AN ADAPTIVE CENTER OF MASS DETECTION SYSTEM EMPLOYING A 2-D DYNAMIC ELEMENT MATCHING ALGORITHM FOR OBJECT TRACKING

Alexander Fish, Dmitry Akselrod and Orly Yadid-Pecht

The VLSI Systems Center, Ben-Gurion University, Beer-Sheva 84105, Israel

ABSTRACT

An adaptive high precision image object centroid - center of mass (COM) detection system employing a 2-D dynamic element matching (DEM) algorithm for object tracking is presented. This proposed system outputs a high resolution COM location of the most salient object in the field of view for tracking and navigation purposes and is suitable for real time applications. Pre-computation object detection is performed by a low resolution mixed signal winner-take-all (WTA) circuit. In this work, the COM computation algorithm parameters are adaptively changed according to the object size, enabling lowering object scanning frequency, but still providing a high precision in the object COM location. Simulation results prove the concept and demonstrate the high precision COM detection result. In addition, a possible hardware implementation is described.

1. INTRODUCTION

An object centroid - Center of Mass (COM) detection is a very important operation in image processing. Representing the position of an object, centroiding is an operation frequently done for tracking and navigation purposes. The tracking of an object position is very important for many scientific, commercial, consumer and military applications. Many approaches to centroid location and tracking have been proposed in the literature [1-8]. Traditionally, the COM location is performed by a digital processor [1]. In this case the COM calculation requires fast access to memory and much on chip "computational power" in order to complete the result calculation in time.

A CMOS sensor array with focal-plane centroid detection system-on-a-chip, i.e. a "smart sensor", can reduce hardware requirements and computation complexity. Previous works have addressed focal-plane COM detection analog circuits in CMOS sensors [2-8]. Shibata et al. [2] introduced a neuron-MOS circuit in which the gate capacitors (implemented as a second polysilicon layer) are scaled according to their relative input position. However, capacitor scaling for a large number of inputs from an imager would not be practical. Deweerth [3] introduced a 1D photoreceptor array that computed the centroid in an analog circuit. He also proposed a 2D extension in which the receptors are alternated spatially so that the currents from adjacent receptors are added to opposing axes [3]. Tartagni and Perona [4] proposed a computational circuit working in the subthreshold regime for implementing the COM function in 1-D. A 9x9 window COM calculation system was proposed in 1999 [5]. It consisted of a 2-D imager array, a switching network, inner-product computation circuits, and an analog divider. On-chip 2-D centroid computation was carried out by first computing the relevant inner-products (weighted sums) for a given row.

In the system described in this paper a pre-computation task, namely object detection, is performed by a low-resolution winner-take-all (WTA) circuit, which was first introduced by Lazzaro [9]. A commonly used WTA selects a single winner out of multiple inputs and is one of the most important building blocks for hardware realizations of neural networks [9]. Recently, we have showed, that the DEM algorithm, commonly used in ADC and DAC circuits, allows reducing the required WTA circuit resolution to 5-6 bits, while still achieving a high precision of the COM output result [8].

This work presents an adaptive high precision object COM detection system employing a 2-D DEM algorithm for real time object tracking. The COM computation algorithm parameters are adaptively changed according to the object size, providing a high precision in the COM location while retaining the same computational complexity for different object sizes.

Section 2 presents the detailed description of the proposed COM detection system with its adaptive operation and outlines the required hardware for system implementation. The performance of the system via simulation results is described in Section 3. Section 4 concludes the paper.

2. THE PROPOSED COM SYSTEM DESCRIPTION

We are interested in tracking the COM location of the brightest object, which usually consists of a large number of bright pixels and is usually found in the captured image together with a noisy background. The COM location can be calculated in the following way: first, the object brightest pixels should be chosen by a WTA circuit – this operation is analogous to the segmentation operation in image processing. Next, we calculate the COM coordinates based on the locations of the bright pixels found in the previous step. If a regular high resolution WTA circuit is used for object detection, a single pixel with the highest signal intensity is selected, thus preventing other object pixels to participate in the following COM calculation, causing possible error in the COM location. The problem intensifies in the case of a large object of interest, when the brightest pixel can be located very far from the pixel whose coordinates represent the actual COM location. A low resolution WTA solves this problem. If a low resolution WTA is used, more than one pixel that can be identified as a potential winner is selected. This occurs when the difference between the potential winner values is less then the resolution of the WTA circuit. Thus, the object of interest is translated eventually to a "perceived object" that consists of a number of potential winners [10]. Therefore more object pixels will participate in the following COM calculation and a better result can be achieved. Fig. 1 (a) and (b) show examples of the object of interest and its "perceived object" respectively.

At this stage only one pixel of the "perceived object" should be chosen by the system and identified as a global frame winner. The DEM algorithm, applied in 2 dimensions, accomplishes one pixel selection from the pixels within the "perceived object", that is the global frame winner. The final COM location is found using the global frame winners from a number of sequentially scanned frames, as described in sub-section 2.1.

0-7803-7762-1/03/$17.00 ©2003 IEEE

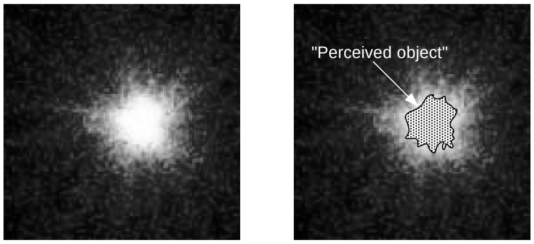

Fig.1. (a) A real image with an object of interest, (b) The "Perceived object" consisting of potential winners with very close values (the value differences are less then the WTA resolution).

2.1. SYSTEM ARCHITECTURE

Fig. 2 shows the block diagram of the proposed 2-D adaptive COM system employing the DEM approach. It should be noted that the WTA resolution here is determined by the D/A resolution. The function of the control unit is to generate all clocks and control signals in the system.

In this system the object of interest is scanned row-by-row using the Row Decoder. The resulting output is a row, consisting of voltage representations of pixels brightness (a high voltage represents a high brightness level). First, the WTA block chooses all potential winners of the scanned row. This is performed in the following way: the ramp module (a D/A controlled by a counter) decreases the reference voltage applied to the comparators from Vdd to 0V by steps corresponding to the DAC resolution. When the ramp reaches the maximum value within the scanned row, one or more comparators output '1'. This causes a '1' at the "OR" output, thus the ramp stops and the value detected is latched. At this point, in the case of a large "perceived object", more than one comparator detect a maximum value pixel in the row, i.e. more than one potential winner exists. In this case there is uncertainty in choosing the row winner out of several possible candidates. The "1-D Row DEM" block performs this task in a desired way.

DEM is a method of dynamic choice of n objects out of m possible, where n <= m. In our case n=1. Since 1973 the subject of DEM has been extensively researched in the literature [11-12] and numerous algorithms have been described. The Data Weighted Averaging (DWA) based technique [12] is used in our system. Recently, we have shown the detailed description of the DEM algorithm application in such a system [10]. A brief description is presented herein.

In order to detect the winner of the current row, we introduce a row winner pointer, which points to the column of the chosen row winner of the last scanned row. Thus the winner of the current row is the first potential winner adjacent to the row winner pointer in the direction of pointer increase. For example, if the row winner points to column no. 5, and we have 4 potential winners in the current row, with column addresses 3, 4, 16 and 18 respectively, the winner of the current row will be the pixel with column address 16. The reset value of the pointer is the first column. Note that the pointer value is not reset between subsequent frames, thus the previous frame influences the results of the next frame. After tracking the next winner, the winner pointer receives its column address, moving only in one permanent direction upon which was decided on up front.

At this stage the system is expanded to a 2-dimensional one, i.e. we are interested in choosing a single global winner for the whole frame. This is done by the "1-D column DEM" block. Assuming that after the "1-D Row DEM" block finds the row winner the control unit stores the winner digital value and the column address in the corresponding place in the memory. This way, at a certain stage (after scanning the whole frame) all row winners are stored in memory. The same method, as described for the row winner detection, is used for detecting the global winner. In this case a column winner pointer has an identical function to that of the row winner pointer and is initialized to the first row address. It should be noted that the potential winners of the whole frame are the pixels with the same maximum frame values. Thus we obtain the coordinates of the "frame winner". In order to find the object COM, several "frame winners" are obtained by sequentially scanning the frame. The final result is calculated by estimation methods using the obtained "frame winners". The simplest estimation method is averaging all "frame winners" and this is the one used. It must be noted that because of the possibility of a dynamic object, an image must be scanned *OSR*F* times per second, where *F* is the video frequency and *OSR* is the over sampling ratio. To find the COM the object is scanned *OSR* times.

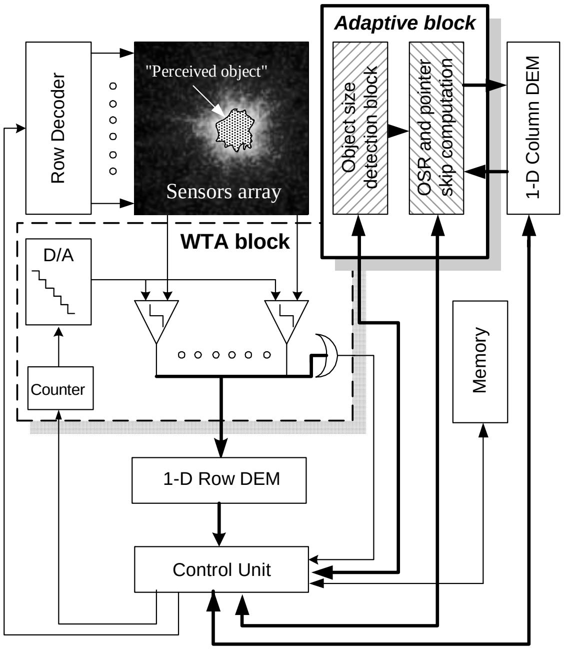

Fig.2 Block diagram of the proposed 2-D adaptive COM system employing the DEM approach.

One of the important advantages of the proposed system is its simplicity of implementation due to small hardware requirements and simplicity in re-design for various sensor array sizes. We have previously assumed that a digital value for every row winner is stored in memory. This way the final frame winner is chosen from these row winners by the "1-D column DEM" block. This possible implementation was presented in order to explain the concept of the proposed system. Actually, only one row and one column pointer values, a temporary winner for the current frame, the final winner of the previous frame and temporary and final COM locations are needed to be stored. The winner of every scanned row in the current frame is compared to the temporary frame winner. If the row winner is less than the temporary frame winner, the temporary winner does not change. If the row winner value is higher than the temporary frame winner value, the temporary winner is replaced by the given row winner. If they are equal, the

"1-D column DEM" block takes a decision depending on the column pointer value according to the DEM algorithm applied, as previously described. At the end of a given frame, the temporary frame winner replaces the final winner for the previous frame and participates in the computation of the COM location. The temporary result of the COM calculation is stored and only after a required number of frames (OSR), this temporary COM result replaces the final COM location and is output. This way the required memory size does not depend on the sensor array size. This is a very important advantage for large sensor arrays.

2.2. ADAPTIVITY TO DIFFERENT OBJECT SIZES

Recently we have shown that the system precision dependence on OSR is as shown in Fig. 3 [8]. Matlab simulations were carried out on a theoretical circular "perceived object" with a certain size (17 pixels), as accepted after the WTA operation. The COM location error has a periodic behavior with a constant period of OSR (equals to the certain size of 17 pixels), that is wrapped in an exponential envelope. The period is determined according to the "perceived object" size (number of rows), since an integer number of whole object sampling reduces the error. This error gets lower with increasing OSR; that is the reason for the wrapping "envelope". For OSR < 17 there is a strong improvement in precision with OSR increase. Following this observation, it seems best to work in the area of the first local minimum, where OSR equals to object size.

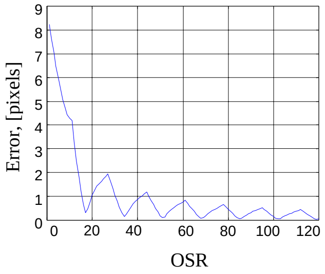

Fig. 3. COM location error vs. OSR for the theoretical circular "perceived object" with a certain diameter of 17 pixels.

Simulations of the location error as a function of OSR were carried out for various circle diameter sizes. The simulation results showed, as expected, that the required OSR for optimal system operation increases linearly with object size. Although the system demonstrated high precision, the high OSR required for large objects made it impractical in the case of large sensor arrays for real time applications. In these applications there is a maximal allowed OSR that is determined by the array size, the required number of frames per second and the chosen WTA resolution. Thus, we have proposed an improved algorithm. Using this algorithm, the required OSR for a given object size is reduced by increasing the DEM pointers skips, both in rows and columns,.i.e. if a pointer skip equals to 2 the algorithm will skip to the winner adjacent to the row winner pointer and will choose the next one. This way, the whole object is scanned more quickly and the required OSR is reduced. Using this algorithm, every object has its optimal OSR and column and row pointers skip, depending on the object size.

In this currently modified system these parameters are adaptively changed depending on the object size. The "Adaptive block" (see Fig. 2) is responsible for this function. The block consists of an "Object size detection block" and of an "OSR and pointer skip computation" blocks. The "object size detection" block detects the number of the rows in the "perceived object". The object size is determined during a test frame scanning in the following way: the winner of the scanned row is compared to the temporary winner of the given frame (as defined in sub-section 2.1). If they are equal, the current row winner is identified as an object row and a special counter, that counts the number of object rows, is increased by 1. If the current row winner value is less than the temporary frame winner value, the counter result is not changed. If the row winner value is higher than the temporary frame winner value, the temporary winner is replaced by the given row winner and the counter is initialized to 1. This way the counter outputs the size of the processed object at the end of this test frame. It should be noted that there is a necessity to accomplish this operation each time the object of interest is changed. The object size is input to the "OSR and pointer skip computation" block that performs the required OSR and row and column pointers skips for a given object size. The OSR and pointers skips computations can be accomplished via simple algebraic calculations according to the following equations:

$$P_{skip} = \left[{object\_size} / {\max OSR} \right] + 1 \qquad (1)$$

$$OSR = \left[{object\_size + 1} / {pointer\_skip} \right] \qquad (2)$$

where max OSR is the maximum allowed OSR, pointer_skip and OSR are the calculated pointer skip and OSR respectively applied to the COM computation algorithm and the sign [] indicates the integer part of the division result. Equations (1) and (2) are the result of an integer number for the whole object sampling requirement fulfillment [12]. Using the OSR and the pointer skip found from (1) and (2), to obtain COM location the object is scanned a number of times that corresponds to the first local minimum as described in Fig. 3.

For example, for a given object of 23 pixels size we need OSR = 23 in order to scan the whole object with pointer skip = 1 (this corresponds to the first local minimum in Fig. 3). If the maximum allowed OSR is 7, the object scanning with pointer skip=1 will cause a large error in the COM result, because only 7/23 of the object is scanned (it corresponds to OSR that is less than the first local minimum in Fig. 3). According to (1) we need a pointer skip of [23/7]+1=3+1=4. Using this pointer skip the whole object will be scanned faster. The required OSR in order to scan the whole object one time (and thus to work in the first local minimum) is determined by equation (2). Thus, OSR = [(23+1)/4]=6, i.e. using an OSR of 6 and a pointer skip of 4 will result in a small error in the COM location.

3. SIMULATION RESULTS

A set of MATLAB simulations on a number of real images has been carried out in order to examine the proposed algorithm for COM calculation. The results for two are presented below. The proposed algorithm was applied to the objects shown in Fig. 4, assuming a real time application system with a maximum allowed OSR of 6. The "percieved objects", derived from the images in Fig. 4 via 5 bit WTA resolution, have sizes of 17 and 14 rows respectively. According to (1) and (2) the pointer skip for the object from Fig. 4(a) is 3 and the required OSR is 6. For the object

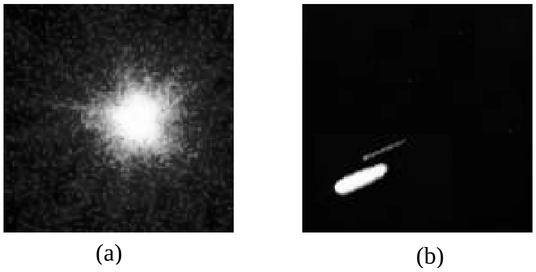

Fig. 4. Images with real objects, (a) A circle-like object (occupying 482 pixels), (b) An ellipse-like object (occupying 160 pixels).

from Fig. 4(b) the results are 3 and 5 respectively. COM computation precision, found from simulations, is 0.49 pixels for the object of Fig. 4(a) and 0.7 pixels for the object of Fig. 4(b). Fig. 5 shows this error in the COM location vs. the OSR with pointer skip = 3 for the object of Fig. 4(a). This simualtion shows that an optimal OSR, calculated by equation (2) is, as desired, located at the first local minimum, causing a small error in the COM location.

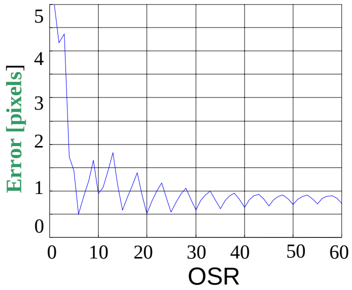

Fig. 5. COM location error vs. OSR for the object from Fig. 4(b) with the applied adaptive algorithm parameters (OSR=6, pointer skip=3).

4. ALGORITHM EFFICIENCY AND LIMITATIONS

In this subsection the cases where the proposed algorithm is more efficient are shown and the algorithm limitations are presented. First, the algorithm is most efficient in cases of large objects of interest with a noisy background, where a large group of object pixels have a similar brightness level and thus are selected as belonging to the perceived object. In these cases the proposed algorithm gives a high precision with low hardware overhead relative to other COM detection techniques. In cases of small objects or objects with a large object intensity distribution the algorithm is less efficient. Second, at this stage the proposed algorithm is not suitable for multiple objects detection. An additional limitation is that the maximum number of the possible "scan paths" (number of sequentially scanned frames with different row winners) is determined by the smallest number of winners in any of the rows in the frame. Choosing OSR, which is greater than the number of "scan paths", is not efficient. Thus the number of "scan paths" should be enlarged. This problem can be solved in a simple way by eliminating the influence of rows with a small number of row winners (< OSR) on the row DEM pointer and introducing an additional row DEM pointer in order to choose row winners in these rows.

5. CONCLUSIONS

An approach for implementing an adaptive high precision image object centroid - COM detection system employing a 2-D DEM algorithm for object tracking purposes was proposed. This proposed system outputs a high resolution COM location of the most salient object in the field of view for tracking and navigation purposes and is suitable for real time applications. The system detects the object size and adaptively calculates the required OSR and pointer skip parameters. The adaptively calculated OSR is less then the maximum allowed OSR enabling the system to be suitable for real time applications even for large objects detection.

A set of MATLAB simulations on a number of real images has been carried out to examine the adaptive algorithm accuracy. The results show that the proposed system enables achieving a high precision COM output for different object sizes. Future plans include full system implementation in hardware.

6. REFERENCES

[1] K. Cameron, J. Canaris, S. Whitaker, "A video image processor for real-time centroid calculation", Proceedings of Fifth Annual IEEE International ASIC Conference and Exhibit, pp. 201 -204, 1992.

[2] T. Shibata, T. Nakai, N. Mei Yu, Y. Yamashita, M. Konda, T. Ohmi, "Advances in Neuron-MOS Applications", ISSCC Proc., pp. 304-305, 1996.

[3] S. P. DeWeerth, "Analog VLSI Circuits for Sensorimotor Feedback", Ph.D thesis, Caltech, Pasadena, CA, 1991.

[4] M. Tartagni and P. Perona, "Computing centroids in current-mode technique", Electronic Letters, Vol. 29, No. 21, pp. 1811-1813, Oct 1993.

[5] C. Sun, G. Yang, C. Wrigley, O. Yadid-Pecht and B. Pain, "A smart CMOS imager with on-chip high-speed windowed centroiding capability", Proc. IEEE workshop on CCDs and advanced image sensors, Karuizawa, Japan, June 10-12, 1999.

[6] R.C. Meitzler, K. Strohbehn and A.G. Andreou, "A silicon retina for 2-D position and motion computation", Proc. ISCAS'95, New York, USA, 1995.

[7] M.Clapp and R..Etienne-Cummings, " A dual pixel-type imager for imaging and motion centroid localozation", Proc. ISCAS'01, Sydney, Australia, May 2001.

[8] A. Fish, D. Akselrod and O. Yadid-Pecht, "A "Smart Sensor" for High Precision Image Centroid Computation via a Low Resolution Winner-take-all Circuit in Conjunction with a Dynamic Element Matching Algorithm", submitted to the Analog Integrated Circuits and Signal Processing journal special issue on "Smart Sensors".

[9] J. Lazzaro., S. Ryckebusch, M. A. Mahowald, and C.A. Mead, "Winner-tale-all networks of O(n) complexity", ed. D.S.Touretzky, Morgan Kaumann, San Mateo, CA, 1:703-711, 1989.

[10] D. Akselrod, A. Fish and O. Yadid-Pecht, "A Mixed Signal Enhanced WTA Tracking System via 2-D Dynamic Element Matching", Proc. ISCAS'02, Arizona, U.S.A, May 2002.

[11] R J. van de Plassche, "Precision current-source arrangement", U.S. Patent 3 982 172, Sept. 21, 1976.

[12] H. S. Jackson, "Circuit and method for canceling nonlinearity error associated with component value mismatches in a data converter," U. S. Patent 5 221 926, June 22, 1993.