(IJRST) 2014, Vol. No. 4, Issue No. II, Apr-Jun **ISSN: 2249-0604**

AN AUTONOMOUS INTELLIGENT ROBOT VISION COMPUTATIONAL SENSORS IN VLSI

M.Chaitanya Suman [1], K.Prathyusha [2]

[1] *Assistant Professor, Department of ECE. Universal College of Engg.Tech, Guntur District, Andhra Pradesh, India.*
[2] *PG scholar, Department of ECM. K.L.University, Vaddeswaram, Guntur District, Andhra Pradesh, India.*

ABSTRACT

Traditional approaches for solving real-world problems using computer vision have depended heavily on CCD cameras and workstations. As the computation power of workstations doubles every 1.5 years, they are now better able to handle the large amount of data presented by the cameras; yet real-time solutions for physical interaction with the real-world continues to be very hard, and relegated to large and expensive systems. Our approach attempts to solve this problem by using computational sensors and small/inexpensive embedded processors. The computational sensors are custom designed to reduce the amount of data collected, to extract only relevant information and to present this information to the simple processor, microcontrollers (µCs) or DSP's, in a format which minimizes post-processing latency. Consequently, the post-processors are required to perform only high level computation on features rather than data. These systems are applied to problems such as target acquisition and tracking for image stabilization and autonomous data driven auto navigation for mobile robots. We present an example a system that uses a pair of computational sensors and a µC to solve a auto navigation problem. The computational sensors, however, have wide applications in many problems that require image preprocessing such as edge detection, motion detection, centroid localization and other spatiotemporal processing. This paper also presents a general-purpose computational sensor capable of extracting many visual information components at the focal plane.

INTRODUCTION

Consider a hostage situation in an urban environment. When the law enforcement individuals arrive on the scene, it would not be prudent to enter the building containing armed terrorists. Instead a group of semi-autonomous robots are placed at the doorway and they search the building to find the captives. The robots are given a variety of visual sensors. They range from passive CCD cameras, IR sensors to intelligent sensors. The CCD and IR cameras are used to simply image the scene and allow the law enforcement officers to inspect the environment. Hence, they are off-the-shelf systems that can be easily interfaced with transmission hardware for viewing on PDAs (personal digital assistants). The computational sensors, on the other hand, are used to preprocess visual images; the extracted visual information is used to guide the robots. The robots autonomously execute the officers' command using the information provided by the computational sensors and decisions made by local/remote processing hardware.

To ensure their survival, the robots must have a variety of specialty skills, such as stealth (quiet, low EM signature), hazard detection (obstacle avoidance and evasion), speech recognition and homing, and cooperative map building (landmark recognition and location), as shown in figure 1. Clearly these are very complex and computationally intensive behaviors and properties. The combination of computational sensors, imagers, local/remote processors and clever algorithms will be instrumental in realizing these behaviors.

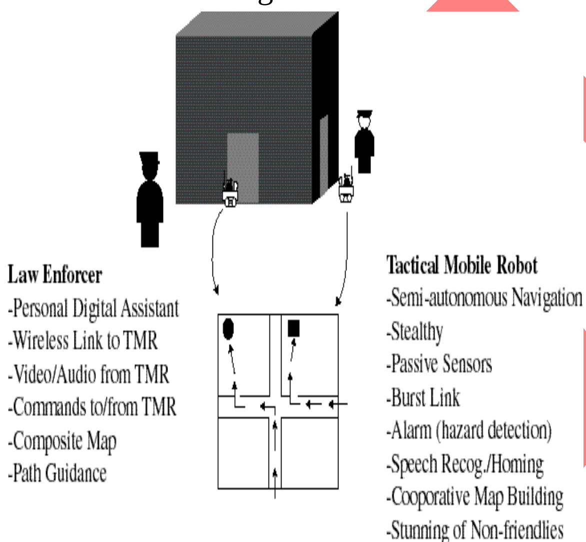

Figure 1: The Semi-Autonomous Urban Search/Locate Robot

Over the past few years, a few information extracting computational sensors have been developed [Koch, 1995]. The application of these sensors to real problems requiring visually guided interaction with the environment is still in its infancy [Kramer, 1998]. Consequently, the problems that have been attempted are small and relatively easy for the robotics community.

They are, however, the sentry problems that evaluate the potential of using computational sensors in difficult and complex robotics systems and tasks [Etienne, 1998]. So far, the results have been encouraging. We are now poised to expand the computational power of these smart sensors such that they can be used in real robot vision systems. Yet, as the speed and computational power of µP/DSP/µC continues to increase while their cost plummets, can custom designed smart sensors compete with the ubiquitous and programmable CCD/CPU robot vision systems? Fundamentally, having a computational sensor in the processing path is beneficial. This is because it relieves the µP/DSP/µC from menial tasks of data preprocessing for information/feature extraction. Consequently, the µP/DSP/µC can be applied to more difficult problems that capitalize on their general purpose computational strengths. Despite the obvious power, speed and efficiency advantages of application specific computational sensors, the man-time and cost associated with their re-design for each new application are prohibiting. Hence, I propose a general-purpose computation sensor (GPCS) that performs considerable and programmable computation at the focal plane. The user has the advantage of choosing which processed image format and which process parameters are required for the problem.

Furthermore, the same sensor can be used in a variety of robotics system, resulting in compatibility across systems and problems. The GPCS can subsequently replace the camera. Similar to other electronics commodities, such as CCDs, CPUs and DSPs, an industry can be constructed around providing these smart sensors with various specialties. Clearly, their interface and programming architecture must be standardized for general use. As can be expected, the GPCS has a considerable design penalty, in terms of man-time and material cost. Consequently, these following questions beg answers. Should sensor designers focus on application specific or general-purpose computational sensors?

- Is the prototyping cost of ASCS out-weighed by their performance benefit?
- Does flexibility of the GPCS out-weighs its redundancies and inefficiencies?
- Will the development of advanced computation sensors advance the state-of-the-art of robotics?

These questions are addressed below.

The examples of computational sensors for robot vision provided below address these two extremes.
The first design uses an application specific approach to realize a sensor with centroid localization properties [Etienne, 1998]. This chip is optimized for area compactness, low power and high speed. To demonstrate the usefulness of this computational sensor, it is combined with a simple µC to realize a data driven autonomously navigating mobile robot. The complete system is low-cost, compact, lightweight and operates at speeds up to 2 m/s. Power consumption is dominated by locomotion and not information processing. On the other hand, to overcome the shortcomings of the application specific approach, a general-purpose computational sensor (GPCS) for visual information processing is proposed. Performing ten complex information (feature) extraction tasks at the focal plane, this sensor is clearly a complicated IC. Consequently, it requires a great deal of design time. Yet, it offers a unique opportunity to standardize the front-end of all robot vision systems. Arguments are offered which highlight the benefits of using the GPSC with a µP/DSP/µC processing core to implement robot vision systems.

MOTION CENTROID LOCALIZATION COMPUTATIONAL SENSOR

A. System Overview

Motion centroid computation is used to isolate the location of moving targets on a 2D focal plane array. Using the centroid computation, a chip which realizes a neuromorphic visual target acquisition system based on the saccadic generation mechanism of primates can be implemented. The biological saccade generation process is mediated by the superior colliculus, which contains a map of the visual field [Sparks, 1990]. Horiuchi has built an analog system that replicates most of the neural circuits (including the motor system) believed to form the saccadic system [Horiuchi, 1996]. Staying true to the anatomy forced his implementation to be a complex multi-chip system with many control parameters. On the other hand, our approach focuses on realizing a compact single chip solution by only mimicking the behavior of the saccadic system, but not its structure. The benefit of our approach is its compactness, low-power consumption and large response dynamic range.

B. Hardware Implementation

Our approach uses a combination of analog and digital circuits to implement the functions of the retina and superior colliculus at the focal plane. The retina portion of this chip uses photodiodes, logarithmic compression, edge detection and zero crossing circuits. These circuits mimic the first three layers of cells in the retina with mixed sub-threshold and strong inversion circuits. The edge detection circuit is realized with an approximation of the Laplacian operator implemented using the difference between a smooth (with a resistive grid) and original versions of the image [Mead, 1989]. The high gain of the difference circuit creates a binary image of approximate zero-crossings. After this point, the computation is performed using mixed analog/digital circuits. The zero-crossings are fed to ON-set detectors (positive temporal derivatives) which signal the location of moving or flashing targets. These circuits model the behavior of some of the amacrine and ganglion cells of the primate retina [Barlow, 1982]. These first layers of processing constitute all the "direct" mimicry of the biological models. Figure 2 shows the schematic of these early processing layers.

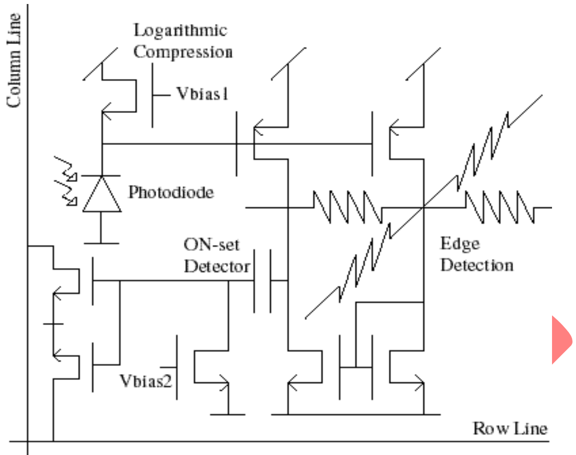

Figure 2: Schematic of the model of the retina.

The ON-set detectors provide inputs to the model of the superior colliculus circuits [Sparks, 1990]. The ON-set detectors allow us to segment moving targets against textured backgrounds. This is an improvement on earlier centroid and saccade chips which used pixel intensity [DeWeerth, 1992]. The essence of the superior colliculus map is to locate the target to be foveated. In our case, the target chosen to be foveated will be moving. Here motion is define simply as the change in contrast over time. Motion, in this sense, is the earliest measurable attribute of the target that can trigger a saccade without requiring any high-level decision making. Subsequently, the coordinates of the location of motion must be extracted and provided to the motor drivers. The circuits for locating the target are implemented entirely with mixed signal circuits. The ON-set detector is triggered when an edge of the target appears at a pixel. At this time, the pixel broadcasts its location to the edge of the array by activating a row and column line. This row (column) signal sets a latch at the right (top) of the array.

The latches asynchronously activate switches and the centroid of the activated positions is provided. The latches remain set until they are cleared by an external control signal. This control signal provides a time-window over which the centroid output is integrated. This has the effect of reducing noise by combining the outputs of pixels which are activated at different instances even if they are triggered by the same motion (an artifact of small fill factor focal plane image processing). Furthermore, the latches can be masked from the pixels' output with a second control signal. This signal is used to de-activate the centroid circuit during a saccade (saccadic suppression). Figure 3 shows a portion of the schematic of the superior colliculus.

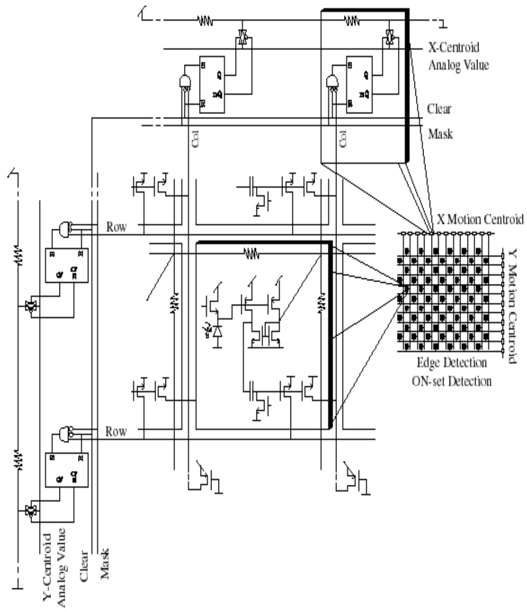

Figure 3: Schematic of the model of the superior colliculus.

Results

In contrast to previous work, this chip provides the 2D coordinates of the centroid of a moving target. Figure 4 shows the oscilloscope trace of the coordinates as a target moves back and forth (at a fixed y- displacement), in and out of the chip's field of view. The y-coordinate does not changes while the x-coordinate increases and decreases as the target .

Technology	1.2um ORBIT
Chip Size	4 mm ^{2}
Array Size	17 x 19
Pixel Size	70 x 65 um ^{2}
Fill Factor	11%
Intensity	0.1u - 100m W/cm ^{2}
Min. Contrast	10%
Response Time	2 - 10 ^{6} Hz (@ 1 m W/cm ^{2})
Power (chip)	5 mW (@ 1 m W/cm ^{2} Vdd = 6V)

Table I: Performance characteristics of the centroid localization computational sensor.

moves to the left and right, respectively. The chip has been used to track targets in 2D by making micro saccades. In this case, the chip chases the target as it attempts to escape from the center. The eye movement is performed by converting the analog coordinates into PWM signals that are used to drive stepper motors. The system performance is limited by the contrast sensitivity of the edge detection circuit, and the frequency response of the edge (high frequency cut-off) and ON-set (low frequency cut-off) detectors. With the appropriate optics, it can track walking or running persons under indoor or outdoor lighting conditions at close or far distances. Table I summaries the characteristics of the chip.

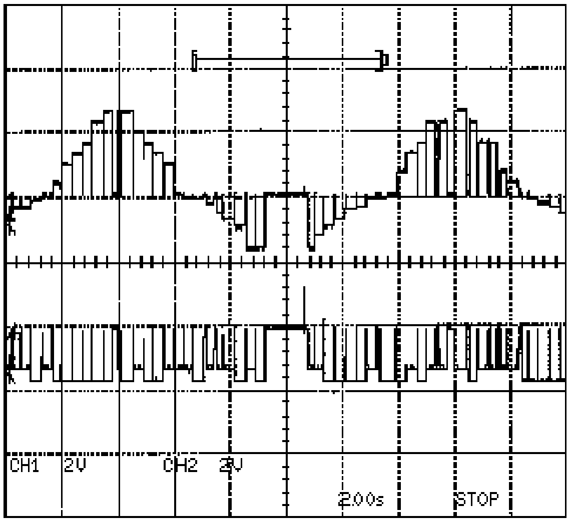

Figure 4: Oscilloscope trace of 2D centroid for a moving target.

AUTONOMOUS NAVIGATION USING THE COMPUTATIONAL SENSOR

A. System Overview

The simplest form of data driven auto-navigation is the line-following task. In this task, the robot must maintain a certain relationship with some visual cues that guide its motion. In the case of the line-follower, the visual system provides data regarding the state of the line relative to the vehicle, which results in controlling steering and/or speed. If obstacle avoidance is also required, auto-navigation is considerably more difficult. Our system handles line following and obstacle avoidance by using two sensors which provide information to a microcontroller (µC). The µC steers, accelerates or decelerates the vehicle. The sensors, which use the centroid location system outlined above, provide information on the position of the line and obstacles to the µC. The µC provides PWM signals to the servos for controlling the vehicle. The algorithm implemented in the µC places the two sensors in competition with each other to force the line into a blind zone between the sensors. Simultaneously, if an object enters the visual field from outside, it is treated as an obstacle and the µC turns the car away from the object. Obstacle avoidance is given higher priority than line-following to avoid collisions. The µC also keeps track of the direction of avoidance such that the

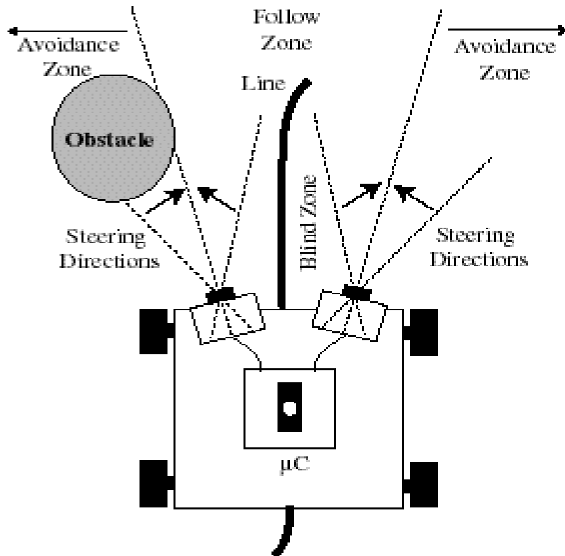

Figure 5: Block diagram of the autonomous line-follower system.

vehicle can be re-oriented towards the line after the obstacle is pushed out of the field of view. Lastly, for line following, the position, attitude and velocity of drift, determined from the temporal derivative of the centroid, are also used. The speed control strategy is to keep the line in the blind zone, while slowing down at corners, speeding up on straight always and avoiding obstacles. The angle which the line or obstacle form with the x-axis also affects the speed. The value of the x-centroid relative to the y-centroid provides rudimentary estimate of the attitude of the line or obstacle to the vehicle. For example, angles less (greater) than +/- 45 degrees tend to have small (large) x-coordinates and large (small) y-coordinates and require deceleration (acceleration). Figure 5 shows the organization of the sensors on the vehicle and control spatial zones. Figure 6 shows the vehicle and samples of the line and obstacles.

B. Hardware Implementation

The coordinates from the centroid localization circuits are presented to the µC for analysis. The µC used is the Microchip PIC16C74. This chip is chosen because of its five A/D inputs and three PWM outputs. The analog coordinates are presented directly to the A/D inputs. Two of the PWM outputs are connected to the steering and speed control servos. The PIC16C74 runs at 20 MHz and has 35 instructions, 4K by 8-b ROM and 80 by 20-b RAM. The PIC program determines the control action based on the signal provided by the sensors. The vehicle used is a four-wheel drive radio controlled model car (the radio receiver is disconnected) with Digital Proportional Steering (DPS).

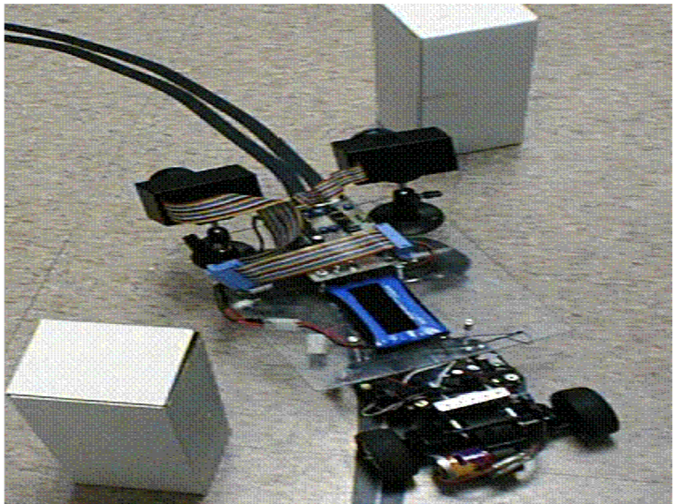

Figure 6: A picture of the vehicle.

Results

The vehicle was tested on a track composed of black tape on a gray linoleum floor with white obstacles. The track formed a closed loop with two sharp turns and some smooth S-curves. The sensors were equipped with 8mm variable iris lens, which limited their field of view to about 9o. Despite the narrow field of view, the car was able to navigate the track at an average speed of 1 m/s without making any errors. On less curvy parts of the track, it accelerated to about 2 m/s and slowed down at the corners. When the speed of the vehicle is scaled up, the errors made are mainly due to over steering and spinning-out in corners.

CONCLUSIONS

The need for and the benefit of having computational sensors in the processing pathway of robot vision systems have been argued. The question on the type of computational sensors to use, i.e. application specific or general-purpose, is somewhat more difficult to answer. To address this issue, an application specific computational sensor for motion centroid localization, modeled after the biological retina and superior colliculus, is presented. Furthermore, two of these sensors, together with a simple µC, are used to realize the solution to a toy line-following with obstacle avoidance auto navigation problem. Despite the simplicity of this problem, this experiment shows that we are currently able to create application specific sensors which can be used to solve real problems in real-time. Generally, application specific sensors are efficient in function, but not in design and re-use. So finally, will this sensor improve the state-of-the-art of robotics? I argue that it must, provided that the *information* provided by the sensor is useful very close to its raw form. If a large amount of post processing is required, then the benefits of information extraction at the focal plane are lost. This implies that the quality of information provided by the sensor must be robotics grade. What is "robotics? grade quality?" This specification must be provided by the user. Furthermore, it is application and algorithm specific. So far, the assumption under which we are operating is that 8-10 bits SNR is sufficient. This is realizable with digitally controlled analog processing using VLSI circuits. We are making progress towards a general-purpose computation sensor that will benefit many robotics attacking a variety of problems.

REFERENCES

[1]Andreou A., Boahen K., Pouliquen P., Pavasovic A., Jenkins R., and Strohbehn K., "Current-Mode Subthreshold MOS Circuit for Analog Neural Systems," *IEEE Transactions on Neural Networks*, Vol. 2, No. 2, pp. 205-213, 1991.

[2]Barlow H., *the Senses: Physiology of the Retina*, Cambridge University Press, Cambridge, England, 1982. DeWeerth, S. P., "Analog VLSI Circuits for Stimulus Localization and Centroid Computation," *Int'l J. Computer Vision,* Vol. 8, No. 2, pp. 191- 202, 1992.

[3]Etienne-Cummings R., M. Abdel Ghani and V. Gruev, "VLSI Implementation of Centroid Localization for Autonomous Navigation," excepted for publication in *Advances in Neural Information Processing Systems 11*, 1998.

[4]Etienne-Cummings R., J Van der Spiegel and P. Mueller, "Neuromorphic and Digital Hybrid Systems," *Neuromorphic Systems: Engineering Silicon from Neurobiology*, L. Smith and A. Hamilton (Eds.), World Scientific, 1998.

[5]Etienne-Cummings, R. and Donghui Cai, "A General Purpose Image Processing Chip: Orientation Detection," *Advances in Neural Information Processing Systems 10*, 1998b.

[6]Etienne-Cummings R., J. Van der Spiegel and P. Mueller, "VLSI Implementation of Cortical Motion Detection Using an Analog Neural Computer," *Advances in Neural Information Processing Systems 9,* M. Mozer, M. Jordan and T. Petsche (Eds.), MIT Press, pp. 685-691, 1997.

[7]Etienne-Cummings R., J. Van der Spiegel and P. Mueller, Mao-zhu Zhang, "A Foveated Tracking Chip," *ISSCC97 Digest of Technical Papers*, Vol. 40, pp. 38-39, 1997.

[8]Horiuchi T., T. Morris, C. Koch and S. P. DeWeerth, "Analog VLSI Circuits for Attention-Based Visual Tracking," *Advances in Neural Information Processing Systems*, Vol. 9, Denver, CO, 1996.

[9] Horn B., *Robot Vision*, MIT Press, Cambridge MA, 1986. Koch C. and H. Li (Eds.), *Vision Chips: Implementing Vision Algorithms with Analog VLSI Circuits*, IEEE Computer Press, 1995.

[10]Kramer, J. and G. Indiveri, "Neuromorphic Vision Sensors and Preprocessors in Systems Application," *Proc. Adv. Focal Plane Arrays Electronics Cameras*, May, 1998.

Mansfield, P., "Machine Vision Tackles Star Tracking," *Laser Focus World*, Vol. 30, No. 26, pp. S21-S24, 1996.