Development of a Mote for Wireless Image Sensor Networks

Ian Downes, Leili Baghaei Rad*, and Hamid Aghajan
Wireless Sensor Networks Laboratory, Department of Electrical Engineering
Stanford University, Stanford, CA 94305
Email: downes@stanford.edu, leili@ieee.org, aghajan@stanford.edu

***Abstract*— This paper presents the design of a new mote for distributed image sensing applications in wireless sensor networks. The processing and memory limitations in current mote designs are analyzed and a simple but powerful new platform is developed. The mote is based on a 32-bit ARM7 micro-controller operating at clock frequencies of up to 48 MHz and accessing up to 64 KB of on-chip RAM. An expansion interface is provided to support multiple mid- and low-resolution image sensors concurrently as well as traditional sensors. Wireless communication is provided by the Chipcon CC2420 radio which operates in the 2.4 GHz ISM band and is compliant with the IEEE 802.15.4 standard. An integrated USB and serial debug interface allows simple programming and debugging of applications. The additional requirements of an image sensor mote are discussed along with a discussion of possible applications and research areas.**

I. INTRODUCTION

Most applications in the field of wireless sensor networks are designed under constraints for communication energy and bandwidth [1]. These constraints have caused most applications to limit their data acquisition mechanisms to low-bandwidth data types. Industrial applications of wireless sensor networks have been considered in developing new wireless motes [2], [3]. A new direction in wireless sensor network application design centers on the idea of enabling the network to learn the behavior of the phenomena in the environment rather than merely making measurements and reporting about a single effect of interest. This new development trend calls for mechanisms to provide the network with more complex forms of awareness about the situational state or the context of the events [4]. At the same time, such application paradigms would also facilitate the incorporation of interactions between the network and the events in progress, perhaps providing different forms of data and information to the observers through recognizing the preferences stated by them [5], [6].

It is natural to envision the need for acquiring more complex or higher bandwidth data types from the environment for facilitating context-aware applications. Visual data forms can often provide a wealth of information about the surroundings of a network.

This paper will investigate several aspects concerning the design and use of image sensors in a wireless sensor network (WSN). Section II provides motivation and background work for image sensing in the context of wireless sensor networks, and discusses the principle differences in the sensing models between image sensor networks and other types of sensor networks. Considerations will be given to the current range of available mote platforms to motivate the need for a new mote platform based on image sensors. In section III, the requirements for developing a wireless mote with multiple image sensors on-board are discussed, and a detailed description of our mote design as well as example applications of low-resolution image sensors are presented. Section IV offers some concluding remarks.

II. WIRELESS IMAGE SENSOR NETWORKS

The trend towards design of energy efficient, low-cost image sensors has been largely driven in the past several years by the rapid expansion of the mobile phone market. Image sensors have been considered in the past as data sources for surveillance and security applications [7], [8], [9]. In these and other applications in which observers are interested in visual monitoring of effects in the environment, the nodes of the network generally act as providers of raw data to a central processing entity, which sifts through the gathered data in order to draw conclusions about the events occurring in the environment [10]. The assumption of a central processing unit may not often be valid in the case of wireless sensor networks. Additionally, continuous transmission of high-bandwidth data from all nodes to a central processor may cause rapid energy depletion or bandwidth congestion. In many applications of wireless sensor networks, only certain attributes of events are of interest to the observer. Detection of situations that may need observer's attention or intervention [7], monitoring the rate at which moving objects flow through the observed environment [11], [12], or registering the types and quantities of certain objects or events [10] are among such applications.

Technology advancements in processor design for embedded applications have allowed for increased in-node processing capabilities at much more cost-effective and energy-efficient levels. As transmission cost is a dominant part of energy expenditure in wireless networks, in-node processing can help avoid constant transmission of raw images between the network nodes by enabling the extraction of event attributes from the acquired data at the node.

In this premise, the application may only require occasional transmission of images to the observer. For example, in an

---

* Leili Baghaei Rad was working with the Wireless Sensor Network Laboratory, Stanford University, under a summer research collaboration.

In Proc. of Cognitive Systems and Interactive Sensors (COGIS), Paris, March 2006.

application to monitor the flow of vehicular traffic in a highway, the nodes of the network would periodically transmit packets containing average speed information in each lane [13].

However, many applications involving visual data may require the nodes to occasionally provide the observer with a number of image frames acquired from an event of interest. For example, in the highway traffic monitoring application, when a node detects a vehicle with a speed outside of a predefined range, it may buffer and transmit a few image frames [14], which can be used for various law enforcement or accident scene analysis purposes.

In many applications of wireless sensor networks, the detection of an event is also associated with reporting its location by the network. Hence, it is important that the location of the network nodes be known. As reported in [15], in addition to visual sensing of the environment, image sensors can provide information that can be used to perform automated network localization and topology construction.

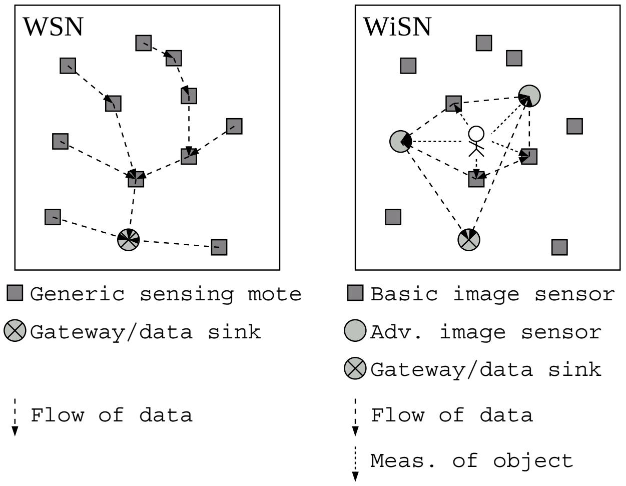

Fig. 1. Models of sensing in WSN and WiSN. In the WiSN, the multi-tier nodes collaborate to identify the object or event of interest. Note that the arrows indicate the general flow of data and not the actual transmission paths.

A. Networked sensing models and application types

Although a wide range of sensor types are currently in use, a large number employ approximately the same sensing model. Using a network of spatially separated sensors, the distribution of an environmental variable is sampled and transmitted to a common data sink. Each sample is a measure of the variable at the location of the sensing node. When combined, the samples can be used to create an estimate of the distribution in the area under observation. By increasing the number of nodes to form a denser network, the measured values will tend to create a more accurate estimate of the distribution. A level of redundancy in the data is achieved provided the distance over which the variable changes is greater than the spacing of the nodes. It should be noted that this is not achieved by multiple independent measurements at the same location but rather by over-sampling the distribution. Fig. 1 illustrates the flow of information from the sensing nodes to the data sink.

The sensing model described can be used for several different applications of WSNs. In [16], the author describes three classes that cover a large number of possible WSN applications. The first class, environmental data collection, uses a distributed sensor network as described above to collect many measurements over an extended period of time. The measurements of the distribution are then analyzed offline to determine trends and interdependencies in the data. The network will have frequent data transmissions but the amount of data in each transmission would be low. As the data is expected to be analyzed offline, latency in transmission is acceptable. This is in contrast to the second application class, security monitoring, where latency is not acceptable. In this application the same arrangement of sensors may be used but the data is not collected. Instead each node processes its measurement to detect if an exceptional event has occurred. Only when such an event is detected is a message sent across the network. For these infrequent transmissions speed and reliability is maximized at the expense of higher energy consumption. The third application type is node tracking, in which an object, tagged with a node, moves through a network of sensors. Each sensor can detect the presence of the tag node and report the information to a data sink.

Even though many differing factors exist between the three application classes described above - network topology and the amount of in-node processing, for instance - all three share the same sensing model. Each node is limited to sensing the environment immediately around it and, possibly after some processing, transmitting data to a common sink.

B. Networked image sensing model

An image sensing model is different in three significant ways from the sensing model mentioned above. First, a single measurement from an image sensor is actually composed of many separate measurements, one for each photosensitive element in the imager. All of the measurements are of the same type (i.e. the intensity of light) but each comes from a different source as determined by the optical lens. The precise number of measurements depends on the structure and may range from six in a very small linear array to many millions for a high-resolution color imager. Simply stated, each measurement provides multiple data points to the node, e.g. a two-dimensional set of values. This is in contrast to the typical one-dimensional signal yielded by other sensing types. Second, due to the spatial span of data represented in an image sensor measurement, besides the mentioned intensity levels as measured data, the image sensor can also provide a sense of directionality to the source of light. This can be used to obtain information about the location of objects in the environment. Thirdly, the sensing range of image sensors is naturally not limited to the immediate vicinity of the node's location and these sensors are often used to make measurements from rather far objects.

It may be argued that other sensors such as acoustic or even temperature sensors can make measurements from distant sources [17]. However, the practical working range of these sensors is much less than that possible with a suitable lens and an image sensor. In addition, these sensors lack any real form of directionality to the source when a single sensor node is employed.

The described properties of image sensors require a slightly modified sensor model for a WiSN. For example, with a set of image sensors, the same object or event can be observed from different view points. This presents opportunities for developing techniques to take advantage of the mentioned properties in a collaborative data processing platform using data obtained at the multiple sensor nodes. Examples include improving the location estimates of an object and better resolving multiple co-located objects. Further possibilities include constructing primitive 3-D models of observed objects based on multiple observations from different angles. Additional information present in the 2-D images could be used to aid in identifying and classifying objects.

C. Related work on image sensor motes

A number of other groups are working in areas related to the development of image sensor motes. The following list briefly describes a selection of related work.

1) Panoptes - Video sensor network: The Panoptes platform is designed as a video sensor platform capable of medium-resolution video at high frame rates [18]. A StrongARM-based embedded Linux board is used, running at 206 MHz and equipped with 64 MB of RAM. The board uses a USB web camera as a video source and 802.11 for wireless communications.

2) Image sensor mote with FPGA for compression: In [19] the authors develop an alternative approach for an image sensor mote where the transmission of compressed images is considered. To support the high memory and processing requirements, an ARM7 CPU is used in conjunction with a field programmable gate array device (FPGA) and additional memory. With a color VGA camera source the FPGA is used to implement a range of wavelet-based compression methods. A Chipcon CC1000 radio is used.

3) Cyclops - Image sensor daughter-board: Another approach to the design of an image sensor mote is undertaken by [20], where an image sensor daughter-board is developed. The board is designed to attach as an external sensor to a mote board such as one from the Mica family, and therefore it does not include a radio. This allows the network and image sensing aspects to be separated. The board uses an Agilent ADCM-1700 CIF camera module as an image source. To provide a frame buffer, a complex programmable logic device (CPLD) is used to help interface to a shared 512 KB SRAM. An ATmega128L processor running at 4 MHz is included to direct operations and perform image processing. In addition, the CPLD can be used for basic operations during frame capture.

4) SensEye - Multi-tier camera sensor network: The significant differences the projects described above have are reconciled in [21], where a camera sensor network is considered as a heterogeneous mix of devices with varying capabilities. At the lowest tier, the most basic motes are equipped with very low-resolution cameras or other detection means such as vibration sensors. At higher tiers more advanced motes use high-resolution cameras with pan-tilt-zoom capabilities. The combination of capabilities allows the network to be optimized to suit the application.

III. HARDWARE DEVELOPMENT

To support further research in algorithm and protocol design for WiSNs, a flexible and expandable mote architecture is desired. With possible research topics in mind in areas such as collaborative processing for object and event tracking, algorithm design for self-localizing networks, routing scheme development for event-driven networks, and supporting applications involving image sensors and mobile agents, the design requirements for the new image sensor mote are described in this section. Our development plans aim to deploy a network of 150 wireless nodes equipped with image sensing capability supporting one or more image sensors and on-board processing.

A. Requirements for an image sensor mote

The design of a mote intended for use with image sensors requires additional considerations over that of a generic sensor mote. Specifically, more local processing is necessary to extract the information from the data. This requires a combination of a powerful power-efficient processing unit plus supporting memory.

A further unique requirement for image motes is the issue of the type of interface to the image sensor itself. In general, the availability of different image sensor interfaces means that an image mote must be designed for a specific image sensor or a family of image sensors.

B. Suitability of current mote platforms

Prior to making the decision to develop a mote, the current mote platforms were investigated to determine if a suitable solution existed. Fig. 2 illustrates that most current mote platforms are either of the generic sensing type or are extremely powerful gateway nodes. The existing generic sensing motes are found to lack adequate processing power and memory sizes for image sensing applications.

To illustrate this, let us consider the data in Table I, which show the memory and number of multiply-and-accumulate (MAC) operations for a single 2-D convolution with a 3x3 kernel. The ATmega128 micro-controller is commonly used for generic sensing motes and shall be considered here. First, it should be noted that ATmega128 uses an 8-bit architecture and is thus not efficient for multiplying two 8-bit numbers when compared to 16- and 32-bit processors. Including cycles to read/write data, ATmega128 requires 74 cycles per pixel. When clocked at 4 MHz, this would require 2 seconds. This can be compared to a 32-bit ARM7 device which requires at most 56 cycles per pixel and takes 0.12 seconds when clocked at 48 MHz. The factor of 16 reduction in execution time is

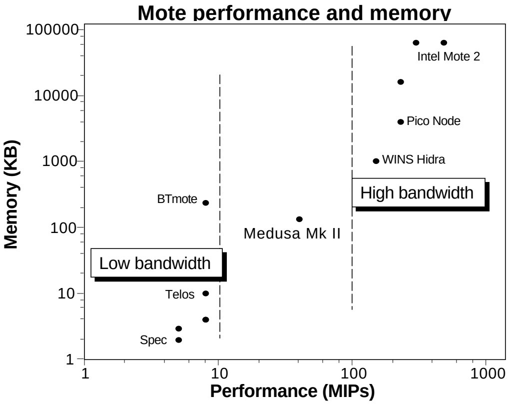

Fig. 2. Plot of processor performance and memory size for current mote platforms.

TABLE I

MEMORY AND MAC OPERATIONS REQUIRED FOR A 2-D CONVOLUTION WITH A 3X3 KERNEL.

Frame resolution	Frame buffer (KB)	MACs for 2-D Sobel
30 × 30	<1	16,200
CIF - 352 × 288	99	1,824,768
VGA - 640 × 480	300	5,529,600

greater than the difference in power consumption between the two devices ($\sim$5 mA vs. $\sim$31 mA, both at 3.3 V) indicating that the ARM7 architecture is more power-efficient for such an operation. Furthermore despite ATmega128's ability to perform the operation, it lacks sufficient memory to store the frame and any results.

At the opposite end of the spectrum, the powerful motes such as Intel Mote 2 [3] exceed the memory and processing requirements but they have higher power consumption and are comparatively more expensive, again making them unsuitable for large network deployments. Between these two extremes the Medusa Mk II was discounted primarily due to the added complexity of two micro-controllers and the use of the TRF1000 radio (a 802.15.4 compatible radio was preferred). The Cyclops daughter-board offers a suitable image sensor with added processing ability, but again it uses a dual CPU configuration. This has the significant drawback of limiting research into cross-layer optimizations. With no suitable options available, it was determined that a new mote platform would be required for addressing the research topics mentioned earlier.

C. *Image sensors for WiSNs*

Before progressing with the design of the new mote it is important to identify which image sensors will be used as they will determine the interfaces that are required. With the mote positioned in the lower tiers the image sensors are expected to be of a lower resolution and quality when compared to cameras used for higher tiers and wired systems. With this in mind, the choice of sensors is then influenced primarily by the ease of interfacing, power consumption and the form factor.

At this early stage in the development of WiSNs, the demand may not be adequate to justify a customized image sensor for mass production. Instead, image sensors which meet the requirements must be sourced within existing markets. Perhaps the most visible use of image sensors is in the consumer digital camera market. These are typically highly mass-produced devices which operate from a battery power supply and are of a reasonably small form factor. However, the continued demand for improved resolution has meant that the current generation of cameras have a minimum resolution of 1280 × 1024 (1.3 megapixels). This requires too much memory and thus it is not feasible for the proposed mote. A lower-resolution imager can be found in another major consumer device - the mobile phone - where the use of image sensors is more recent. Restrictions on the physical size, low-resolution displays and the need for wireless transmission mean that current image sensors are of CIF (352 × 288) and VGA (640 × 480) resolutions. In addition, the sensors are intended for low power usage meaning that they are a suitable candidate for use with an image sensor mote. The Agilent ADCM-1670 has been selected as one such device. A summary of relevant specifications for ADCM-1670 is given in Table II. Of particular note is that the interface is a serial connection and that the module incorporates a lens assembly and an internal frame buffer.

Despite the seemingly small size of a CIF image it still represents a human-readable image in which everyday objects are easily recognized. As the resolution of an image decreases, the image is still recognizable to the human brain but it becomes increasingly difficult for computer algorithms to process. It is of great interest to investigate the minimum resolutions at which an image contains useful information that can be extracted. To allow this area of research a second much lower resolution image sensor is desired. A suitable sensor, the Agilent ADNS-3060, is used in computer optical mice. The sensor is a specialized integrated circuit that contains a 30 × 30 pixel sensor and a small digital signal processor. Table II lists some relevant specifications for the sensor. The device is somewhat unique in that it can easily be reconfigured to capture and store frames which can then be read by a host processor over a serial interface. The low resolution of this imager will provide an additional avenue of research where the sensor can be used either as the main image sensor or as a trigger for the CIF sensor. A preliminary investigation into the output from the ADNS-3060 has shown that it may be useful for basic detection, triggering and counting of objects as described in the next section. It should be noted then that this implies that the mote must support the connection of both types of sensors concurrently.

D. *Example applications of low-resolution image sensors*

To illustrate the use of image sensors in a distributed network, it is useful to examine a simple demonstration in greater depth. The proposed application consists of monitoring the internal pedestrian corridors within a building. Traditionally a security room would be required to monitor a relatively small number of cameras selectively placed throughout the facility.

TABLE II

SPECIFICATIONS OF THE LOW- AND MID-RESOLUTION IMAGE SENSORS USED. A HIGH-RESOLUTION SENSOR IS INCLUDED FOR COMPARISON.

Model	ADNS-3060	ADCM-1670	Micron MT9D011
Purpose	Optical mouse sensor	Mobile imaging (CIF CMOS)	PC camera, mobile imaging
Frame size (pixels)	30 × 30	352 × 352	1600 × 1200
Frame rate (per sec.)	6469 int., 100 ext.	15	15
Frame buffer (KB)	1.5	48	none
Output formats	6-bit grayscale	YUV, RGB, JPEG	10 bit RGB
Interface	SPI	UART	10-bit parallel
Supply voltage (V)	3.1-3.6	2.65-3.3	1.7-1.9 & 2.5-3.1
Supply current (mA)	30 @ 500 fps	26 @ 15 fps QCIF	40 @ 15 fps
Power down current (μA)	5 typ.	10 typ.	10
Power up time (ms)	75 max.	200	
On-board capabilities	Pixel sum, max. pixel value	F/2.6 lens, JPEG compression, windowing, subsampling	Windowing

Human operators would observe the continuous video streams and act accordingly to any events of interest.

A multi-tier image sensor network presents an alternative system. Large numbers of simple, inexpensive wireless sensors would provide dense coverage ensuring that no area of the facility would be unmonitored. When a potential event is detected, higher levels of the network would be notified. This will allow high-resolution image sensors to confirm the event, capture data, and pass it on to human personnel or other modules of the security system as appropriate. Such a system would provide advantages on multiple levels. First, the improved coverage will be achieved at a lower installation and administration cost. The system will also be inherently redundant in both communication and monitoring, creating a more robust system. Finally, the system is capable of advanced data gathering and processing. This presents possibilities of integrating the system into other aspects of the building services. For example, accurate information on the pedestrian flow into and out of a room could aid the control of the air conditioning and lighting systems.

To explore how even very simple low-resolution image sensors could be used for this purpose, a demonstration was performed using a single sensor node. The image sensor used produced frames of resolution 30 × 30 pixels and 6-bit grayscale depth at a rate of five frames per second. The sensor was positioned above and to the side of a narrow pedestrian pathway. The view spanned a six foot long segment of the pathway.

Fig. 3 shows a sequence of frames captured by the sensor which show passing pedestrians, followed by the results of image processing with the goal of determining the direction and speed of the motion of the pedestrians. Two people are seen in the sequence walking in opposite directions. An algorithm was developed to detect and determine the direction of the pedestrian movement. Movement is detected by first subtracting the background image and then applying a threshold on the gradient magnitude of the resulting image. Objects are identified based on this threshold and their centers of gravity are determined. Multiple objects in a vertical direction are considered together and are combined. Using the center of gravity locations across multiple frames, all possible paths connecting the locations are computed. A linear regression is performed on each path and a threshold is applied to the sum of the squared residuals. Those paths below the threshold are considered valid objects crossing the camera's field-of-view. The slope of the fitted line reveals the object's direction and approximate speed. The application of the algorithm is shown in Fig. 3, where the two remaining paths after applying the threshold correctly match the motion of the two pedestrians. In a short test of 50 events, the algorithm correctly identified 100% of the events. However, this was achieved under the following test conditions:

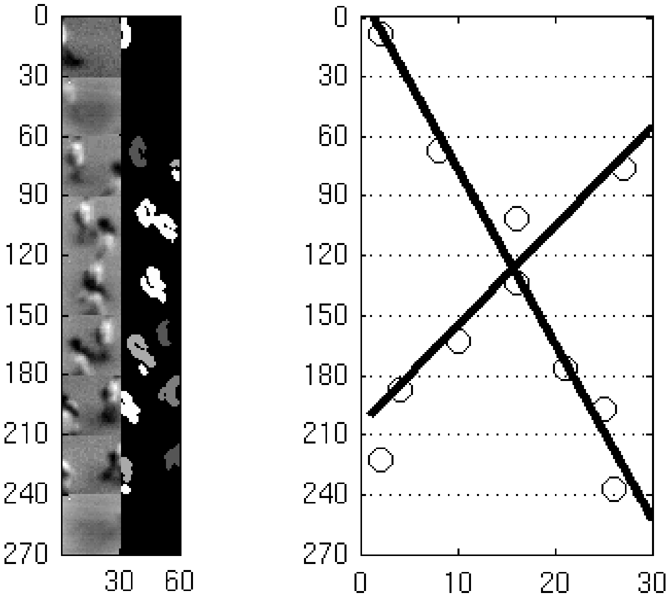

Fig. 3. Demonstration of an image sensor node used to count pedestrians passing a walkway. Direction and estimates of speed are obtained.

1) The speed of movement was limited to walking and jogging speeds.
2) People passing were separated such that no two people moving in the same direction were seen within a single frame.

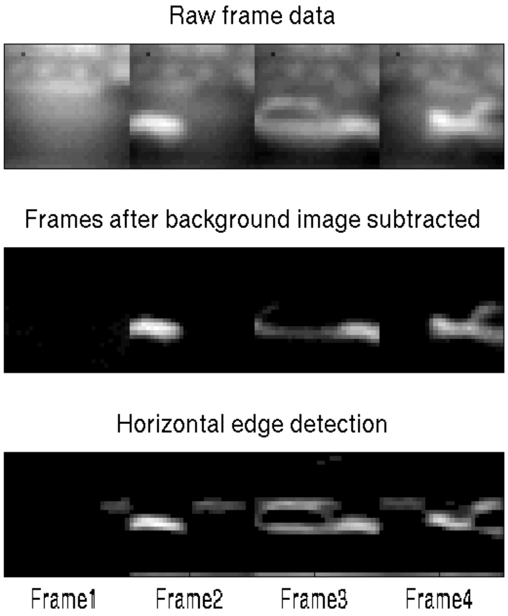

Fig. 4. Sequence of frames showing two passing cars. The lower sequence has the background image removed and a 2-D edge detection filter applied.

3) Simultaneous movement of two people in opposite directions was permitted.
4) People stayed within the view of the camera at all times when moving across the field-of-view (no significant vertical movement in the image plane).

Although these assumptions may be restrictive in some cases, it must be noted that this is a simple sensor that would be part of a multi-tier image sensor network. By allowing communication between the nodes, the network can track multiple people moving from one node's field-of-view to another's. Furthermore, if multiple closely-spaced objects need to be resolved, the low-resolution sensor node could trigger a node equipped with a higher-resolution camera to obtain the required information. This may, for example, be a pan-tilt-zoom camera which would receive data about where the target was located allowing it to quickly capture the necessary frames.

Fig. 4 shows a second example for detecting cars on a roadway and estimating their speed using a low-resolution image sensor node. The data was captured via a PC parallel port which limited the frame rate to approximately 5.2 frames per second. When connected to a micro-controller, frame rates of up to 100 frames per second are possible.

The images of Figs. 3 and 4 are encouraging as they show that despite the very low-resolution sensor used, there is information that can be easily extracted. The simple processing steps include thresholding and 2-D convolution, operations that are well within the capabilities of a power-conscious embedded device. The small size of image frames allows for more frames to be processed, or more elaborate image processing functions to be applied to the frames. The individual information contributed by the mote may be small; however, a dense network might be capable of sophisticated detection by in-node processing and collaborative decision making. For example, a single high-resolution camera might track an object moving within its field-of-view, but it will have limited knowledge of its full 3-D motion due to using a single observation point. A calibrated cooperative sensor network will be able to triangulate the object's position by combining information from a number of spatially separated observation points.

E. Development of a wireless image sensor mote

The new mote has been developed with the following goals in mind:

1) Broadly speaking, the new mote would have capabilities similar to the Medusa Mk II, but it would use a 802.15.4-compatible radio and have suitable interfaces for connecting multiple image sensors.
2) In the hierarchy of a multi-tier camera network it would lie near the lowest tier - equipped with low- and/or medium-resolution cameras and be intended to be deployed in a dense network.
3) In addition to interfacing to cameras, the mote should be able to connect to other sensors (passive infrared, temperature, pressure, humidity, etc.).
4) The mote needs to have a sufficiently low power consumption such that extended battery operation is viable.
5) As a final goal, the mote should be easy to develop applications with, providing programming and debugging over standard communication interfaces.

The system diagram for the entire mote board is shown in Fig. 5. As indicated by the low number of blocks, the mote board has been kept simple with a minimum of components. This was in part due to the requirement for low power consumption but also to help reduce the mote size and manufacturing cost. The following discussion will briefly look at each of the system blocks in turn, noting design decisions and their implications. Fig. 6 shows the prototype of the mote.

1) Processor: To provide the required processing power and memory for the mote, it was determined that a device based on the ARM7TDMI core would be suitable. The ARM7 is a 32-bit core (with support for a reduced 16-bit instruction set), which can typically operate at clock frequencies up to 50 MHz and address up to 256 MB of memory (much less used in practice).

The specific family that has been chosen is the AT91SAM7S series from Atmel Corporation. The devices within this family include the same peripheral set and are contained in the same package[1], but differ in the amount of RAM and Flash memory provided, allowing for an easy upgrade path. The Atmel devices were chosen in preference to other devices such as the Philips Semiconductor LPC21XX devices due to the USB slave peripheral present in the Atmel series. The benefits of this will be discussed shortly.

---

[1] With the exception of the AT91SAM7S32.

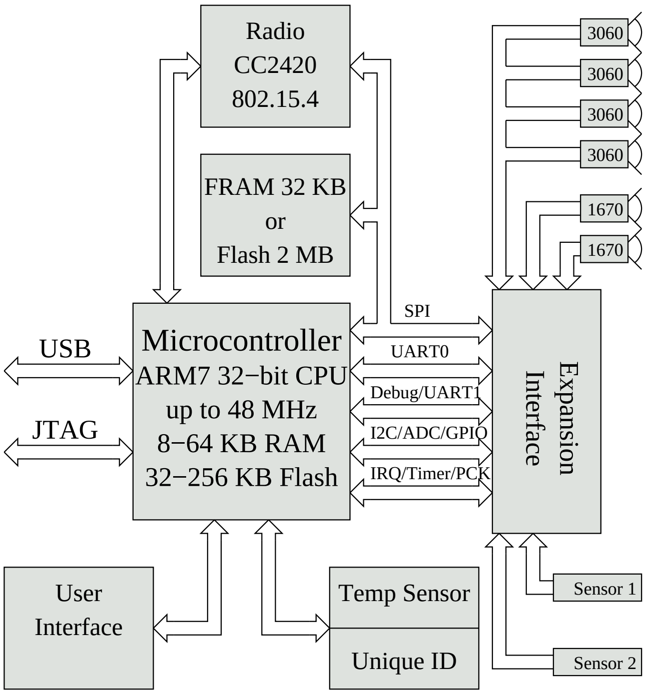

Fig. 5. System diagram of the mote board.

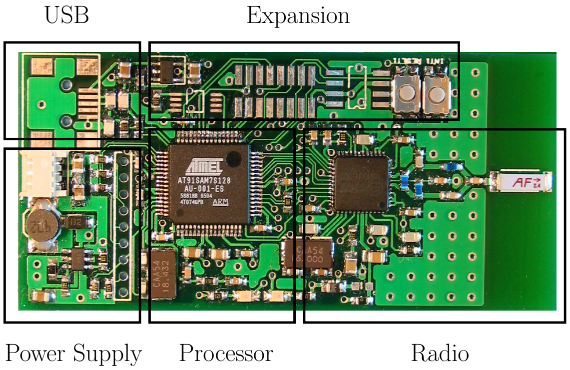

Fig. 6. Current prototype of the mote platform with the major functional units outlined.

2) Radio: The decision of the radio system is critical to the wireless network as a whole. When choosing the radio interface, the performance must be evaluated not just for the individual mote but also for the network as a whole. Narrowband radios for example may consume less power for a mote due to fast start up times but their lower noise tolerance may impose more power drain on the network since all nodes may need to transmit at higher power levels. In addition, the mote must be considered as part of a sensor network which is likely to consist of several different mote types (a multi-tier image sensor network for example). It is not practical for each mote to implement its own radio protocol and thus a standard interface is much preferred. The IEEE 802.15.4 standard defines a physical communications layer for low-power, low data rate (250 kbps) communication. The Chipcon CC2420 combines low power and efficient operation with support for 802.15.4 radio and has been selected.

3) Expansion interface: To connect the image sensors and other sensors to the mote, an expansion connector is provided on the board. A simple connector allows for a mechanically robust connector suitable for cable attachment to multiple cameras. The reduced number of connections also simplifies the PCB design of the board. The connector supports a maximum of two Agilent ADCM-1670 CIF image sensors and four Agilent ADNS-3060 image sensors concurrently using two independent UARTs and a shared SPI bus. Additional functions are multiplexed using the remaining pins. These include an I2C (TWI) serial bus, inputs to the analog to digital converter (ADC), timer inputs and outputs, programmable clock outputs, interrupt request lines, plus standard general purpose I/O pins. Several of the GPIO pins are high drive (16 mA) and can be used to power attached sensors instead of using the main board supply. Possible devices are not limited to sensors but can include memory, ADCs/DACs and GPIO expansion devices.

It would be beneficial to examine the improvements that have been made to other mote platforms as they have evolved. One goal for the new Telos platform has been to improve the process of development and programming for the individual motes [22]. For this purpose, the Telos platform introduced the standard USB interface, which can be used for programming applications and for data retrieval. This was achieved using a separate external USB transceiver with no impact on power consumption (it is bus powered). The same benefit is obtained in our new image mote platform using the integrated USB peripheral. The AT91SAM7S micro-controllers are preloaded with a bootloader stored in the ROM, which allows the system to be booted from an image supplied across the USB interface. This allows the initial programming of the device (and later recoveries if necessary). Once an application is loaded, it can also be used for standard data transfers. Together with the debug interface, this functionality mitigates the need for the traditionally required JTAG interface.

4) External memory: The AT91SAM7S family offers devices with RAM sizes between 8 and 64 KB and Flash memory sizes between 32 and 256 KB. A standard SPI memory device footprint has been included to allow for external memory. Depending on the requirements, the footprint can be used for two different devices. First, if more RAM is necessary, a FRAM memory chip could be used. These are currently limited to 32 KB but offer unlimited write/erase cycles and no wait states when writing. The memory access would be much slower than on-chip memory or parallel external memory, but it may be acceptable for frame buffering for instance. The second use would be for a Flash memory device. These are currently available in sizes of up to 2 MB. Due to the slow write speed and limited erase cycles, the memory is most suited for program and data storage (pattern matching templates, etc.). If the memory were to be used as a frame buffer, the lifetime of the mote may be restricted depending on the frequency of data writes. For example, if a 2 MB Flash device was specified for 100,000 write/erase cycles with one 100 KB frame written every 10 seconds, the device would be expected to fail after approximately 230 days. Neither of these devices are typically low power and their impact as frame buffers would need to be further investigated.

5) Power supply: At the time of the design, the AT91SAM7S was specified to operate from a core voltage of 1.8 V and an I/O voltage of 3.0 - 3.6 V. It was believed that it would operate correctly at lower I/O voltages which would allow the use of the unregulated battery directly supplying a single linear regulator. However, it was decided that the first prototype would follow the specifications and a 3.3 V regulated supply would be used. The Linear Technology LTC3400 synchronous boost converter was selected. It can start up and operate from a single cell and achieves $>$90% efficiency over a 30 - 110 mA current draw range. The mote is expected to operate in this range both when only the processor is operating and also when the processor, radio, and image sensors are operating together. The converter switches to burst mode operation when the mote enters the sleep state. In this mode, the converter has a 19 μA quiescent current draw and can supply up to approximately 3 mA. After the design was completed, the specifications of the device were revised and a second operating range for the I/O voltage was declared at 1.8 V. This would allow for the use of a single linear regulator. In a future version, the boost converter may not be required at all, reducing the sleep current draw (though the 3.3 V supply is still required for programming the Flash memory).

6) User interface: A basic user interface is provided on the mote using a pair of push-buttons and a pair of LEDs. One of the push-buttons is connected to the micro-controller reset input and provides a way to reset the system to a known state. The second button shares one of the interrupt request lines with the expansion connector and can be used to interrupt the processor. Red and green LEDs are connected to GPIO pins and can be used to signal the status of the system.

IV. CONCLUSIONS

This paper has presented the development of a new mote platform for wireless image sensor networks. It has investigated potential applications and determined necessary characteristics for an image sensor mote, primarily sufficient processing capability and memory. These characteristics have been used to show that current mote platforms are not suitable for image sensing applications. A new mote is proposed which is capable of interfacing with up to six separate cameras of different resolutions simultaneously. The cameras can be of medium resolution (CIF) or low resolution (30 $\times$ 30 pixels). A proposed network of these motes will be used to enable further research on wireless image sensor networks. Potential areas include investigating more intelligent sensor networks with the ability to learn from an environment, and to control agents based on visual observations. Applications such as surveillance and roadway and facility monitoring can also be explored using the proposed mote.

V. ACKNOWLEDGMENTS

The authors gratefully acknowledge provision of components by Agilent Technologies and Chipcon AS for our planned network deployment.

REFERENCES

[1] A. J. Goldsmith and S. B. Wicker, “Design challenges for energy-constrained ad hoc wireless networks,” in *IEEE Wireless Commun. Mag.*, vol. 9, no. 4, Aug. 2002, pp. 8–27.

[2] M. Horton, D. E. Culler, K. Pister, J. Hill, R. Szewczyk, and A. Woo., “Mica: The commercialization of microsensor motes,” in *Sensors Online*, Apr. 2002.

[3] L. Nachman, R. Kling, R. Adler, J. Huang, and V. Hummel, “The intel mote platform: a bluetooth-based sensor network for industrial monitoring.” in *IPSN 2005*, Apr. 25–27 2005, pp. 437–442.

[4] M. Baldauf and S. Dustdar, “A survey on context-aware systems.” [Online]. Available: http://citeseer.ist.psu.edu/baldauf04survey.html

[5] G. Abowd, E. Mynatt, , and T. Rodden, “The human experience,” in *IEEE Pervasive Computing, V.1, No.1*, Jan-March 2002, pp. 48–57. [Online]. Available: http://csdl.computer.org/dl/mags/pc/2002/01/b1048.pdf

[6] A. K. Dey and G. D. Abowd, “Towards a better understanding of context and context-awareness,” in *Proc. of the CHI 2000 Workshop on The What, Who, Where, When, and How of Context-Awareness*, April 2000. [Online]. Available: ftp://ftp.cc.gatech.edu/pub/gvu/tr/1999/99-22.pdf

[7] C.-K. Chang and J. Huang, “Video surveillance for hazardous conditions using sensor networks,” in *Proc. of the IEEE International Conf. on Networking, Sensing and Control*, March 2004, pp. 1008–1013.

[8] L. Jiao, Y. Wu, G. Wu, E. Y. Chang, and Y.-F. Wang, “The anatomy of a multi-camera security surveillance system,” in *ACM Multimedia System Journal Special Issue, V.10, No.2*, Oct. 2004.

[9] A. Jain, Y.-F. Wang, and E. Y. Chang, “A collaborative camera system for surveillance,” in *UCSB Technical Report*, Nov. 2004.

[10] D. Yang, H. Gonzalez-Banos, and L. Guibas, “Counting people in crowds with a real-time network of image sensors,” in *Proc. of IEEE ICCV*, Oct. 2003.

[11] Z. Sun, G. Bebis, and R. Miller, “On-road vehicle detection using optical sensors: a review,” in *Proc. of the IEEE Intelligent Transportation Systems Conference*, Oct. 2004, pp. 585–590.

[12] O. Sidla, L. Paletta, Y. Lypetskyy, , and C. Janner, “Vehicle recognition for highway lane survey,” in *Proc. of the IEEE Intelligent Transportation Systems Conference*, Oct. 2004, pp. 531–536.

[13] Z. Yin, F. Yang, H. Liu, and B. Ran, “Using image sensors to measure real-time traffic flow parameters,” in *TRB Annual Meeting*, 2004. [Online]. Available: http://www.topslab.wisc.edu/resources/publications/ran_2004_2174.pdf

[14] N. S. Love, I. Masaki, and B. K. P. Horn, “Efficient traffic monitoring,” in *MIT ITRC Report*, 2003. [Online]. Available: http://www-mtl.mit.edu/researchgroups/itrc/ITRC_report/MTL_2003_report/nsl_mtl_annual_2003.doc

[15] H. Lee and H. Aghajan, “Collaborative self-localization techniques for wireless image sensor networks,” in *Proc. of Asilomar Conference on Signals, Systems, and Computers*, 2005.

[16] J. P. Hill, “Design and implementation of wireless sensor networks for habitat monitoring,” M.S. Thesis, U.C. Berkeley, 2003.

[17] J. C. Chen, L. Yip, J. Elson, H. Wang, D. Maniezzo, R. E. Hudson, K. Yao, and D. Estrin, “Coherent acoustic array processing and localization on wireless sensor networks,” in *Proc. of the IEEE*, vol. 91, no. 8, Aug. 2003, pp. 1154–1162.

[18] W. Feng, E. Kaiser, W. Feng, and M. L. Baillif, “Panoptes: Scalable low-power video sensor networking technologies,” *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 1, pp. 151–167, May 2005.

[19] Z. Cao, Z. Ji, and M. Hu, “An image sensor node for wireless sensor networks,” in *ITCC’05*, vol. 2, 2005, pp. 740–745.

[20] M. Rahimi, D. Estrin, R. Baer, H. Uyeno, and J. Warrior, “Cyclops: image sensing and interpretation in wireless networks,” in *Proc. of ACM SenSYS 2004*, Nov. 2004.

[21] R. Kulkarni, D. Ganesan, and P. Shenoy, “The case for multi-tier camera sensor networks,” in *NOSSDAV’05*, Washington, USA, June 13–14 2005.

[22] J. Polastre, R. Szewczyk, and D. Culler, “Telos: Enabling ultra-low power wireless research,” in *IPSN/SPOTS 2005*, Apr. 25–27 2005.