Image Sensor with Focal Plane Change Event Driven Video Compression

Yu M. Chi[1,2,3], Ralph Etienne-Cummings[3] and Gert Cauwenberghs[2,1]

[1] Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093

[2] Division of Biological Sciences, University of California, San Diego, CA 92093

[3] Department of Electrical and Computer Engineering, Johns Hopkins University, MD 21218

Abstract— An image sensor with focal plane based hardware acceleration of video compression is presented. The 90×90 pixel CMOS image sensor provides in-pixel processing of intensity changes, serving as an analog memory and processor for temporal image difference computation. Surveillance quality videos of up to a 48:1 compression ratio via a temporally compensated DCT based compression algorithm is attained with just an 18.5 MHz micro-controller. Power consumption is 225mW during full operation and 6mW during a sleep mode that continuously monitors for change events to trigger encoding.

I. INTRODUCTION

Video compression algorithms are among the most computationally intensive tasks in current imaging technology. Sophisticated compression schemes like H.264 provide, simultaneously, high compression rates and low visual distortion. Implementation, however, is costly in both terms of power consumption and hardware complexity. For applications like low power, long term surveillance, an alternative approach is justified for two reasons. First, most scenes are static - necessitating a sensor platform that does not expend unnecessary bandwidth and energy during these periods. Conventional video compression algorithms are successful in the first regard, but not the second. Secondly, the tradeoffs between bandwidth, power and distortion are different. The first two needs to be prioritized with the provision of maintaining a visual quality sufficient to identify events of interest.

Our approach takes a fundamentally different philosophy to the traditional motion estimation/motion compensation based video compression algorithms. We combine traditional video compression techniques with the analog processing capability of the focal plane processing in CMOS image sensors. The change detection imager used [3] is capable of storing and detecting image brightness changes at the pixel-level autonomously and without digital processing or memory. Not only does this provide a "wake-up" trigger for data transmission, it serves as a co-processor for a conditional replenishment compression algorithm [6] [7] to provide for full motion video.

In this paper, we significantly improve and describe in detail the video compression algorithm briefly introduced in [9]. The compression scheme is generalized to different environments as well as refined to achieve markedly improved visual quality and performance.

II. IMAGING HARDWARE

A. Image Sensor

Each active pixel sensor in the 90×90 imaging array includes the standard photodiode, reset, buffer and access transistors along with an additional three transistor, two capacitor comparator circuit. The operation of this imager, which can perform both in-pixel change detection and ADC has been presented at previous conferences [3] [8]. As output, the imager provides a standard analog image intensity output as well as a digital flag (Fig. 1) indicating the presence and direction of intensity changes. The imager consumes 4.2mW at 30fps.

A typical output for the change detection is shown in Fig. 1. Thresholds for change detection is user adjustable with a rejection band in order to eliminate spurious readings. The change detection output is completely independent of any external processing and serves as a low power wake up interrupt.

B. Test Board

To examine the performance of the change detection imager in a low cost, low complexity hardware environment, the imager is integrated with a micro-controller (dsPIC33) on a 4 AA battery powered circuit board (Fig. 2). The micro-controller operates at a clock rate of 18.5 MHz and includes just 16 kilobytes of onboard SRAM. Since the image sensor handles temporal de-correlation via the change detection output, the remaining tasks in an compression scheme, spatial de-correlation and variable length coding, are handled by the micro-controller. The standard interface is through a RS232 link which enables easy interfacing with a variety of hosts.

III. COMPRESSION ALGORITHMS

The overall design goals for the compression algorithm are targeted for operation over a typical low bandwidth wireless network and are as follows:

1) Full video capability
2) Provision for high quality compressed still image transfer
3) Operation over extremely low bandwidth
4) Constant data rate output regardless of image content

Fig. 1. Imager change detection output [8] showing a person moving in the frame. Positive change threshold events are shown in white, and negative change threshold events are shown in black. Change events of either polarity trigger DCT encoding and transmission of imager intensity output over an 8x8 block of pixels.
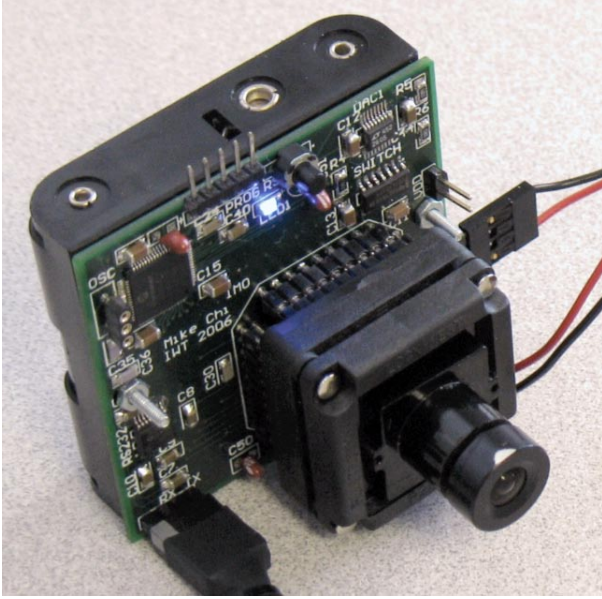

Fig. 2. Camera circuit board showing image sensor, lens and microcontroller.

A. Video Coding

The main challenge in real-time video coding on a platform with limited processing power and bandwidth is maintaining a high image quality while keeping a constant low bitrate. Because of processing power limitations in addition to the constraints imposed by real-time operation, an elaborate bit-rate controller is impossible. In most low bandwidth platforms, data is sent in fixed packet sizes at regular intervals. Thus our video encoder must be designed to guarantee that each frame in the video must fit inside the defined data size to provide a consistent and regular video stream.

To maximize image quality and work within these constraints, a temporally compensated, progressively updated DCT framework is utilized (Fig. 3). By updating only changing blocks, the encoder is able to take advantage of the large temporal correlation between frames and reduce the data rate. The progressive DCT coding prioritizes the most important visual features (the low frequency content). Since the dsPIC has limited computational capabilities, a fast binary DCT approximation [4] which can encode an 8x8 block in less than
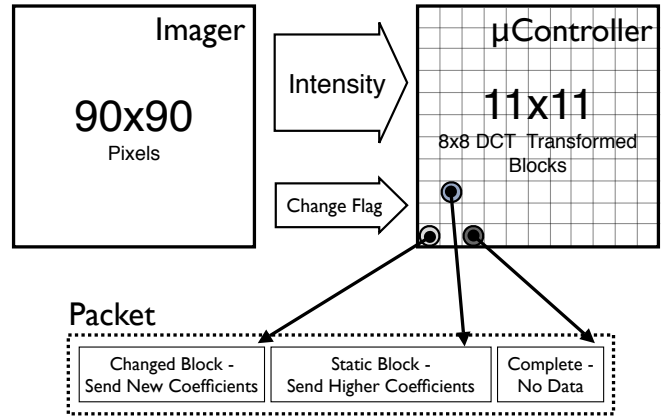

Fig. 3. Conditionally updated progressive DCT based video compression algorithm. The microcontroller conditionally updates or refines 8x8 blocks with assistance from the image sensor's change detection output.

$200\mu s$ is used in lieu of standard floating or fixed point DCT.

The encoder partitions each 90 by 90 frame into 11 by 11 blocks of the standard 8 eight pixels in each dimension, throwing away a one pixel border. Encoding begins with imager readout where the dsPIC stores the image along with tabulating the number of pixel changes in a block. The blocks are split into three types. The first are blocks which contain a number of changed pixels greater than a set threshold. The second are blocks that have not changed but still need higher frequency DCT coefficients to complete the image. Finally, the third are simply blocks that have not changed and have already been fully transmitted.

The first and second types are flagged for encoding. To pack the entire frame in a packet, the number of DCT coefficients allocated to a block is simply the packet size divided by the number of blocks to be sent. Blocks that have changed are transformed whereas the data for blocks that need only higher frequency coefficients are pulled from memory to avoid encoder-decoder mismatches from small, below threshold, drifts. Coefficients are quantized by a factor of 16 to fit coefficients as 8-bit signed integers. Finally, the encoder also checks to see if the remaining coefficients in a block are all zeros to avoid sending superfluous information.

The primary challenge [9] with this encoding scheme compared to traditional MPEG-like algorithms is the lack of an encoder feedback loop which eliminates the steady state error due to the distortion error introduced in compression. This is solved by two additional steps. First blocks are marked as changed if the number of pixels that report a temporal difference exceed a threshold in not only the current frame, but also the previous. This eliminates the artifacts at the edges of a block due to moving object. Secondly, each block also has an associated age counter that triggers a refresh after a set period. To avoid flushing all blocks at the same time in a static environment and incurring a severe momentary degradation in image quality, the counter is incremented with a random component to disperse the refresh over an average of frames

Fig. 4. Video output from sensor board at 50:1 (right) and 5:1 (left) compression rates.

equal to the compression ratio (time to fully transmit every DCT coefficient for every block).

A short header containing a map of blocks and their status is attached alongside the data stream. Each block has a 2-bit value indicating whether it is changing, updating or finished. For the 90x90 imager, this adds an addition 31 bytes per frame. The decoder reads this map and updates it's copy of the DCT coefficients as directed before taking the inverse transform and displaying the image.

The maximum feasible compression rate for this encoder is equal to the number of blocks in bytes which guarantees that every block receives at least one coefficient in the worst case. For the 90x90 imager, this corresponds to 121 bytes for the data plus the 31 byte header. At 10 frames a second *the effective data rate is only 1.5kbps*, which meets the constraints of many low bandwidth links.

B. Still Image Encoder

In certain circumstances it is also desirable to have a snapshot mode to acquire an image of interest at a high image quality. A possible scenario involves using the video encoder to monitor an area with the low definition, yet high frame rate video stream. When an event of interest occurs, a high quality still can be stored and later transmitted. Since the still image transmission mode need not be real-time, the requirement of constant size packets is eliminated. Thus, variable length coding along with a more optimized quantization scheme is possible which greatly improves the image quality.

A JPEG-like algorithm, simplified to run on the microcontroller is used to compress the image. The same fast DCT approximation [4] is applied to the 8x8 blocks followed by the JPEG perceptual quantization matrix. Entropy encoding is facilitated by a combined RLE and fixed table Huffman code. For most images a 3.5:1 to 5:1 compression ratio, at a minimal loss in visual quality, can be achieved depending on the image's characteristics.

IV. PERFORMANCE AND IMAGE QUALITY

The video compression algorithm were implemented in C on the micro controller and operates as expected. The imager is set to detect pixel intensity changes of 100mV (corresponding to a brightness change of 20 in an 8-bit pixel) with a block flagged as changed if more than 5 pixels report a change. Two sample images are show in in Fig. 4. The left image
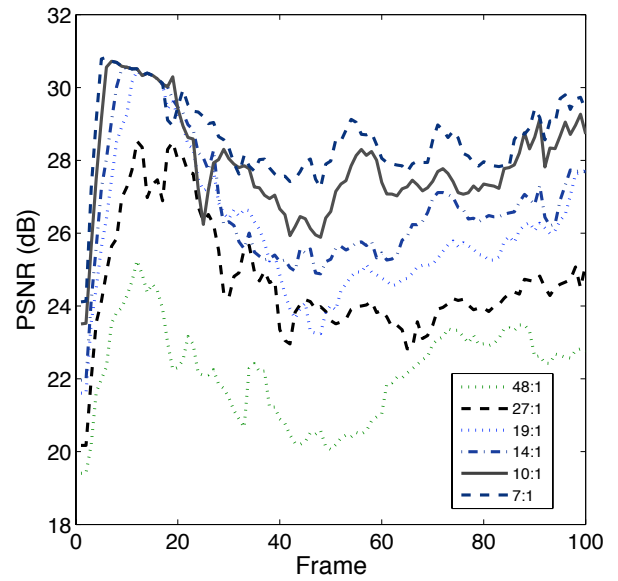

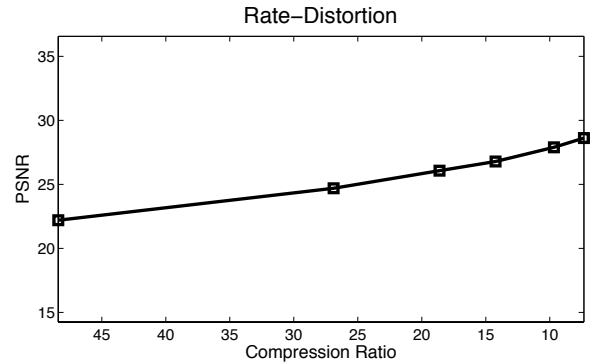Fig. 5. PSNR for each frame in the test sequence at different compression rates.

Fig. 6. Rate-Distortion curve for the video sequence. The PSNR for each bit-rate is the average of all 100 frames in the sequence.

was coded with the high quality mode and the right with the maximum compression rate. It is important to note that even though the highly compressed picture suffers from a high level of artifacts and distortion, it is still fully usable in a security context, especially given that it is part of a full motion video stream.

In order to further evaluate the performance of the video sensor architecture and compression algorithm, the same algorithm was ported to a PC environment that simulates the change detection imager with the aforementioned parameters using pre-made movies. This allows the algorithm to be better characterized by enabling distortion measurements with commonly used test sequences.

The *hall* sequence is a good test bench, corresponding to a typical surveillance scenario with a static camera and background, punctuated by a person entering the frame. The performance of the encoder at different rates is plotted in Figs. 5 and 6 along with a sample frame in Fig. 7. As seen from the PSNR plot, the image quickly converges to a
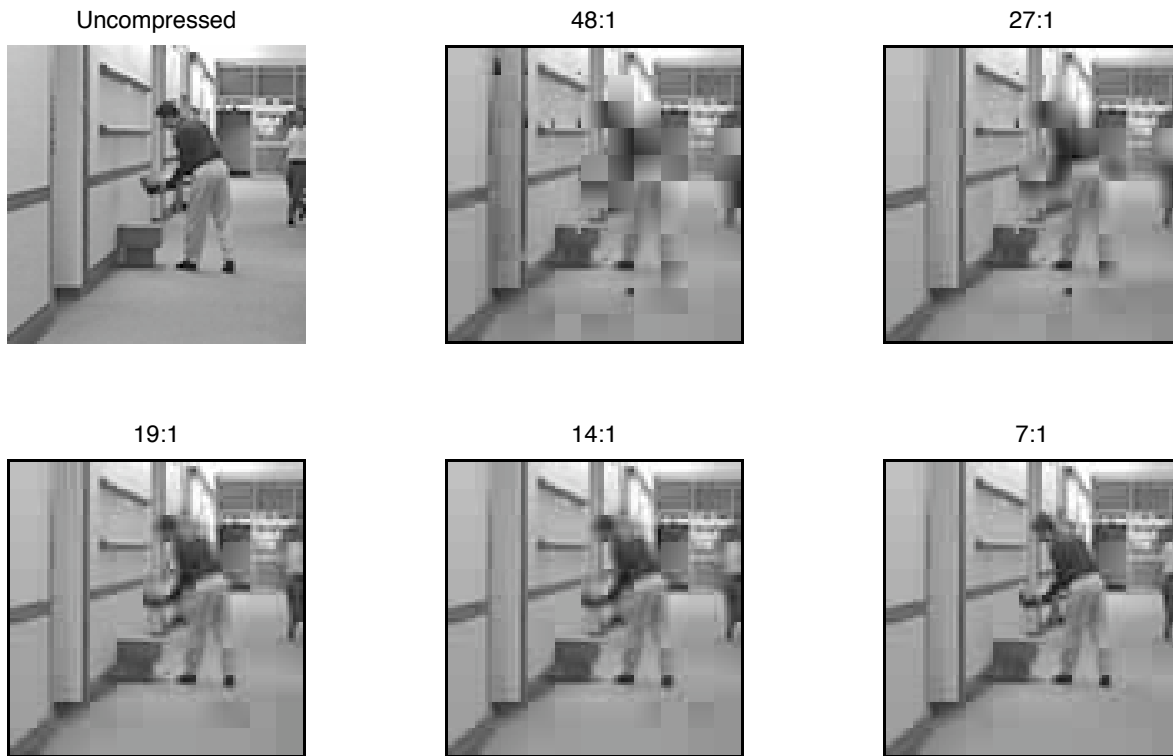
Fig. 7. Final frame of the example video coded at different bitrates showing the visual distortion in the presence of motion.

high quality approximation in the beginning since no motion is present, allowing all of the bandwidth to be allocated towards transmitting higher frequency DCT coefficients. In the presence of motion after the static period at the start, the encoder is able to maintain a high image quality except in the 48:1 case due to insufficient bandwidth. Regardless of bitrate, the encoder maintains a constant average error without error accumulation [9] due to the periodically dispersed block refresh to guarantee that even static blocks are updated to avoid glitches and long term drifts.

All compression rates below 48.5:1 provide images that enable a user to clearly discern the action in image. Even though the maximum compression rate results in large blocking artifacts, it is still sufficient to indicate motion as well as events of interest.

V. CONCLUSION

We demonstrate a low power video sensor platform for low bandwidth applications using a minimal of hardware by leveraging focal plane processing. By using a computational image sensor with focal plane motion processing, the overall complexity of the system is reduced, making it suitable for ad-hoc deployment and battery powered operations. In the proposed scenario, the image sensor itself can autonomously monitor for change events through the pixel level circuits without external digital processing. When video transmission is desired, we also demonstrate that despite the limited processing facilities of the board's microcontroller, a DCT based

temporally compensated video encoder can be implemented with assistance from the focal plane processing capabilities of the imager. This design illustrates the possibility of surveillance type images in applications heavily limited by power and bandwidth.

REFERENCES

[1] R. Puri, A. Majumbar, P. Ishwar, K. Ramchandran, "Distributed video coding in wireless sensor networks," *IEEE Signal Processing Magazine,* Jul. 2006.
[2] E. Culurciello, A. Savvides, "Address-Event Image Sensor Network," *IEEE International Symposium on Circuits and Systems,* May. 2006.
[3] U. Mallik, M. Clapp, E. Choi, G. Cauwenberghs and R. Etienne-Cummings, "Temporal change threshold detection imager," *IEEE Solid State Circuits Conference* Feb. 2006.
[4] J. Liang and T. D. Tran, "Fast Multiplierless approximation of the DCT with the lifting scheme," *IEEE Transactions on Signal Processing*, Vol. 49, No. 12, pp.3032-3044, Dec. 2001.
[5] W.C. Feng, B. Code, E. Kaiser, M. Shea, L. Bavoil, "Panoptes: Scalable low-power video sensor networking technologies," *Proc. 11th ACM Int. Conf. Multimedia,* Nov. 2003.
[6] Y. Chiu, T. Berger, "A Software-Only Videocodec Using Pixelwise Conditional Differential Replenishment and Perceptual Enhancements," *IEEE Trans. Circuits and Systems for Video Technology,* April 1999.
[7] E. Amir, S. McCanne, M. Vetterli, "A layered DCT coder for Internet Video," *Proc. International Conference on Image Processing* Sept. 2006
[8] Y. M. Chi, U. Mallik, M. Clapp, E. Choi, G. Cauwenberghs and R. Etienne-cummings, "CMOS Camera With In-Pixel Temporal Change Detection and ADC," *IEEE J. Solid State Circuits,* vol 42, pp. 2187-2196, October, 2007.
[9] Y. M. Chi, P. Carpenter, K. Colling, G. Cauwenberghs and R. Etienne-Cummings, "Video Sensor Node for Low-Power Ad-hoc Wireless Networks," *IEEE Conference on Information Sciences and Systems* March, 2007.