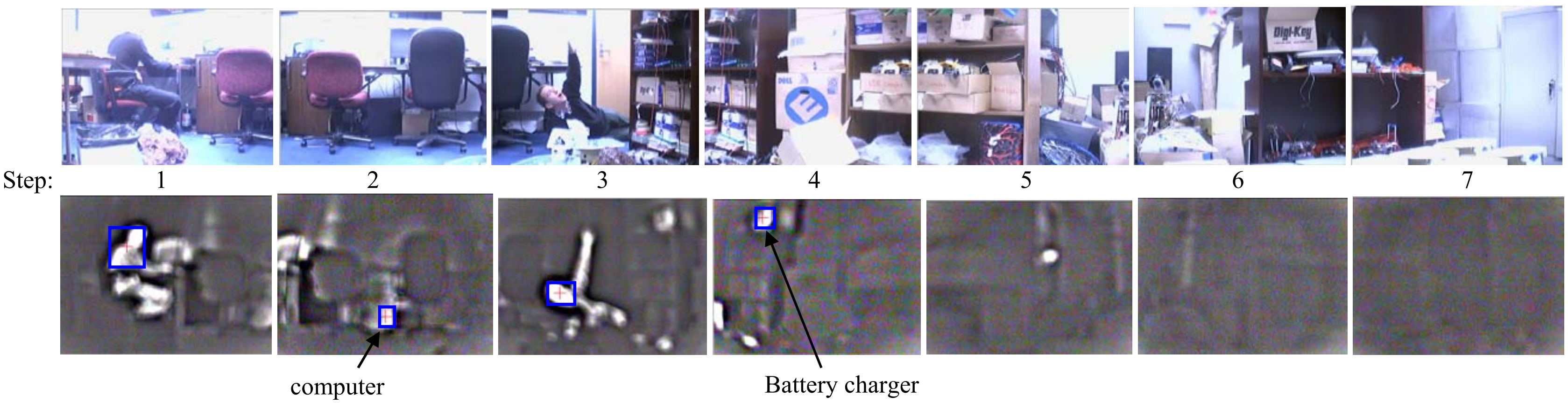
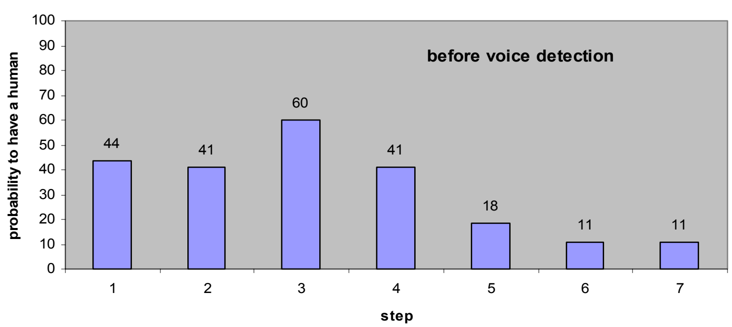


Figure 26: Experiment 1, panorama with normal and infrared view

x = measurement; p = probability

Step	1		2		3		4		5		6		7	
	x	p	x	p	x	p	x	p	x	p	x	p	x	p
Pyro	1	47	0	5	0	5	0	5	1	47	1	5	0	5
Motion	0	5	0	5	5	80	0	5	0	5	0	5	0	5
Size Heat	3172	70	650	90	1445	90	793	90	0	1	0	1	0	1
Motion IR	0	1	3	70	2	70	1	70	0	1	0	1	0	1
Final prob		**44**		**41**		**60**		**41**		**18**		**11**		**11**



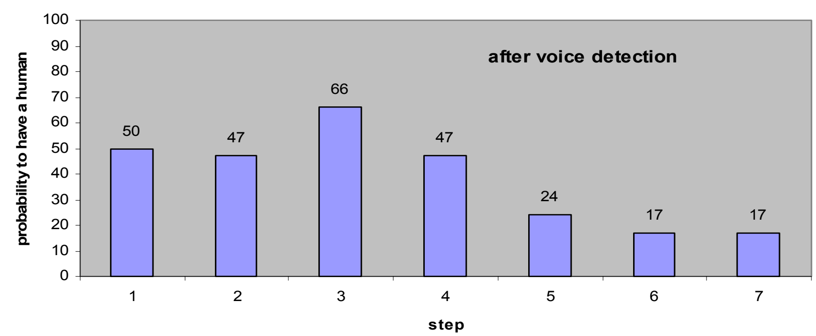


Table 6: experiment 1: measurements and results

1 2 3

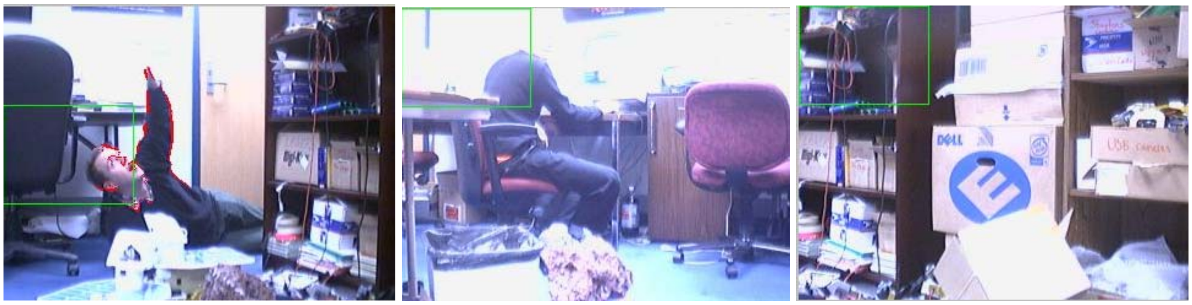


Figure 27: experiment 1, final results

The right histogram on Table 2 represent the probability at every step, before recording sound, and on the left, just after recording and do sound processing. During this experiments the second people was shouting. So we observe that the final results have a higher probability at every step than before sound processing.
We can observe that the two humans are well detected by the robot. The first highest probability to have a human is where the human has his arm moving. Even if the pyro sensor has not seen this people, the final probability is quite high because the other sensors have good results and the confidence of the pyro sensor is lower than the other. The third best result has a quite high probability too but it is not a human. It is a warm object (battery charger) that has nearly the same size than a human head. It involves that the probability is high even if it does not move, because the IR camera has a high confidence value. So at the end of the scan, the user can easily see that there is no human at this step and he does not take in account that direction. Moreover, if first we know the environment maybe we should have lowered a bit the confidence of IR pictures because there is some other warm object, and also put higher the confidence for motion because there is no other moving object than human. However, with the default value for confidence, this final result is correct.

6.4.2 Experiment 2: in a hall

The second experiment is done in the corner of a hall where two people are standing. One is moving his arm but the other one is staying without moving. On the ceiling there are artificial light sources. For this experiment nobody is shouting.

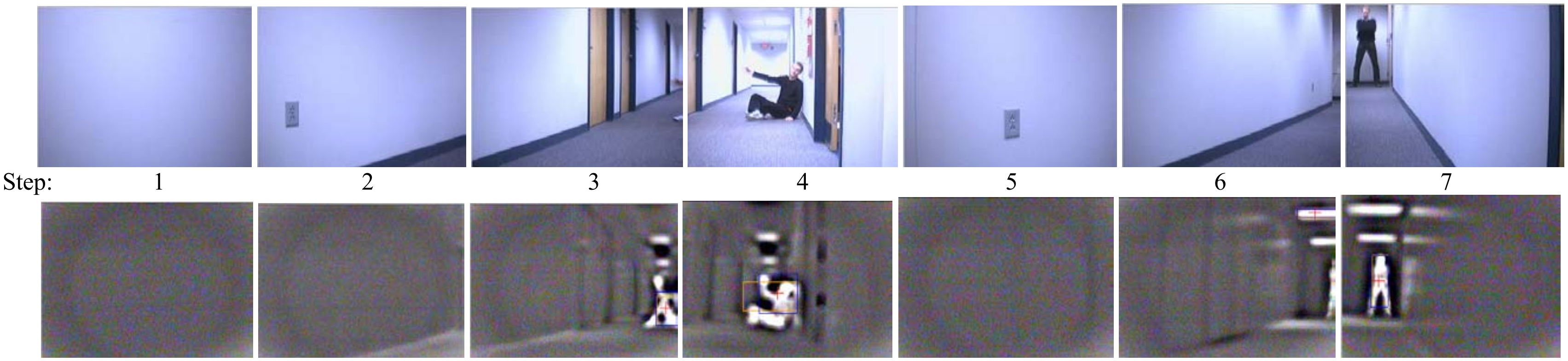
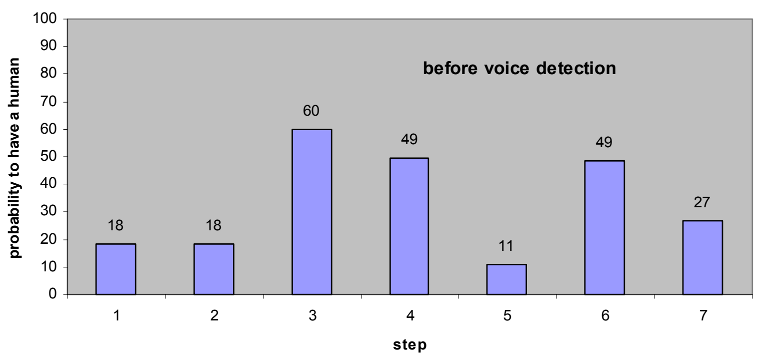


Figure 28: Experiment 2, panorama with normal and infrared view

m = measurement; p = probability

Step	1		2		3		4		5		6		7	
	M	p	m	p	m	p	m	p	m	p	m	p	m	p
Pyro	1	47	1	47	1	47	0	5	0	5	1	47	0	5
Motion	0	5	0	5	1	50	2	40	0	5	0	5	0	5
Size Heat	0	1	0	1	1529	90	3544	70	0	1	668	90	1835	90
Motion IR	0	1	0	1	7	70	26	90	0	1	150	70	2	70
Final prob		**18**		**18**		**60**		**49**		**11**		**49**		**27**



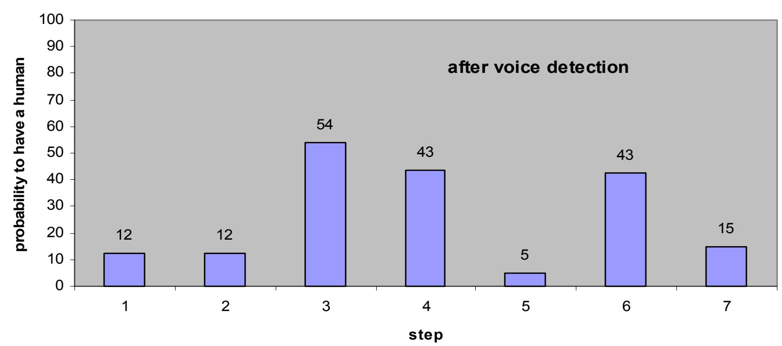


Table 7: experiment 2: measurements and results

1 2 3

Figure 29: experiment 2, final results

On that experiment observe different little failure of the data fusion can be observed. First, to do the panorama, on two successive pictures, there is a part of the image which is the same. We see (Figure 28 step 3 and 4) that the motion of the first people has a part in the step 3 and also in step 4. So on the first result picture (Figure 29) there is only a hand moving and we cannot easily guess that there is a human.
On the 6th step in the IR picture (Figure 28 step 6), the warmest part is the ceiling light. It has about the same size than the human head measured during the calibration. Furthermore, during the IR motion detection the blobSize was not constant and a motion is measured. then with the IR processing there is a high probability to have human on that step. There is also the field of view of the USB camera and the IR camera that are not exactly the same. In the second result picture (Figure 29, 2) it is difficult to understand what the IR processing has detected. Eventually, with these two errors, one person is missing in the final result.
In the two final histograms (Table 7), we observe that there is no voice detected. Indeed, all the probabilities are lower after the voice detection than before.

6.4.3 Experiments 3: in a disaster area

The third experiment is performed in the NIST (U.S. National Institute of Standards and Technology) arena at Carnegie Mellon University. This arena simulates a disaster area with a maze of walls, doors, and elevated floors recovered with dust, paper, stones, and remains. This arena provides various tests for robot navigation, mapping capabilities and victim detection. Variable flooring, overturned furniture, and problematic rubble provide obvious physical obstacles. Each simulated victim is a clothed mannequin emitting body heat and other signs of life, including motion and sound.
During this experiment two humans and one simulated victims are present on the 180° scan of the robot. The body and the arm of the simulated victims is hot. Its fingers are moving too.

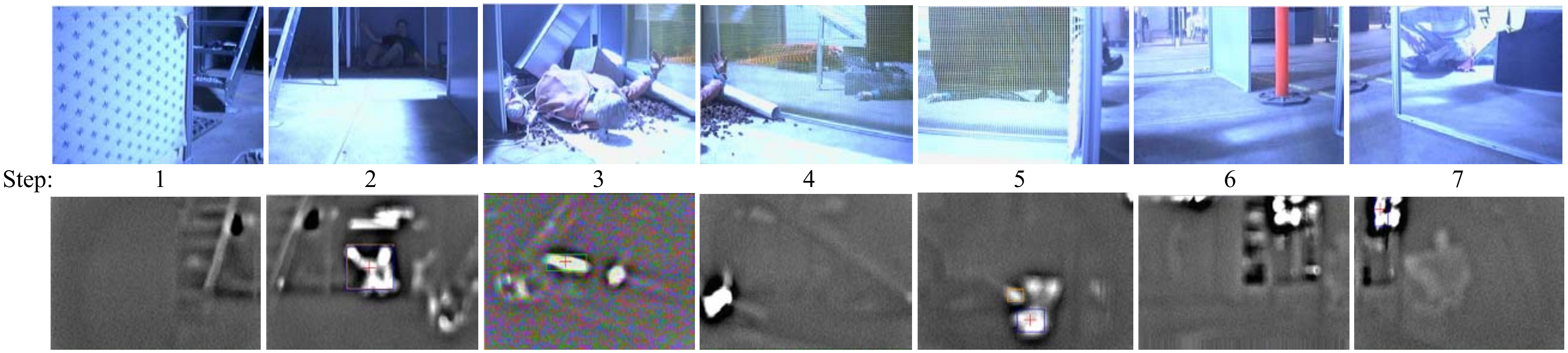
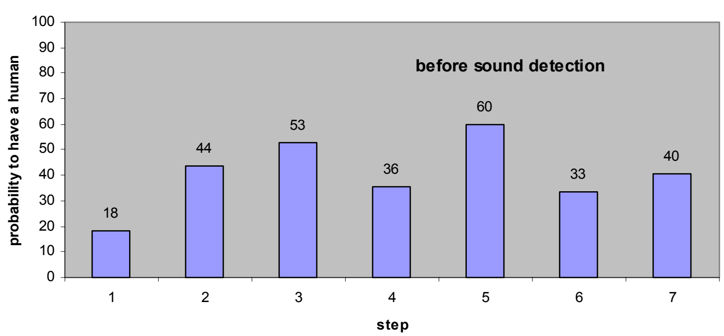


Figure 30: experiments 3, panorama with normal and infrared view

m = measurement; p = probability

Step	1		2		3		4		5		6		7	
	M	p	m	p	m	p	m	p	m	p	m	p	m	p
Pyro	1	47	1	47	1	47	0	5	1	47	0	5	0	5
Motion	0	5	0	5	0	5	0	5	1	50	0	5	3	40
Size Heat	0	1	3103	70	1421	90	821	90	887	90	1642	90	1050	90
Motion IR	0	1	9	70	3	70	0	1	91	90	0	1	0	1
Final prob		**18**		**44**		**53**		**36**		**51**		**33**		**40**



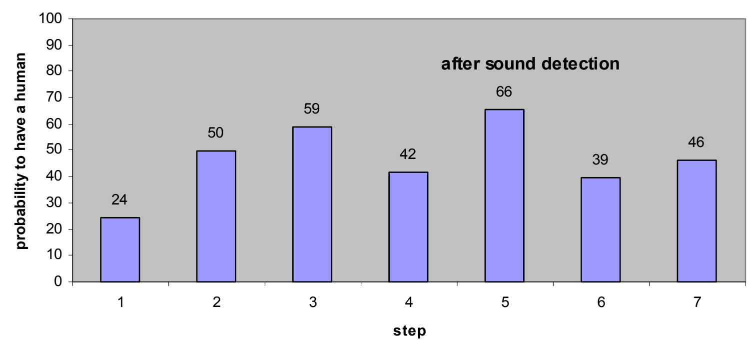


Table 8: measurements and results experiments 3

1 2 3

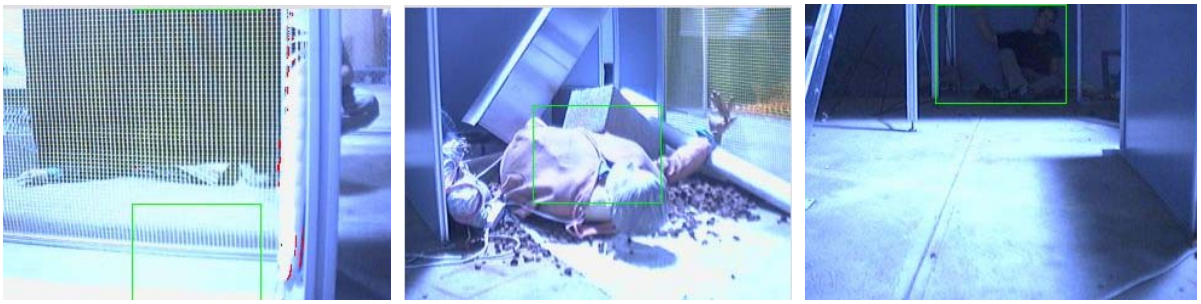


Figure 31: experiment 3, final results

This is the most complex environment of the three experiments but the most real. The results are interesting. On step two (see Figure 30) a human is at 6 meters far from the robot moving his arm. He is in the shadow. He is easily detected by the IR camera but no motion is seen because his arm has almost the same color than the background. The hot part of the victim in step 3 is also detected by the IR camera but the infrared picture of this simulated victim has not the shape of a human because there is only warm blanket inside the mannequin to simulate body heating. Its finger motion is so little than the algorithm does not detected it as well, the robot is too far. On the three last steps, a human is hidden behind a wall and a curtain, but the robot can see him through the mirror on the last step. The heat reflection is not strong enough to have a very bright shape of the reflection. So the IR camera does not detect something. The pyro sensor is quite accurate on that experiment because it detects the two humans and the simulated victim an does not detect the reflection heat in the mirror on the last step.

The first highest result (see Figure 31, 1) is an accumulation of false positive results. There is a motion in the plastic curtain in front of the robot. The human is partially hidden with a black curtain but the IR camera can see him through. Moreover the blob tracked is not the same between the two IR pictures and it involves a motion that does not exist. All the sensors have detected something but they don't belong to the same object. So the probability is high even if the human can not be reach directly.
In the opposite the two other results are correct. The body of the mannequin is well seen specially with the IR camera. But the motion of its finger was not detected. It was under 1 ‰.
For the third case, even if there is no motion detected, the probability is quite high because the confidence of the IR detection is high and the pyro has also detected the human.

In this area, which simulate disaster situation, there is a lot of different texture in the wall or objects that can interfere the results of the sensors. As we see, often one sensor do not have a good results alone. Its measurement is on an environmental object and not the human. In this area we can have a good overview how hard is the human detection in real condition.

6.5 Strengths and weaknesses of fusion

According to these experiments, data fusion is essential to have good results for human detection. Using only one sensor would involve many false results that can not be differentiated with right one. Then, using multiple sensors as we do, decrease the number of these false results. The relation between the sensors i.e. their confidence value is very dependant of the environmental condition. It is difficult to find a way, or an exact method to proceed for chosen this value. It needs a lot of tests and experiments and of course the more we know where the robot will evolve, the easier the choice of these values will be.
To improve the result of fusion, it is necessary to link spatially the measurement of each sensor to know if a measurement corresponds to the same object than the other measurement with another sensor. It can be done specially with the motion detection and the infrared picture.
Some parameters like confidence value are difficult to set because there is no precise method to calculate them.
The function $f_i(x)$ or lookup table could also be improved by having more possibilities of output value depending of the input. The result would be more precise and would have better resolution compare to every step. But we have to keep in mind that the Stayton board cannot use floating number, thus this improvement is limited.
Sensor fusion is very useful in complex environment to avoid detecting false results with only one sensor. Because no sensor is perfect, data fusion is very powerful to compensate the weakness of every sensor.
importance

7 INTERACTION BETWEEN USER AND ROBOT

Currently, in the heart of the USAR project at Carnegie Mellon is RETSINA. This is an intelligent multi-agent[2] System as explained in [12], [13] and [18]. The goal is to create a system where humans, software agents and robot can work together in teams to coordinate information, goals, plans and tasks in order to provide aid as quickly and safely as possible in the event of an urban disaster. The human must be able to work with a team of heterogeneous robots. As described in [5], experiences to determine the capabilities of humans working with USAR robots show that, the data the robots present to the human operator is very important in determining the success of the mission. Information must be distilled so that the human can make quick decisions. More recommendations for interaction between user and robot can be found in [4].

7.1 User interface for the sensors

Our approach is to have the robot complete a panorama of the environment, fuse the information from all of its sensors, and provide back to the human the three highest probabilities where a human may be found along with pictures of each location. The user can then select the interesting directions and tell the robot where to move.

The user interface for the sensors must support following commands:

- Change the parameters of each sensor manually
- Calibrate each sensor in an independent way
- Use each sensor in an independently and receive their information
- Survey the environment around the robot with all the sensors together and provide back any interesting information

The user interface was designed to meet these criteria and is shown in Figure 32:

[2] An agent is an autonomous, intelligent, collaborative, adaptive, computational entity. Given certain goals, an agent needs to have abilities to execute needed actions and seek and incorporate relevant information

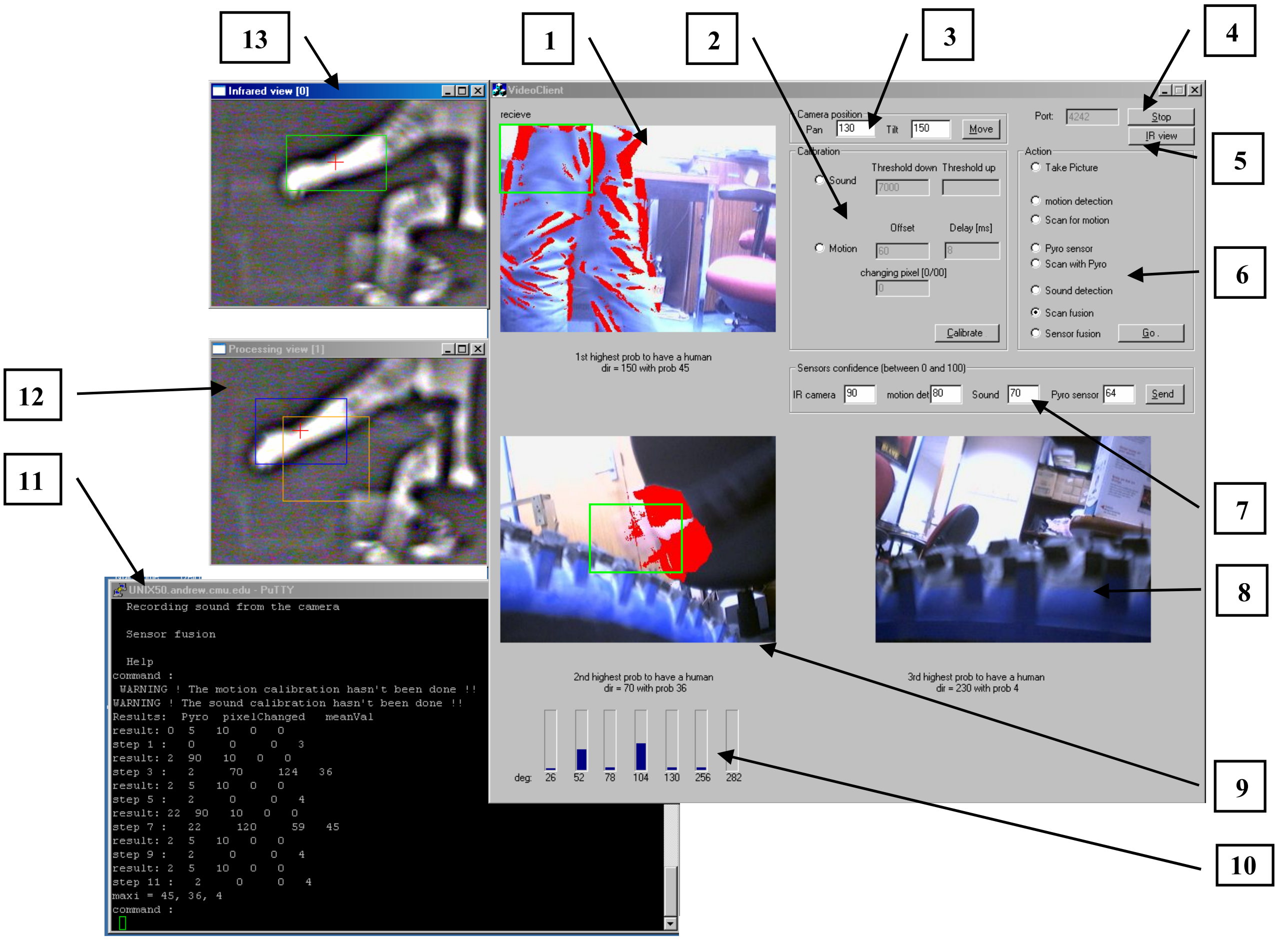


Figure 32: User interface for sensors on USAR robot

1) Result view 1: where there is the highest probability to find a human. Motion is overlaid in red and a green rectangle is displayed to locate where the warmest part on this picture is.
2) Calibration part
3) To move the pan tilt head
4) Open the connection with the robot. The port can be chosen
5) Start the Infrared view
6) Action to send to the robot
7) Confidence fields: the user can enter manually a confidence for each sensor and send them to the robot
8) Result view 3: where there is the 3rd highest probability to find a human. Same display settings than 1).
9) Result view 2: where there is the 2nd highest probability to find a human. Same display settings than 1).
10) Graphic bar: display the probability to have a human at each step of the 180° panorama scan
11) Robot terminal window
12) Infrared view 2: display the result of the infrared motion detection

13) Infrared view 3: display the infrared view in realtime and the tracking of the warmest part

As shown in Figure 32, the largest part of the interface is used to display pictures. The IR images are displayed in real time (13[3]) and the three pictures where there is the highest probability to have a human (1, 8, 9) are displayed on the main window. Then, the user has a fast good overview of the environment around the robot. He can easily select in which direction the robot can go because the robot has already selected the interesting direction (i.e. where a human could be).

7.2 User manual

First the user has to open the *videoClient* application on the external computer. In 4) the port is chosen and then Start is pressed. Then the user has to start the executable file *humanDet* on the robot with the IP address of the external computer and the port to open. The connection between the robot and the external computer is established. Then the user has three possibilities: calibrate the sensors in 2), move the pan tilt head in 3), change manually the confidence of each sensor in 7) or send a command on the robot to perform an action in 6).

- Calibrate the sensors:
  The user can choose between sound and motion calibration. For sound calibration he can put a threshold value manually in the field and then select the *Calibrate* button or execute an automatic calibration clicking only on *Calibrate*. For motion calibration, the user first selects a time delay between the pictures acquisition and an offset for the difference between these two pictures. After that, *Calibrate* is selected and the user can choose between an automatic or manual calibration.

- Move the pan-tilt head:
  The user first enters a value between 0 and 255 in the pan and tilt fields and then click on *Move* to change the pan-tilt head position on the robot.

- Change manually the sensor confidence:
  The user first enters a value between 0 and 100 in the sensor field of which he wants to change the default value. Then he clicks on *send* to send these values to the robot.

- Send a command to the robot
  In this field (6), the user can choose between different actions to execute. After that he clicks on *Go* to start this action on the robot. To do a 180° scan and find people *Fusion All* has to be selected after starting the IR view with 5) (*IRview* button).

[3] This number and all the next one in this paragraph refer to Figure 32

8 IMPROVEMENTS & FUTURE WORK

8.1 Hardware

After having the infrared camera, the pyroelectric sensor has become almost useless. The infrared camera provide much better information, it is more robust and more reliable. To replace it, another sensor which could be complementary to the IR camera is an infrared non-contact thermosensor used for object temperature measurement like use in [30]. It will be an interesting sensor to distinguish, when a warm object is detected with the IR camera, if it is on body temperature. However these sensors are more expensive than the pyroelectric. More information can be found in [33].

A important lack in sensor, is a long distance sensor to distance know how far the victim is. With this sensor it would be possible to have a function that give the size of known motion depending on distance and the same for the size of the hottest part detected in IR view. The final probability of human presence could be more accurate.

To have an easier hardware interface with the robot, it would be useful to connect the infrared camera directly on the embeded board of the robot and use its digital output. Then the framegrabber and the analog emitter/receiver would be no more useful and every computer could use to teleoperate the robot. With this improvement there are two solutions:

- Do not use tLib anymore for processing the IR images because it does not run very well on the stayton board (very slow and can not use float number) and do very simple image processing directly on the robot
- use a new board for the robot instead of the stayton, like PC104 board to connect the infrared camera and use tLib on it

8.2 Software

First, it would be useful to add a function in sound detection to detect regular banging. Indeed, in disaster area, sometimes victims are banging on wall or on ground because they can't shout. Moreover sound propagates itself better in structure than in the air. So if the robot could be able to detect regular banging it will have one more useful function for human detection.
With the actual sound detection it would be done quite easily by detecting several time a short high noise above a threshold higher than now. Then by adding a counter it would be possible to detect such a kind of sound.

The Infrared camera is a very efficient device for human detection. More work can be done on it to have better human detection. If the digital output are used or the serial connection, more settings can be changed to have better image quality depending on the environment. Furthermore detection of human shape could be done and overall a better fusion with the normal image to superpose the both could be done. There is already a lot of project using these both cameras and fusing the data.

For now the robot suggest the user in which direction there could have a human. After seeing the pictures the user chooses the best direction. But the robot could be more autonomous and take more decision without human intervention. After doing the panorama scan the robot could go back in the direction where there was the highest probability to have a human and take again measure in that direction to improve the reliability of the first measurements.

8.3 Extension to other applications

Currently this set of sensor is mounted on the USAR robot at Carnegie Mellon, but if the hardware platform is the same, it could be put on another robot. It is planned to use it on the PER (Personnal exploration Rover) at Carnegie Mellon for the next Urban Search and Rescue RoboCup. They will work with a little team of robot to do the search of victims in the arena. These two robots do not have the same physical shape but they are complementary. One is big and can drive fast to explore the arena whereas the PER is more little and drive slower. so it can perform more precise tasks.

Moreover human detection is not only used in urban search and rescue task. There are different fields of application that can use robot and then these kind of sensors to find people. It is possible to use this set of sensor on a completely different robot like robot for search and rescue in Earthquake, flooding or avalanche situation. Robot with this set of sensor could also be used for surveillance and security in buildings or other high security area.

9 CONCLUSION

The goal of this project was to provide a sensor suite for human detection in the urban disaster environment. The integration of these sensors on the existing USAR robot at Carnegie Mellon and their evaluation to detect victims was the second part of the project.

Many researches were done to survey the state-of-the-art in USAR robotics with special importance on sensors for victim detection. A list of all the currently available sensors was established to know which kind of sensor it was possible to use for victim detection.
For our application and compare to existing project, the following sensors were chosen with low-cost and lightweight as main criteria:

- USB camera with build-in microphone
- Pyroelectric sensor
- Infrared camera

	Algorithm selected	Advantage	Limitations
Pyro Sensor	-	only human heat detection	environmental dependencies, binary output
USB camera	motion detection	computationally not expensive	change in light intensity
microphone	long duration and high amplitude sound detection	computationally not expensive	ambient noise
IR camera	hottest part extraction / motion detection	human heat detection	other hot moving object

Table 9: summary of selected sensors main characteristics

The Table 9 shows the main characteristics of selected sensors. To have global information about human presence the data of all these sensors are fused with a method using confidence level for each sensor. This technique using multiple sensors improves the robustness of the final results.

The choice of confidence value is essential to have good results but they depend strongly of the environment. With experiments we see that this technique is efficient to find victims. The robot is able to give useful information to the user, and by selecting interesting direction it can suggest the user which place to explore.

The future work would be to improve detection having more reliable confidence function for each sensor and improve the choice of confidence value. To have a better quality of human detection, it would be a good solution to add a long distance sensor.
Finally, the most challenging part would be to maximize the autonomy of the robot to limit user attention on it.

10 Acknowledgment

Here are the people who have contributed to this project in a way or another. I would like to address my gratitude to them.

- Prof. Illah Nourbakhsh of the Robotics Institute of CMU, supervisor, for giving me a lot of advises and following this project very carefully.

- Prof. Reymond Clavel of LSRO of EPFL, supervisor, and Dr Charles Baur, VRAI-Group EPFL, for having accepted me in their Lab and given me the possibility to do my diploma abroad.

- All the USAR team from Carnegie Mellon University, especially Kevin Oishi for his help in software, Mark Yong for his help in hardware and Mary Berna for her help during all the project.

- Terry Fong, responsible assistant, VRAI-Group EPFL, for having set up the project and following it from Switzerland. Sébastien Granges, VRAI-Group, EPFL, for his help with the library tLib.

Pittsburgh, 20th of February 2004

Steve Burion

11 REFERENCES

- **Lectures**

[1] R. Siegwart, I. Nourbakhsh, "*Introduction to Autonomous Mobile Robotics*", EPFL & CMU, Lausanne, Pittsburgh, 2002

[2] H. Durrant-Whyte "*Introduction to sensor Data Fusion*", lessons, Australian Centre for Field Robotics, University of Sydney, 2002. Also at : http://www.acfr.usyd.edu.au/teaching/graduate/Fusion/index.html

- **Papers**

[3] A. Rogalski, K. Chrzanowski, "*Infrared devices and techniques*", Warsaw, 2002.

[4] J. Casper "*Human-Robot Interactions during the Robot-Assisted Urban Search And Rescue Response at The World Trade Center*", MS Thesis, Computer Science and Engineering, USF, South Florida, 2002.

[5] J. Casper and R. Murphy, "*Workflow Study on Human-Robot Interaction in USAR*", ICRA 2002, pp 1997-2003.

[6] Murphy R, Casper J, Hyams J, Micire M, and Minten B "*Mobility and Sensing Demands in USAR*", (invited), IECON 2000, Nagoya, Japan, 2000.

[7] L. Matthies, Y. Xiong, R. Hogg, "*A Portable, Autonomous, Urban Reconnaissance Robot*", The 6th International Conference on Intelligent Autonomous Systems, Venice, Italy, 2000.

[8] K. Kuhnly, "*A technical overview of pir motion detection*," Tech. Rep., ITI Technologies Inc, 2000.

[9] S. Bahadori, L Iocchi "*Human Body Detection in the Robocup Rescue Scenario*", Roma

[10] P. Bernasconi, "*Perception et communication pour robot de sauvetage*", Semester Project, ASL2 EPFL, Lausanne, 2003.

[11] A. Noth, "*Développement d'un système auditif pour le robot humonïde Robota*", ASL3 EPFL, Lausanne, 2003.

[12] Sycara, K., Paolucci, M., van Velsen, M. and Giampapa, J., "*The RETSINA MAS Infrastructure, in the special joint issue of Autonomous Agents and MAS*", Volume 7, Nos. 1 and 2, 2003.

[13] Paolucci, M., Onn Shehory and Sycara, K., "*Interleaving Planning and Execution in a Multiagent Team Planning Environment*". In the Journal of Electronic Transactions of Artificial Intelligence, 2001.

[14] P.C. Cattin, "*Person Detector for Mobile Robots*", EPFZ, Zürich

[15] D. Blank, G. Beavers, "*A Robot Team that Can Search, Rescue, and Serve Cookies: Experiments in Multi-modal Person Identification and Multi-robot Sound Localization*", University of Arkansas, Fayetteville, 2001.

[16] P.C. Catin "*Biometric Authentification System Usinng Human Gait*", EPFZ, Zurich 2002.

[17] S. Grange, T. Fong, C. Baur, "TLIB : a real-time computer vision library for HCI", VRAI-Group, EPFL, Lausanne, 2003

- **Web sites** (available on 5th of february 2004)

[18] CMU RETSINA Homepage: http://www-2.cs.cmu.edu/~softagents/index.html

[19] CRASAR Homepage: http://crasar.csee.usf.edu/

[20] CRASAR Project: http://crasar.eng.usf.edu/MainFiles/index.asp

[21] UVS Project (Japan): http://www.r.cs.kobe-u.ac.jp/uvs/

[22] Tokyo University: http://www.titech.ac.jp/publications/e/publications.html

[23] NASA urban robot: http://robotics.jpl.nasa.gov/tasks/tmr/homepage.html

[24] iRobot Homepage: http://www.irobot.com/

[25] Inuktun Compagny: http://www.inuktun.com

[26] Kentree Robotics : http://www.kentree.com/default.htm

[27] Different Infrared cameras:
http://www.ircameras.com/PDF/brochure_omega.pdf
http://www.infraredsolutions.com/html/IR160DataSheet.pdf
http://www.msanet.com
http://www.flir.fr/
http://www.raytheoninfrared.com/
http://www.raytheoninfrared.com/productcatalog/proditem19.html

[28] Glolab pyro sensor: http://www.glolab.com/pirparts/infrared.html

[29] SpO2 sensor and gas sensor:
http://www.micrelec.fr/catalogue/produitsvt.asp?num=390
http://www.geneq.com/catalog/en/q-check_co2-co_meter.htm
http://www.microsens.ch/products/gas.htm

[30] USAR Robocup contest:
http://robotarenas.nist.gov/competitions.htm
http://www.r.cs.kobe-u.ac.jp/robocup-rescue/

[31] Guide to the Stayton Board :
http://www-2.cs.cmu.edu/~mberna/stayton/tutorial.html#Downloads
http://openrobotics.ece.pdx.edu/

[32] Infrared camera information:
http://www.marlow.com/Applications/DSP/temperature_stabilized_uncooled.htm
http://shop.store.yahoo.com/spytechagency/thermalimaging.html
http://x26.com/infrared/images/fpa.htm

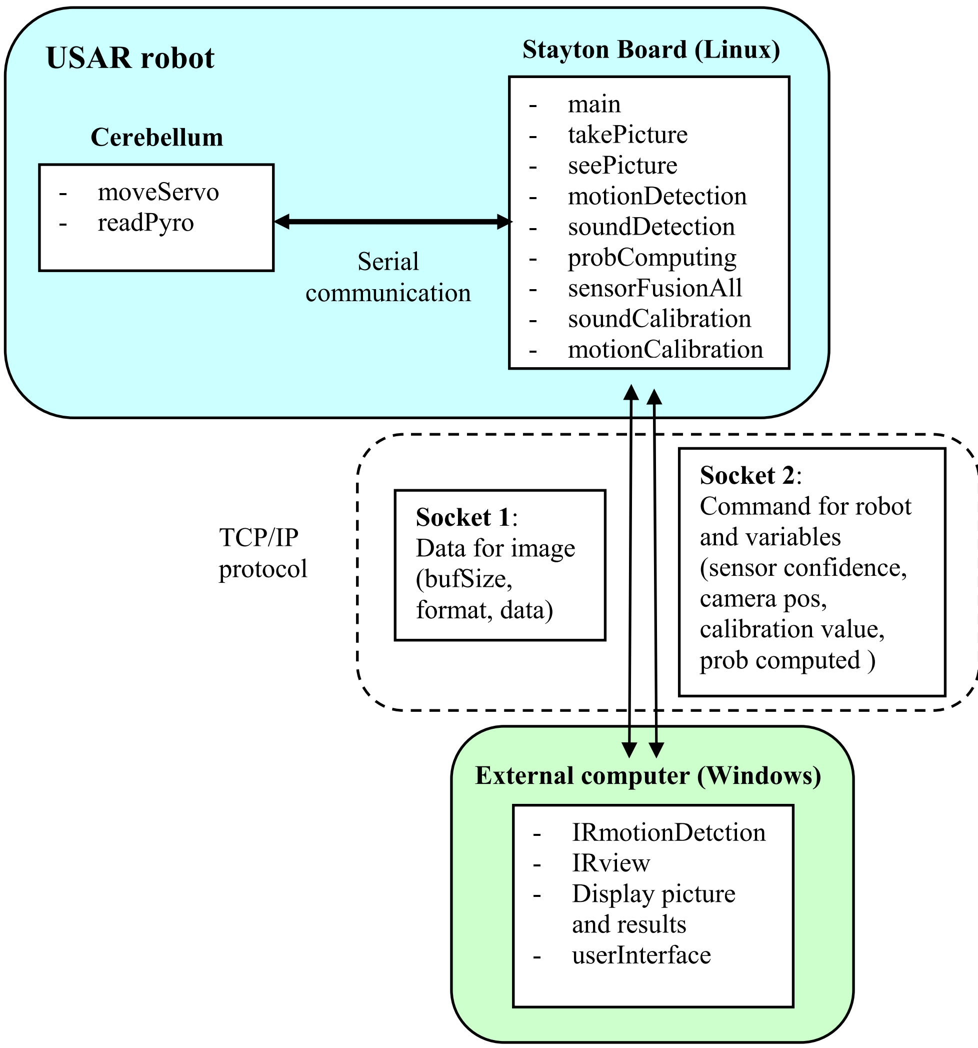
[33] Raytek Non contact infrared temperature mesurement :
http://www.raytek-northamerica.com/index2.html

[34] Millimeter-radar: http://www.getradar.com

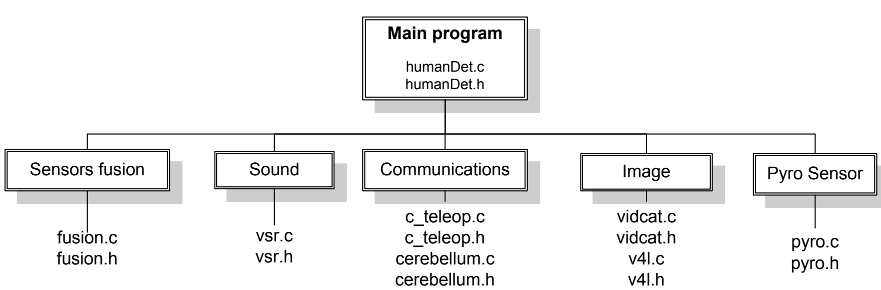
APPENDIX A - HARDWARE

APPENDIX B – SOFTWARE

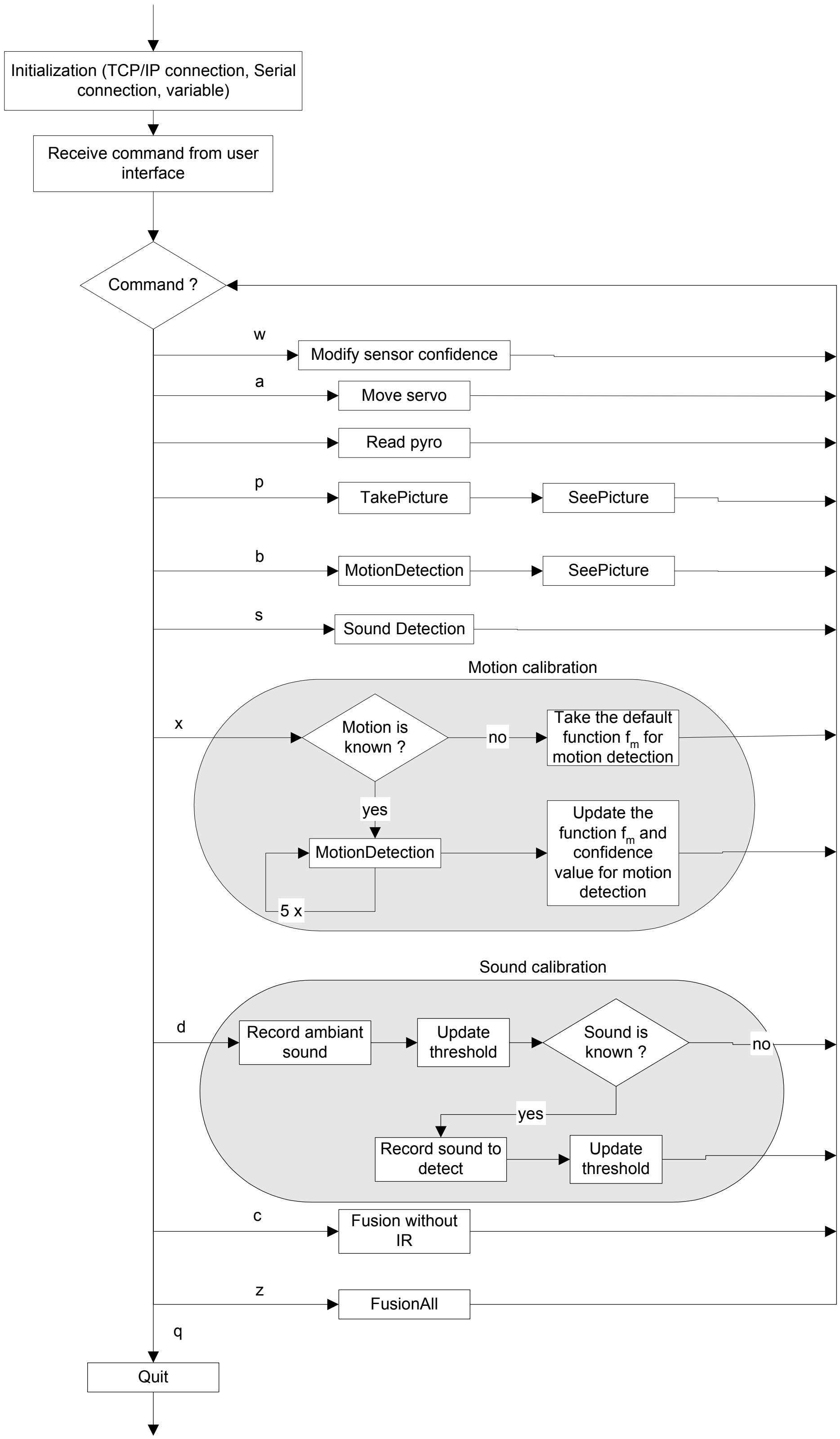
- Block diagram for main functions and communication protocol between the user interface and the robot



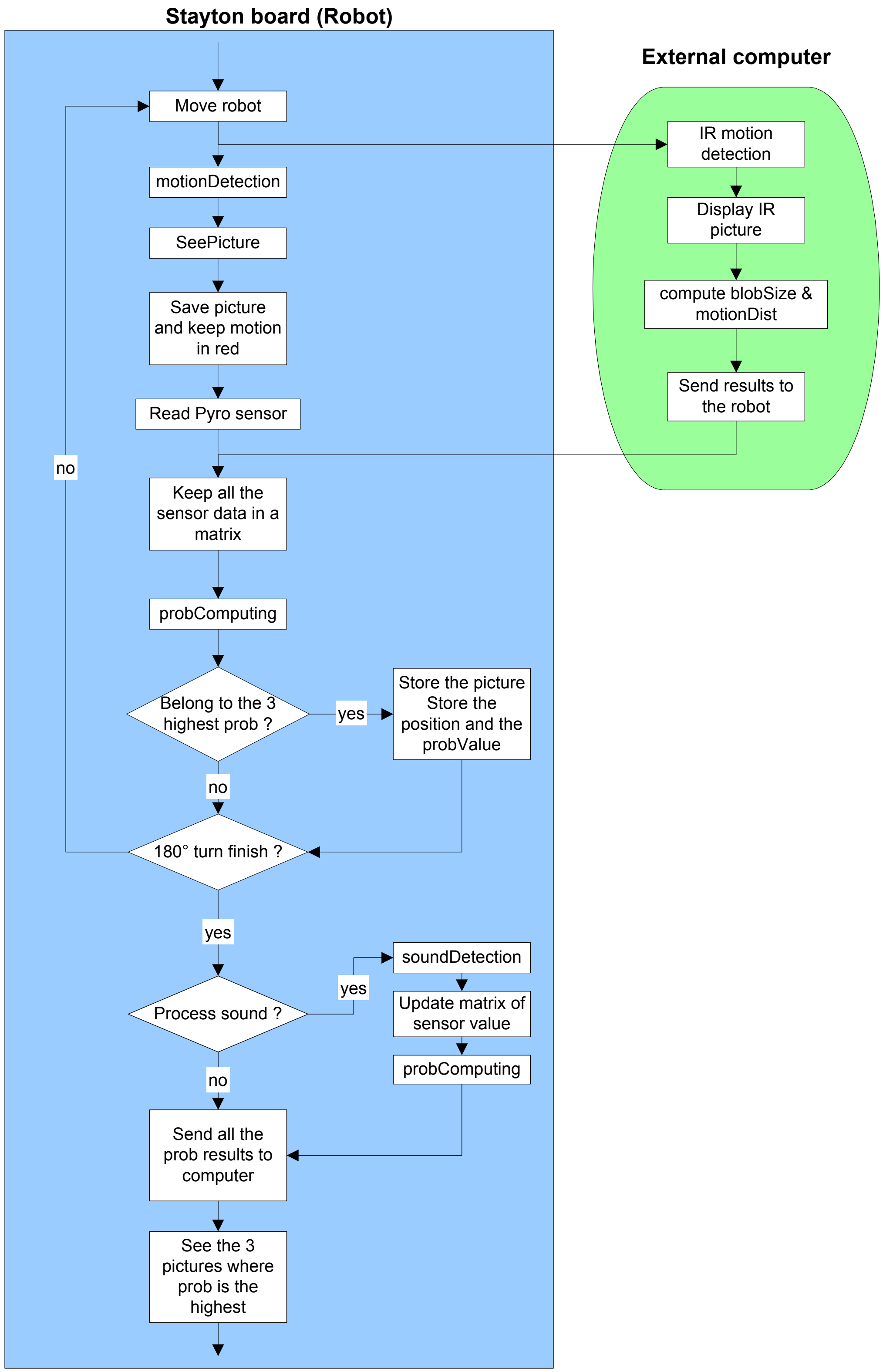
- List of the files with their belonging



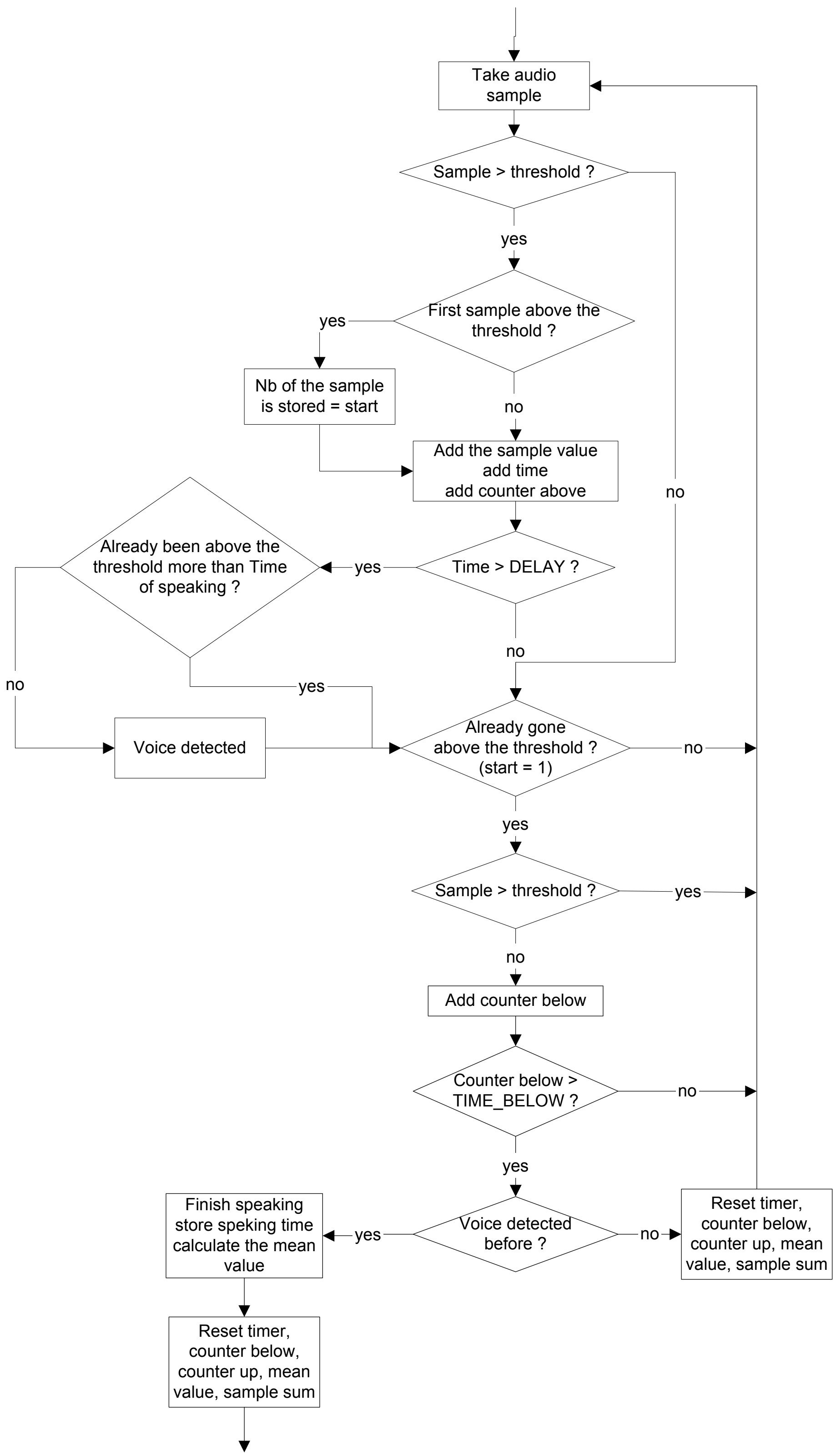
- Flow chart of `main` function in the file humanDet.c



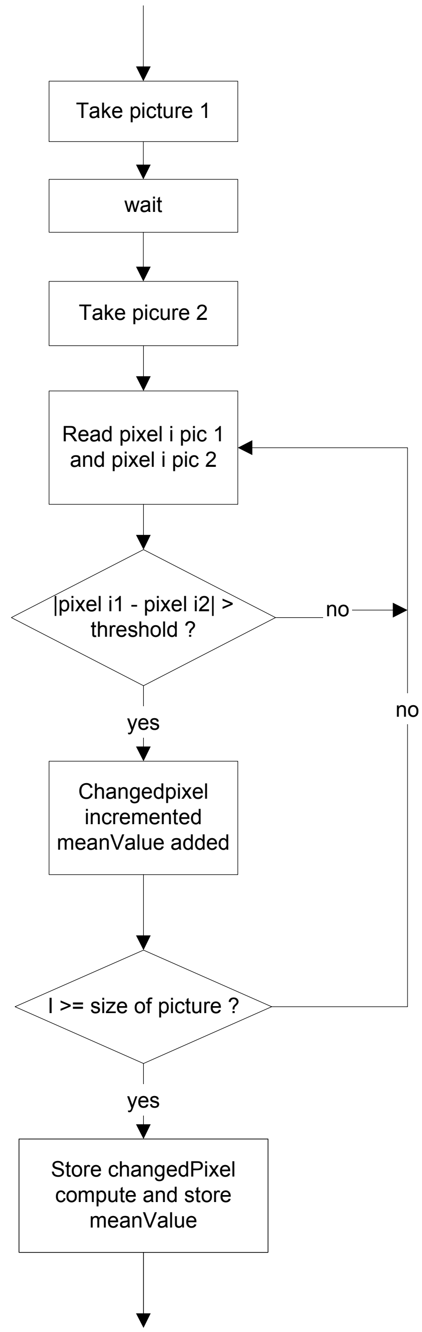
- Flow chart of `sensorFusionAll` function in the file humanDet.c



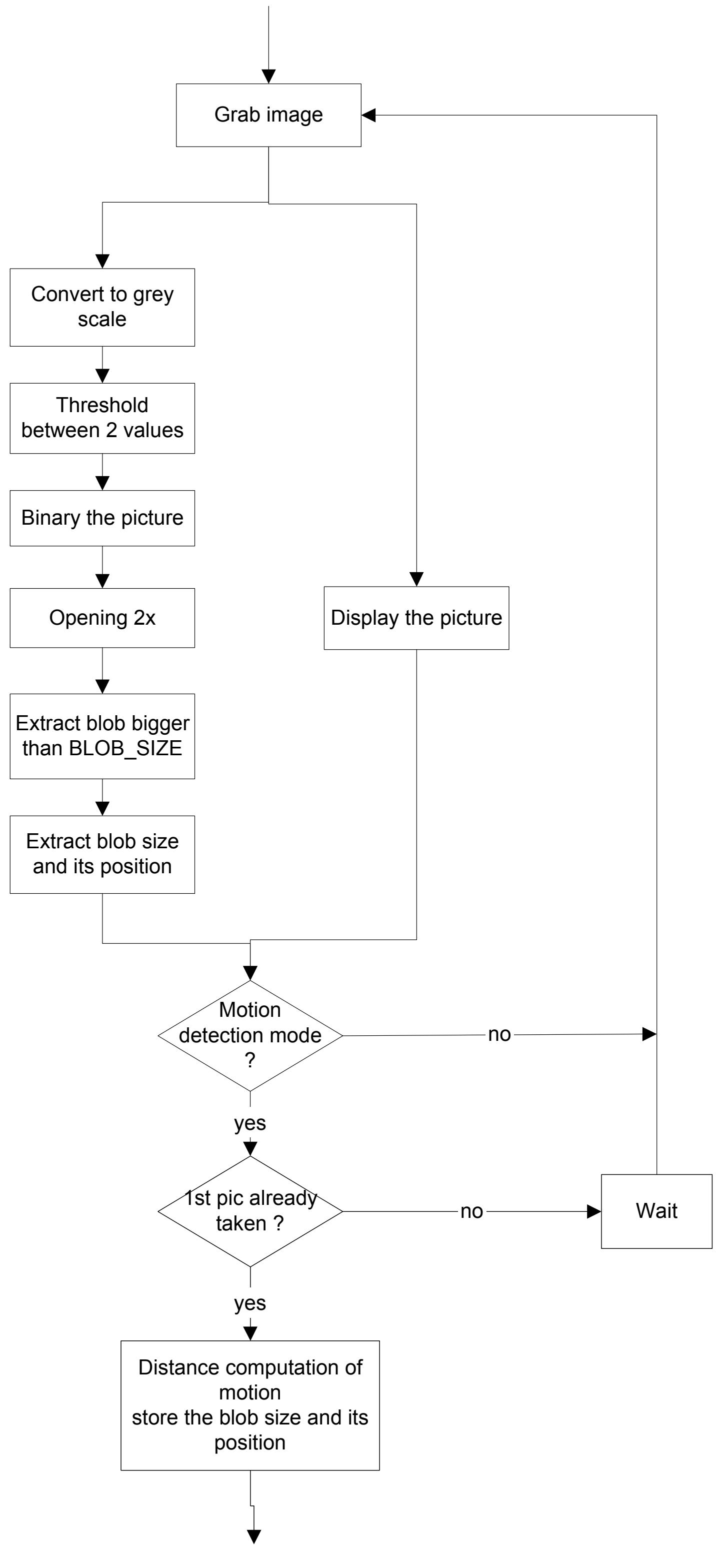
- Flow chart of `voiceDet` function in the file vsr.c



- Flow chart of `motionDetection` function in the file vidcat.c



- Flow chart of `IRcapture` function in the file IRvideo.c



APPENDIX C - List of the main functions

- **In File c_teleop.c**

`void wait_ms (int time)`
Wait during `time` in millisecond

`void initSerial (void)`
Initialize serial connection with cerebellum

`void initTcpip (int argc, char **argv)`
Initialize tcp/ip connection. Open two socket connections

`int initServo (int centerPan, int centerTilt)`
Put the pan tilt head straight

`runInitialization(int arc, char **argv)`
Call all the initialization functions

`void sendImageToServer(char *pic, char *fmt, int bufSize, char window)`
Send the picture in JPEG format to videoClient and choose in which window it will appear

`int seePicture(char *picture, char window)`
Convert picture in JPEG and send it to the server

- **In file Pyro.c**

`int pyroMotionDetection(void)`
Wait for a signal from the pyroelectric sensor and take a picture

`int pyroScanHuman(int startPan, int stopPan, int startTilt, int step)`
Do a horizontal scan with the pan tilt head and take picture when the pyro sensor has detected something

- **In file vidcat.c**

`void array_to_matrix(char *line)`
Transform a char raw picture to a matrix [colomn][line][value (R, G or B)]:

`char *matrix_to_array (void)`
Transform a matrix RGB picture in a char raw picture

`void BGRtoRGB(char *picture1)`
Transform a BGR raw picture in a RGB raw picture

`void drawCross (int centerX, int centerY)`
Draw a red cross at centerX, centerY

`void drawRectangle(int x, int y, int w, int h)`
Draw a green rectangle with upper left corner at (x,y) and w width and h height

`int initializeCamera ()`
Initialize the USB camera with the following settings
width = WIDTH_D, height = HEIGHT_D, palette = VIDEO_PALETTE_YUV420P

`char * takePicture (void)`

Grab an image from device in RGB char row format

`void motionDetectionCamera(int delay, int offset, int *changed, int *m)`

Compute motion detection by subtraction of 2 pictures. The pictures are taken in an interval of `Delay` in millisecond. `Offset` is a threshold to select when a pixel as changed. The function returns the number of changing pixel and their mean value

- **In file vsr.c**

`void sound(int recTime, int calib)`

Open the sound threads for recording and writing data; process sound in real time
if calib is 1, this is the calibration mode
Returns the mean sound amplitude and the speaking time

`void voiceCalib(int amp)`

For calibration: measure ambient noise

`int voiceDet(int amp)`

Process basic human voice detection in quiet environment

`int read16bit(unsigned char input1, unsigned char input2)`

Take the sample in wave format and convert in an integer value to process it

- **In file humanDet.c**

`int probComputing(int pyro, int change, int mean, int sizeHeat, int motionHeat)`

Compute the probability to have a human depending on the measure of each sensor. Take the value in every lookup table with confidence

`int processSound(int recordTime)`

record sound

`int sensorFusionAll (void)`

Do a 180° scan with the robot. At each step take a measurement with every sensor (pyroelectic, motion detection, receive value from IR image processing) and fuse all the data of every sensors then store the 3 images and positions where there is the highest probability to have a human

`int main (int argc, char **argv)`

Wait for a command from the user interface in VideoClient and execute the correspondent function

`void calibrateSound (int threshold)`

Record the ambient noise and a sound if the sound to record is known. Return the value of the threshold

`void calibrateMotion (int delay, int offset, int pixelChanged)`

Calibrate the motion detection by applying 5 times the motionDetction function on a known motion

APPENDIX D – SUPPLEMENTARY THEORY

- **More information about IR camera**

Here is a summary about the information found in [3], [27] and [32] to have a better idea how the infrared camera is working.

Infrared is the portion of he electromagnetic spectrum beyond the visible (blue to red, 0.4-0.75 µm) response of the human eye. IR wavelengths extend from 0.75 µm to 1000 µm, where the shortest microwaves (radar) begin. Because IR radiation is generated by heat, it is called thermal radiation.
With the introduction of new technology associated with modern focal-plane-array (FPA) infrared (IR) systems has become more and more used and powerful device. FPA is a plane of sensitive element. But not the entire surface of the detector is sensitive to IR energy. Around the rows and columns of individual IR detectors making up the array is an inactive region surrounding each of the detector. The inactive areas can serve as pathways for electronic signals. The ratio of active IR sensing material to inactive row and column borders is called the fill factor. An ideal detector would have a very high fill factor because it would have a large percentage of its area dedicated to collecting IR photons and a very small area dedicated to detector segregation. There are two types of infrared FPAs:

- monolithic
- hybrid

Monolithic FPAs have both IR-sensitive material and signal transmission paths on the same layer. Conversely, monolithic FPAs generally have lower performance than their hybrid counterparts because having the detector material and signal pathways on the same level results in a significantly lower fill factor (~55%).
A **hybrid** array has the IR-sensitive detector material on one layer and the signal-transmission and processing circuitry on another layer. The two layers are bonded together to transmit the signal from each detector element to its respective signal path on the multiplexer below. Although this process requires more steps and can be more expensive, it results in FPAs with a significantly higher fill factor (~75%-90%) and then a much higher thermal sensitivity than the monolithic FPAs. Some hybrid FPA cameras provide sensitivity down to 0.02°C.

In their camera, Raytheon uses an uncooled detector which converts the focused LowWavelength IR energy into an electrical signal. The detector material is a pyroelectric ceramic composed of barium strontium titanate (BST). It uses a change in the dielectric constant of the material with temperature changes and the resulting change in the capacitance is termed the ferroelectric effect. Because this reaction is optimized at room temperature, the detector is referred to as an "uncooled" sensor due to the lack of a cryogenic cooling system. But to use it in different environmental conditions with temperature changing, they use a thermoelectric cooler based on the Pelletier effect. The cooler must be put into the heat mode when using the camera in a cold environment or into cooling mode to reduce the detector temperature when the environment is warm. Thus the thermoelectric cooler is thermodynamically reversible.