論文 86-23-6-16

A Study on the Extraction of Feature by State-Space Concept with Euclidean Distance Operator

(Euclidean 거리연산자와 결합된 상태공간 기법에 의한 영상추출)

崔 甲 石* 尹 東 漢**

(Kap Seok Choi and Dong Han Yoon)

要 約

윤곽선 추출을 위한 합리적이고 효율적인 방법을 제안한다. 본 논문에서의 방법은 euclidean 거리 연산 산자와 결합된 상태공간 기법을 이용한다. 제안 방법을 sobel 연산자와 비교 하였고 그 결과 본 논문에서 제시한 방법이 sobel 연산자를 이용한 것 보다 효율적임을 입증하였다.

Abstract

An efficient and reliable method for the extraction of features is presented. The method utilizes by a state space technique with Euclidean distance operator. The proposed method is compared with the Sobel operator. Simulation results show that our method performs as well as the Sobel operator.

I. Introduction

The extraction of features such as edges and curves from an image is useful for a variety of purposes. Edges and similar locally defined features play important roles in texture analysis (1-4). The interpretation of image edges as arising from various types of discontinuities in the scene plays an important role in the inference of 3-dimension surface structure from an image (5-13). Edges are useful in image matching for obtaining sharp matches that are insensitive to grayscale distortions (14-19). Edges can be used in conjunction with various segmentation techniques to improve the quality of a segmentation (20-28). It should also be mentioned that linear curves are often of importance in their own right, e.g. roads or drainage patterns on low resolution remote sensor imagery.

The general approach to edge detection makes use of digital versions of standard isotropic derivative operators, such as the gradient or Laplacian. A closely related approach is to linearly convolve the image with a set of masks representing ideal step edges in various directions. Lines and curves can be similarly detected by linear convolutions. However linear operators are not specific to features of a given type many also respond in other situations involving local intensity changes. An alternative approach is to use gated operators respond only when specific relationships hold among the local intensities. In all of these approaches the output is a quantitative edge or curve value. The final detection decision can be made by thresholding this value

*正會員：明知大學校 電子工學科

(Dept. of Elec. Eng., Myong Ji Univ.)

**正會員，金烏大學校 電子工學科

(Dept. of Elec. Eng., Kum-oh Inst. of Tech.)

接受日字：1986年 6月 20日

New classes of operators have been defined based on fitting polynomial surfaces to the local image intensities and using the derivatives of these polynomials as edge value estimates. This method which was first proposed by Prewitt allows edges to be located to subpixel accuracy (29). Another method proposed by Hueckel is to find a best fitting step edge to the local intensities.

The edge detectors were based on small image neighborhoods, typically 3 by 3. A more powerful approach is to use a set of first or second difference operators based on neighborhoods having a range of sizes and combine their outputs, so that discontinuities can be detected at many different scales. Here the edges are localized at maxima of first differences or at zero-crossings of second differences. Oprators based on large neighborhoods can also be used to detect texture edges, at which the statistics of various local image properties change abruptly. Cooperation between operators in different positions can be used to enhance the feature values at points lying on smooth edges or curves. This was one of the first applications of relaxation methods to image analysis at the pixel level (34-35).

The standard approaches to edge detection are based on a very simply model in which the image is regarded as ideally composed of essentially constant regions separated by step edges. Recent study by Haralick is based on the more general assumption of a piecewise linear, rather than piecewise constant, image, which allows simple shading effects to be taken into account.

In this paper an efficient and reliable method for automatic edge detection is presented. This method is based upon template matching using state space techniques with a Euclidean distance. The recursive characteristic of the proposed method yields, computational efficiency.

The paper is composed into four parts. Part 11 provides the concept of the method. In Part 111, the method is evaluated in comparison with the Sobel operator. Conclusions are included in part IV.

II. Edge Detector Using State Space Techniques

Let the template weights be described by the coefficients Tij for i,j=1,2,3, Ii(j) be the intensity of the image at line i and column j, and Yn(k) be accumulated result at line n and column k. The representation of the operation is shown in Figure 1. The template is placed over each input pixel. The pixel and its eight neighbors are multiplied by the their respective template weighting coefficients and summed. The result is placed in the output image at the same center pixel location. This operation occurs for each pixel in the input image, e.g. 256 X 256=65,536 times. Each operation requires nine multiplications and nine additions. A full image convolution requires on the order of a half-million multiplications and addition-not a quick process.

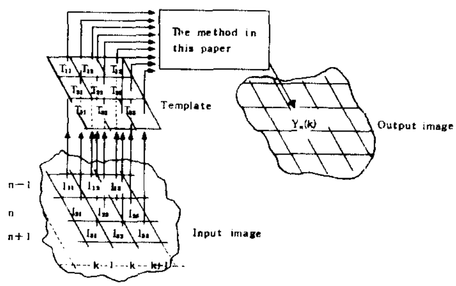

Fig. 1. The representation of the operation.

For using state space technique, this process can be realized by a moving average operator mat H(S), define as:

$$Yn(k) = \text{mat } H(S) \text{ mat } Cn(k+1) \quad (1)$$

where

$$\text{mat } H(S) = ((T_{11} S^2 + T_{12} S + T_{13}) \; (T_{21} S^2 + T_{22} S + T_{23}) \; (T_{31} S^2 + T_{32} S + T_{33}))$$

$$\text{mat } C(k+1) = \begin{bmatrix} I_{n-1}(k+1) \\ I_n(k+1) \\ I_{n+1}(k+1) \end{bmatrix}$$

and S is the horizontal shift operator defined by $S^i I_j(k) = I_j(k-i)$

Replacing S by 1/Z, equation (1) can be modified as follows:

$$Y_n(k) = (T_{11} Z^{-2} + T_{12} Z^{-1} + T_{13}) I_{n-1}(k+1) + (T_{21} Z^{-2} + T_{22} + T_{23}) I_n(k+1) + (T_{31} Z^{-2} + T_{32} Z^{-1} + T_{33}) I_{n+1}(k+1) \quad (2)$$

mat H(Z) can be realized by a multi-input single-output state space model as shown in Figure 2 with equation (2).

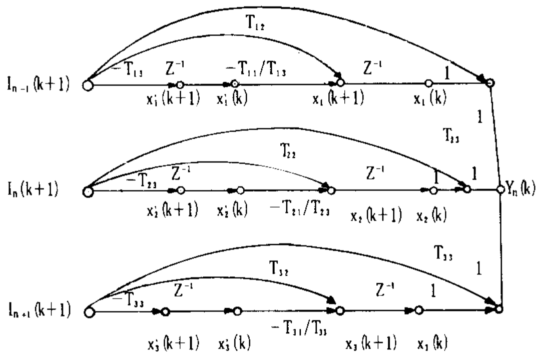

Fig. 2. The multi input single output state space model for realizing mat H(Z).

Employing this realization yields:

$$\begin{bmatrix} x_1(k+1) \\ x_1'(k+1) \\ x_2(k+1) \\ x_2'(k+1) \\ x_3(k+1) \\ x_3'(k+1) \end{bmatrix} = \begin{bmatrix} 0 & -T_{11}/T_{13} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & T_{21}/T_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -T_{31}/T_{33} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_1'(k) \\ x_2(k) \\ x_2'(k) \\ x_3(k) \\ x_3'(k) \end{bmatrix}$$

$$+ \begin{bmatrix} T_{12} & 0 & 0 \\ -T_{13} & 0 & 0 \\ 0 & T_{22} & 0 \\ 0 & -T_{23} & 0 \\ 0 & 0 & T_{32} \\ 0 & 0 & -T_{33} \end{bmatrix} \begin{bmatrix} I_{n-1}(k+1) \\ I_n(k+1) \\ I_{n+1}(k+1) \end{bmatrix}$$

$$Y_n(k) = (1\ 0\ 1\ 0\ 1\ 0) \begin{bmatrix} x_1(k) \\ x_1'(k) \\ x_2(k) \\ x_2'(k) \\ x_3(k) \\ x_3'(k) \end{bmatrix} + (T_{13}\ T_{23}\ T_{33}) \begin{bmatrix} I_{n-1}(k+1) \\ I_n(k) \\ I_{n+1}(k+1) \end{bmatrix}$$

Therefore, state space equation is obtained

mat $X_n(k+1)$ = A mat X(k) + B mat $C_n(k+1)$ (3)

mat $Y_n(k+1)$ = C mat X(k) + D mat $C_n(k+1)$ (4)

where mat $X_n(k)$ is 6 X 1 state vector and A, B, C, D are 6 X 6, 6 X 3, 1 X 6, 1 X 3 matrices.

The image is vector scanned horizontally from left to right and the 3 X 1 vector mat C_n (k+1) is fed as an input to the state space model yielding the output $Y_n(k)$. The vector scan moves downwards in increments of one line. mat $X_n(0)$ represents the background intensity to the left of the image. Each template will correspond to a state space model, thus, four state space models will operate in parallel.

To illuminate how the method utilizes the function $Y_n(k)$, let's consider without the loss of any generalities that the vertical line edge shown in Figure 3 is to be detected using template weight coefficients as shown in Figure 4.

n − 1	5	5	5	25	5	5	5
n	5	5	5	25	5	5	5
n + 1	5	5	5	25	5	5	5
		k − 2	k − 1	k	k + 1	k + 2	

Fig. 3. The intensity of input image for vertical line.

− 1	2	− 1
− 1	2	− 1
− 1	2	− 1

Fig. 4. Template weight coefficients.

The state space model will yield the following sequence of output: $Y_n(k-2)=0$, $Y_n(k-1)=-60$, $Y_n(k)=120$, $Y_n(k+1)=60$, $Y_n(k+2)=0$

A 5 by 1 pattern vector mat $Y_n(k)=(Y_n(k+2)\ Y_n(k+1)\ Y_n(k)\ Y_n(k-1)\ Y_n(k-2))$ can be formed to be utilized by the Euclidean distance. mat $Y_n(k)$ can be generated recursively as follows:

mat $Y_n(k)$ = mat R(S) mat $C_n(k+2)$ (5)

where mat $R(S)^t = (H(S)\ SH(S)\ S^2H(S)\ S^3H(S)\ S^4H(S))$. mat R(S) can be decomposed as follows: mat R(S) = mat U(S) H(S) (6)

where mat $U(S)^t = (1\ S\ S^2\ S^3\ S^4)$.

Equation (5) can be written as follows:

$$\text{mat } Y_n(k) = \text{mat } R(S) \text{ mat } C_n(k+1)$$
$$= \text{mat } U(S) H(S) \text{ mat } C_n(k+2)$$
$$= \text{mat } U(S) Y_n(k+2) \qquad (7)$$
$$\text{mat } Y_n(k) = \text{mat } U(S) Y_n(k+2)$$

Replacing S by 1/Z mat U(Z) can be realized by a single input multi output state space model as shown in Figure 5.

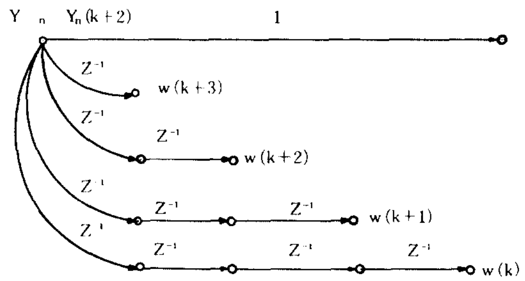

Fig. 5. The single input multi output state space model for realizing mat U(Z).

Employing this realization yields:

$$\text{mat } W(k+1) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w(k) \\ w(k+1) \\ w(k+2) \\ w(k+3) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} Y_n(k+2)$$

$$\text{mat } Y_n(k) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w(k) \\ w(k+1) \\ w(k+2) \\ w(k+3) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} Y_n(k+2)$$

Therefore, state space equations is obtained

$$\text{mat } W(k+1) = E \text{ mat } W(k) + F\, Y_n(k+2) \qquad (8)$$
$$\text{mat } Y_n(k) = G \text{ mat } W(k) + J\, Y_n(k+2)$$
$$n, k = 1, 2, 3, \ldots \qquad (9)$$

where mat W(k) is a 4 X 1 state space vector and E, F, G, J are 4 X 4, 4 X 1, 5 X 4, 5 X 1 matrices.

The two state space models represented by equations (3), (4) and (8), (9) can be augmented into one state space model as follows:

$$\begin{bmatrix} \text{mat } X(k+1) \\ \text{mat } W(k+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ P & E \end{bmatrix} \begin{bmatrix} \text{mat } X(k) \\ \text{mat } W(k) \end{bmatrix} + \begin{bmatrix} B \\ Q \end{bmatrix} \text{mat } C_n(k+2) \qquad (10)$$

$$\text{mat } Y_n(k) = (R\ G) \begin{bmatrix} \text{mat } X(k) \\ \text{mat } W(k) \end{bmatrix} + M \text{ mat } C_n(k+2) \qquad (11)$$

where 0 is 6 by 4 zero matrix and

$$P = \begin{bmatrix} 000000 \\ 000000 \\ 000000 \\ 101010 \end{bmatrix} \quad Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ T_{13} & T_{23} & T_{33} \end{bmatrix} \quad R = \begin{bmatrix} 101010 \\ 000000 \\ 000000 \\ 000000 \\ 000000 \end{bmatrix} \quad M = \begin{bmatrix} T_{13} & T_{23} & T_{33} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The output vector mat $Y_n(k)$ is compared to a predefined set of prototype vectors that represent the different types of edges that can be detected using the template that is associated with the state space model. Classification of the pattern vector is based on a Euclidean distance concept as follows:

Consider M pattern classes and assume that these classes are representable by prototype patterns mat $Z_1, Z_2, \cdots, Z_m$. The Euclidean distance between the i-th prototype which represents the i-th type of edge associated and pattern vector mat $Y_n(k)$ is

$$D_i = \| \text{mat } Y_n(k) - \text{mat } Z_i \|$$
$$= \sqrt{(\text{mat } Y_n(k) - \text{mat } Z_i)^t (\text{mat } Y_n(k) - \text{mat } Z_i)} \qquad (12)$$

Euclidean operator computes the distance from a pattern mat $Y_n(k)$ of unknown classification to the prototype of each class, and assigns the pattern to the class to which it is closet. In other words, the pattern vector is assigned to the i-th edge if

$$D_i(\text{mat } Y_n(k)) < D_j(\text{mat } Y_n(k)) \text{ for all } i=j$$

The pattern vectors from the other three state space models are compared to the prototype of the associated templates. Upon the completion of the classification process, a mapping function to map the edges will be invoked.

The method associated state space equations requires that the number of multiplications needed to obtain mat $Y_n(k)$ is only 15 multiplications .

III. Evaluation of the method

The method presented in the paper was evaluated using the input image of 3M. For each template used by the recursive edge detector Figure 6, the state space model is developed as described by equation (10) and (11).

mask 1

−1	2	−1
−1	2	−1
−1	2	−1

mask 2

2	−1	−1
−1	2	−1
−1	−1	−2

mask 3

−1	−1	−1
2	2	2
−1	−1	−1

mask 4

−1	−1	2
−1	2	−1
2	−1	−1

Fig. 6. The mask of edges that are stored as prototypes.

For each mask there are about 20 types of edges that are stored as prototypes, and are used by the Euclidean distance operator. The threshold value "T" was chosen to be 40 and an edge detection is displayed if the outcome of the mask is larger than 1 3T 1.

Figure 7 shows the input image of the letter 3M and Figure 8 shows the produced image which is applied Figure 7 by the method of state space concept. The Sobel operator is applied to Figure 7 and the result is shown in Figure 9. It is evident that the method in the paper performs as well as the Sobel operator.

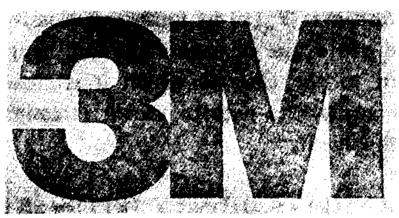

Fig. 7. The input image of the letter 3M.

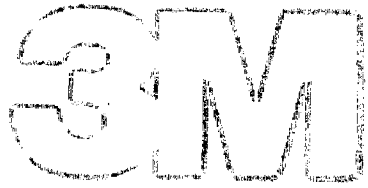

Fig. 8. The produced image by the method in the paper.

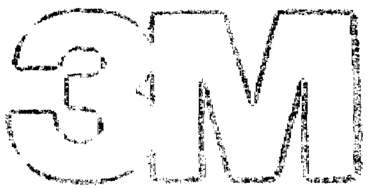

Fig. 9. The produced image by the Sobel operator.

Fig. 10. The input image of the cup.

Fig. 11. The produced image by the method in the paper.

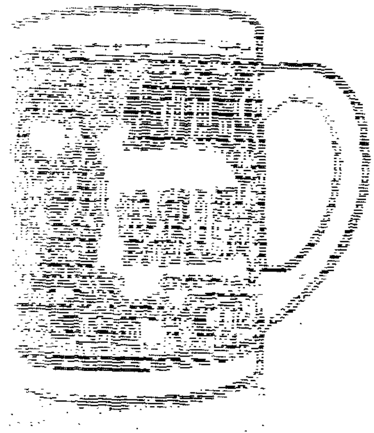

Fig. 12. The produced image by the Sobel operator.

IV. Conclusion

In this paper a recursive edge detection method based on the state space concept with an Euclidean distance operator is proposed. The simulation result demonstrates that the method in the paper performs as well as the Sobel operator.

References

[1] B. Julesz, *Visual Pattern Discremination.* IRE Trans. Inf. Theory IT-8, 84-92, 1962.

[2] R.M. Haralick, "Textural features for image classification", *IEEE Trans. Syst. Man Cybernet.* SMC-3, 610-621, 1973.

[3] L.S. Davis, "Texture analysis using generalized cooccurrence matrices", *IEEE Trans. Pattern Anal. Mach.* Intell. PAMI-1, 251-259, 1979.

[4] J.T. Maleson, *Understanding Natural Texture.,* Proc. DARPT Image Understanding Workshop, 19-27, 1977.

[5] D. Waltz, *Understanding Line Drawings of Scenes with Shadows,* The Psychology of Vision, 19-91, M-H, 1975.

[6] T, Kanade, *Recovery of the Three Dimensional Shape of an Object From a Single View,* Artof. Intell. 17, 409-460, 1981.

[7] T.O. Binford, *Inferring Surfaces from Images.,* Artif. Intell. 17, 205-244, 1981.

[8] H.R. Barrow, *Interpreting Line Drawings as 3-Dimensional Surfaces.,* Artif. Intell. 17, 75-116, 1981.

[9] K.A. Stevens, *The Visual Interpretation of Surface Contours.,* Artif. Intell. 17, 47-73, 1981.

[10] A.P. Withkin, *Recovering Surface Shape and Orientation from Texture.,* Artif. Intell. 17, 17-45, 1981.

[11] B. Horn, *Obtaining Shape from Shading Infermation.,* The Psychology of computer Vision, M-H, 115-155, 1975.

[12] K. Ikeuchi, *Numerical Shape from Shading and Occluding Boundaries., Artif.* Intell. 17, 141-184, 1981.

[13] R.J. Woodham, *Analyzing Images of Curved Surfaces.,* Artif. Intell. 17, 117-140, 1981.

[14] M.A. Fischier, "The representation and matching of pictorial structure", *IEEE Trans. Computer.* C-22, 67-92, 1973.

[15] L.S. Davis, *An application of Relaxation Labelling to Spring Roded Template Matching.,* Proc. 3rd Int. Joint conf. on pattern Recognition, 591-597, 1976.

[16] S. Ranade, *Point Pattern Matching by Relaxation.,* Pattern Recognition 12, 268-275, 1980.

[17] O.D. Faugeras, *Semantic Description of Aerial Images using Stochastic Labelling.,* Proc. 5th Int. Conf. on Pattern Recogintion, 352-357, 1980.

[18] K. Price, "Matching segments of images", *IEEE Trans. Pattern Anal. Mach. Intell. PAMI-1,* 110-116, 1979.

[19] D.H. Ballard, *Generalizing the Hough Transform to Detect Arbitrary Shapes.,* Pattern Recognition 13, 111-122, 1981.

[20] W.E.L. Grimson, From Images to Surfaces, MIT Press, Cambridge, MA, 1981.

[21] B.K.P. Horn, *Determining Optical Flow.,* Artif. Intell. 17, 185-203, 1981.

[22] S. Ullman, *The Interpretation of Visual Motion, MIT* Press, Cambridge, MA, 1979.

[23] D.R. Reddy, *Picture Segmentation using a Recursive Region Splitting Method.,* Computer Graphics Image Process. 8. 313-333, 1978.

[24] J.O. Eklundh, "A relaxation method for multispectral pixel classification", *IEEE Trans. Pattern Anal. Mach. Intell.* PAM-2,

72-75, 1980.
[25] A. Rosenfeld, "Thresholding using relaxation", *IEEE Trans. Pattern Anal Mach. Intell.* PAMI-3, 598-606, 1981.
[26] T. Pavlidis, Structural Pattern Recognition, Springer, NY, 1977.
[27] J.A. Feldman, *Decision theory and artifical intelligence.,* Artif. Intell. 5, 349-371, 1974.
[28] J.M. Tenenbaum, *Experiments in Interpretation-Guided Segmentation.,* Artif. Intell. 8, 241-274, 1977.
[29] J.M. S. Prewitt, *Object Enhancembent and Extraction.* Picture Processing and Psychopictorics, Academic Press, 75-149, 1970.
[30] M.H. Hueckel, *An operator which locates edges in digital Pictures.,* J. Ass. Computer Mach. 18, 113-125, 1971.
[31] M.H. Hueckel, *A local visual operator which recognizes edges and lines.,* J. Ass. Comput. Mach. 20, 634-647, 1974.
[32] A. Rosenfeld, "Edge and curve detection for visual scene analysis", *IEEE Trans. Comput.* C-20, 562-568, 1971.
[33] D. Marr, *Theory of Edge Detection.,* Proc. R. Soc. B207, 187-217, 1980.
[34] S.W. Zucker, "An application of relaxation labeling to line and curve enhancement", *IEEE Trans. Comput.* C-26, 394-403, 922-929, 1977.
[35] B.J. Schachter, "An application of relaxation method to edge reinforcement" *IEEE Trans. Syst. Man Cybernet.* SMC-7, 813-816, 1977.