

## PRECISION MEASUREMENT IN LOW RESOLUTION IMAGERY

D. C. Douglas Hung  
New Jersey Institute of Technology  
Newark, NJ 07102

O. R. Mitchell  
University of Texas at Arlington  
Arlington, TX 76019

### ABSTRACT

The significant disadvantage for traditional contour representation is when the resolution is reduced the effort of undersampling is proportional enlarged. The goal of this study is to improve edge detection results, especially for those corner points in low resolution. This study describes a method, which is based on 4-connected pixel-wise linearization, for finding contours from low resolution video images. This allows more accurate inspection and identification of objects from image data. In practice, geometrical models are used to manipulate the linearization. A hypothesis is generated for examining the corner points. It provides a solution to recognition of low resolution imagery.

### INTRODUCTION

In computer vision, a simple edge is a border between two regions, each of which has approximately uniform brightness. Edges often result from the occluding outer contour of objects and one of the regions should be considered belonging to the object. Much work that has so far been concerned with detection of edge fragments is concerned with the inaccuracy due to sampling resolution. The boundaries that are extracted from low resolution images are highly distorted even in ideal computer synthesized images.

There has been quite a number of different approaches toward development of edge detection over the years which includes such operators as the Sobel [3], Prewitt [8], the Hueckel [5], [6], [7]. Since ideal step edge may not be present in real image, a thick contour most likely to be found. Although a non-maximum suppression method of edge thinning [RosA78] or a local extrema extraction [MitO79] can be used to thin the edge, it only obtain edge location to pixel accuracy. When the pixel resolution is reduced, the distortion is enlarged especially for those segments with high curvature.

Our study estimates the contour using the gray-level images. An precision measurement based on the concept of pixel-wise linearization is used to generate two-dimensional coordinates of points with high accuracy. As an object is imaged using fewer pixels (due to the camera covering a larger field of view), the exact shape still can be determined, and precise measurements of orientation, location, perimeter, and area can be made.

### STRAIGHT EDGE MEASUREMENT

#### A. Ideal Straight Edge Model

Let us assume that quantization of the original image is performed with a square sampling aperture of side length one, and we obtain  $n$  gray levels ranging from  $f_{\min}$  to  $f_{\max}$ . It is assumed that a sampling grid is small enough to allow the boundary to be represented as a straight line as it through this region. In general, an edge point is a location where a brightness change is taking place. Considering an ideal step edge with a random orientation which is passing through a window of image as shown in Figure 1. The orientation of the boundary is defined as the angle between boundary normal vector and the  $x$ -axis. It is only necessary to consider the angle in a range of  $\pi/4$  radians due to symmetry. The translation of the boundary is defined as the shortest distance from the lower left corner of the sampling grid to the boundary. Thus, the boundary can be characterized by the parameters: edge orientation  $\theta$ , edge translation  $\rho$ , background intensity  $g_b$ , and edge contrast  $c$ .

When the image,  $g(x,y)$ , is digitized, the brightness is known only at a discrete set of location. We can think of the result as defined by a discrete grid of impulse. Let use  $\omega_{ij}$  to denote the region of integration. It is assumed that the straight edge is to be located at a random position through out a one pixel square area i.e. two independent uniform probability distributions should be introduced for  $\theta$  and  $\rho$ , respectively. Let us consider the case that a straight edge is passing through  $\omega_8=\omega_{ij}'$  with orientation  $\theta$  and translation  $\rho_\theta$ . Let  $\{\omega_k; k=0, \dots, 7\}$  be a 8-connected neighbor set of  $\omega_8$ . The order of the labeling is according to the relationship shown as Figure 1.

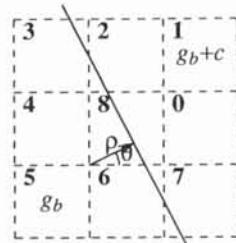


Figure 1 An idealized edge model with square sampling grid superimposed is a line separating two regions of constant gray levels. All nine pixels are labeled for reference.

Let  $a_k$  be the area of  $\omega_k$  covering by the region with higher brightness ( $g_b+c$ ) i.e. the region to the left hand side of the edge, and  $a_k$  can be expressed as

$$a_k = \begin{cases} 1 & \text{if } p_k < 0 \text{ or } q_k < 0 \\ 1 - \rho_k^2 \csc 2\theta & \text{if } 0 < \rho_k \leq \rho_L \\ 1 + 0.5 \tan \theta - \rho_k \sec \theta & \text{if } \rho_L < \rho_k \leq \rho_M \\ 1 + (1 + \rho_k^2 - 2\rho_k \rho_H) \csc 2\theta & \text{if } \rho_M < \rho_k \leq \rho_H \\ 0 & \text{if } \rho_k > \rho_H \end{cases} \quad (1)$$

where  $\rho_k$  be the translation of  $\omega_k$ ,  $\rho_L = \sin \theta$ ,  $\rho_M = \cos \theta$ , and  $\rho_H = (\sin \theta + \cos \theta)$  for  $k=0, \dots, 8$ . From the above expression,  $a_k$  is function of two parameters  $\rho$  and  $\theta$ . In image processing, area measure can be obtained from gray level data i.e.

$$a_k = \frac{f(i, j) - g_b}{c}. \quad (2)$$

## B. The Basic Models and Edge Estimation

Consider the following condition: suppose that we do not know neither the translation nor the orientation i.e.  $(\rho, \theta)$  are unknown. The only information we have is the gray level data i.e.  $f(i, j)$ . There has infinitely many combinations of  $(\rho, \theta)$  can be found. Figure 2 shows the plots of  $\rho$  versus  $\theta$  with different  $a_k$ . To solve the above problem, the concept of slope coherence, i.e. an edge segment crossing two consecutive pixels is linear except for potential corner pixels, is introduced for pixel-wise linealization.

According to the previous assumption that a sampling grid  $\omega_k$  is small enough to allow the boundary to be represented as a straight line as it through the region. Patterns that have been created by a linear edge segment crossing a pixel are classified as  $d$ - and  $\bar{d}$ -cells. We propose the set of crossing patterns as edge primitives. They are defined as:  $d$ -cell has an edge crossing adjacent sides of the pixel (shown as  $\omega_2, \omega_3, \omega_6$ , and  $\omega_7$  in Figure 1), while  $\bar{d}$ -cell has an edge crossing opposite sides of the pixel (shown as  $\omega_8$  in Figure 1).

We now describe the basic models, the  $d$ -cell and  $\bar{d}$ -cell, each of which have a continuous curvature extend from its 4-connected neighborhood.

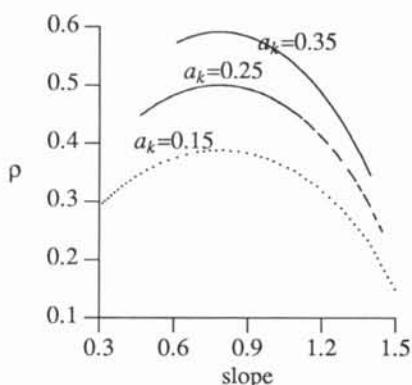


Figure 2 Plot of translation  $\rho$  versus slope.

a. The estimation of a  $d$ -cell which has a 4-connected  $d$ -cell neighbor shown as Figure 3a. Let  $\alpha$  and  $\beta$  be the effective area for  $\omega_8$  and its neighbor, respectively. According to the similarity of two triangles and the geometry, we have

$$\frac{y}{x} = \frac{1-y}{z} = \frac{(1-y)^2}{2\beta} \quad (3)$$

where  $z=2\beta/(1-y)$ . Since  $2\alpha=yx$ , we can have

$$y = \frac{K_1}{1+K_1}, \quad x = \frac{2\alpha}{y} \quad (4)$$

where  $K_1=\sqrt{\alpha/\beta}$ . Thus, we can obtain the local slope,  $m$

$$m = \tan \theta = \frac{y}{x} = 2\alpha(1 + \frac{1}{K_1})^2 \quad (5)$$

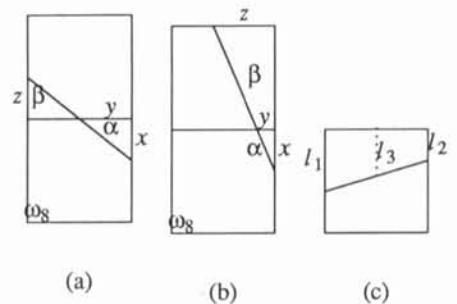


Figure 3 The piecewise estimation of a (a)  $d$ -cell with a 4-connected  $d$ -cell neighbor, (b)  $d$ -cell with a 4-connected  $p$ -cell neighbor, and (c)  $p$ -cell.

To represent this estimation, the center point of the hypotenuse is marked.

b. The estimation of a  $d$ -cell which has a 4-connected  $p$ -cell neighbor shown as Figure 3b. Similar to previous analysis, it follows

$$z = \frac{2(\alpha + \beta)}{1+x} \quad (6)$$

$$\frac{y}{x} = \frac{z}{1+x} = \frac{2(\alpha + \beta)}{(1+x)^2} \quad (7)$$

$$x = \frac{K_2}{1-K_2}, \quad y = \frac{2\alpha}{x} \quad (8)$$

and

$$m = \tan \theta = \frac{y}{x} = 2\alpha(1 - \frac{1}{K_2})^2 \quad (9)$$

where  $K_2=\sqrt{\alpha/(\alpha+\beta)}$ .

c. The estimation of a  $\bar{d}$ -cell which is shown as Figure 3c. Let  $l_1$  and  $l_2$  be the length of the parallel sides of a trapezoid. The effective estimation area can be computed as  $a=(l_1+l_2)/2$ . It is the same as the definition for the length of the center parallel line  $l_3$ . To present the estimation, the ending point of  $l_3$  is used.

Consider the experiment of estimating an ideal step edge with the constraint of  $(\rho, \theta)$ . Only the quantization, 0.5 gray level, error will affect the estimation result. In general, the gray level image use 256 levels to represent the brightness i.e. the error is  $1.95 \times 10^{-3}$  pixel square. Thus, our algorithm can locate modeled edge to hundreds of a pixel.

## CONTOUR ESTIMATION

### A. The Discontinuities

We state that a corner is a location where two edges are joined. It is defined that the model of a corner, consisting of two linear sectors having constant slopes, enclosing an angular discontinuity. Let  $\omega_c$  be a corner pixel;  $v_1$  and  $v_2$  be the crossing points for the outgoing and incoming edges, respectively. A general model is shown in Figure 4. Let  $p_1$  be the estimated node of the right-hand side neighbor of  $\omega_c$ . Then, the precise location of  $c$  can be estimated by

$$v_2 + (s \sin(\theta+\phi), s \cos(\theta+\phi)) \quad (10)$$

where  $\phi$  is the angle between  $\vec{v_1 v}$  and  $\vec{v_1 v_2}$ ,  $\theta$  is the angle between  $\vec{v_1 v_2}$  and  $\vec{p_1 v_1}$ , and  $s$  is the length of  $\|v_1 c\|$ .

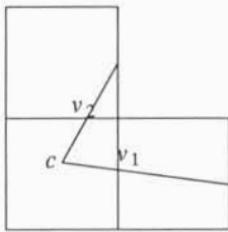


Figure 4 The general model for corner estimation.

In low resolution images, the spatial dimension is compressed. It is very likely that several corners become 4-connected neighbors and form a *corner stack* (Figure 5). To estimate the corner stack, a recursive corner finding procedure is used. The prediction is started from the top-most layer of the stack and then down to the lowest layer.

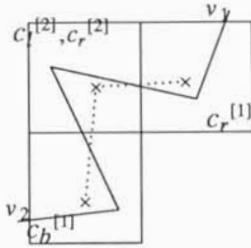


Fig 5. The estimation of a corner stack.

Let  $c_l^{[j]}$  and  $c_r^{[j]}$  be a pair of corners in  $j$ th layer. If the border tracing is performed in the counter-clockwise direction, then  $c_l^{[j]}$  will be in the left-pile and  $c_r^{[j]}$  in the right pile of the stack. Let  $v_1^{[j]}$  and  $v_2^{[j]}$  be the crossing points on this layer. The recursive corner finding algorithm uses the previous estimation as the initial guesses (indicated as 'x' in Figure 5). The procedure can be expressed as:

1. If  $j=1$ ,  $c^{[0]}$  is replaced by the  $c^{[1]}$ 's left-hand or right-hand side 4-neighbor.
2. Update  $c^{[j]}$  by the corner estimation method.
3. If  $j$  is not the deepest layer, set  $j=j+1$  and go to step 2.

4. An error of  $\epsilon = \max(d(v_1^{[j]}, v_1^{[j-1]}), d(v_2^{[j]}, v_2^{[j-1]}))$  is computed.. If  $\epsilon > \tau$  then set  $j=1$  and go to step 1, else stop.

### B. The Collision

A *collision* has occurred if more than one edge has been detected within a non-corner pixels i.e. a pixel has been traced more than once. A pixel with a collision is called a *x-cell*. A *x-cell* has three regions: two backgrounds and one object (it could be two objects and one background for a concave curve) as shown in Figure 6. To manipulate the *x-cell*, two *pseudo pixels* are introduced to interpret them, separately.

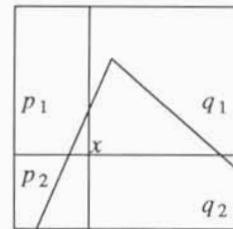


Figure 6 Corner with single collision.

Using the concept of slope coherence, the slope of the left-side pseudo pixel is estimated by the average slope of its two consecutive neighbors, say  $p_1$  and  $p_2$ , and the geometry rules. Similarly, the right-side one is predicted by  $q_1$  and  $q_2$ .

## EXPERIMENTAL RESULTS

The triangle of Figure 7 was used for the first test. Given a sequence of decreasing resolutions  $r_j$  for all possible  $j$  where  $r_0$  is the resolution of the original image and  $r_k$  is the resolution after the image is shrunk by a factor of  $2^k$ . Figure 7a-7e demonstrates the shape preserving effect of our algorithm for decreasing resolutions.

A most challenging shape for our algorithm is a circle, especially when the resolution is very small. A circle with a 3.3 pixels radius and its origin point randomly shifted to make the synthesized image non-symmetric is shown in Figure 8a. To measure the compactness of a circle, a measure *curve factor* [1] is used. The curve factor is dimensionless, scale-independent quantity related to a predetermined ideal shape i.e. circle. Let  $\{P_1, \dots, P_N\}$  are equally spaced points of a curve, then the curve factor is defined as

$$\zeta_n \equiv \frac{1}{R} \sqrt{\frac{1}{N} \sum_{i=1}^N (P_i - \bar{P}_c - R)^2} \quad (11)$$

where

$$\bar{R} = \frac{1}{N} \sum_{i=1}^N |P_i - \bar{P}_c|$$

$$\bar{P}_c = \frac{1}{N} \sum_i P_i$$

Since  $\zeta_n$  is normalized by  $R$ , it is scale-independent and positive defined quantities which is related to an ideal circle. For an ideal circle,  $\zeta_n$  is equal to zero. Our technique gives a  $\zeta_n$  value of 0.00576 as well as an area error of 0.734 and a perimeter error of 0.156 (Figure 8b). The measurement indicates an imperfect compactness of 0.5%.

## CONCLUSION

This paper has presented an algorithm based on slope coherence for sub-pixel edge estimation. It operates based on simplest models which is associated with the geometric linearization idea. Corners and collisions problem are properly solved and implemented. Since corners

are of interest by advanced recognition and inspection systems, one advantage of this technique is the corners are located when the edge estimation is done.

The estimation power of our algorithm is demonstrated through the experimental results. It is useful for low resolution images especially for printed circuit inspection under a controlled environment.

## REFERENCES

- [1] P. E. Danielson, "A new shape factor," *Computer Graphics Image Processing*, vol. 7, pp. 292-299, 1978.
- [2] H. Freeman, "Computer processing of line drawing images," *Computer Survey*, vol. 6, pp. 57-97, Mar. 1974
- [3] R. O. Duda and P. E. Hart, *Pattern Classification and Scene analysis*, New York:Wiley, 1973.
- [4] K. A. Dunkelberger and O. R. Mitchell, "Contour tracing for precision measurement," *IEEE Int. Conf. Robotics and Automation*, 1985.
- [5] Manfred H. Hueckel, "An operator which locates edges in digitized pictures," *J. Assoc. Comp. Mach.*, vol. 18, pp. 113-125, Jan. 1971.
- [6] Manfred H. Hueckel, "A local visual operator which recognizes edges and lines," *J. Assoc. Comp. Mach.*, vol. 20, pp. 634-647, Oct. 1973.
- [7] Manfred H. Hueckel, "Erratum for [6]," *J. Assoc. Comp. Mach.*, vol. 21, pp. 350, Apr. 1974.
- [8] J. M. S. Prewitt, "Object enhancement and extraction," in *Picture Processing and Psychopictorics*, B. S. Lipkin and A. Rosenfeld, Eds. New York:Academic Press, 1970.

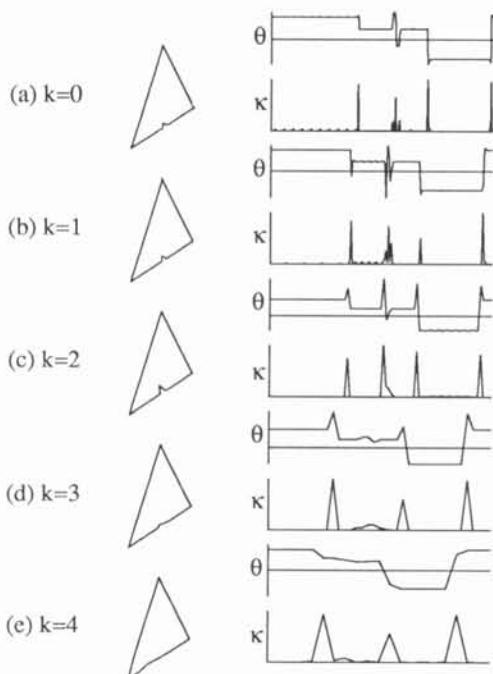


Figure 7 Multiresolutional information extraction.

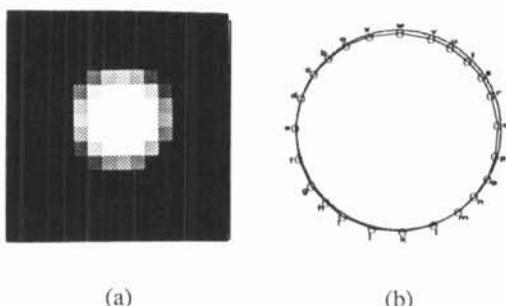


Figure 8 (a) Computer generated image for circle with a 3.3 pixel radius. (b) Estimated curve (solid line) and a dotted circle with a radius of 3.3 pixels.