Proceedings of the 6th WSEAS International Conference on Robotics, Control and Manufacturing Technology, Hangzhou, China, April 16-18, 2006 (pp177-180)

Logical Projection Vectors Method To Label Connected Components In Binary Images For Simple Real-Time Applications

M. SALMAN CHAUDHRY, JAVAID IQBAL AND AKHTAR NAWAZ MALIK
Departments of MTS and Computer Engineering
College of Electrical and Mechanical Engineering
Peshawar Road, Rawalpindi
PAKISTAN

Abstract: - This paper proposes a new method to efficiently label connected components in binary images. There exist many methods for this but they all are complex enough to prohibit implementation for simple real-time applications such as robotics. The proposed method is very simple in terms of implementation complexity and extremely fast, at the cost of a small fraction of accuracy. This approach is best suited for simple real-time robotic applications.

Key-Words: - Binary images, Connected components, Labeling, Real-time, Robotics.

1 Introduction

To count and label connected components in an image is a vital task in most image processing algorithms (such as image segmentation). The output of these algorithms is usually in binary format i.e. 1s and 0s. In these outputs, the high (1) pixels are found in groups that are separated by low (0) pixels. Within the groups, the high pixels are 4 or 8 connected to each other. 4-connectivity is based on the connection between two pixels when they both have a common x or y location index. Whereas the 8 connected pixels exist in diagonals. A connected component labeling algorithm finds these connections between pixels and assigns them a label that is same for all the pixels in one group and different from that of the other groups. An algorithm that has the property of assigning labels in sequential order is always appreciated as it provides with the count of connected components in the image.

The connected component labeling algorithms are found in two categories; 1) sequential and 2) parallel. The sequential algorithms are conventional and are run on single processing unit machine whereas the parallel algorithms take the help of more than one processing units to perform this operation. Most sequential algorithms work in raster scan manner.

Generally the most basic algorithm works like this; In the first scan (normally top-left to bottom-right), if two connected pixels are found and at most, any one has an old label, they are labeled with a new or found label. If both have different labels, then those labels are recorded in an equivalence table. In the second scan, all the equivalent labels are replaced by their smallest equivalent label. The process of second scan is known as re-labeling. The enhancements to the basic algorithm revolve around the re-labeling' pass distribution in spatial domain whereas iterative algorithms also exist [1]-[3]. The real-time implementation of such methods is rather complex and is expensive [7]. The presented novel method does not employ any raster scans or equivalence tables rather it performs logical operations in predefined directions and calculates projection vectors. This implementation is limited to simple real-time applications like robotics as there is a chance of error i.e. in very complex images, multiple components maybe labeled as same.

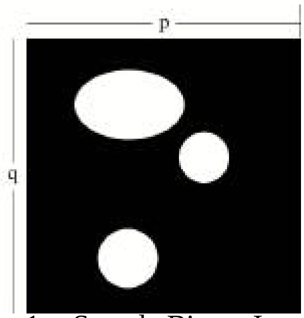

Fig. 1 – Sample Binary Image

2 Logical Projection Vectors

The presented algorithm is based on 'logical projection vectors' to label the 'visibly' separated

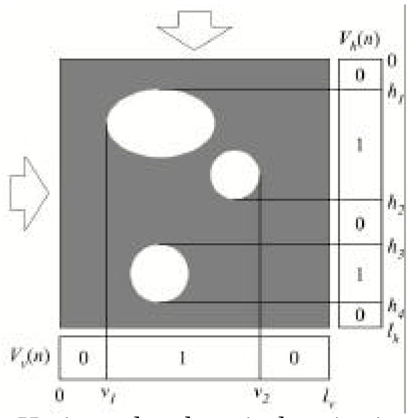

Fig. 2 – Horizontal and vertical projection vector construction

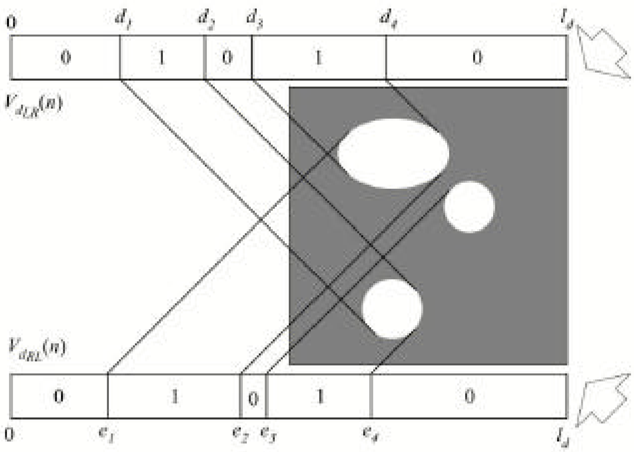

Fig. 3 – Diagonal projection vectors construction

components. These projections are obtained using the logical 'OR' operation between all the pixels that are intersected by the 'projected ray'. Thus these are binary vectors. We have defined four projection vectors; 1) horizontal, 2) vertical, 3) left to right diagonal, 4) right to left diagonal. There are a certain number of projected rays for each projection vector. For horizontal vector, the no. of rays is equal to the height of image (q), for vertical vector, it equals the width of image (p). Both diagonal vectors have length a unit less than the sum of both dimensions of image, i.e.

$$l_h = q$$

$$l_v = p$$

$$l_d = p + q - 1$$

We use the notation of logical OR '$\vee$' in the manner that summation of a range is usually written. The limits define the elements that are OR operated with each other. Then the vectors are defined as;

$$V_h(n) = \bigvee_{j=0}^{p-1} i(j,n) \;, \qquad n = 0 \; to \; l_h$$

$$V_v(n) = \bigvee_{j=0}^{q-1} i(n,j) \;, \qquad n = 0 \; to \; l_v$$

For the above two vectors j is not a function of n, whereas it is for the diagonal vectors. The R-L (right to left) diagonal vector is found by;

$$V_{d_{LR}}(n) = \bigvee_{\substack{j=l_s \\ k=r_s}}^{\substack{l_e \\ r_e}} i(j,k) \;, \quad n = 0 \; to \; l_d \qquad (1)$$

Where;

$$l_s = \begin{cases} n & n \le q \\ q & n > q \end{cases}, \qquad r_s = \begin{cases} 1 & n \le q \\ n-(q-1) & n > q \end{cases}$$

$$l_e = \begin{cases} 1 & n \le p \\ n-(p-1) & n > p \end{cases}, \qquad r_e = \begin{cases} n & n \le p \\ p & n > p \end{cases} \qquad (2)$$

It can be seen in equation (1) that j=l_s:l_e is a decrementing and k=r_s:r_e is an incrementing operation. The L-R diagonal vector can simply be obtained by flipping the image around vertical axis (horizontally) and passing through same process as the R-L vector. However the direct way to calculate is;

$$V_{d_{RL}}(n) = \bigvee_{\substack{j=l_s \\ k=r_s}}^{\substack{l_e \\ r_e}} i(j,k) \;, \quad n = 0 \; to \; l_d \qquad (3)$$

Where;

$$l_s = \begin{cases} q & n \le p \\ q-(n-p) & n > p \end{cases}, \quad r_s = \begin{cases} n & n \le p \\ p & n > p \end{cases} \qquad (4)$$

$$l_e = \begin{cases} q-(n-1) & n \le q \\ 1 & n > q \end{cases}, r_e = \begin{cases} 1 & n \le q \\ n-(q-1) & n > q \end{cases}$$

It can be seen in the equation (3) that both j=l_s:l_e and k=r_s:r_e are decrementing operations. Also note that in equations (1) and (3), the limits for j and k i.e. l_s, l_e, r_s and r_e are not linear ranges rather piecewise functions of n.

3. The Algorithm

After the computation of logical projection vectors, the intersections of these vectors are calculated to estimate the connected components in the image. The removal of induced error (non-existing areas) is automatic as the zeroes in the binary image are not labeled with any number even if they are indicated as estimated components.

The horizontal and vertical vectors are at right angles to each other but both are at an angle of p/4 radians to the diagonal vectors. Therefore the processing is started using either pair. First, the vertical and horizontal vector pair is processed to

Proceedings of the 6th WSEAS International Conference on Robotics, Control and Manufacturing Technology, Hangzhou, China, April 16-18, 2006 (pp177-180)

find the intersected regions of the groups of both then the diagonal vectors are used to subdivide these regions (if necessary).

The start and end of each group in a vector is marked and recorded. The notations are used as;

Horizontal: $h_1, h_2 \ldots h_{nh}$
Vertical: $v_1, v_2 \ldots v_{nv}$
Diagonal L-R: $d_1, d_2 \ldots d_{nd}$
Diagonal R-L: $e_1, e_2 \ldots e_{ne}$

Then the intersected regions for first pair will be;

$(h_1,v_1 – h_2,v_2)$, $(h_1,v_3 – h_2,v_4)$ … $(h_1,v_{nv-1} – h_2,v_{nv})$
$(h_3,v_1 – h_4,v_2)$, $(h_3,v_3 – h_4,v_4)$ … $(h_3,v_{nv-1} – h_4,v_{nv})$
:
$(h_{nh-1},v_1 – h_{nh},v_2)$, $(h_{nh-1},v_3 – h_{nh},v_4)$ … $(h_{nh-1},v_{nv-1} – h_{nh},v_{nv})$

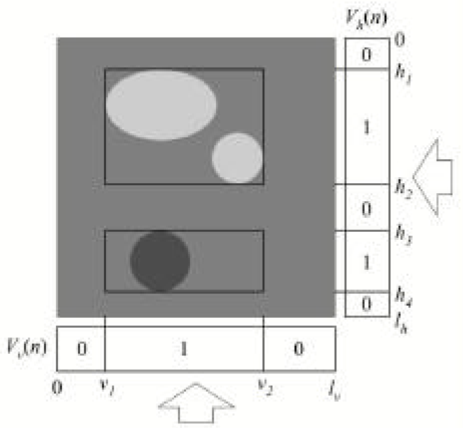

Fig. 4 – Estimated regions and labeling from hz. and vt. vector groups

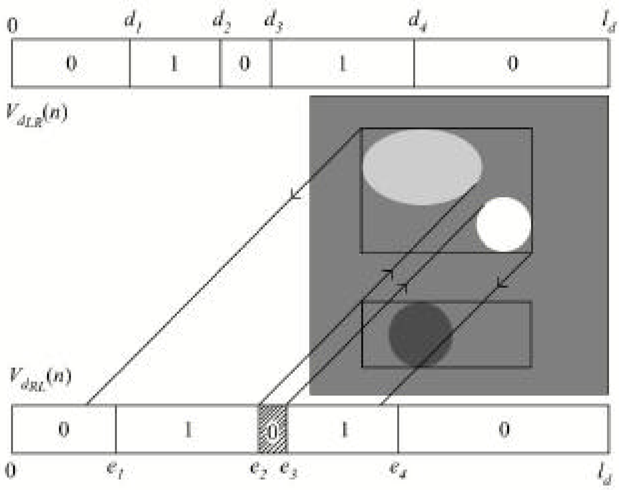

Fig. 5 – Separation and re-labeling of estimated regions

A number of these estimated regions will have to be cancelled out due to the fact that no ‘high’ pixel is defined by them in the image. Therefore all such regions are discarded.

Now the remaining procedure involves checking to see if any of the estimated regions from one pair divide some estimated regions of the other. This is done by projecting the y axis limits of each estimated region on the diagonal vectors in the following way;

The elements of diagonal vector are also 1 pixel apart as the image therefore it is easy to project any point on to it. Since the diagonal vector is at p/4 radians to the horizontal axis, the projected point location can be found as;

$$\tan\left(\frac{p}{4}\right) = \frac{perp}{base} = \frac{y}{b}$$
$$\Rightarrow b = y$$

Where b is the displacement between x axis location of projected point and its projection on the diagonal vector (figure 2) from which the actual location is easily found.

If any non-group (zero) area in the diagonal vector is found ‘bounded’ by the limits of the projection of an estimated region then there are two separate connected components in that estimated region separated by a diagonal. This diagonal is specified by the bounded non-group. The limits of this non-group are one by one considered as n and definitions from the set of equations (2) are used to separate the connected component label. The same procedure is then used for L-R diagonal vector using (4) to check if there are more separations to do.

An enhancement to this method can be implemented if the image is sub-divided into equal sized blocks and the same algorithm is applied to each of them. To obtain the result, the labels from all sub-images can be merged together to get the final labeled image.

11. Results/Conclusion

We have presented a new method to efficiently label connected components in binary images. Implementation of the available methods is complex and expensive thus not suitable for simple real-time applications such as robotics. The proposed method is simple to implement especially on hardwired techniques. The method was successfully tested and found to outperform the raster scan based algorithms even using the serial processors.

11. REFERENCES

[1] L. D. Stefano and A. Bulgarelli, “A simple and efficient connected components labeling algorithm,” in Proceedings of International Conference on

Image Analysis and Processing, pp. 322-327, 1999.

[2] K. Haris, S. Efstratiadis, N. Maglaveras, J. Gourassas, and G. Louridas, "Artery skeleton extraction using topographic and connected component labeling," in Proceedings of International Conference on Image Processing, pp. 339-342, 2001.

[3] A. Rosenfeld and J. L. Pfaltz, "Sequential operation in digital picture processing," Journal of Association for Computing Machinery, Vol. 13, pp. 471-494, 1966.

[4] R. M. Haralick, "Some negihborhood operations," in Real Time/Parallel Computing Image Analysis, M. Onoe, K. Preston, and A. Rosenfled, eds., New York: Plenum Press, pp. 11-35, 1981

[5] R. Lumia, L. Shaprio, and O. Zuniga, "A new connected components algorithm for virtual memory computers," Computer Vision, Graphics and Image Processing, Vol. 22, pp. 287-300, 1983

[6] K. Suzuki, I. Horiba, and N. Sugie, "Fast connected-components labeling based on sequential local operations in the course forward raster scan followed by backward raster scan," in Proceedings of 15th International Conference on Pattern Recognition, Vol. 2, pp. 434-437, 2000.

[7] K. B. Wang, T. L. Chia, Z. Chen and D. C. Lou, "Parallel Execution of a Connected Component Labeling Operation on a Linear Array Architecture", *Journal Of Information Science And Engineering 19*, pp. 353-370, 2003