

An Address-Event Image Sensor Network

Thiago Teixeira, Eugenio Culurciello
Electrical Engineering
Yale University
New Haven CT 06520
eugenio.culurciello@yale.edu

Andreas G. Andreou
Electrical and Computer Engineering
Johns Hopkins University
Baltimore MD 21218

Abstract—We discuss an imaging architecture for sensor network applications, that employs a 32×32 Address-Event Representation (AER) imager. At the sensor level, pixels convert light intensity into a pulse density modulated stream of address events. Two different types of COTS wireless radio nodes are used, along with two separate approaches to wireless data transmission – one as a train of AER addresses, and the other as a histogram of the active pixels. Information transmitted in the limited-bandwidth network yields effective means for detection and partial recognition of the object even at very low bit rates, yielding a maximum experimental frame-rate of close to 6fps.

I. INTRODUCTION

Restricted by stringent power budgets, Wireless Sensor Networks (WSN) are limited to a selection of components that consumes power sparingly [1], [2]. This is why the majority of applications of WSNs confine themselves to the scalar sensing domain, employing sensors such as thermometers, accelerometers and photodetectors. For vectorial data sensors, the energy cost is twofold: sensors themselves are typically not designed for low-power applications, and, what is more, such vast amounts of data represent an extra power expenditure when transmitted through a radio. Therefore, WSNs must, all at once, address the issue of power management, component-wise power reduction, and minimization of transmitted data.

In this paper, we report on a WSN system that focuses on the latter two issues: first, we employ an Address-Event Representation (AER) CMOS imager (the *ALOHA* imager) [3], which presents low power requirements; second, we analyze the impact of AER on transmission cost, proposing and comparing two different transmission algorithms.

The system uses commercial, off-the-shelf (COTS) WSN hardware [4] using standard *TinyOS* software interfaces. Other groups working on image sensor network have used COTS image sensors [5], [6], which typically scan and transmit every pixel, whether they carry relevant data or not. Since AER imagers are event-based, they guarantee data relevance and minimize data payload in situations when events are scarce. This translates in an overall power benefit, due to reduced processing and transmission needs.

II. EVENT-BASED CMOS IMAGER

The *ALOHA* imager [3] is a 32×32 AER grayscale image sensor with high dynamic range. It is inspired on sensors in the human cornea, which produce electrical spikes with a rate proportional to the amount of light gathered by them. Each

pixel in the *ALOHA* signals an event when a certain amount of photons are received, and due to physical restrictions of integrated circuits, their output is multiplexed into a 10-bit bus. This constitutes the Address-Event Representation, since each event is represented by the 10-bit address of the originating pixel (5 bits for the x coordinate and 5 for the y).

The *ALOHA* utilizes a standard Acknowledge-Request protocol to communicate events to the outside world: every time a new event is signaled, the *ALOHA* latches its address onto the bus and sets the request bit (*Req*) high. It will only lower the request and wait for another event after the acknowledge bit (*Ack*) is raised by the reader (Figure 2). If, while the bus is in use, one or more pixels signal events, the *ALOHA* simply drops them. This does not produce any noticeable anomalies since the arbiter selects, in effect, a random sample of all events. The proportionality between event-frequency and pixel-brightness guarantees that the probability of an event being transmitted is as high as the importance of the event.

At this point it should be stated that although AER is just a data representation format, it imposes a degree of imager-level information filtering as well. This is because an “event” is, in itself, the presence of a relevant feature. Therefore, AER sensors are typically suited for data processing. Additionally, the advantages of AER vary from one application to another. One example situation where the *ALOHA* would thrive is in the localization of light-emitting events (car headlights, forest



Fig. 1. System overview: WSN node (mote) with *ALOHA* sensorboard streams captured images to a base node connected to a PC.

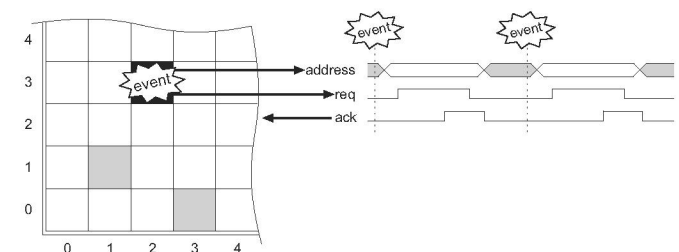


Fig. 2. *ALOHA* control signals: Once an event is fired, the address bus is latched and *Req* goes high. The chip, then, waits for an *Ack* before it sends another event.

fires, explosions) in dark scenes.

If one desires to display an image from a stream of events output by the ALOHA, the brightness of each pixel can be calculated from how frequently the pixel's address occurs in the event stream. This frequency can be obtained in one of two ways: by counting the number of occurrences in a given time; or by counting the time between two consecutive occurrences. In this project, this is done in the first manner. The second approach requires precise timing measurements, which are unavailable at the remote side (due to power, cost, etc) as well as the base side (due to the lags associated with wireless transmission).

III. SENSOR NETWORK HARDWARE

Two similar types of sensor nodes (or *motes*) were fitted with the same custom daughter board (or *sensorboard*) and tested for comparison. The motes employed were the Mica2 (Fig. 3) and MicaZ, which differ mainly in their radio controllers.

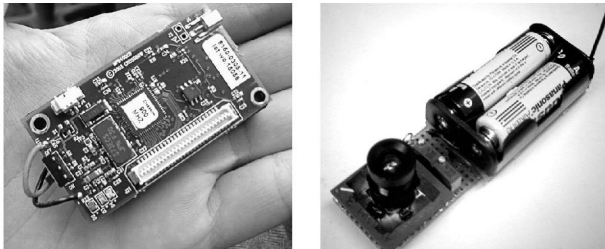


Fig. 3. Mica2 mote (left) and Mica2 with ALOHA sensorboard (right)

While Berkeley's Mica2 motes sport a 900MHz radio with a maximum data-rate of 38.4kbps , the ZigBee-capable version of the Mica2, dubbed MicaZ, achieves up to 250kbps through the 2.4GHz band. The inclusion of the former in this experiment is mainly to provide a frame of reference for comparison against the previous implementation of the ALOHA-based sensor node, which operated on that hardware platform. Their other features include, the Atmel Atmega 128L micro-controller, TinyOS compatibility, and support for the same sensorboard interface.

In our setup, the imaging node was a single hop away from the base station. This provides a best-case scenario, or an upper-bound in network performance. In future implementations, the effect of multihopping shall be included.

IV. BASE SOFTWARE

On the local side, a *base* mote and a PC are connected through UART. The base mote runs the standard TinyOS TOSBase software which automatically forwards all received radio packets to the PC, and all UART messages to the WSN.

On the base PC, a custom Java program reads these packets and displays images in real time. The user has the option to configure the number of events from which an image is formed (using the method already discussed). This number is called the *integration period* (Fig. 4), and it can be defined as either a function of time or the total number of events to be collected.

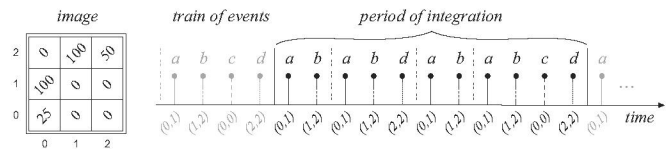


Fig. 4. A train of addresses as received on the base from a theoretical 3×3 imager similar to the ALOHA. The integration period can be changed on-the-fly.

V. MOTE-SIDE SOFTWARE

On the remote side, a custom TinyOS software controls the ALOHA imager, collecting events as they are available. Instead of immediately forwarding each event to the radio channel, the program alternates between a *gathering phase* and a *transmission phase* (Figure 5). This approach aims to dissolve the cost of transmission overheads throughout a larger number of events, minimizing the energy-per-bit ratio.

In the paragraphs that follow, the two operating states are described in further detail.

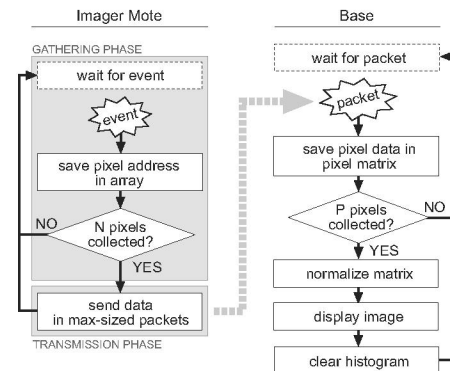


Fig. 5. Software overview: diagram of event-based nature of the problem.

During the gathering phase, the mote continuously acquires data from the ALOHA until a predetermined amount N of events is collected. This amount is called the *acquisition period*, since it may alternatively be defined as a function of time. Once the imager data has been collected, it is usually desirable to have the network provide some level of data interpretation. To transmit the address-event data to a viewing station in a compact form, two different transmission formats were investigated. This aims to control the power consumption or allow for higher frame-rates – whichever is most important to the application.

One approach (dubbed *DETA* and discussed in further detail in Section V-A) is to simply send the acquired events in the same way as they are read, that is, in address-event representation. This keeps any processing overhead to a minimum, but may not be optimal in some cases. Namely, when exposed to a bright scene, each pixel would likely signal multiple events. This would result in the same address being transmitted multiple times, in an seemingly redundant manner.

The issue here is that number of times an address appears is *not* irrelevant data, since it reflects the brightness of a pixel (or the rate at which it fired events). Thus, it may be useful in these circumstances to transmit both the each address active

pixel and number of events that it has signaled. This approach is discussed with greater detail in Section V-B. Note that, as only active pixels are sent, it implies that all transmitted pixels have collected at least 1 event, and 0-valued pixels are discarded. Therefore, this is the same as transmitting only non-zero-valued pixels from a regular imager's output.

Naturally, a third approach is to send the entire scanned image, much like a traditional imager would. Intuitively, this format is useful in scenes where the vast majority of pixels fire events (high number of active pixels per total number of pixels), since the cost of transmitting a zero-valued pixel is counterbalanced by the profit due to the omission of each pixel's address.

The first two approaches were investigated in this work. The next sections describe them in more detail.

A. Direct Event Transmission Algorithm

The first approach described in the previous section, the Direct Event Transmission Algorithm (DETA), simply collects events from the ALOHA imager into an array (or train of addresses), and sending that data raw. Due to the maximum payload size for TinyOS packets being by default 29 Bytes, the array is divided in 28-Byte slices prior to transmission. Figure 6 shows the DETA data formatted into a TinyOS packet.

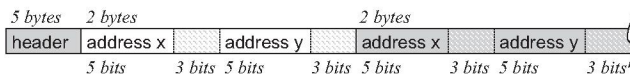


Fig. 6. DETA Packet format: notice the 3 unused bits for each data byte.

Since the ALOHA events are comprised of 10-bit addresses, and because TinyOS packets organize data in the traditional 8-bit manner, a second version of the DETA was implemented, removing the 6-bit waste per address (as seen in Fig 6). This leads to higher power efficiency and data transfer rates. Dubbed DETA2, this optimized version of the DETA reduces the packet structures to that seen in Figure 7.



Fig. 7. DETA2 Packet format: the waste is gone.

As mentioned before, the DETA family of algorithms become less efficient in scenarios where only a handful of pixels are active and highly excited. This is, though, a basic characteristic of AER, and may simply suggest that the event type has been poorly chosen for that specific task. Additionally, given that the sensitivity of the ALOHA imager can be externally tweaked, one may either adjust it depending on the expected brightness profile of the subject or employ a small control circuitry to dynamically do so.

B. Sparse Frame Transmission Algorithm

This second algorithm takes advantage of the scenario where active pixels are likely to trigger multiple events. It works by recording the number of events signaled by each pixel,

and transmitting this data after the acquisition period. This is, in fact, a Data-Address-Event representation, since both the address and the data of each active pixel are transmitted (Fig. 8).

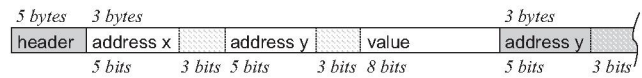


Fig. 8. SFTA Packet format.

As was the case with the DETA, further optimization of the packet format is possible, and such an approach is explored with the SFTA2, as shown in Figure 9.

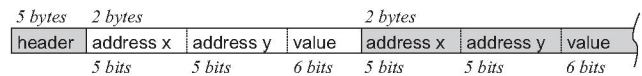


Fig. 9. SFTA2 Packet format.

In the SFTA2, though, the data consists of 6-bit numbers, as opposed to the 8 bits used in the SFTA. In order to compensate for this loss in bit-depth, any time the data d for a pixel requires more than 6 bits, the SFTA2 sends that pixel twice: once with the $d = 2^6$ and another time with $d = d - 2^6$. This situation was not expected to occur often in our tests, though, as the probability of a pixel triggering 2^6 events given our acquisition period is negligible.

It is important to notice that neither the SFTA nor the DETA carry the original timing information contained in the AER data stream. Furthermore, the SFTA goes a step further and also discards the order in which the events are signaled – only the event-rate is kept. In this experiment, no adverse effect is produced from this, as the discarded data is irrelevant to the visualization algorithm employed on the PC.

VI. RESULTS AND ANALYSIS

With the ALOHA pointed directly toward a fluorescent light source placed at a distance of nearly 6 feet from the mote (as shown in Figure 11), the two families of algorithms were employed with acquisition periods ranging from 500 to 3000 events per transmission. The plots in Figures 10, 12 display the frame latency for each setting. For this experiment, the integration period was always the same as the acquisition period.

As can be seen in the plots, the maximum frame rate achieved was with the DETA2 running on a MicaZ: 5.95 frames per second, for an acquisition period of 500 events. The SFTA2 performed similarly, coming at 5.55 *fps*. At 2000 events per transmission, a frame rate of 2.20 *fps* was detected for the MicaZ running SFTA2. For longer acquisition periods, the tendency is that the SFTAs out-perform the DETAs, since the transmitted data for the former is bounded above by the dimensions of the imager. As is typical of many AER sensors, though, the definition of an event is desired to be such that a low event-rate is produced. These circumstances call for the DETA family, and Figure 11 demonstrates that a recognizable image may be formed from a small number of events.

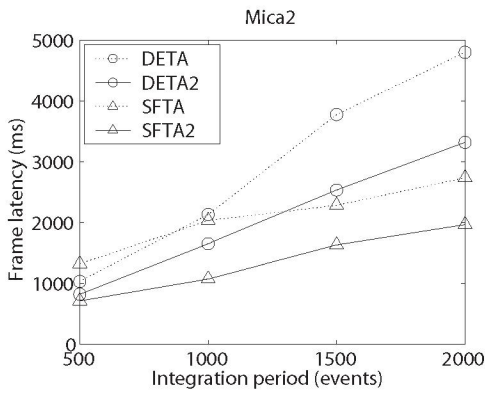


Fig. 10. Mica2 frame latency as a function of the acquisition period.

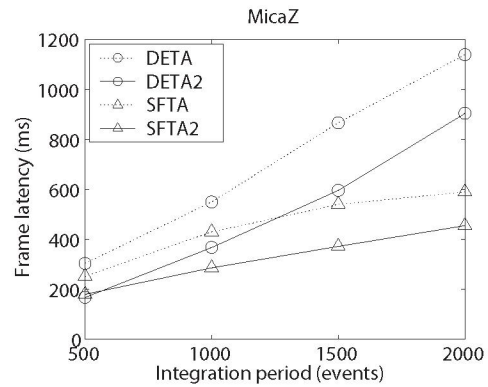


Fig. 12. MicaZ frame latency as a function of the acquisition period.

Given that the Mica family of motes is powered by 2 AA batteries when remotely deployed, power consumption was measured for the source voltages of 2.25V, 3.0V and 3.3V. The result is shown in Table I.

Mica2	2.25V	3.0V	3.3V
collection phase	41mW	63mW	76mW
transmission phase	55mW	117mW	155mW

MicaZ	2.25V	3.0V	3.3V
collection phase	55mW	75mW	96mW
transmission phase	65mW	117mW	145mW

TABLE I

Assuming a 50% duty cycle between the transmission and gathering phases, two 1800mAh AA batteries with perfect 3.0V potentials throughout their lifetime, would span

$$2 \times 1800mAh \times 3.0V \times \frac{2}{63mW + 117mW} = 120hours,$$

or the equivalent of 5 days of non-stop operation and radio communication. The current draw for the Mica2 has been found [7] to be 11mA when simply listening to the radio. This is equivalent to 33mW of power consumption. Assuming that the LEDs on the mote consume around 12mW, this implies the sensorboard alone is responsible for 63mW – 33mW – 12mW = 45 mW.

VII. CONCLUSION

We have reported on an address-event imaging WSN able to stream pictures formed from 500 events at frame-rates



Fig. 11. Picture of fluorescent lamps taken by the ALOHA (left). Pictures taken with an acquisition period of 200 events (middle) and 500 events (right).

as high as 6fps. Power consumption was measured to an average of 90mW for the entire system running on 2 AA batteries. This allows 5 days of uninterrupted use of the imager and radio. Two different approaches for data transmission were compared, each yielding favorable results under mutually exclusive conditions: the SFTA family outperforms DETA when the number of events per active pixel is high; conversely, the DETA is preferable for the opposite situation. Moreover, frame-rates can be further improved for the SFTA by transmitting one row of active pixels at a time and removing the redundant y-coordinate from each pixel's address.

VIII. ACKNOWLEDGEMENTS

Fabrication was provided by MOSIS. This work was supported in part by the National Science Foundation grants *Integrated Sensing: Cell clinics on a chip*, ECS-0225489 and *A Comparative Study of Information Processing in Biological and Bio-inspired Systems: Performance Criteria, Resources Tradeoffs and Fundamental Limits*, EIA-0130812.

REFERENCES

- [1] I. Akyildiz, W. Su, Y. Sankarasubramanian, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 42, no. 5, pp. 102–114, August 2002.
- [2] Various Authors, *Proceedings of IEEE Special Issue on Sensor Networks*. IEEE, August 2003.
- [3] E. Culurciello and A. G. Andreou, "ALOHA CMOS imager," in *Proceedings of the 2004 IEEE International Symposium on Circuits and Systems ISCAS '04*, May 2004.
- [4] Crossbow, *MPR/MIB Mote Hardware Users Manual*, Crossbow Technology Inc., San Jose, CA 95134-2109, 2004, <http://www.xbow.com/>.
- [5] M. Rahimi, D. Estrin, R. Baer, H. Uyeno, and J. Warrior, "Cyclops: image sensing and interpretation in wireless networks," in *Second ACM Conference on Embedded Networked Sensor Systems, SenSys*, Baltimore, MD, November 2004.
- [6] T. Ko and N. Berry, "Distributed feature extraction for event identification," in *Ambient Intelligence: Second European Symposium, EUSAI Proceedings*, vol. 3295, Eindhoven, The Netherlands, October 2004, pp. 136–147.
- [7] B. C. G. W. A. V. Shnayder, M. Hempstead and M. Welsh, "Simulating the power consumption of largescale sensor network applications," *Proc. the 2nd international conference on Embedded networked sensor systems (SenSys 2004)*, November 2004.