Technical Report: isda06-002
June 1, 2006

Peter Bajcsy, pbajcsy@ncsa.uiuc.edu
Rob Kooper, kooper@ncsa.uiuc.edu
David Scherba, dscherba@ncsa.uiuc.edu
Martin Urban, murban@ncsa.uiuc.edu

Image Spatial Data Analysis Group
National Center for Supercomputing Applications
1205 W Clark, Urbana, IL 61801

TOWARD HAZARD AWARE SPACES: KNOWING WHERE, WHEN AND WHAT HAZARDS OCCUR

Abstract

While considering all existing hazards for humans due to (a) natural disastrous events, (b) failures of human hazard attention or (c) intentional harmful behaviors of humans, we address the problem of building hazard aware spaces (HAS) to alert innocent people. We have researched and developed components of a prototype HAS system for detecting fire using wireless "smart" micro electro-mechanical systems (MEMS) sensor networks, such as the MICA sensors manufactured by Crossbow Inc., and spectral cameras, for instance, thermal infrared (IR), visible spectrum and multi-spectral cameras. Within this context, the paper overviews technical challenges and prototyped scientific solutions to (1) robotic sensor deployment using remote control, (2) synchronization of sensors and cameras, (3) localization of sensors and objects, (4) calibration of measurements from sensors and spectral cameras, (5) proactive camera control, (6) hazard detection, (7) human alert, (8) hazard confirmation and understanding, and (9) hazard containment. The work presents innovative signal processing applications of sensor network technology, and includes also theoretical and practical limitations that have to be understood while working with novel technologies.

1. Introduction

Wireless sensing devices are frequently used in smart spaces, ubiquitous and proactive computing, and situation awareness applications [35], [44], [47]. One could list a plethora of applications suitable for the use of wireless sensor networks and other sensing instruments, for instance, health care (wellness system for aging), environmental monitoring (pollution of air, water, and soil), atmospheric science (sever weather prediction), structural health monitoring (equipment or material fatigue detection), military surveillance (vehicle movement detection), wild life monitoring (animal migration), or intelligent vehicle design (obstacle detection) [11], [16], [24], [45], [32]. The list of on-going wireless sensor network projects is also growing every day (see NSF and NIST projects [26], [31], [38], [39], such as NSF NEON, LOOKING, SCCOOS, ROADNet, USArray, TeraBridge, ORION, CLEANER or NIST SHIELD). We have been interested in the hazard awareness application scenarios [4], [5] that concern humans due to (a) natural disastrous events, (b) failures of human hazard

attention or (c) intentional harmful behaviors of humans. Our focus is on the problems related to building hazard aware spaces (HAS) to alert innocent people.

While building a real-time HAS system, one has to address the issues of (1) setting up the system to achieve desired accuracy and (2) operating it to achieve reliable performance with or without human intervention. In order to setup a HAS system, one ought to find ways how to deploy sensors, synchronize them, localize sensors and other instruments in the environment, and calibrate measurements coming from wireless sensors and instruments to obtain values represented in engineering units (for example, a raw value of temperature has to be converted to degrees of Celsius). These steps for HAS preparation allow us to answer questions about when, where and what hazards occur during the operation of a HAS system. In addition, one has to understand the limitations of 'smart' wireless sensor networks (WSN), such as low-power, broadcast range, available on-board memory and CPU, to optimize the layout of sensor networks in terms of minimal wireless loss, minimal energy consumption and maximal information content received from the network. From the perspective of operating a HAS system, the objective is to perform reliable proactive data acquisition, hazard detection, human alert, hazard confirmation and possible understanding of specific hazard characteristics, and finally hazard containment. These building steps have been reflected in our research and development, and are illustrated in the overall HAS schema in Figure 1.

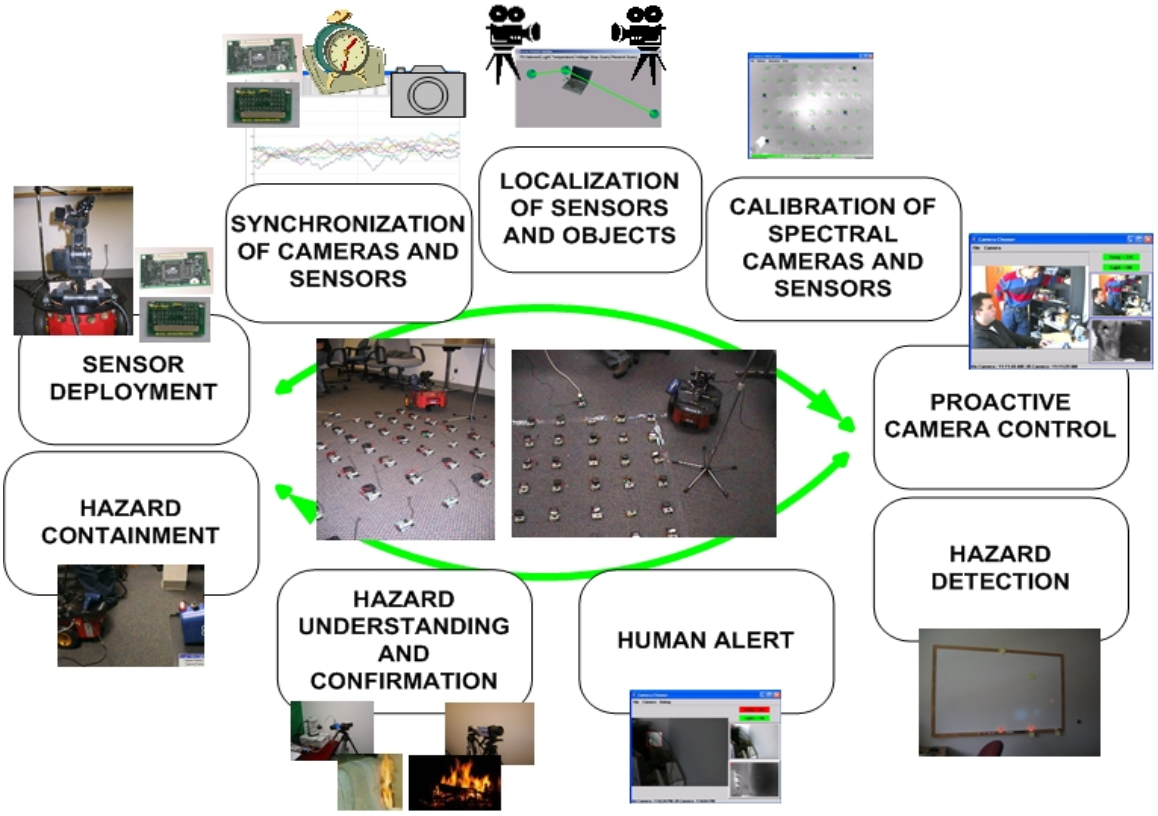

Figure 1: An overview of several components of the hazard aware spaces (HAS) prototype. The top components represent the setup of HAS while the bottom components correspond to the operation of HAS.

2. Research and Development Challenges

Research and development of HAS poses several fundamental challenges in the areas of sensing, remote and proactive control, wireless data acquisition, wireless communication, signal and image processing, and detection and pattern recognition. It is the understanding of these issues that leads to an optimal real-time system design of HAS. In this paper, we elaborate on the research themes about (a) adaptive remote HAS setup, (b) measurement accuracy and sampling rate, (c) communication bandwidth control and information selection, (c) hazard understanding from sensor and image data, (d) human-computer interfaces for human alert, and (e) the use of robotics in HAS application domains. The research themes map into a development of technology components illustrated in Figure 1 and include (1) deployment of point sensors using remote robot control, (2) synchronization of sensors and cameras, (3) localization of sensors and objects, (4) calibration of measurements from sensors and spectral cameras, (5) proactive camera control, (6) hazard detection, (7) human alert, (8) hazard confirmation and understanding, and (9) hazard containment. The rest of this paper presents hardware and software for building a prototype HAS system, and theoretical and experimental solutions to the aforementioned research and development technology components.

3. Hardware and Software Description

In our HAS system design, we used the MICA hardware that is manufactured by Crossbow Inc. The MICA hardware consists of (1) 4MHz Atmega 128L processor, (2) 128K bytes Flash, 4K bytes SRAM and 4K bytes of EEPROM, (3) 916MHz radio transceiver with a maximum data rate of 40Kbits/sec, (4) attached AA (2) battery pack and (5) plug-in sensor boards like the MTS101CA, attached through a 51-pin expansion connector. For more details, see [25], [14], [15], [39], [40].

The MICA sensors are deployed using an intelligent wheeled robot P2DX8 made by ActivMedia Robotics, Amherst, NH, [1] and an on-board computer-processing unit for real-time processing. The robot has a ring of 8 forward sonar sensors that can be used for obstacle avoidance and a two-wheel drive plus balancing caster for smooth motion. The robot is connected to a laptop that is either directly cable-connected to the local area network or wirelessly connected with other computers. For the gesture-driven remote control of the robot, we used the IS-300 Pro Precision Motion Tracker by InterSense, MA, [18] with the update rate of 500 Hz, weight of 15 oz, and angular resolution of 0.02 Deg. It measures yaw, pitch and roll using a miniature solid-state integrated inertial instrument “InertiaCube” and these temporal signals serve as inputs to our gesture recognition algorithm. For the voice-driven remote control, we used wireless audio sensors by Audio-technica Corp. To obtain video feedback from the robot, we mounted a pair of wireless miniature color cameras by Samsung on the robot’s platform.

The hazard aware space was equipped with a visible spectrum camera (Network Color Camera SNC-RZ30N PTZ Pan/Tilt/Zoom by Sony, and Canon PowerShot SD100 digital camera) and by a thermal infrared (IR) camera, (the Omega model by Indigo Systems Corporation, Goleta, CA). The thermal IR camera is a long-wavelength (7.5-13.5 microns) uncooled microbolometer camera designed for infrared applications. It is

controlled via RS232 serial port and the analog NTSC video output is digitized using a Hauppauge WinTV board. For temperature calibration experiments, we used a regular thermometer used by chemists as the temperature gauge. It is measuring temperature directly in engineering units of degrees Celsius and providing temperature readings in the range [-40°C, 150°C] with a reading uncertainty equal to ±1°C. A set of preliminary experiments to discriminate burning materials was performed with a hyperspectral camera by Opto-Knowledge Systems Inc. This camera is based on a liquid crystal tunable filters (LCTF) technology and operates in two wavelength ranges, such as visible [400nm, 720nm] and near infrared [650nm, 1100nm].

4. Sensor Deployment

When it comes to deploying a large number of sensors in a potentially hazardous environment, one would like to engage robots for sensor deployment. The challenges lie in combining the capabilities of robots with a remote robot control that would be human friendly. We addressed the above challenges by exploring multiple inputs for remote robot control including voice, gestures and human-computer interfaces (HCI), and developing real-time recognition algorithms [21], [42], [43].

The remote robot control system consists of three basic software components including (a) acquisition and recognition of control commands from multiple inputs, (b) client-server network communication, and (c) command fusion and execution by a robot and its arm. The inputs for recognition of control commands come from (1) wired or wireless microphones, (2) wired orientation sensors mounted on human arms, and (3) HCI devices, such as a mouse, a keyboard or a text file with the sequence of control commands. The set of gesture commands is based on the US Navy lexicon for navigating aircrafts on the ground. Fusion of multiple commands is performed by (a) analyzing time delays and (b) assigning different priorities to commands and the clients issuing those commands. Consistent and conflicting commands are considered before a selected command is executed by a robot. For an emergency control, a video signal is sent to a monitoring station. The overview of the system with multiple inputs is presented in Figure 1.

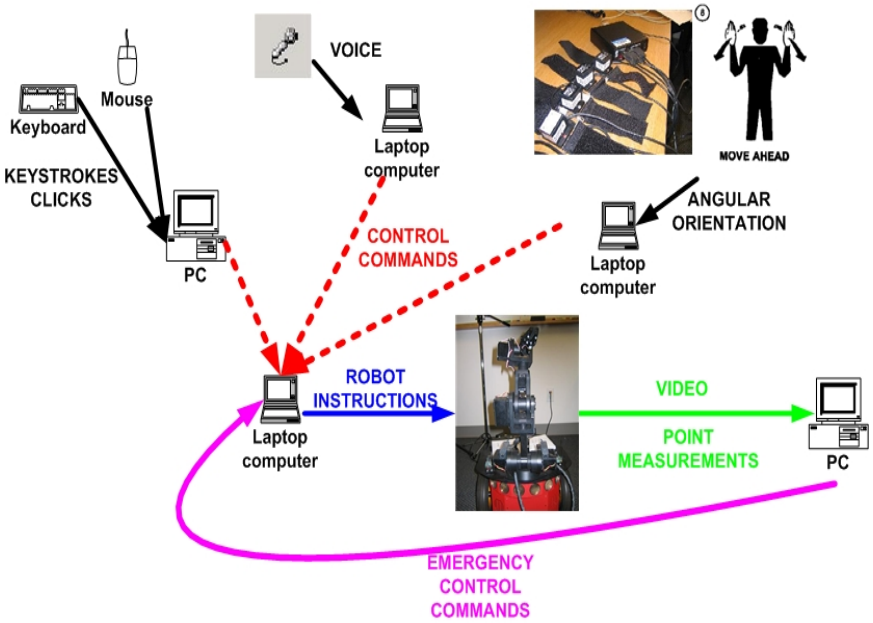

Figure 2: An overview of a system for remote robot control using sound, gesture and human-computer interface inputs.

First, we implemented robot control using HCI inputs. A user can use a keyboard and type in commands of his choice and their corresponding parameters. Second, we developed a template based speech recognition system so that typing can be replaced by more user friendly interface. Third, we added a gesture recognition system to accommodate remote control execution in very noisy environments, for instance, a carrier deck. Finally, we enabled robot arm control via mouse and keyboard interfaces in order to perform simple loading and unloading operations. For emergency control purposes, we mounted a wireless camera on the platform of a robot to obtain video feedback.

In terms of system architecture, the software is designed based on a client-server paradigm. All input devices (microphones, orientation sensors, keyboard and mouse) are attached to multiple computers that represent the clients in the developed system. In our laboratory experiments the robot is connected to a laptop using the RS232 connection. This laptop acts as a server and accepts TCP client connections over the network. Each client can issue control commands to the robot by sending commands to the server laptop. The server fuses the commands from all clients, and resolves any conflicts that may occur. After command conflicts are resolved the commands are translated to a set of robot instructions. These instructions are sent to the robot via the RS232 connection, and are then executed.

The command signal processing flow is illustrated in Figure 2, and it presents the fusion of multiple robot controls. According to Figure 2, data are acquired first by using one of the previously mentioned controls. After the data are collected, they are classified using a recognition algorithm into one of the 21 possible taxiing commands (text and arm inputs do not require classification). Once the command is recognized, it is sent as a packet over a local area network (LAN) to a server that is directly connected to the robot. Next, the server fuses all incoming commands and decides which one will be sent to the robot.

These commands are then translated to robot instructions which are executed by either the robot or its arm. Finally, a camera mounted on the robot provides video feedback that acts as another client controlling the robot. The server can handle multiple client connections at once and decides what commands should be executed.

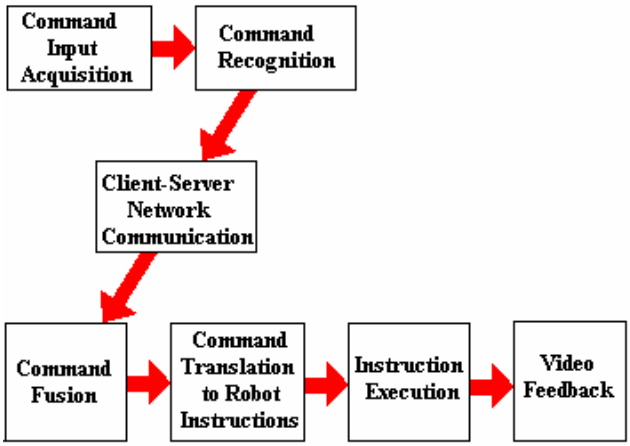

Figure 3: The signal processing and communication flow in a remote robot control system.

We are currently working on a new client application that will track an object and the robot will follow it. This can be used in two different scenarios. In the first scenario the robot will follow a person (see **Figure 4** left) tracking the color of the shirt of the person. In the second scenario the robot will scan the room for flames and move towards the flame (see **Figure 4** right). In both cases the system will use the camera mounted on the robot platform to take video snapshots of the environment. The video snapshots are analyzed in real-time to detect objects of known color and bound by changes of the projective region size that is adaptively adjusted over time to accommodate for "zoom effects." (object is growing as it comes closer to a camera) The tracking system is trained with a specific color of an object of interest prior to execution.

In order to use multiple color cameras without re-training the colors, one has to develop mappings from one RGB camera to another RGB camera since the cameras report different color values even if the viewed object is the same. For example, one encounters the problem of color camera mapping when a camera mounted on the robot platform is broken and replaced, or the tracking system is passing the tracking task from mobile camera to a static pan/tilt/zoom camera in a room. This leads to an interesting problem of how to make sure that red viewed by camera 1 is the same red viewed by camera 2. We will show a solution to this problem in section 10.1.

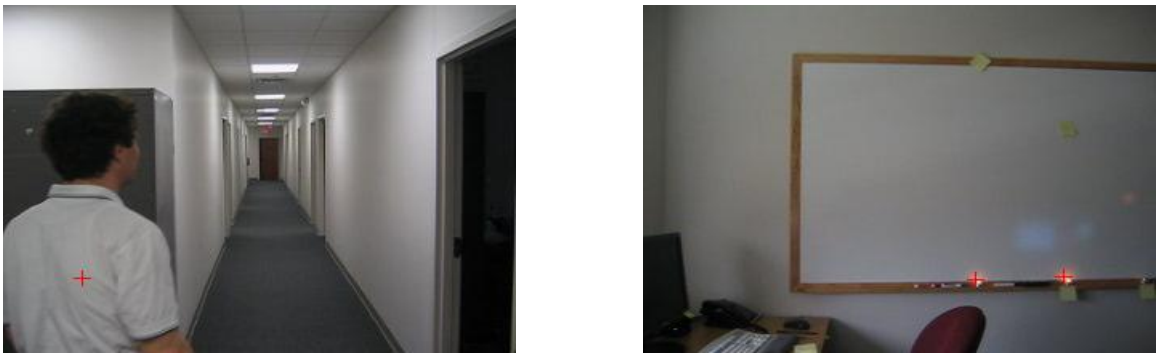

Figure 4 Robot tracking a person on left, and robot tracking flames in a room on the right.

5. Synchronization of Sensors and Cameras

In order to obtain useful sensor readings, one would like to know when the sensor readings were taken. Thus, our goal was to synchronize cameras with the deployed MICA sensors, or the MICA sensors with cameras.

5.1. Temporal synchronization

To achieve the synchronization goal we have to collect the data from both cameras and MICA sensors with accurate timestamps. These timestamps combined with the location of each sensor and camera would allow us to associate a MICA sensor value with an image pixel value. Based on our synchronization needs, we designed and developed a temporal calibration technique that follows the schema in Figure 5.

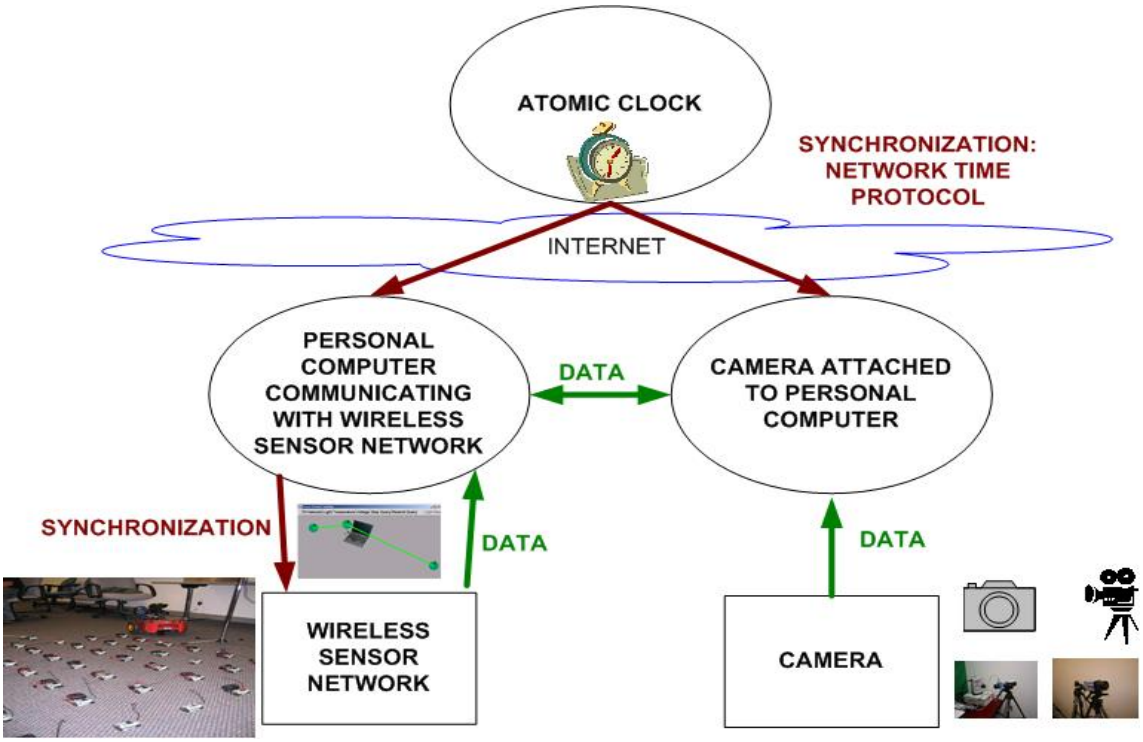

Figure 5: A schema for time synchronization of MICA sensors and cameras.

Cameras are attached to a PC and the PC uses its internal clock to set a timestamp for every captured image (the time the first bit of the image is received). If cameras are attached to multiple PCs then the PCs are synchronized using the standard NTP synchronization protocol. The MICA sensors communicate through an interface board with either the same PC as one of the cameras, or a different PC. In the case of the different PC we assume the camera PCs and the MICA PC are synchronized using the standard NTP synchronization protocol.

The MICA sensors can be viewed as small autonomous PCs. We need to synchronize the time of these small PCs with the PC receiving data. One approach is let the PC time-stamp the incoming messages. However, if there are some delays in the network or due to processing of the incoming messages, then the timestamps of the MICA sensor readings would be incorrect. We approached this problem by implementing a simple time synchronization of the MICA sensors with the PC. At the MICA initialization, the PC will send the current time to each of the MICA sensors. A MICA sensor will update this timestamp every Xms (X=10 in our case). Each reading on the MICA sensor will be sent back to the PC with the timestamp of the reading. One can also compensate for the network transfer time (or round trip time). If the difference between when the reading is taken by the MICA sensor and the time when the reading is received by the PC is larger than Yms (Y=350 in our case), then the PC and the MICA sensors will resynchronize their time.

5.2. MICA sensor time measurements

With the TinyOS development environment comes a generic application called TinyDB [40]. This application is a general purpose tool which makes it easy to obtain sensor readings from the MICA sensors. The syntax for requesting MICA readings is a SQL like language. For example, to request the current temperature on the MICA sensors every 500ms we can simply issue a TinyDB query "`SELECT temp FROM sensors EPOCH DURATION 500`".

Unfortunately, we observed after some initial experiments that the time between the sensor reading and the PC time was increasing. **Figure 6** shows the time difference between MICA sensor and the PC clock (Y-axis) as a function of the time when the reading was taken (X-axis). We also noticed that with "`EPOCH DURATION`" less than 400ms we would not receive as many readings as expected. These two problems made us discard this general purpose application and create a custom application for the MICA sensors.

The described time problem is caused by the MICA sensors using 1024 ticks per second, while all TinyDB applications are using 1 tick equal to 1ms. This discrepancy leads to 2.5% error in the timing. The time problems were corrected in our custom application as it is illustrated in **Figure 7** (removed temporal dependency of variables). In addition, the use of our custom application simplifies the request and data transfer and speeds up the time between requesting data being transmitted and receiving the first data.

Regarding data transmission, data can be transmitted across a network using TCP (Transmission Control Protocol), UDP (User Datagram Protocol) or a custom protocol. When data is transmitted using TCP the network layer will make sure no messages are lost and any lost data are retransmitted. UDP on the other hand does not make any guarantees about delivery of the data, often referred to as best effort. We know that data are transmitted at a high rate. Any lost message can be safely ignored since we know more up-to-date information will be transmitted shortly. It is our understanding that TinyDB uses a TCP like protocol where it guarantees no message is lost. From the data transmission perspective, we ignored lost messages since (a) we do not need all messages and (b) we are only interested in the most recent measurement.

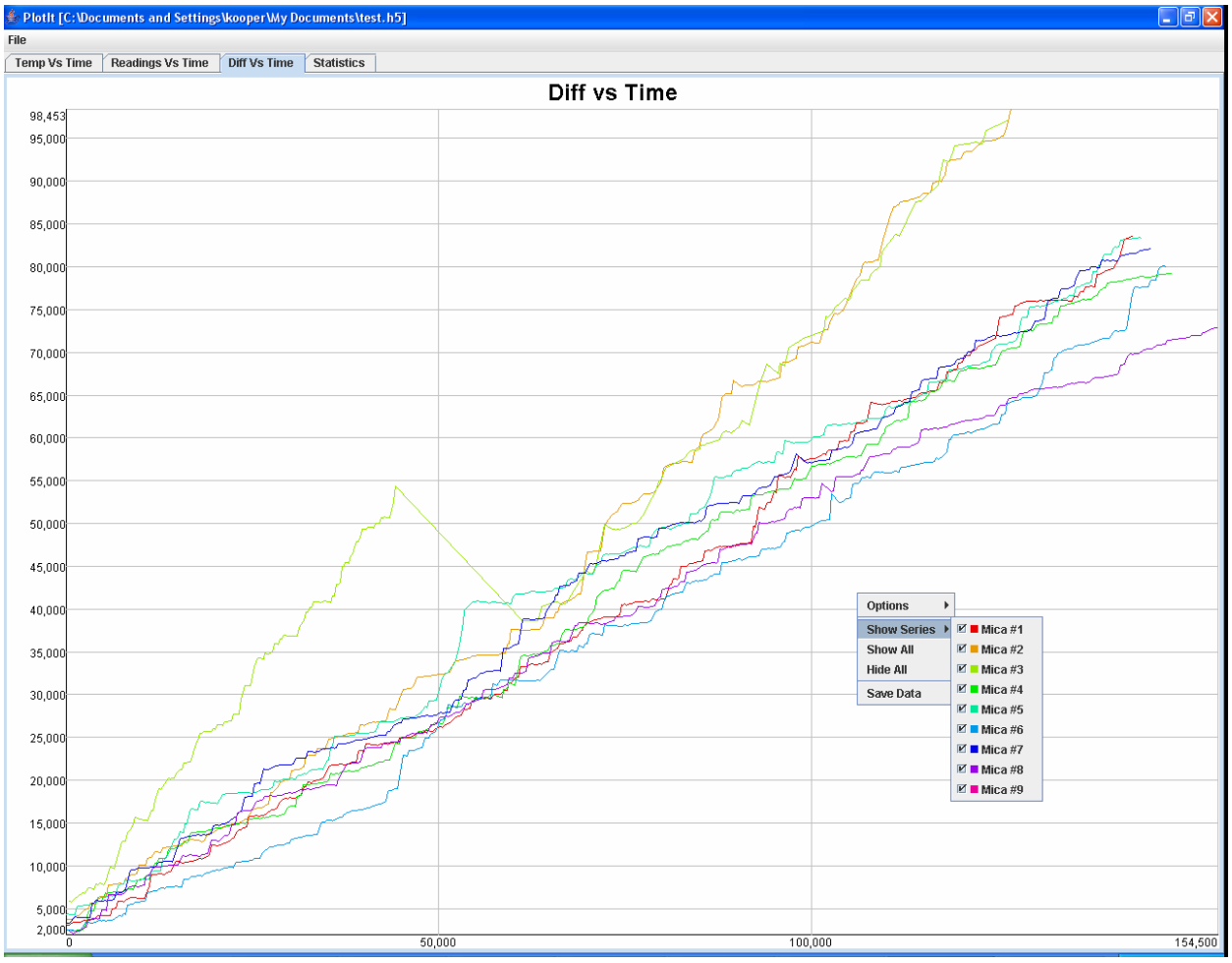

Figure 6: Time difference between MICA sensors and the PC clock (Y-axis) as a function of the time when the reading was taken (X-axis). Results obtained using TinyDB software.

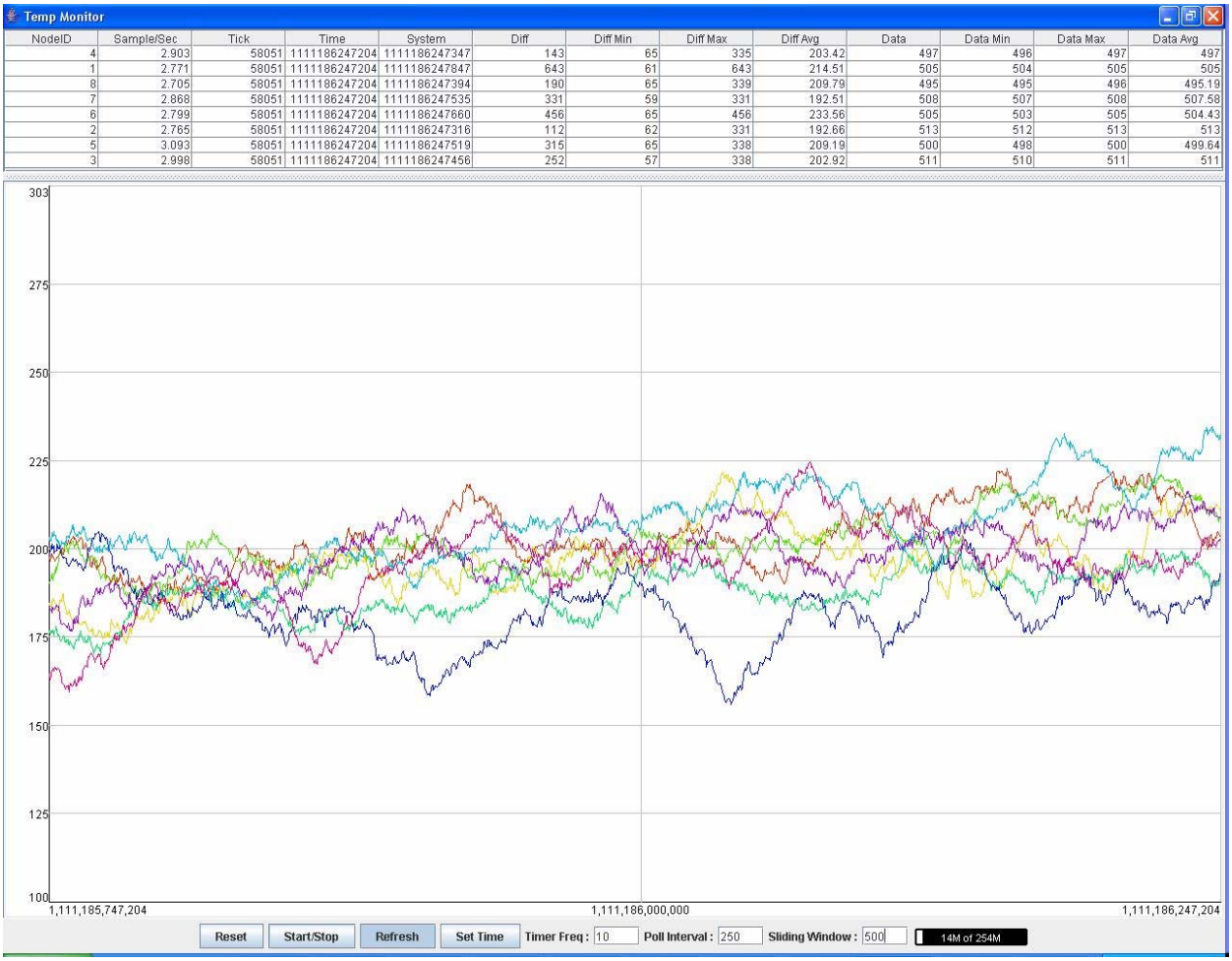

Figure 7: Time difference between MICA sensors and the PC clock (Y-axis) as a function of the time when the reading was taken (X-axis). Results obtained using custom software.

6. Localization of Sensors and Objects

Another important aspect of any HAS system is the knowledge of where hazards occur. The general problem of 3-D information recovery has been addressed in the past by many researchers in the computer vision, machine vision and signal/image processing communities [9], [23], [29], [46] and in the wireless communication community [3], [27], [49], [50]. The motivation for obtaining 3-D information often comes from applications that require object identification, recognition and modeling. In the case of HAS, one would like to determine locations of the deployed wireless MICA sensors and of other objects in the scene assuming that the camera locations are known.

The problem of 3-D information recovery is difficult regardless of whether it addresses static or dynamic object location estimation. In the past, the problem of depth recovery was approached, for example, (a) by vision techniques referred to as shape from cues [28] where cues can include stereo, motion, shading, etc…, and (b) by communication techniques frequently referred to as location sensing (radio or ultrasound time-of-flight lateration or signal strength analysis [14]). Although the vision and location sensing

techniques have been proposed, very few methods are robust and accurate enough to be used in real-time applications. It is well known that many of the depth estimation algorithms are computationally expensive with limited robustness and accuracy in most unconstrained, real-life applications. The need for improved robustness and accuracy of depth estimation motivated our work on stereo and wireless sensor location fusion.

Our approach to the 3D information recovery problem is based on fusing localization data from wireless sensor networks with depth maps obtained through computer vision stereopsis [36]. One could envision performing (1) depth map calibration, (2) sensor location calibration, or (3) depth map and localization fusion. A flowchart depicting the entire process from raw data to calibrated or fused information is shown in Figure 8. We have performed several experiments with synthetic and measured data using the Crossbow MICA2 motes, TinyOS, and Image to Knowledge (I2K) implementation of the stereo algorithm [8].

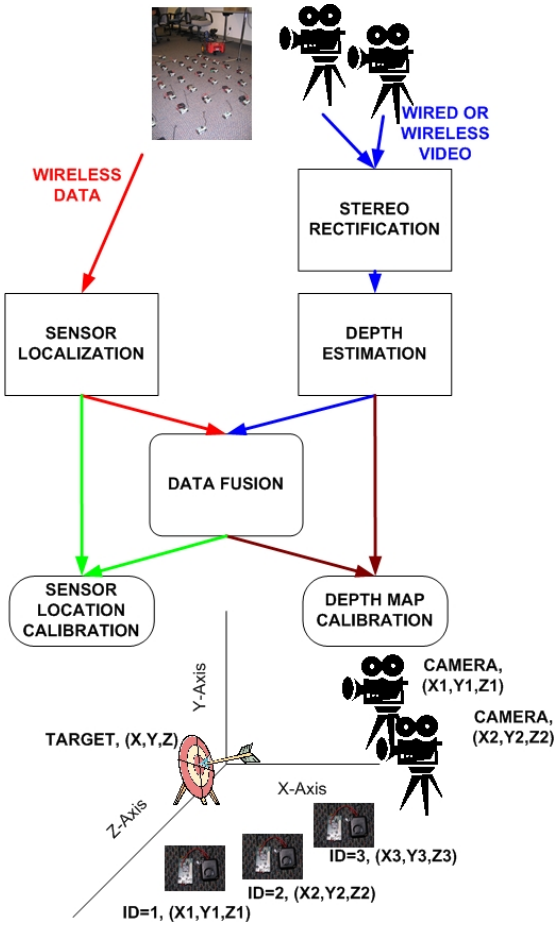

Figure 8. Flowchart of Sensor Fusion

The first component of our fusion system is a pair of visible spectrum cameras (Canon PowerShot SD100 digital cameras). Contrary to wireless sensor networks (WSNs), cameras are viewed as traditional sensors and have proven to be reliable, relatively

inexpensive, and suitable for collecting a dense set of measurements (a raster image) from their environment. Many techniques have been developed in the past two decades that can extract shape information from images and video [46], and using multiple cues [28]. In our HAS system, we use the stereopsis algorithm implemented in I2K software [8] to derive a depth map.

The second component of our fusion system is the localization of wireless sensors. In our work, we use the MICA sensor capability to record sound with a microphone and broadcast sound with a speaker. MICA sensor locations are determined via an acoustic time-of-flight ranging technique. Acoustic time-of-flight ranging is both an accepted and easily implemented ranging technique. Figure 9 shows the strategy of time-of-flight ranging we implemented. The first step is to send a message to a ranging endpoint node. The endpoint node, after receiving the message, simultaneously broadcasts a radio ranging message with a 4 kHz chirp. Every node in the network is configured to listen for the radio ranging messages and starts a timer which stops when the audible chirp is heard. A broadcast message announcing the distance between the endpoint and receiving nodes is then sent for all who are interested.

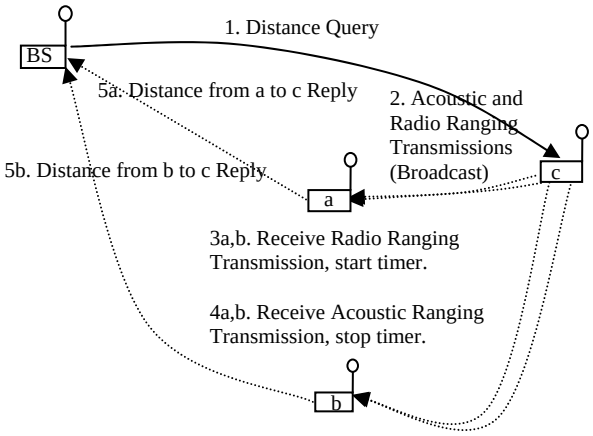

Figure 9**.** Acoustic Time-of-Flight Ranging. BS is the base station connected to a computer. The MICA2 motes are labeled a, b and c.

Finally, the third component is the fusion of sensor localization and stereo depth map results. This operation is performed by (a) registering the localization and depth map data, (b) estimating the uncertainty of localization and depth map data as a function depth distance, and (c) fusing the two data by minimizing the uncertainty over the entire depth range. The registration problem is approached by either global optimization or local model-based fitting. The global optimization is achieved by minimizing the difference between depth values and localization values in the least squares sense by solving a non-linear set of equations (number of MICA sensors is equal to the number of equations) using a downhill simplex search. The local model-based fitting approach assumes that a set of apriori known sensors is co-planar. Then, the registration is performed by (a) fitting

a 3-D surface to a set of apriori known co-planar sensor locations, and (b) computing the registration transformation parameters.

In order to fuse the data, the uncertainty of localization and depth map data as a function depth distance is estimated theoretically and verified experimentally. The theoretical uncertainty estimates are derived from a stereo depth disparity equation and from modeling point-point ranging/localization error. Figure 10 shows the determination of a fusion threshold for a particular choice localization and depth map uncertainties. We developed simulation capabilities for any range of input uncertainties to determine desired fusion thresholds as illustrated in **Figure 11**.

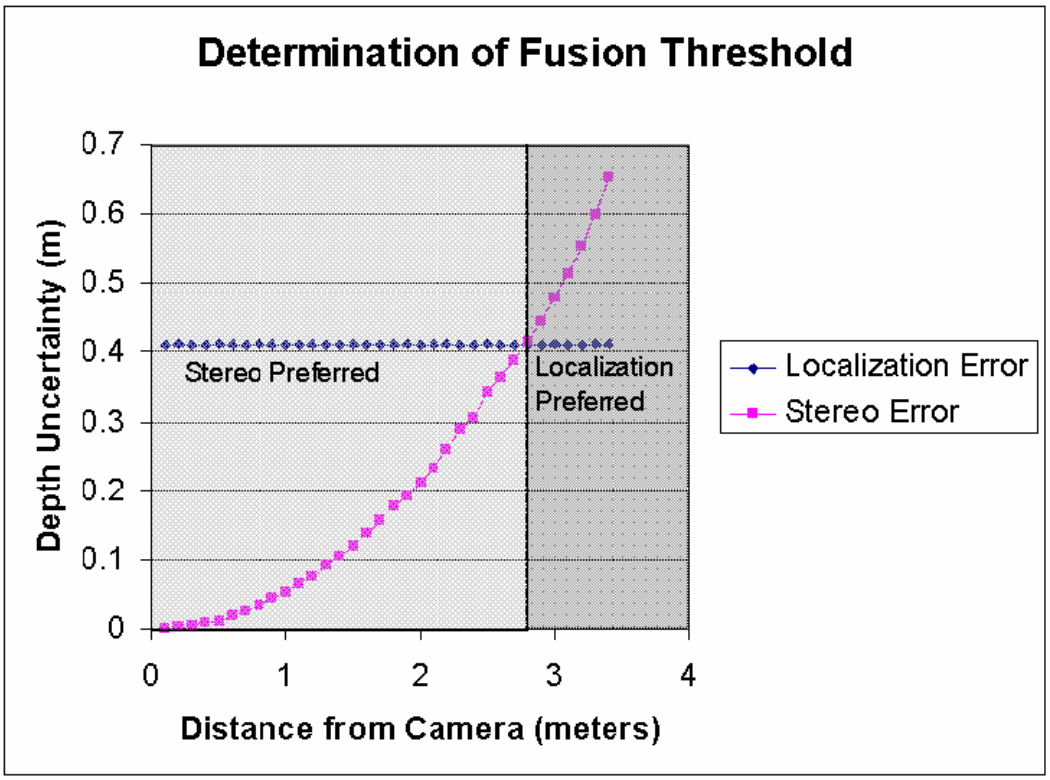

Figure 10: Uncertainty of localization and stereo depth map as a function of depth (distance from camera).

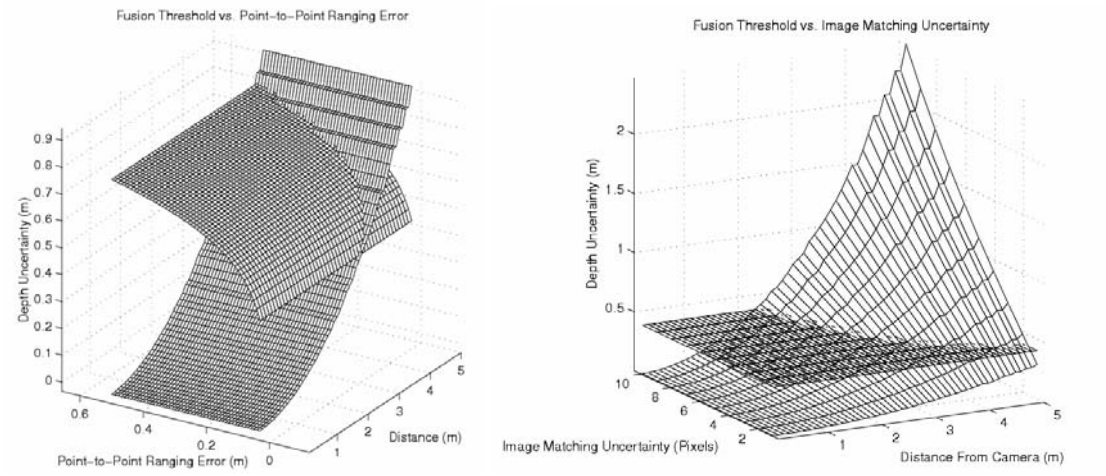

Figure 11: Theoretical predictions of fusion threshold from the intersection of two surfaces representing depth map uncertainty and localization uncertainty as a function of distance from camera. Left – intersecting surfaces for variable point-to-point ranging/localization error and fixed depth map error. Right – intersecting surfaces for variable stereo depth map error (proportional to image matching uncertainty) and fixed point-to-point ranging error.

To illustrate the last step of the fusion component, we performed laboratory experiments according to **Figure 12**. The pair of input stereo images is shown in **Figure 13**. We fused the localization and stereo depth map data sets by minimizing the uncertainty over the entire depth range. The resulting depth map is shown in **Figure 14** and the accuracy improvements are summarized in **Table 1**. The results in **Table 1** demonstrate a significant promise of the prototyped solution and its effectiveness.

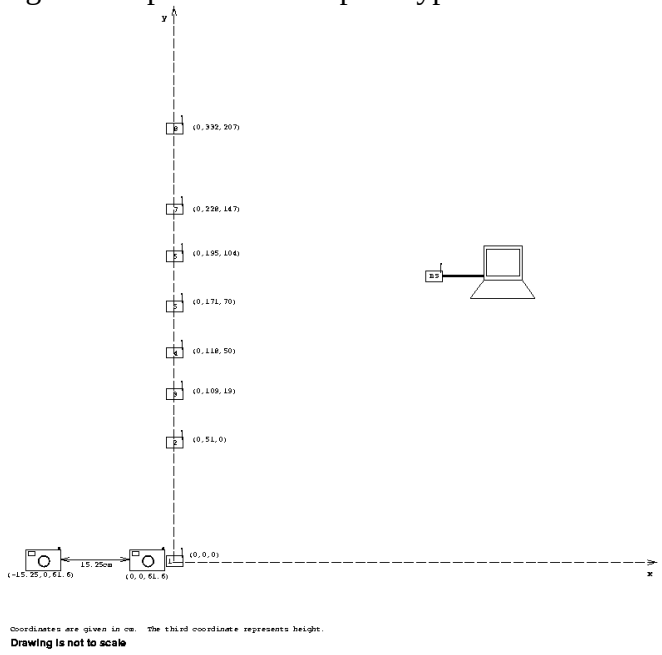

Figure 12: Schema of laboratory experiments to demonstrate localization and stereo depth map fusion. The MICA sensors are arranged in a (y,z) plane and communicate with

the base station attached to a computer illustrated above. The stereo pair of pictures is taken along the horizontal x-axis that is perpendicular to the sensor plane.

Figure 13: A stereo pair of images taken for the fusion laboratory experiments.

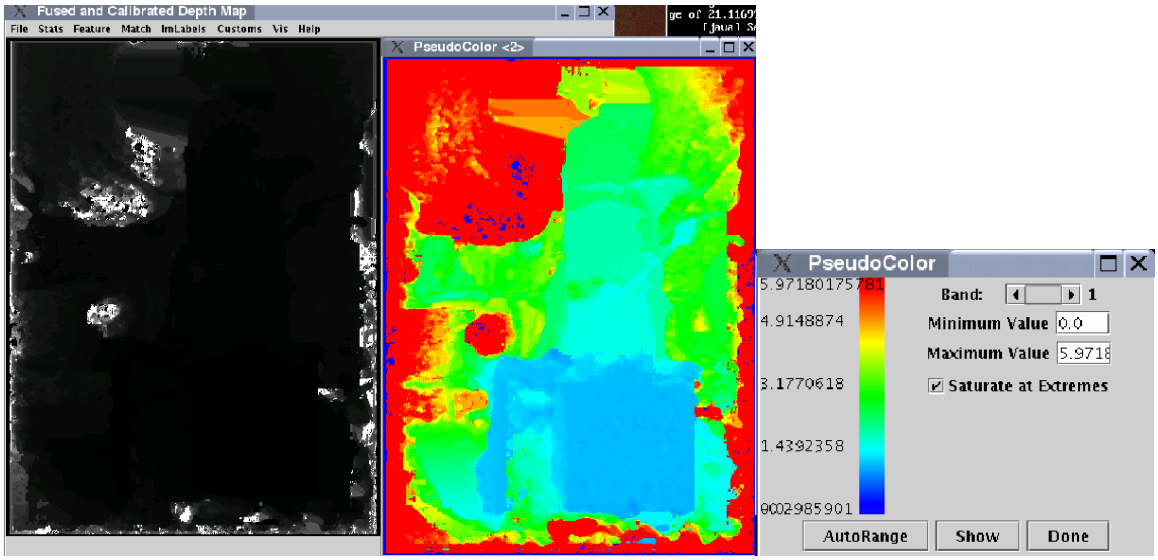

Figure 14: Resulting depth map after fusing localization and stereo depth map data. Left – input depth map, middle – pseudo colored depth after fusion, and right – the color legend showing the range of depth values in the pseudo colored depth map.

Table 1: Fusion results evaluated in terms of accuracy improvement.

Data types	Error [m]	Accuracy Improvement [%]
Localization error before fusion	1.19	27%
Localization error after fusion	0.87	
Depth map error before fusion	31.06	98%
Depth map error after fusion	0.72	
Sum of data errors before fusion	32.25	95%
Sum of data errors after fusion	1.59	

7. Calibration of Measurements from Sensors and Spectral Cameras

Our objective is to detect and recognize hazards, for instance fire. A combination of MICA temperature sensors and thermal IR cameras provides a desired test bed of fire sensing capabilities. We strive to perform continuous low bandwidth data acquisition using MICA sensors to monitor fire hazards and then trigger high bandwidth data acquisition using thermal IR cameras to recognize types of fire hazards (e.g., shape of flames and their spatial extent).

The use of both, MICA sensors and thermal IR cameras, requires accurate calibration of raw sensor readings and camera pixel values. One has to convert sensor and camera raw values into engineering units, such as degrees of Celsius or Fahrenheit or Kelvin, otherwise the raw values cannot be used for detection and recognition purposes. This conversion is also denoted as a spectral calibration since temperature (a variable representing thermal wavelength range) could be replaced by any other spectral variable, for instance, a variable that represents visible spectrum, near infrared or radar wavelengths. **Figure 15** shows our proposed schema for calibrating raw values. We view this step as one of the many necessary steps leading to scene and hazard understanding.

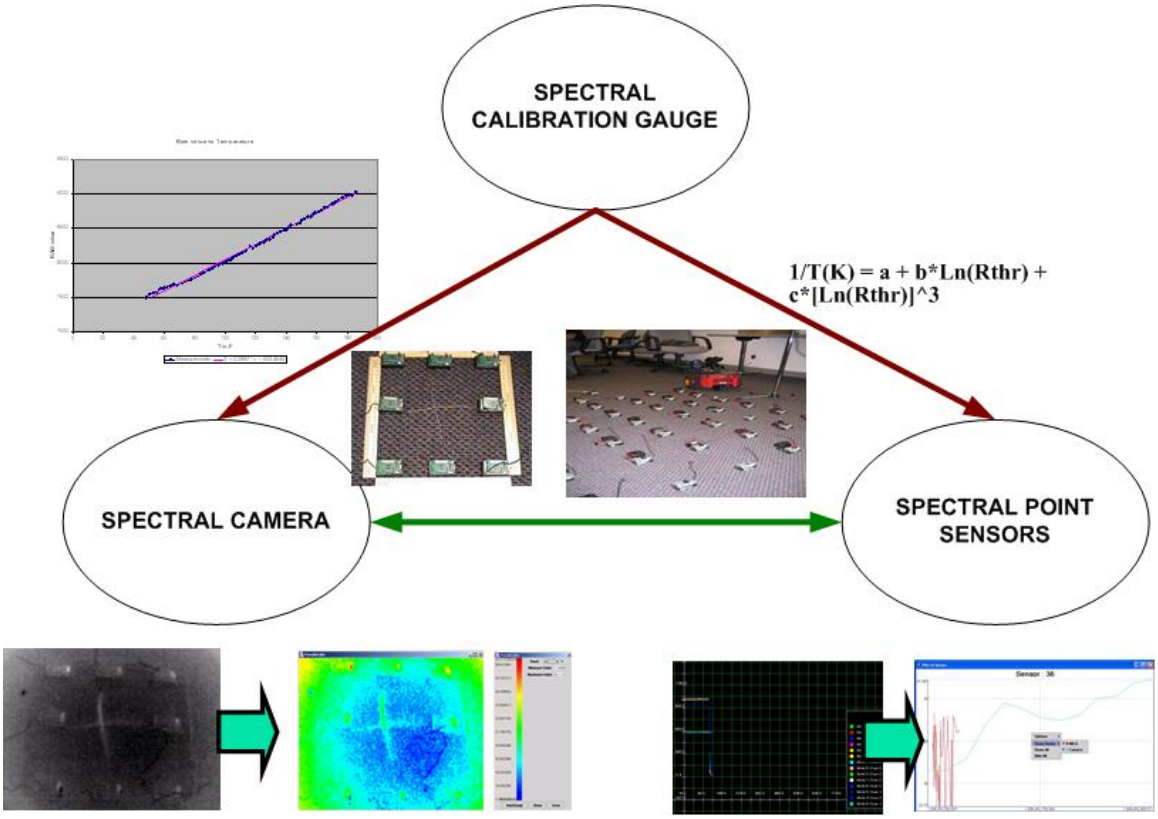

Figure 15: Calibration schema for MICA sensors and spectral cameras using an appropriate spectral gauge. We used temperature sensors on a MICA sensor board, thermal IR camera and a thermometer as a calibration gauge in our experiments.

7.1. Calibration of Thermal IR Camera using MICA Sensor Readings

First, we explored the calibration of thermal IR images using pre-calibrated MICA sensor readings. The problem of thermal IR camera calibration occurs in all application domains where automation using machine vision is highly desirable. One can find the need for thermal IR camera calibration in many other areas, for instance, in remote sensing (radiometric and photogrammetry calibration of aerial and satellite imagery), robotics (vegetation detection using near calibrated infrared and red wavelength imagery), astronomy (brightness estimation of stars using thermal IR imaging of the sky) or military (battlefield analyses). We foresee the use of widely distributed and deeply embedded “smart” micro electro-mechanical systems (MEMS) sensors as potential thermal IR calibration gauges for thermal IR cameras in future.

A simple calibration hardware setup is shown in Figure 16. The calibration procedure can be described as follows assuming that all sensors and cameras were synchronized and their locations are known. First, MICA sensors are programmed to sense and send temperature readings over a certain time period. Second, during the same time period, temperature measurements are collected with a thermometer (a calibration gauge). Third, a calibration transformation is established for MICA temperature sensors using a factory recommended formula and verified with thermometer measurements. Fourth, both thermal IR camera and MICA sensors are initiated to acquire data by broadcasting a

RESET signal to MICA sensors and triggering thermal IR camera acquisition. Fifth, MICA sensors transmit every set (packet) of temperature measurements with the state of the internal counter (time stamp) to the base station attached to a personal computer (PC). In meantime, the thermal IR camera acquires data with the time stamp of the CPU clock counting from the RESET signal. Sixth, the MICA raw temperature measurements are received and transformed into degrees Celsius. Seventh, MICA temperature sensor locations in the thermal IR image are identified, and statistics of the transformed MICA temperature measurements and the thermal IR image pixel values at the MICA sensor locations are related to form the final calibration transformation. In this step, if the entire scene viewed by a thermal IR camera is temperature homogeneous then MICA temperature sensor locations in the thermal IR image do not have to be identified and statistics of the thermal IR image can be computed over the entire image. Illustrations of raw and calibrated thermal IR images are in **Figure 17** and **Figure 18**.

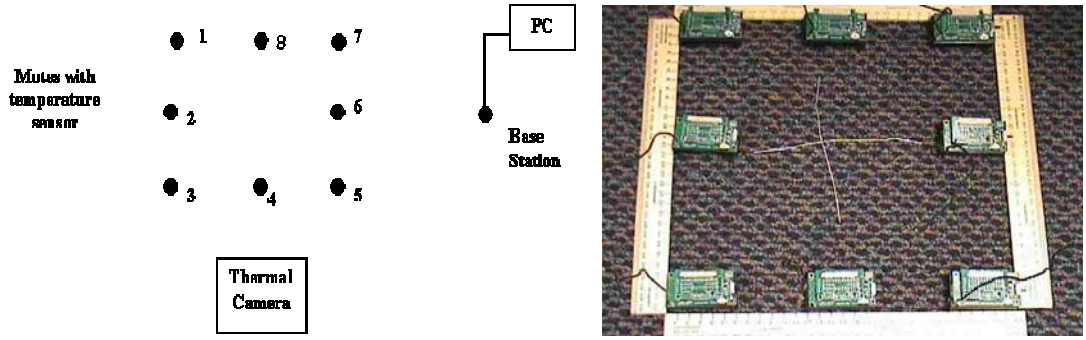

Figure 16 Left – A schema of the experimental setup. Right - The MICA sensor locations viewed with a visible camera. The tiny black spot in the middle (slightly towards the right) of a sensor mote is the thermistor, whose thermal IR pixel value we need to determine.

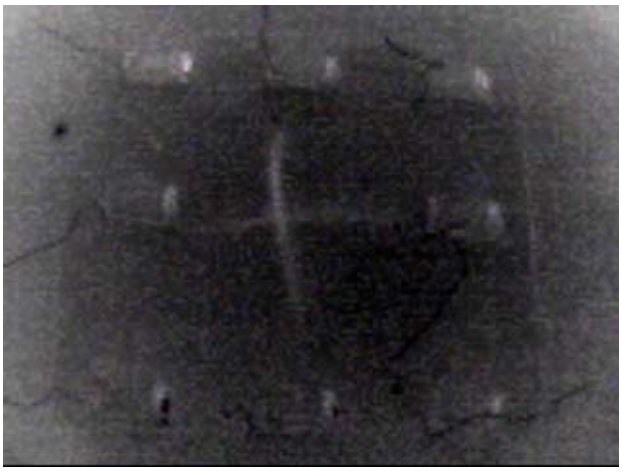

Figure 17 Temporally averaged raw thermal IR image.

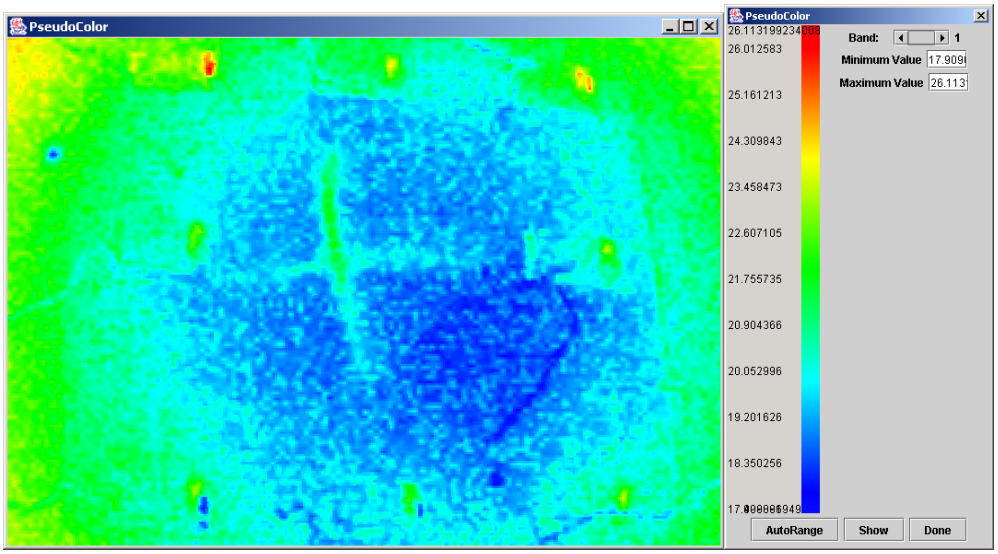

Figure 18 Pseudo-colored calibrated thermal IR image. The color bar on the right side shows the value of each pixel in degrees Celsius in contrary to a raw (uncalibrated) image where values would correspond to the output of an Analog to Digital (A/D) converter, e.g., [0, 256].

7.2. Calibration of MICA Sensor Readings using Thermal IR Camera

Second, we explored the calibration of MICA sensor readings using pre-calibrated thermal IR images. This problem turns out to have a great application for large scale sensor deployment scenarios. Calibrating a large number of MICA sensors is a very tedious and time-consuming process since every sensor has to be treated separately. One should be aware that although MICA sensors come with a manufacturer's recommended calibration formula, each sensor has its own hardware characteristics. Thus, inaccurate values in engineering units would be obtained if only the calibration formula was applied. This is illustrated in **Figure 19** by showing a spatially dense spatial arrangement of MICA sensors and the inter-sensor variations of raw temperature readings. In order to avoid false hazard alarms, it is critical to obtain accurate temperature readings in engineering units and hence provides a motivation for our calibration approach.

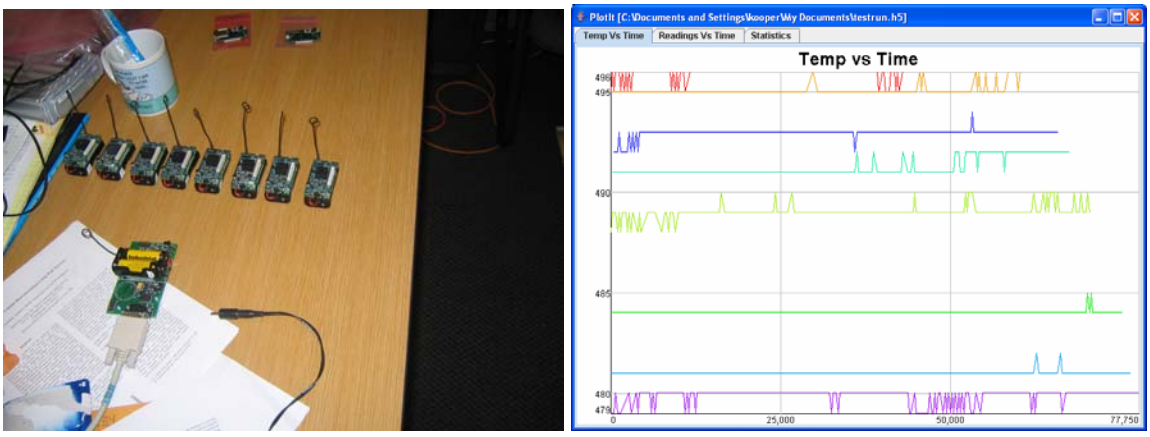

Figure 19: Left – MICA sensor spatial arrangement. Right – Variations of raw temperature readings obtained from the MICA sensors shown in the left picture.

A similar calibration hardware setup to **Figure 19** was used for our experiments. The calibration procedure can be described as follows assuming that all sensors and cameras were synchronized and their locations are known. First, we calibrate a thermal IR camera according to a procedure described in the next paragraph. Second, both thermal IR camera and MICA sensors are initiated to acquire data by broadcasting a RESET signal to MICA sensors and triggering thermal IR camera acquisition. Third, MICA sensors transmit every set (packet) of temperature measurements with the state of the internal counter (time stamp) to the base station attached to a personal computer (PC). In meantime, the thermal IR camera acquires data with the time stamp of the CPU clock counting from the RESET signal. Fourth, the MICA raw temperature measurements are received and transformed into degrees Celsius. Fifth, MICA temperature sensor locations in the thermal IR image are identified. All calibrated thermal IR image pixel values at the MICA sensor locations are used to form the spatially dependent calibration transformation for MICA sensor readings.

To calibrate thermal IR images as needed for the MICA sensor calibration procedure, we acquired thermal infrared images of a blackbody, such as a cup of water, while measuring its temperature with a regular thermometer (see **Figure 20**). The cup of water is heated up or cooled down while the thermometer is submerged in water. We took a thermal infrared snapshot of the cup of water each time the temperature has changed. Then, we averaged a random set of pixels from the image region that corresponded to water. **Figure 21** shows the raw thermal infrared values as a function of temperature. The graph in **Figure 21** proves linear dependency between raw thermal infrared values and temperatures in engineering units. The calibration mapping between the raw thermal infrared image intensities and the temperature in engineering units is established using a linear model with estimated parameters from the experimental data.

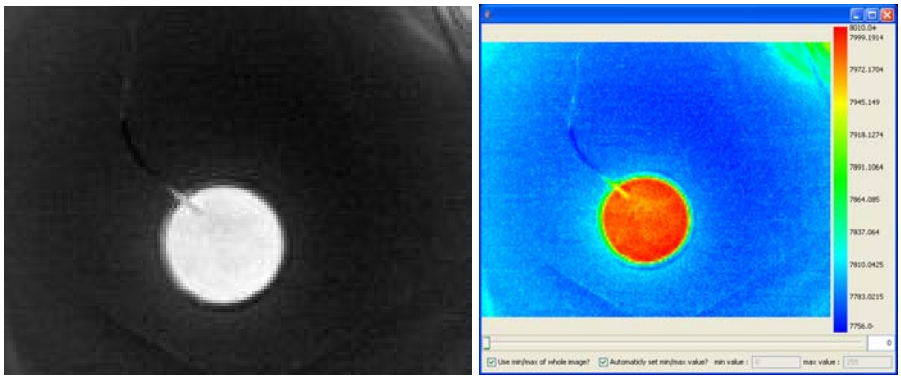

Figure 20: Thermal IR image of a cup of hot water. Black denotes cold and white is hot. Notice the probe of the thermometer in the upper left corner of the cup. The same image is displayed on the right in pseudo color, where red is hot and blue is cold.

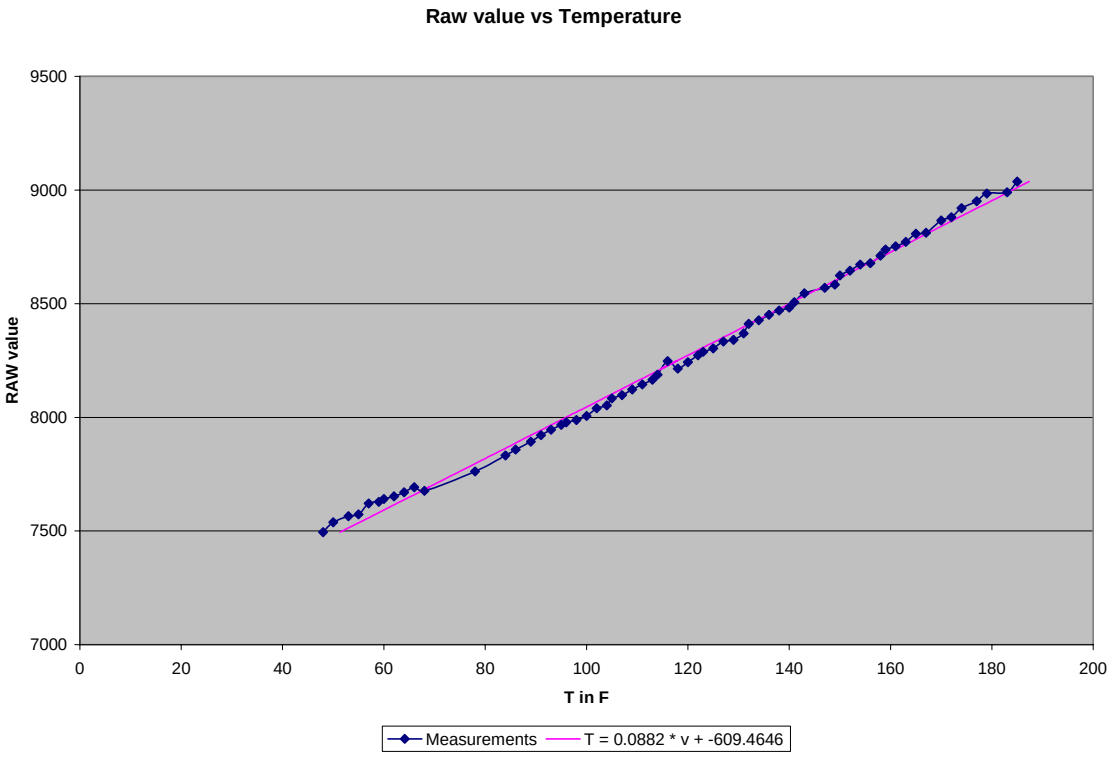

Figure 21: Raw thermal infrared camera value as a function of temperature.

In order to quantify the goodness to the linear calibration model, we performed the least squares error evaluations. **Table 2** summarizes our error analysis. The coefficient of determination r2 is used for comparing estimated and actual y-values (raw camera values), and ranges in value from 0 to 1. A value of r2=1 means that there is a perfect correlation in the sample (no difference between the estimated y-value and the actual y-value). At the other extreme, a value of r2=0 means that the linear regression equation is not a good model in predicting y-value (raw thermal infrared camera value as a function of temperature). Our value of r2=0.99589 demonstrates that the linear calibration model is a very good approximation of the calibration dependency.

Additional values in **Table 2** can be explained as follows. The F statistic or the F-observed value is used for determining whether the observed relationship between the raw thermal infrared camera values and temperature values occurs by chance. It is possible to compute the confidence level for the linear model by using the degrees of freedom df value and F statistics, and a statistical lookup table. Based on the values dF=62 and F=14802.02 in **Table 2**, the F-distribution critical (lookup table) value is 190.5 for Alpha = 0.005, and the degree of freedom of nominator v1=60 and denominator v2=2, Since the F-distribution critical value is smaller than the F-observed value (190.5< 14802.02), the hypothesis that the observed linear relationship occurred by chance can be rejected.

Table 2: A summary of the calibration errors for the experimental results shown in Figure 21.

y=m*x+b	0.0881758	-609.465	Linear calibration model
se_m and se_b	0.0007248	5.995442	Standard error for constants m and b
r2	0.9958289		Determination coefficient
Sey	2.522087		Standard error for the y estimate
F	14802.02		F-observed value
dF	62		Degrees of freedom
ss_reg	94154.482		Regression sum of squares
ss_resid	394.3771		Residual sum of squares.

7.3. Calibration Challenges

Our calibration application is unique in the sense that it deals with continuous data collection from MICA sensors and a thermal IR camera at high sampling rates. Most of other applications known to us assume much lower sampling rates, for example, sampling in environmental and atmospheric science applications. However, there are several challenges when it comes to (1) acquiring data continuously from wireless sensor networks, (2) dealing with large numbers and high spatial density sensor networks, and (3) performing real time calibration of thermal IR images. We will outline these challenges in the following sub-sections.

7.3.1. Acquiring Data From Wireless Sensor Networks

The aforementioned calibration procedures have to take into account wireless information loss from MICA sensors to the base station. Thus, we formulated the thermal IR calibration problem theoretically as an optimization problem and solved it by minimizing wireless information loss and maximizing information content. We investigated experimentally the impact of several communication protocols, spatial sensor arrangements, MICA's antenna orientations, presence of other wireless devices and the dependencies on the number of active MICA sensors to minimize the wireless information loss. Our results were summarized in our past publications [33], [34], [37].

In addition, we investigated the issues related to temporal sampling. One would like to understand the tradeoffs between the frequency of sensor sampling and data broadcasting, and the associated cost in terms of energy, memory storage, CPU utilization, and information loss due to on-board pre-processing and/or wireless transmission. Our tradeoff analysis is summarized as follows.

We are using broadcast to obtain the information back from the MICA sensors to the PC. This could result in two sensors sending their data at the same time resulting in a collision. Due to these collisions we will not receive all readings however, a newer reading will be following the previous reading. If we only have a few sensors transmitting, or we do not transmit too often then the chance of a collision would be small. If we increase the temporal sampling rate, or the number of sensors in the network then we increase the chance of collisions. If we have many sensors transmitting as fast as possible, then we would have many collisions and receive very few readings. Thus, for a given number of sensors there must exist an optimum temporal sampling rate and vise versa.

Figure 22 shows how the number of samples received per second decreases with an increasing number of sensors. If there is only one sensor and we sample every 128ms, then we receive almost eight readings per second (1000/128). If we add more sensors then the number of readings decreases because of the collisions in the network. With seven sensors transmitting we obtain the same number of samples per second using 128ms sample rate (and a lot of collisions in the network) or 256ms sample rate. If we would increase the number of sensors then we would continue finding these tradeoff configurations (number of sensors and sample rate configuration assessed by the number of reading lost due to network congestion). From these experimental studies one can make determinations when it is better to switch to a lower sampling rate and still receive the same number of samples per second as at the higher sample rate (while saving battery power). Although unclear from these experiments, it is hypothetically possible that with 128ms sample rate and many sensors in the network one would receive fewer samples per second than with the same number of sensors and 256ms while wasting battery power. Thus, knowing the optimal sampling rate for a given number of active sensors will not only decrease the number of collisions in the network but also save sensor energy. The MICA sensors are battery operated and have a limited supply of energy. Transmission is in most cases the most expensive operation the sensor performs and thus minimizing the number of transmissions will increase the lifetime of the sensors.

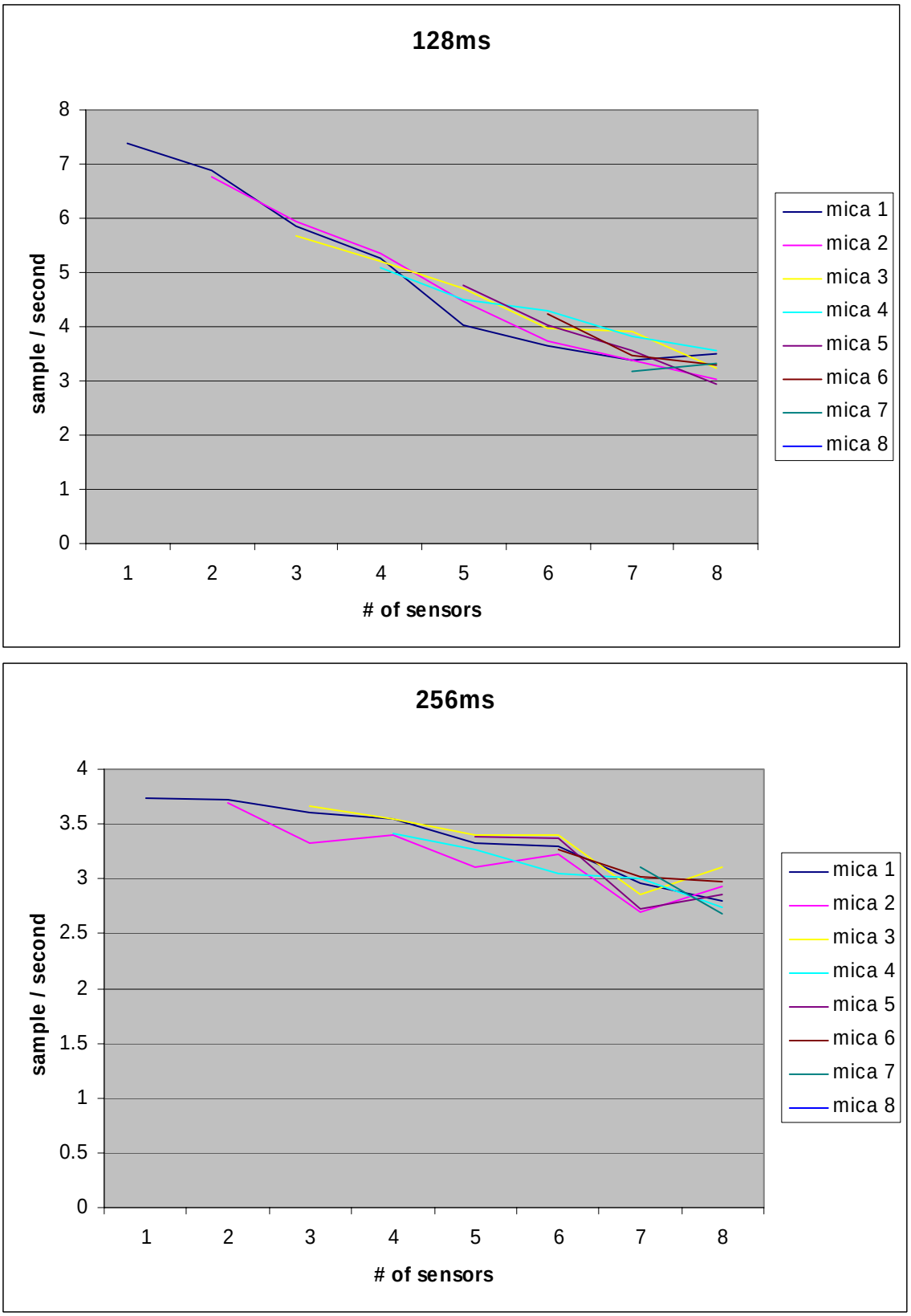

Figure 22: Samples (sensor readings) received per second as a function of the number of sensors for the temporal sampling rate equal to 128ms (top) and 256 ms (bottom).

There are applications other than hazard aware spaces where the time delay between sensor readings and the PC attached to the base station is not critical. In this case, for instance for data mining purposes, we could collect multiple sensor readings and transmit multiple readings in a single broadcast. This would reduce the number of transmission resulting in less collisions in the network, higher number of samples per second (but with loss of timeliness) and less transmission time (since the network overhead is amortized over multiple readings) resulting in less battery usage. We did not investigate this type of application scenario.

Finally, one could consider MICA sensing and on-board data processing to broadcast only data summaries or data triggered events. While this seems as a very efficient approach, the appropriateness depends solely on an application scenario. For instance, it might be appropriate for temperature triggered hazard event detection but it will prohibit us from doing any analysis of temperature values preceding the hazard event to remove false alarms. For calibration purposes, one would need all readings to correlate them with thermal IR values and hence we did not investigate this type of wireless data acquisition, pre-processing and transmission schema.

7.3.2. Large Number And High Spatial Density Wireless Sensor Networks

We have investigated the scalability issues of a system that would have large numbers of MICA sensors forming a high spatial density sensor network. For this purpose, we borrowed 46 MICA2 sensors from the CS Department at UIUC since our current number of MICA sensors is only 20 (10 MICA1 and 10 MICA2). These MICA2 sensors come with a temperature sensor, as well as a powerful sounder (see Figure 23). Our experimental setup is shown in Figure 24.

Figure 23: MICA2 sensor with a powerful sounder that was borrowed from the CS Department, UIUC.

Figure 24: Experimental calibration setup with MICA2 sensors on the floor, thermal IR camera located close to the ceiling and pointing straight down, and visible spectrum camera located close to the door and providing these setup pictures. A robot with a hair dryer was used to simulate temperature changes (see the left picture).

These experiments revealed several challenges regarding wireless traffic (related to information loss), incompatibility of MICA2 sensors in terms of their broadcast frequency, time stamping issues embedded in TinyDB software and the reliability of MICA sensor readings. **Figure 25** demonstrates a few of the problems. One would have to invest more time to provide solid conclusions about the scalability of the wireless sensor sensing mechanism using MICA hardware.

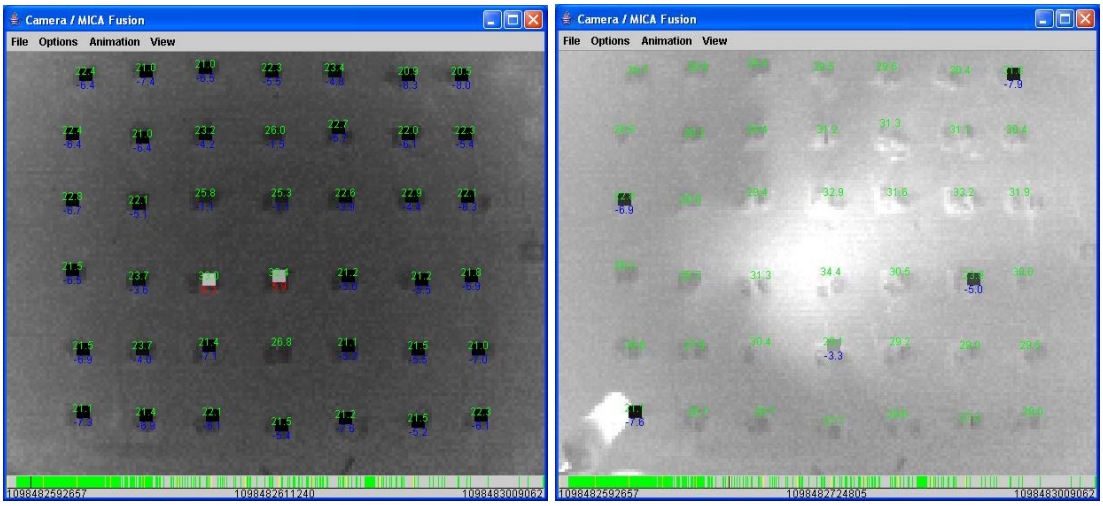

Figure 25: Visualization of calibration challenges including (a) irregularity of packets received from MICA sensors and (b) spatially varying MICA temperature readings. Temporal irregularity of packets received is shown as green bars at the bottom of each image. Spatial irregularity is visible in the right image by showing missing readings from most MICA sensors (missing blue or red values). Images correspond to snapshots taken by the thermal IR camera with dark being cold and white being hot. Calibrated temperature readings of the thermal IR camera are displayed as green numbers in the image. We computed the difference between calibrated MICA temperature readings

according to manufacturer's formula and the calibrated thermal IR pixel values. The differences are displayed as either blue (MICA reading is colder than thermal IR camera reading) or red (MICA reading is warmer than thermal IR camera reading). The MICA temperature reading is shown at MICA location as a small square (5x5 pixels) using the temperature color scheme of the thermal IR camera. The left picture is taken with homogeneous ambient temperature. The right picture is acquired with heterogeneous ambient temperature introduced by a hair dryer (see the white blob in the left lower corner).

7.3.3. Performing Real Time Calibration Of Thermal IR Images

We spent extra time on thermal IR image acquisition since the image data acquisition could operate in 8-bit or 12-bit mode. Unfortunately, the camera vendor FLIR Systems (the vendor was originally Indigo Inc. but it was sold in January 2004) did not provide a developer's kit with a low level application programming interface (API) for 12-bit image acquisition without an additional fee. Due to our financial constraints, we decided to implement a 12-bit data acquisition mechanism using the RS 232 (serial port) interface. The drawback of the current 12-bit data acquisition mechanism using RS 232 is its slower sampling rate (30 seconds per frame for 128x162 pixels) than it could be using the FireWire (IEEE1394) interface (25 frames per second). The issues of (1) 8-bit versus 12-bit data, (2) data transfer and conversions from camera to a personal computer, and (3) real-time acquisition, are outlined next.

The issue of 8-bit versus 12-bit thermal IR data acquisition: The Indigo Omega camera used in our experiments uses internally a 12 bit representation of temperature values. To obtain data from the camera to the PC, a special hardware module can be used to translate the signal to a serial port signal, an analog video or a fire wire (IEEE 1394) video. When the camera data are transmitted over the serial port, the 12 bit representation is transmitted as a 16bit signal (equal to 2 bytes). The same is true for a fire wire video. When transmitting the data across the analog video the 12 bit signal is first converted to an 8 bit representation which is then converted to an analog signal.

To perform the 12 bit to 8 bit conversion, a technique called smart scene is used. This technique will find the coldest and hottest temperature in the image and scale all values in the image with respect to these values. For instance, if we have an image with a range from 100 to 500 then the value 100 is mapped to 0, and the value 500 is mapped to 255 and all other values are mapped correspondingly. Now, if we have a scene that has values ranging from 1000 to 2000 these will also be mapped into the values 0 to 255. If we do not know what mapping is used we cannot distinguish between the value 100 from image 1 and 100 from image 2. It is also apparent from these examples that we loose precision, in the case of image 2 we are mapping the values 1000 to 2000 in a range from 0 to 255, resulting in almost 4 distinct values being mapped to the same value.

Data transfer and conversions from camera to personal computer: Figure 26 shows the process when using a video capture board on the PC to get the data from the IR camera. The 12 bit data from the camera are first converted to 8 bits (see previous

paragraphs), and then are converted to an analog video signal. This video signal is transmitted and converted back to a digital signal. This conversion happens in hardware resulting in 30 frames per second. However, the video capture board does not know the bits represent originally a grayscale data set and it will treat the data as a color image. Losses due to (a) converting the signal to analog, (b) during transmission and (c) during the final analog-to-digital conversion, the image values received as red, green and blue are not identical. This consideration combined with the facts that we loose (a) precision during the 12 bit to 8 bit conversion, and (b) knowledge about what value 0 maps to, we decided to use the 12 bit data transfer.

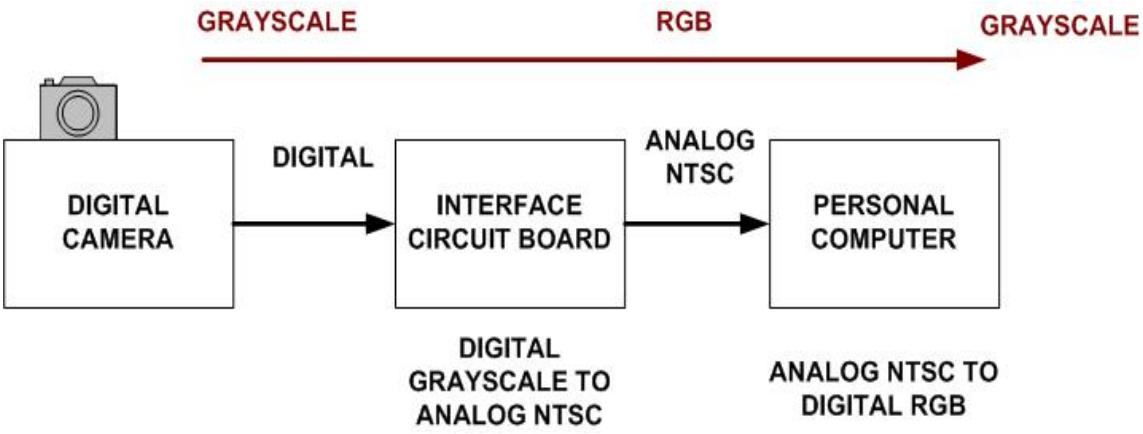

Figure 26: Data transfer and conversions for 8-bit acquisition mode of the thermal IR Omega camera.

Figure 27 shows the process when using either fire wire or serial port. In both cases the data is transmitted from the camera to the interface circuit board and transmitted either across the serial port or fire wire. Both of these data transfers can handle the 12 bit data, by pretending the data is actually 16 bit. However, we can only receive 1 frame every 20 seconds using the serial port due to its low bandwidth, while with the fire wire port we can receive 30 frames per second. Unfortunately, unlike many other manufacturers, Indigo does not provide a free toolkit to read the data from the fire wire port and we did not have any expertise and resources to develop data reading from fire wire. Thus, we decided to implement the 12 bit data transfer using the serial port. In the future, given a software support for FireWire IEEE 1394 enabling 12-bit data acquisition, one would achieve higher image sampling rate than with the current serial RS232 connection (viewed as a temporary alternative solution for acquiring 12-bit data).

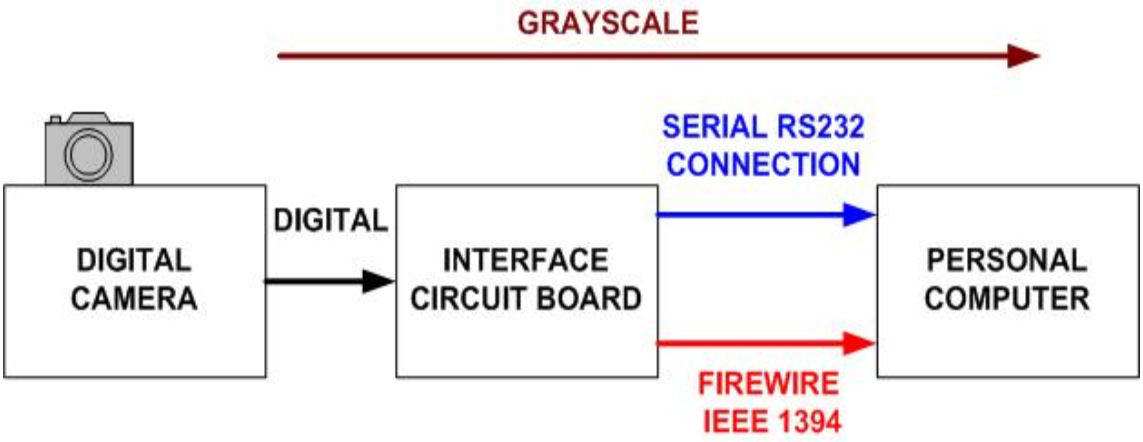

Figure 27: Data transfer for 12-bit acquisition mode of the thermal IR Omega camera.

Real-time data acquisition: In the 8-bit mode, the data acquisition speed is defined by a video capture card in a personal computer that performs Analog NTSC to RGB conversion (25 frames per second). In the 12-bit mode, the data acquisition speed is defined by a serial RS232 connection (1 frame per 20 seconds). The implementation tradeoff can be described as a real time but "inaccurate" 8-bit acquisition versus a non-real time but accurate 12-bit acquisition. Based on our calibration objective to obtain the most accurate measurements, we decided to use the more accurate data acquisition over the real-time data acquisition solution.

8. Proactive Camera Control, Hazard Detection and Human Alert

We have investigated proactive approaches to camera control, spectral image analysis and human alert mechanisms. We have completed a design phase of a proactive camera control system that can trigger visible spectrum and thermal infrared spectrum cameras based on luminance and temperature sensors mounted on the available MICA sensor boards. Our current design addresses the problems of (a) efficient bandwidth management by proactive camera control (low bandwidth monitoring of hazard awareness spaces with MICA sensors and high bandwidth monitoring of hazard awareness spaces with cameras), (b) hazard understanding (multi-spectral sensing including visible and thermal infrared information) (c) choice of the best spectral modality to capture data about the environment based on the data provided by the MICA sensors and (d) human alert mechanism (image analysis to highlight areas of potential hazard). The proactive camera logic can be described as follows.

If light = on then visible camera = on
If light = off then thermal IR camera = on
If temp > thresh & light = on then visible camera = on & hazard region enhancement
If temp > thresh & light = off then thermal IR camera = on & hazard region enhancement

Examples of proactive camera control and human hazard alert mechanism are shown in **Figure 28** and **Figure 29**.

Figure 28: Proactive camera control. Depending on the temperature and light readings from a wireless network of motes, either a visible spectrum video or a thermal infrared spectrum video is sent to a hazard monitoring station.

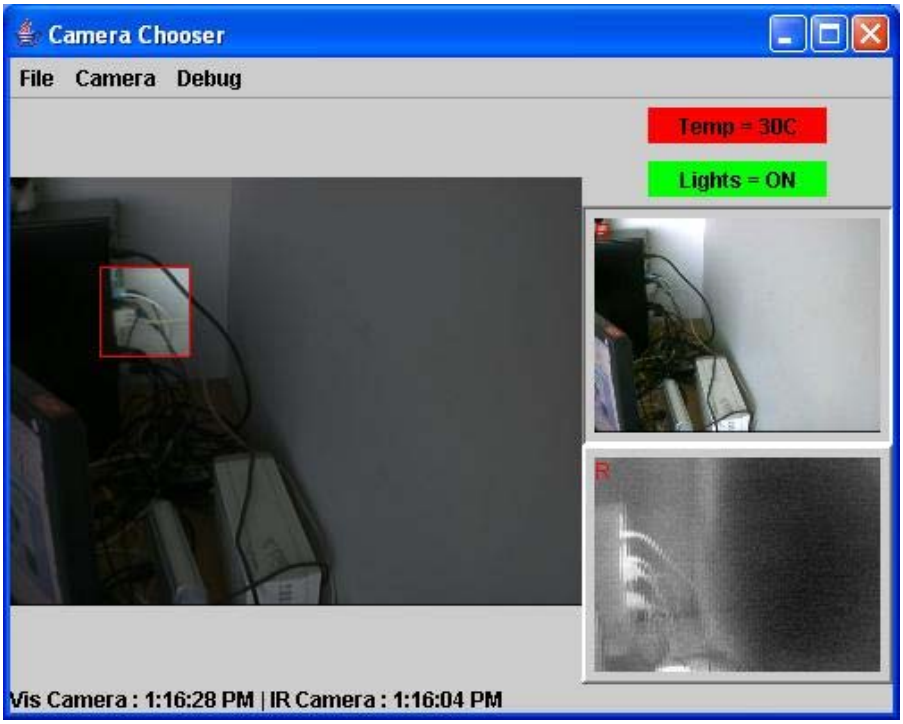

Figure 29: Illustration of hazard alert by automated region selection and image enhancement. Note how the system uses the RGB camera when the office light is on, and combines the information about white (hot) regions from the thermal IR camera to highlight the RGB region of interest (the hazard region).

9. Hazard Confirmation and Containment

We have also investigated the possibilities of using the robot and its robotic arm for hazard confirmation and containment. Hazard confirmation is needed when MICA sensors report inconsistent values or when hazard measurements are too sparse and it is desirable to increase spatial density of sensor readings. Deployment of additional MICA sensors to confirm hazards is executed the same way as the deployment of new MICA sensors by using remotely controlled robot with multiple human-computer interfaces.

Hazard containment requires studying hazard types, containment methods, hazard accessibility and many other application specific constraints. We have prototyped a demonstration showing that a robot could be used for the purpose of hazard containment when fire hazards in an office building are still relatively small in their extent. It is our belief that constant monitoring of hazards and an immediate hazard containment action could prevent large scale fire hazards and significant financial damages. In our demonstration, a robot is remotely navigated to reach its destination (desired proximity and orientation with respect to a lit candle) and then it contains the fire hazard by pressing the hair dryer “on” button with the robotic arm to extinguish the flame. A time frame of the demonstration is shown in Figure 31.

Figure 30: A time frame from an experiment that shows how a robot can be remotely navigated to reach its destination (desired proximity and orientation with respect to the lit candle). It can then contain the fire hazard by pressing the hair dryer “on” button with the robotic arm and extinguish the flame.

10. Hazard Understanding

To better understand hazards we try to use as many different sensors as possible. One problem is how can we use different cameras to few the same scene and do a simple comparison between the images captured using each camera. First we need to make sure

all images captured are the same color, this is address in 10.1 and then we need to compare the images, a generic framework for this is discussed in Section 10.2. Once we have this we can look at how to use multi band camera's to distinguish different hazards, this is described in Section 10.3

10.1. Calibrating Color Cameras using a Hyperspectral Camera

Our objective is to perform hazard analysis from multi-sensor and multi-instrument inputs. In this environment, we would like to utilize high-spectral resolution imagery for improving low-spectral resolution imagery. In our analysis, we assumed that an acquisition of high spectral resolution images provides more accurate spectral predictions of low spectral resolution images than a direct acquisition of low spectral resolution images. We illustrated the advantages by focusing on a specific case of images acquired by a hyperspectral (HS) camera and a color (red, green, and blue or RGB) camera. First, we identified two directions for utilization of HS images, such as (a) evaluation and calibration of RGB colors acquired from commercial color cameras, and (b) color quality improvement by achieving sub-spectral resolution. Second, we elaborated on challenges of RGB color calibration using HS information due to non-ideal illumination sources and non-ideal hyperspectral camera characteristics.

The core of our work lied in developing camera calibration approaches that compensate for wavelength and spatial dependencies of real acquisition systems. We evaluated two color cameras by establishing ground truth RGB values from hyperspectral imagery and by defining pixel-based, correlation-based and histogram-based error metrics. Our experiments were conducted with three illumination sources (fluorescent light, Oriel Xenon lamp and incandescent light); with one HS OptoKnowledge camera and two color (RGB) cameras, such as Sony and Cannon. We showed a data-driven color-calibration as a method for improving image color quality.

Given the assumption that an acquisition of high spectral resolution images provides more accurate spectral predictions of low spectral resolution images than a direct acquisition of low spectral resolution images, the two approaches to utilization of hyperspectral imagery are illustrated in Figure 31. One can utilize high spectral resolution imagery either for evaluating and calibrating colors of low spectral resolution cameras, or for increasing spectral resolution of low resolution cameras using additional information about scenes. In hazard awareness spaces, we expect to have many color (RGB) cameras since they are inexpensive and operating real-time. Hyperspectral cameras, that are quite expensive and known to have slow image acquisition, would be utilized for improving spectral values (RGB) and spectral resolution (number of bands) of images acquired with color cameras.

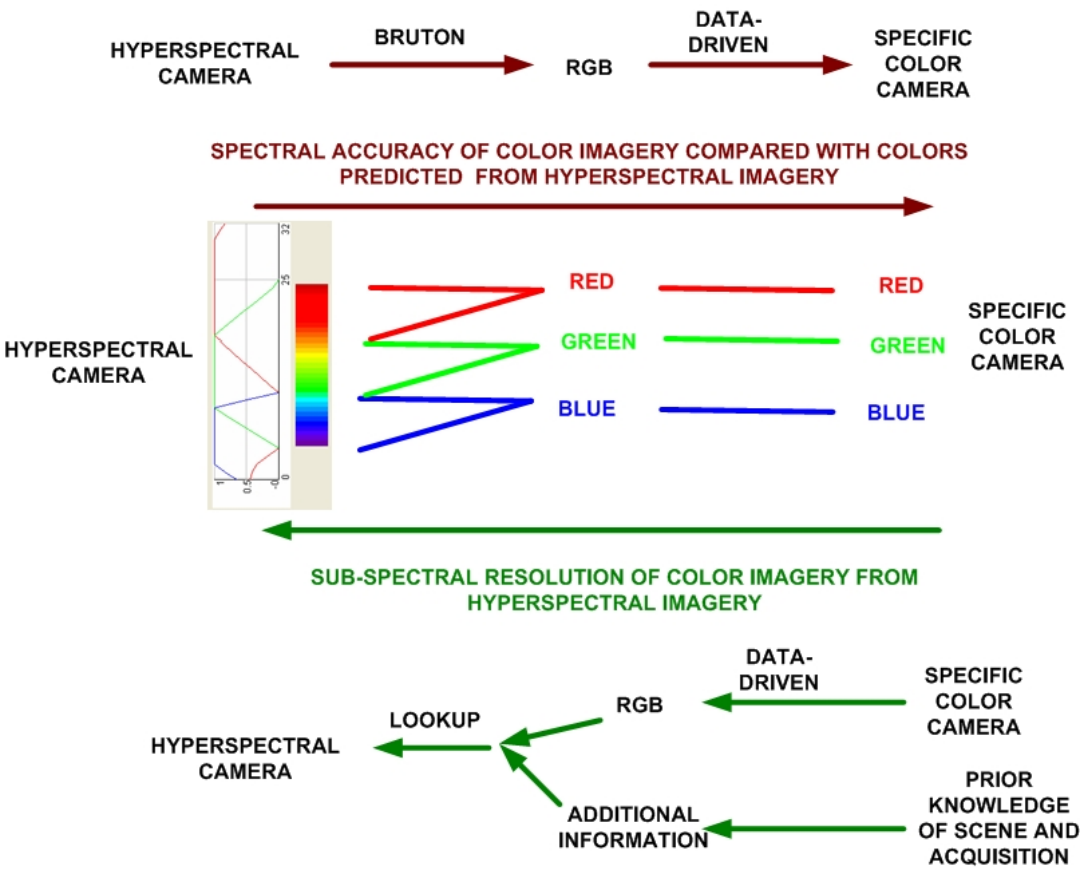

Figure 31: Two approaches how to utilize high spectral resolution imagery for improving color (RGB) imagery.

10.2. Framework for Image Comparison

To compare images taken with different cameras as well as images taken at different times we have developed image comparison techniques. This component is important for hazard understanding and detection since our goal is to search for hazards with minimum human intervention. In addition, humans might not be able to discriminate hazards with the same accuracy as computer algorithms with built-in prior hazard models. From this perspective, modeling hazards and evaluating hazard prediction accuracy are key technologies for operational hazard awareness spaces. These techniques led us to develop a more generic methodology for evaluating image comparison techniques, especially for evaluating statistically predicted versus measured multi-modal imagery.
Since statistical prediction models are frequently chosen as more appropriate modeling approaches, there is a need to evaluate the accuracy of statistically predicted versus measured imagery. This problem poses challenges in terms of selecting quantitative and qualitative evaluation techniques, and establishing a methodology for systematic comparisons of synthetic and measured images. We therefore developed a methodology and showed experimental results for color (red, green and blue) and HS imaging modalities. The methodology includes several evaluation techniques for comparing image samples and their similarity, image histograms, statistical central moments, and estimated probability distribution functions (PDFs). Particularly, we assessed correlation,

histogram, chi-squared, pixel and PDF parameter based error metrics quantitatively, and related them to a human visual perception of predicted image quality. The evaluation schema is illustrated in Figure 32.

.

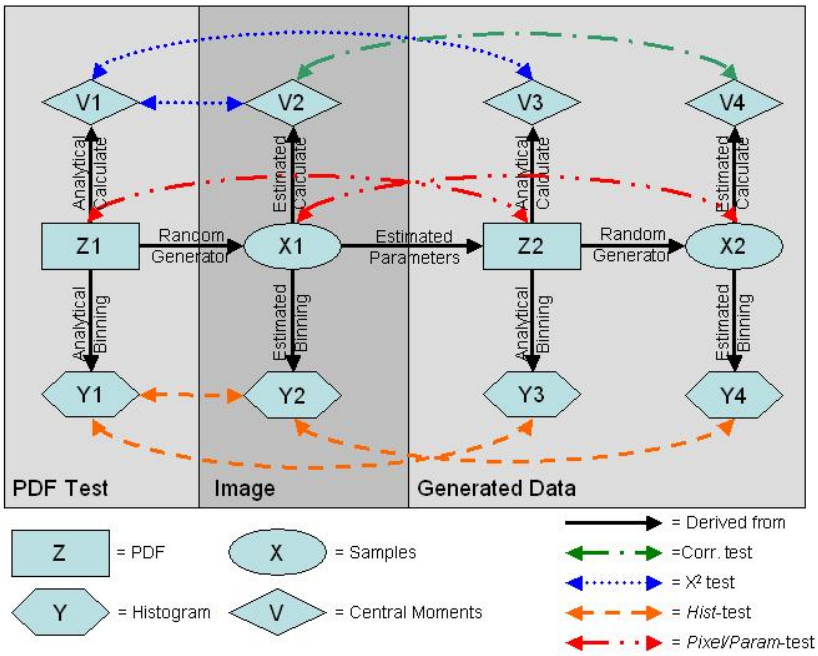

Figure 32: An overview of evaluation comparisons.

10.3. Future Work on Hazard Understanding

It is important to not only detect a hazard, but also understand it. For example in **Figure 33** there is a picture of a couch burning in the left picture, which is bad, and a fire in the fire place on the right which is good. To understand what is burning we are looking at the possibility of using the hyperspectral camera to take images of a hazard. Analysis of the hazard at the different wavelengths will hopefully give us an idea of what is burning.

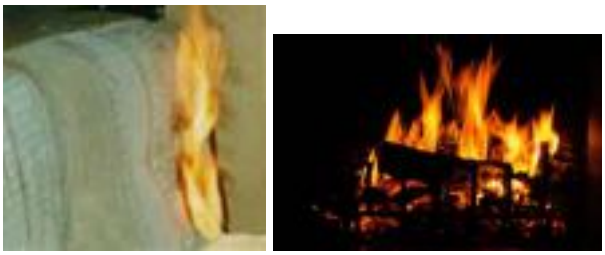

Figure 33: Examples of “bad” (left) and “good” (right) fire hazards.

Not only will the information of what is burning help us detecting if something is a hazard, we can use this information also to assist any fire fighters. If we can tell the fire fighter what is burning, for instance oil vs. wood they can use specific techniques and equipment to contain and extinguish the fire.

11. Summary

Our objective was to research any technical challenges in building a prototype HAS system for detecting fire, vision impairing light, extreme sonic waves, or any other hazards that can be sensed by spectral sensors. Our initial focus was on testing and deploying sensors, and combining wireless “smart” micro electro-mechanical systems (MEMS) sensors and spectral cameras for camera calibration, hazard detection and hazard identification purposes.

We explored the issues related to a large sensor network where the bandwidth available was limited and unlimited broadcasting would be a waste of resources. We have performed several experiments to demonstrate that sometimes acquiring fewer readings leads to higher system efficiency since too many readings might cause collisions in the network resulting in fewer reading reaching the base station. Another constraint of the sensor networks is the energy usage. Again, sending fewer readings will reduce the energy consumption of a sensor.

The readings coming from the sensor can be delayed in the network, or during processing on the host computer. To be able to correlate the readings from the MICA sensors with the readings from other sensors, like a camera, we have implemented a simple time synchronization protocol that will enable us to timestamp each reading at the sensor.

In multi-sensor and multi-instrument systems, the data we receive at a central computer come from multiple sources (e.g. MICA sensors and cameras). Thus, we have investigated methods how to fuse the incoming data and maximize our information gain from the use of diverse sensing capabilities. We have researched specifically a fusion of (a) temperature readings from the MICA sensors with the thermal IR camera and (b) depth information from the MICA sensor localization with the stereo camera depth map.

Finally we have developed a method for deploying sensors remotely with a robot. To allow us to remotely control the robot we have looked at different modalities of controlling the robot (e.g. voice, gestures, a mouse or a keyboard). We have addressed the problems related to issuing conflicting commands to control the robot by using simple heuristic rules derived from the importance of commands (e.g. stop is more important than move ahead).

Our work provides a prototype system with several solutions to outlined problems and a future test bed for developing a functional hazard aware space.

Acknowledgements

This material is based upon work partially supported by the NAVY STTR N03-T003, the National Center for Advanced Secure Systems Research (NCASSR).

References

[1] ActivMedia Robotics, Amherst, NH, http://www.activrobots.com/ROBOTS/p2dx.html#components
[2] Aircraft Signals NATOPS Manual, NAVAIR 00-80T-113.
[3] AT&T Laboratories Cambridge, "The Bat Ultrasonic Location System," Aug. 2004; http://www.uk.research.att.com/bat/.
[4] Bajcsy P. "Toward Hazard Aware Spaces: Knowing Where, When and What Hazards Occur," NCSA Private Sector Program May 2005, http://www.ncsa.uiuc.edu/Conferences/2005Meeting/agenda.html.
[5] Bajcsy P. "Understanding Multi-Instrument Measurement Systems," Understanding Complex Systems Symposium, May 17-20, 2004, UIUC, http://www.how-why.com/ucs2004/abstracts.html#bajcsy
[6] Bajcsy P., and S. Saha, "A New Thermal Infrared Camera Calibration Approach Using Wireless MEMS Sensors," Communication Networks And Distributed Systems Modeling And Simulation Conference (CNDS 2004), January 19-22 2004, San Diego, California.
[7] Bajcsy P and R. Kooper, "Prediction Accuracy of Color Imagery from Hyperspectral Imagery," Proceedings of SPIE on Defense and Security 2005, Conference: Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XI, Vol. 5806-34, 28 March - 1 April 2005, Orlando (Kissimmee), Florida USA.
[8] Bajcsy P. et al., "Image To Knowledge", documentation at http://alg.ncsa.uiuc.edu/tools/docs/i2k/manual/index.html.
[9] Dario P., M. Bergamasco, A. Bicchi, and A. Fiorillo, "Multiple Sensing for Dexterous End Effectors," Robots with Redundancy: Design, Sensing and Control, A.K. Bajcsy, ed., Berlin: Springer Verlag, 1992.
[10] Dinguirard, M. and P.N. Slater, 1999; "Calibration of Space-Multithermal IR Imaging Sensors: A Review"; Remote Sensing of Environment, 68(3): 194-205.
[11] Dishman E., "Inventing Wellness Systems for Aging in Place," Computer, May 2004, pp. 34-41.
[12] Elson J. and Estrin, D. "Time Synchronization for Wireless Sensor Networks". In 2001 International Parallel and Distributed Processing Symposium (IPDPS), 2001.
[13] Heidemann J., F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin and D. Ganesan, "Building Efficient Wireless Sensor Networks with Low-Level Naming". 18th ACM Symposium in Operating Systems Principles, October 21-24, 2001.
[14] Hightower J. and G. Borriello, "Location Systems for Ubiquitous Computing," IEEE Computer, Vol. 34, Num. 8, Aug. 2001, pp. 57-66.
[15] Hollar, S. E., "COTS Dust," Master Thesis at University of California, Berkeley, Fall 2000.
[16] Gupta P. and P.R. Kumar, "The capacity of wireless networks". IEEE Trans. On Information Theory, 46(2): 388 – 404, March 2000.
[17] Intanagonwiwat C., R. Govindan and D. Estrin. "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," ACM/IEEE International Conference on Mobile Computing and Networks (MobiCom 2000), August 2000, Boston, Massachusetts.

[18] InterSense, Burlington, MA, http://www.isense.com/products/prec/is300/is300pro.htm
[19] Krishnamachari B., D. Estrin and S. Wicker. “Modeling Data-Centric Routing in Wireless Sensor Networks”. IEEE Infocom 2002, (also Tech report CENG 02-14), http://www2.parc.com/spl/members/zhao/stanford-cs428/readings/Networking/Krishnamachari_infocom02.pdf.
[20] Lam S. S. “A carrier sense multiple access protocol for local networks.” In Computer Networks, volume 4, pages 21 – 32, 1980.
[21] Lementec J-C. and P. Bajcsy, "Recognition of Arm Gestures Using Multiple Orientation Sensors: Gesture Classification," the 7th International IEEE Conference on Intelligent Transportation Systems, Washington, D.C., October 3-6, 2004.
[22] Long Term Ecological Research Network (LTER), http://www.lternet.edu/
[23] Marr D., Vision, San Francisco: W.H. Freeman, 1982.
[24] Mainwaing A., J. Polastre, R. Szewczyk, D. Culler and J. Anderson: “Wireless Sensor Networks for Habitat Monitoring”.
[25] MICA sensor description, http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf
[26] National Ecological Observatory (NEON), http://www.neoninc.org/
[27] Network and Mobile Systems Group, CSAIL, MIT, “The Cricket Indoor Location System,” Aug. 2004; http://nms.lcs.mit.edu/projects/cricket/.
[28] Pankanti S. and A.K. Jain, “Integrating Vision Modules: Stereo, Shading, Grouping, and Line Labeling,” PAMI, Vol. 17, Num. 9, Sep. 1995, pp 831-842.
[29] Priyantha N.B., H. Balakrishnan, E. Demaine, and S. Teller, “Anchor-Free Distributed Localization in Sensor Networks,” Tech Report 892, Laboratory for Computer Science, MIT, 2003.
[30] Raghavendra C.S and Suresh Singh: “PAMAS – Power Aware Multi-Access protocol with Signaling for Ad Hoc Networks”. ACM Computer Communication Review, vol. 28, no. 3, pp. 5—26, July 1998.
[31] Real-time Observatory, Applications, and Data management Network (RoadNet), http://roadnet.ucsd.edu/
[32] Roush W., A.M. Goho, E. Scigliano, D. Talbot, M.M. Waldrop, G.T. Huang, P. Fairley, E. Jonietz, and H. Brody, “10 Emerging Technologies,” Technology Review, Vol. 106, No. 1, 2003, pp 33-46.
[33] Saha S., and P. Bajcsy, “System Design Issues in a Single-Hop Wireless Sensor Network,” 2nd IASTED International Conference on communications, Internet and information technology (CIIT), pp. 743-748, November 2003.
[34] Saha S., and P. Bajcsy, “System Design Issues for Applications Using Wireless Sensor Networks,” Technical Report NCSA-ALG-03-0003, August 2003.
[35] Satyanarayanan, M. "Pervasive Computing: Vision and Challenges." IEEE Personal Communications, August 2001.
[36] Scherba D. and P. Bajcsy, “Depth Map Calibration by Stereo and Wireless Sensor Network Fusion”, The Eighth International Conference on Information Fusion, July 25 - 28, 2005, Philadelphia, Pennsylvania.
[37] Scherba D. and P. Bajcsy. "Communication Models for Monitoring Applications Using Wireless Sensor Networks," Technical Report NCSA-ALG-04-0003, April 2004.

[38] The SHIELD project, the advanced Structural Health Integrated Electronic Life Determination (SHIELD) systems, URL: http://jazz.nist.gov/atpcf/prjbriefs/prjbrief.cfm?ProjectNumber=00-00-5306) and the URL of the main lead (Caterpillar, Inc., IL, USA): http://www.cat.com/
[39] Smart Dust Project at http://www.cs.berkeley.edu/~awoo/smartdust/
[40] TinyOS website: http://webs.cs.berkeley.edu/tos
[41] Teillet, P.M., B.L. Markham, J.L. Barker, J.C. Storey, R.R. Irish, and J.C. Seiferth, "Landsat Sensor Cross-Calibration Using Nearly-Coincident Matching Scenes"; Proceedings of SPIE Conference 4049, Orlando, Florida, 2000.
[42] Urban M. and P. Bajcsy, “Fusion of Voice, Gesture, and Human-Computer Interface Controls for Remotely Operated Robot,” The Eighth International Conference on Information Fusion, July 25 - 28, 2005, Philadelphia, Pennsylvania.
[43] Urban M., P. Bajcsy, R. Kooper and J-C Lementec, "Recognition of Arm Gestures Using Multiple Orientation Sensors: Repeatability Assessment," the 7th International IEEE Conference on Intelligent Transportation Systems, Washington, D.C., October 3-6, 2004.
[44] Vildjiounaite E., Malm E-J. , Kaartinen J. and Alahuhta P., “ Smart Things in a Smart Home,” the Fifth International Conference on Ubiquitous Computing, pp. 215-216, Seattle, WA, October 2003.
[45] Wang H., D. Estrin and L. Girod: “Preprocessing in a Tiered Sensor Network for Habitat Monitoring, EURASIP Journal on Applied Signal Processing 2003:4, 392–401.
[46] Wechsler H., Computational Vision, Boston: Academic Press, 1990.
[47] Weiser M., "Some computer science issues in ubiquitous computing." Communications of the ACM, July 1993. Pages 75-84.
[48] Whitehouse K., Culler D. "Macro-calibration in Sensor/Actuator Networks." Mobile Networks and Applications Journal (MONET), Special Issue on Wireless Sensor Networks. June, 2003. ACM Press
[49] Whitehouse K. "The Design of Calamari: an Ad-hoc Localization System for Sensor Networks." Master's Thesis, University of California at Berkeley, 2002, URL: http://www.cs.berkeley.edu/~kamin/pubs/whitehouse02calamari.pdf
[50] Whitehouse K. and X. Jiang, “Calamari: A Localization System for Sensor Networks,” Aug. 2004; http://www.cs.berkeley.edu/~kamin/calamari/.
[51] Woo A. and D. Culler, A transmission control scheme for media access in sensor networks, Proc. 7th ACM/IEEE International Conf. on Mobile Computing and Networking, Rome, Italy, 2001, 221-235.