AN AUTOMATED FACE RECOGNITION SYSTEM FOR INTELLIGENCE SURVEILLANCE: SMART CAMERA RECOGNIZING FACES IN THE CROWD

Y. M. Mustafah [a,b], *A. W. Azman* [a,b], *A. Bigdeli* [a], *B. C. Lovell* [a,b]

[a] NICTA, 300 Adelaide Street, Brisbane, QLD 4000, Australia
[b] ITEE, University of Queensland, Brisbane, QLD 4072, Australia

ABSTRACT

Smart Cameras are rapidly finding their way into Intelligent Surveillance Systems. Recognizing faces in the crowd in real-time is one of the key features that will significantly enhance Intelligent Surveillance Systems. The main challenge is the fact that the high volumes of data generated by high-resolution sensors can make it computationally impossible for mainstream processors. In this paper we report on prototyping development of a smart camera for automated face recognition using high resolution sensors. In the proposed technique, the smart camera extracts all the faces from the full-resolution frame and only sends the pixel information from these face areas to the main processing unit. Face recognition software that runs on the main processing unit will then perform the required pattern recognition.

Index Terms— Smart Camera, Surveillance, High Resolution, FPGA

1. INTRODUCTION

Smart cameras are becoming increasingly popular with the advances in both machine vision and semiconductor technology. In the past, a typical camera was only able to capture images, while with the smart camera concept, a camera will have the ability to generate specific information from the images that it has captured.

So far there does not seem to be an established definition of what is a smart camera. In this paper, we define a smart camera as a vision system which can extract information from images and generate specific information for other devices such as a PC or a surveillance system without the need for an external processing unit.

Figure 1 shows a basic structure of a smart camera. Just like a typical digital camera, a smart camera captures an image using an image sensor, stores the captured image in the memory and transfers it to another device or user using a communication interface. However, unlike the simple processor in a typical digital camera, the processor in a smart camera will not only control the camera functionalities but it is also able to process the captured images so that extra information can be obtained from them.

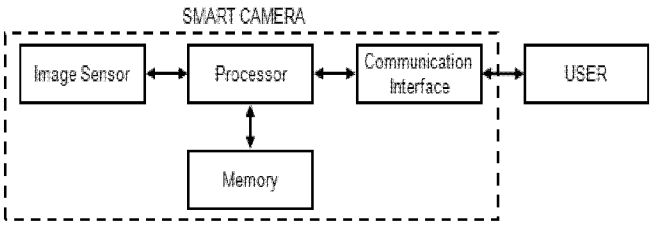

Figure 1*: Basic Smart Camera Architecture.*

Smart cameras have been widely used in many applications. The two major areas where smart cameras are being used are industrial automation and video surveillance. The application of smart cameras for industrial automation is well established, whilst in the video surveillance area, the smart camera is still generally at the research and development stage. In contrast to the manufacturing environment, the use of smart cameras in video surveillance faces a much greater challenge due to the complexity of the objects of interest such as humans, faces, cars, and so on. Moreover the operating environment includes variable lighting, object movement, random numbers of objects, and other artifacts.

2. SURVEILLANCE AND SMART CAMERAS

Video surveillance is becoming more and more essential nowadays. Societies rely on video surveillance to improve security and safety. For security, such systems are usually installed in areas where crime can occur such as banks and car parks. For safety, the systems are installed in areas where there is the possibility of accidents such as on roads or motorways and at construction sites.

Currently, video data is used predominantly as a forensic tool, thus losing its primary benefit as a proactive real-time alerting system. For example, the surveillance systems in London managed to track the movements of the four suicide bombers in the days prior to their attack on the London Underground in July 2005, but the footage was only reviewed after the attack had occurred. What is needed is continuous monitoring of all surveillance video to alert security personnel or to sound alarms while there is still

1-4244-1354-0/07/$25.00 ©2007 IEEE

time to prevent or mitigate the injuries or damage to property. The fundamental problem is that while mounting more video cameras is relatively cheap, finding and funding human resources to observe the video footage is very expensive. Moreover, human operators for surveillance monitoring rapidly become tired and inattentive due to the dull and tiring nature of the activity. Therefore, there is a strong case for automated surveillance systems where computers monitor the video feeds — even if they only help to keep human operators vigilant by sending relevant alarms.

Smart cameras can improve video surveillance systems by making autonomous video surveillance possible. Instead of using surveillance cameras to solve a crime after the event, a smart camera could recognize suspicious activity or individual faces and give out an alert so that an unwanted event could be prevented or the damage lessened. From another perspective, smart cameras reduce the need for human operators to monitor all the video feeds just to detect the activities of interest, thus cutting down operating costs and increasing the effectiveness of the surveillance.

3. RELATED WORK

While there are many smart camera products already available in the market today from a variety of manufacturers such as Tattile, Cognex, Matrix Vision, Sony, Philips, EyeSpector, PPT Vision, and Vision Components, it is still a very active area of research because of the wide range of capabilities of smart camera that could be improved.

One of the earliest publications on smart cameras was by Wolf et al. where they introduced a system that can build a complete model of the torso and recognize various gestures made by a person [1]. The work started with research on a human activity recognition algorithm and soon evolved to the implementation of the software algorithm onto hardware, including Hi8 cameras and Trimedia video capture boards. Their prototype camera is a PC-based system that operates at 30 frames per second (fps) on a 352 x 240 RGB video stream. Another well-known research project was by Bramberger et al [2]. They built a prototype camera called SmartCam which is a fully embedded smart camera system targeted for various surveillance applications such as traffic control. The SmartCam is built by using a VGA resolution CMOS image sensor, several Texas Instruments TMS320C64x DSPs and Intel XScale IXP425 network processor.

Some research utilized the existing smart camera products available in the market. For example, Kleihorst et al. [3] proposed a system that is able to detect and recognize faces implemented on a Phillips's INCA+ camera. Their system runs at rate of 3 fps at a 640 x 480 resolution and can recognize one face in 500ms. Fleck et al. presented a smart camera for embedded and adaptive object tracking in real-time [4]. The implementation was done on Matrix Vision's mvBlueLYNX 420CX Smart Camera. Their camera is capable of robustly following the target over time at a frame rate of over 15 fps at a sensor resolution of 640x480 pixels.

FPGA-based smart cameras are becoming more popular today because of advantages in terms of performance, cost, and rapid conversion to a mass market product. Matsushita et al. proposed an ID CAM, a FPGA and CMOS sensor based smart camera that captures a scene and recognizes the ID of a beacon emitted over a long distance [5]. The system was built using a CMOS image sensor and Virtex-1000 FPGA and runs at a rate of 30 fps and resolution of 192 x 124 pixels. Chalimbaud and Berry [6] proposed an FPGA and CMOS imager based smart camera with template tracking capability. The Stratix EPIS60 FPGA was used as the main implementation platform for the camera and the CMOS image sensor used for the camera capable of capturing image with a resolution of 640 x 480 pixels. In [7], Shi et al. presented the design and implementation of a smart camera, called Gesture Cam, which can recognize simple human hand and head gestures. Their system is based on a Virtex-II Pro FPGA and a CMOS image sensor and works with 320 x 240 resolution.

Table 1 summarized the specification of some smart cameras that have been used in various research works.

Table 1: *Specification of smart camera platforms used in various research works*

Smart Camera	Sensor	Res. (pixel)	Processing Platform	Comm. Interface
Wolf [1]	-	352 x 240	Trimedia TM1300	PCI
Bramberger [2]	CMOS	640 x 480	TMS320C64x DSP	Ethernet, GSM
Kleihorst [3]	CMOS	640 x 480	Xetal, Trimedia	Firewire, RS232
Fleck [4]	CCD	640x480	Spartan-2E FPGA, MPC 8241 PowerPC	Ethernet 100
Matsushita [5]	CMOS	192 x 124	Virtex-1000 FPGA	USB
Chalimbaud [6]	CMOS	640 x 480	Stratix-EPIS60 FPGA	USB2
Shi [7]	CMOS	320 x 240	Virtex-II Pro FPGA	Ethernet, RS232

4. HIGH RESOLUTION SYSTEM

Most of the existing smart camera designs utilize a VGA resolution image sensor (640 x 480 pixels) and some use lower resolutions. Compared to the existing image sensor technology, VGA can be considered as low resolution and not suitable for many video surveillance applications

especially in crowd surveillance. Crowd surveillance usually surveys a wide area with many of objects of interest in view, thus requiring a high resolution camera. High resolution images provide much more detailed information regarding objects in view. For example, in applications such as face recognition, higher resolution images will help to improve the recognition rate — indeed a very high resolution camera could even read nametags and other insignia. Figure 2 shows an example of a region of interest (ROI) extracted from a scene image of a crowd of people (a). The face (b) extracted from a 7 MP (MegaPixel) high resolution image is much more recognizable than (f) extracted from the lower resolution (VGA) scene. The extracted faces were also tested for suitability for automatic detection using a Viola-Jones face detection module [8]. The images (c), (d) and (e) taken with 5, 3, and 1MP sensors were suitable for face detection. However, face could not be detected in the VGA image (f) because the image does not have enough resolution for the face detection module to work.

5. IMPROVING SMART CAMERA DESIGN

We propose a smart camera system that can be used as an aid for face recognition in crowd surveillance. The camera utilizes a high resolution CMOS image capture device and an FPGA based processor for ROI extraction. The system architecture proposed is as shown in Figure 3.

Our system has an internal processor to perform face detection to extract faces from the captured images in real-time. The main motivation to extract faces inside the camera is to conserve as much bandwidth as possible and to save processing time and memory on the client PC which performs the face recognition task.

Note that even in the dense crowd shown in Figure 2, the faces suitable for recognition only represent a very small proportion of the image area. In many scenes, faces would represent less than 1% of the image. Thus the smart camera would not overload the PC by transmitting huge amounts of high-resolution image data which is destined to be discarded immediately after face detection. Such massive data reduction at source by up to two orders of magnitude is an immediate and significant benefit of this approach.

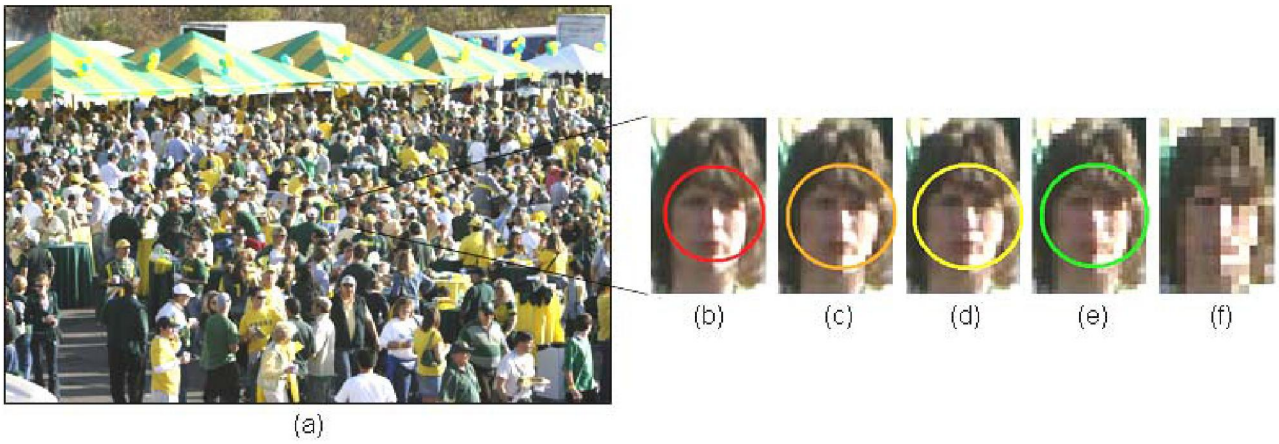

Figure 2*: Overall Scene (a), ROI extracted from scene with resolution of 7MP (b), 5 MP(c), 3MP (d), 1MP (e) and VGA (f).*

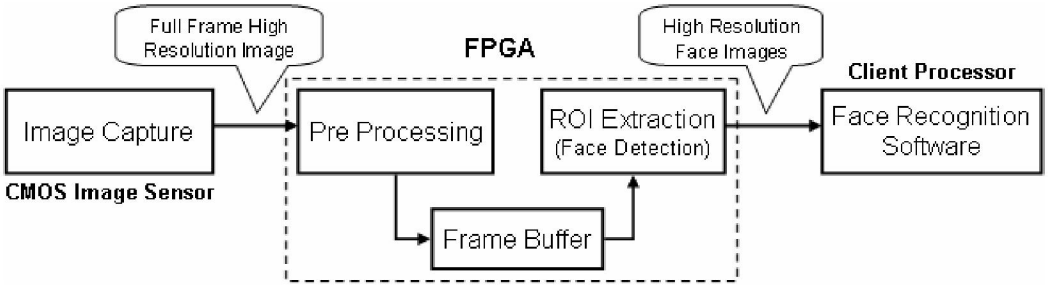

Figure 3*: Proposed System Architecture.*

5.1. High Resolution Image Sensor

Like most of the smart camera prototypes listed on Table 2, we also utilize a CMOS image sensor as it offers high resolution and low noise output. Due to the low power and high speed of CMOS, it is expected in the future that CMOS based image sensors will outperform CCD based image sensors [9].

There are plenty of CMOS image sensors in the market. Table 2 shows the highest resolution sensor product from three leading CMOS image sensor manufacturers; OmniVision [10], Micron [11] and Kodak [12]. It is noticeable that the frame rate is inversely proportional to the resolution of the camera. For wide angle surveillance, since no rapid movements of object of interest is expected, 5 high resolution frames per second can be considered as acceptable baseline performance for the prototype.

Table 2: *High Resolution CMOS Image Sensors*

Manufacturer	CMOS Sensor	Resolution (pixel)	FPS at Full Frame
OmniVision	OV5620	2608 x 1952	7.5
Micron	MT9E001	3264 x 2448	10
Micron	MT9P001	2592 x 1944	14
Kodak	KAC-5000	2592 x 1944	6

5.2. High Bandwidth Communication Interface

High resolution image sensors require a high data transfer rate. Although a smart camera could preprocess the captured images before they are sent to the external devices, it is sometimes necessary for the camera to send the RAW captured images out. For a high resolution camera, a high bandwidth communication interface would be required. This is because, for example, a 5MP sensor working at the rate of 10 fps would require 50 MP of data to be transferred every second if no compression is performed on the image. One pixel might represent several bits of data depending on the color depth of the image. Therefore without compression, a camera with a 5MP high-color (24-bit color depth) frame image working at 10 fps, would need a communication link with sustained transfer rate of 1200 Mega bits per second — no garden variety PC could keep up with this.

However, currently, there are five popular high bandwidth video interface standards available: FireWire 400 or IEEE 1394a, FireWire 800 or IEEE 1394b, USB2, Gigabit Ethernet or GigE, and Camera Link. Table 3 shows the general specification on the five interfaces. USB 2 and FireWire 400 can be considered as unsuitable in term of data transfer speed if we compare them with the current resolution of CMOS image sensor technology. While Camera Link is suitable for very fast data transfer, it only supports one-to-one device connection. This means a network of cameras would not be supported by this interface. GigE and FireWire 800 interfaces can be considered as the most suitable interfaces for the purposed high resolution surveillance as they both have a considerable data transfer speed and allow the networking of the cameras.

Table 3: *Video Interface Standards*

Interface	Data Transfer Rate (Mbps)	Max Cable Length	Max Devices
FireWire 400 (1394a)	400	4.5	63
FireWire 800 (1394b)	800	100	63
USB 2	480	5	127
GigE	1000	100	no limit
Camera Link	3600	10	1

5.3. Reconfigurable Platform for Hardware and Software Processors

Acquiring the right targeted hardware for a smart camera processor can be considered as an important issue. While Application Specific Integrated Circuit (ASIC) provides a high performance and power efficient platform, it suffers from lack of flexibility and can be very expensive due to the high non-recurring engineering (NRE) cost. Digital Signal Processor (DSP) on the other hand only has a single flow of control which could pose problem to meet real-time constraints. The general-purpose processor (GPP) also faces problem in meeting real-time constraint due to poor execution time predictability. Garcia et al. in [13] suggested that reconfigurable hardware as the best option and cost effective for embedded system. Presently, Field Programmable Gate Arrays (FPGA) is one of the most widely used and competitive reconfigurable hardware in the market.

One of the key aspects of FPGA is it has large number of arrays of parallel logics and registers which enable designer to produce effective parallel architecture. Parallel processing is an important feature especially for embedded system that involves high-level computation in real-time, for example, face detection on smart camera processor. Parallel processing allows information to be transferred effectively and obtained end result faster since processing tasks are segregated to be carried out concurrently. At the same time, parallel processing reduce power consumption considerably especially in process which involves back-to-back memory access. Besides that, FPGA allows to incorporate microprocessor on its chip. For our smart camera prototype, Spartan-3 FPGA was chosen as a main processor of the camera. We believe that Spartan-3 platform would impose an interesting challenge so that optimum hardware resource will be utilized in every aspect of the design.

In many cases, the smart camera system involves operations in both hardware logic and software level of implementation. The term "software level" is referring to the smart camera related application software. These include face detection software, gesture detection software and etc. Hardware logic implementation usually performs at higher processing speed compared to software implementation. However, software design is usually cheaper and offers much flexibility. Hence, hardware-software (HW/SW) co-design solution could provide an optimum solution between hardware and software implementation. For our smart camera design, we will employ a PicoBlaze microcontroller architecture. Similar to PowerPC and MicroBlaze, PicoBlaze is another software processor available for Xilinx's FPGAs that is optimized for efficiency and low deployment cost. PicoBlaze is an 8-bit RISC based architecture that is specifically designed for the Spartan and Virtex Xilinx FPGA families. Unlike any other processor, Picoblaze occupies only 96 slices of the FPGA and is extremely flexible and fully embedded within the FPGA. On Spartan-3 FPGA with -4 speed grade, the PicoBlaze is able to run at 88 MHz or 44 MIPS (2 clock cycles per instruction). It has a 1024 x 18-bit wide instructions memory and 16 byte wide general purpose registers.

5.4. Robust Face Recognition System

The uncontrolled environment of crowd surveillance makes a robust face recognition system a necessity. Ideally, a robust face recognition system would be able to recognize faces regardless of the face's expression, angle, features and lighting conditions.

A face recognition system consists of face detection and a face classification part. In order for the system to recognize a particular face, the face must first be detected and then extracted from the captured scene image. The face is then normalized and forwarded to the face classification processor where it could be recognized by comparing it to the faces stored in the database.

The face recognition to be implemented on the system is as proposed by Shan et al in [14]. Their system is comprised of three major components: 1) a Viola-Jones face detection module [8] based on cascaded simple binary features to rapidly detect and locate multiple faces, 2) a normalization module based on the eye locations, 3) Adaptive Principal Component Analysis to recognize the faces. As stated earlier, the face detection part (1) will be implemented on the FPGA platform of the camera while the rest of the module will be implemented on the client PC.

The face detection and face recognition processes usually require a lot of computing power and could consume a lot of time when running on a standard PC. In hardware implementation however, processes can be broken apart and run in parallel so that less time will be taken to execute the processes. FPGA's provide a flexible and suitable reconfigurable platform for applying suitable architecture of the processor. If sufficient parallelism is applied, it is possible for the processes to run in real-time.

6. NICTA SMART CAMERA PROTOTYPE

Our smart camera platform was designed based on the principles outlined in section 5. A board with an XC3S5000 FPGA and a DDR SDRAM slot is used as the main processing board for the camera. We used a 1GB DDR SDRAM as the main frame buffer of the camera. As for the image sensor, 5MP resolution (2592x1944) CMOS type sensor was chosen. The camera sensor could operate up to 14fps at full resolution. As for the camera communication interface to host PC, a FireWire 800 was selected. A board consists of Texas Instrument's 1394b Link Layer and Physical Layer controller chips and 3 FireWire 800 ports is used. All the mentioned boards were interfaced together and powered using a custom designed PCB board. Figure 4 shows our smart camera prototype while Table 4 summarizes the basic descriptions of our prototype.

The algorithm for the camera is programmed in modules to ensure reusability. For example, a basic camera would require an image sensor interface module, a memory interface module, a FireWire Link Layer interface module, a colour interpolation module, a grey-scale module as well as a down-sampling module. It is expected that as the research advances, more modules would be produced. This technique would enable us to switch between software based and hardware logic based module implementation.

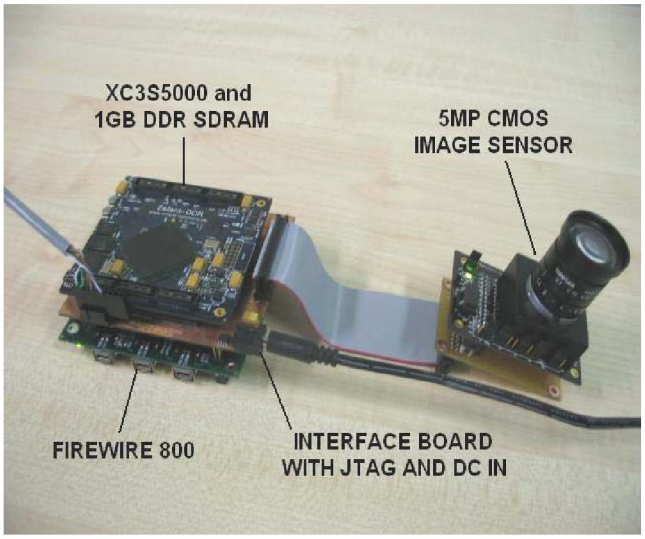

Figure 4*: Smart Camera Prototype.*

Table 4: *Specification of Smart Camera*

Parameter	Value
Sensor Type	CMOS
Resolution	2592 x 1944
Processing	Spartan-3 FPGA
Comm.	Firewire800
Dimension	90 x 90 x 150 mm³

7. CONCLUSION AND FUTURE WORK

Smart Cameras are slowly being introduced in the emerging surveillance systems. They usually perform a set of low-level image processing operations on the input frames at the sensor end. This paper reported on our prototype development of a smart camera for automated face recognition using high resolution (5 MP) sensors. In the proposed technique, the smart camera extracts all the faces from the full-resolution frame and only sends the pixel information from these face areas to the main processing unit. Face recognition software that runs on the main processing unit will then perform the required pattern recognition algorithm.

The main challenge in this project is to build a stand-alone and low power smart camera system that integrates real-time face detection for crowd surveillance. The future work would involve implementing a robust face detection algorithm on the camera and to explore the best HW/SW co-design configuration for it. As discussed by Giovanni et al. [15], although FPGA-based processing platform offers an excellent HW/SW solution (due to its flexibility, reusability and ability to integrate microprocessor architecture on itself), HW/SW co-design would be an important issue to be dealt with in order to achieve the optimal system. The major task in HW/SW co-design is indentifying the balance between the hardware logic and software implementation i.e. finding the optimal partitioning. Poor partitioning will result to excessive usage of resources and poor speed performance. Another problem in HW/SW co-design is developing an efficient communication interface between the hardware and software solution.

8. ACKNOWLEDGEMENTS

This project is supported by a grant from the Australian Government Department of the Prime Minister and Cabinet. National ICT Australia is funded by the Australian Government's Backing Australia's Ability initiative and the Queensland Government, in part through the Australian Research Council.

9. REFERENCES

[1] W. Wolf, B. Ozer, and T. Lv, "Smart cameras as embedded systems," *Computer,* vol. 35, pp. 48-53, 2002.

[2] M. Bramberger, A. Doblander, A. Maier, B. Rinner, and H. Schwabach, "Distributed embedded smart cameras for surveillance applications," *Computer,* vol. 39, pp. 68-75, 2006.

[3] R. Kleihorst, M. Reuvers, B. Krose, and H. Broers, "A smart camera for face recognition," in *International Conference on Image Processing 2004 (ICIP '04)*, pp. 2849-2852, 2004.

[4] S. Fleck and W. Strasser, "Adaptive Probabilistic Tracking Embedded in a Smart Camera," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 134-134, 2005.

[5] N. Matsushita, D. Hihara, T. Ushiro, S. Yoshimura, J. Rekimoto, and Y. Yamamoto, "ID CAM: a smart camera for scene capturing and ID recognition," in *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 227-236, 2003.

[6] P. Chalimbaud and F. Berry, "Design of an imaging system based on FPGA technology and CMOS imager," in *IEEE International Conference on Field-Programmable Technology*, pp. 407-411, 2004.

[7] S. Yu, P. Raniga, and I. Mohamed, "A Smart Camera for Multimodal Human Computer Interaction," in *IEEE Tenth International Symposium on Consumer Electronics 2006 (ISCE '06)*, pp. 1-6, 2006.

[8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 511-518, 2001.

[9] D. Litwiller, "CCD vs. CMOS: Facts and Fiction," in Photonics Spectra, 2001.

[10] OmniVision. http://www.ovt.com/

[11] Micron Technology. http://www.micron.com/

[12] Kodak. http://www.kodak.com

[13] P. Garcia, K. Compton, M. Schulte, E. Blem, and W. Fu, "An Overview of Reconfigurable Hardware in Embedded Systems," *EURASIP Journal on Embedded Systems*, pp.1-19, 2006

[14] T. Shan, B. C. Lovell, S. Chen, and A. Bigdeli, "Reliable Face Recognition for Intelligent CCTV," in *2006 RNSA Security Technology Conference* Canberra, pp. 356-364, 2006.

[15] G. D. Micheli, and R. K. Gupta, "Hardware/Software Co-Design," *Proceedings of the IEEE*, pp. 349-365, 1997.