퍼지 및 지능시스템학회 논문지 2001, Vol. 11, No. 6, pp. 454-458

디지털 영상의 퍼지시스템 표현을 이용한 Edge 검출방법

An edge detection method for gray scale images based on their fuzzy system representation

문병수 · 이현철 · 김장열

Byung Soo Moon, Hyun Chul Lee, and Jang Yeol Kim

한국원자력연구소

요 약

이 논문에서는 디지털 영상의 퍼지시스템 표현으로부터 유도된 Edge검출 알고리듬에 대하여 기술한다. 이 알고리듬은 Gradient를 기반으로 한 것으로서 Convolution Kernel이 기존의 Roberts, Prewitt 또는 Sobel등이 제안한 Gradient Kernel과 다른 새로운 것이다. 사용한 퍼지시스템은 디지털 영상을 근사적으로 표현한 Bicubic Spline 함수를 퍼지시스템 화한 것으로서 2차 도함수가 연속이기 때문에 Gradient나 Laplacian 연산이 가능하다. Grid 점들에서 이 함수의 Gradient는 두 개의 축 방향으로 각각 한 개의 3×3행렬과 원래 영상과의 Convolution에 의하여 산출됨을 보였으며 이를 이용하여 검출된 Edge들은 기존의 다른 방법을 사용하여 검출된 Edge영상보다 훨씬 선명함을 확인하였다. 이 알고리듬 적용사례 2개에 대한 기술이 포함되어 있다.

Abstract

Based on a fuzzy system representation of gray scale images, we derive an edge detection algorithm whose convolution kernel is different from the known kernels such as those of Roberts', Prewitt's or Sobel's gradient. Our fuzzy system representation is an exact representation of the bicubic spline function which represents the gray scale image approximately. Hence the fuzzy system is a continuous function and it provides a natural way to define the gradient and the Laplacian operator. We show that the gradient at grid points can be evaluated by taking the convolution of the image with a 3×3 kernel. We also show that our gradient coupled with the approximate value of the continuous function generates an edge detection method which creates edge images clearer than those by other methods. A few examples of applying our methods are included.

Key Words : Fuzzy System, Edge Detection Algorithm. Convolution Kernel. Gradient Operator, Laplacian

1. Introduction

Edge detection is a technique used in image pre-processing for image enhancement or for pattern recognition by reducing an image to show only its edge details. Edge enhancement is implemented through spatial filters such as shift and difference, Prewitt gradient, and Laplacian edge filters. In addition to the above, many other filters have been developed including Sobel, Kirsch, and Robinson filters[1,2,4]. These filters are mostly of size 3×3 and they are applied to a gray scale image by taking the convolution of the image with a selected kernel. In gray scale images of real pictures, however, there are many factors that degrade the edges including the photon noise, blurring or defocusing, and irregularities of the surface structure of the objects[1]. These noises can also be removed by applying a convolution with properly chosen 3×3 kernels[6]. In this paper, we derive a 3×3 kernel different from any of the above based on a cubic spline function which represents the gray scale image approximately. Note that, instead of using the exact cubic spline interpolation function for the gray scale image, we use an 'approximate' representation whose values at grid points or at pixels may differ from the original. This will not only allow us to define the cubic spline function without the lengthy computations necessary to obtain the cubic spline interpolation function but also to remove some of the noises described above.

We will show in the following that for the approximate representation, we use the original gray scale image as the rule table for a fuzzy system so that no computation is necessary at all to set up the fuzzy system. It is shown[7] that the fuzzy system is an exact representation of the cubic spline function representing the gray scale image and it enables us to evaluate the complicated spline function in a very efficient manner.

First, we show how a cubic spline function can be defined to represent a gray scale image. Let $\{f_{i,j}|i,j=$

접수일자 : 2001년 9월 15일
완료일자 : 2001년 12월 1일

$1,2,\cdots N\}$ be a gray scale image, i.e. $f_{i,j}\in\{0,1,2,\cdots 255\}$ assuming 256 gray levels are used, and let $B_i(x)$ be the cubic B-spline defined on $[x_{i-2}, x_{i+2}]$ where x_j's are defined so that $x_{j+1}-x_j=h$ for all $j=0,1,2,\cdots N+1$. Thus, $B_i(x)$ is defined as $\frac{1}{6h^2}\times$

$$\begin{cases}(x-x_{i-2})^3 \\ h^3+3h^2(x-x_{i-1})+3h(x-x_{i-1})^2-3(x-x_{i-1})^3 \\ h^3+3h^2(x_{i+1}-x)+3h(x_{i+1}-x)^2-3(x_{i+1}-x)^3 \\ (x_{i+2}-x)^3 \\ 0\end{cases} \quad (1)$$

Note that we may assume that $f_{i,j}=f(x_i, y_j)$ for some continuous function $f(x,y)$ where y_j's are also assumed to be equally spaced points. The following theorem justifies that we may use cubic spline function $S(x,y)=\sum_{i,j=-1}^{N} f_{i,j}B_i(x)B_j(y)$ can be used as an approximate representation of a gray scale image $\{f_{i,j}|i,j=1,2,\cdots N\}$.

Theorem 1 [8]. If $f(x,y)$ is a three times continuously differentiable function and if $S(x,y)=\sum_{i,j=-1}^{N} f_{i,j}B_i(x)B_j(y)$ where $f_{i,j}=f(x_i,y_j)$ with $B_i(x)$, $B_j(y)$ defined on the intervals $[x_{i-2},x_{i+2}]$ and $[y_{j-2},y_{j+2}]$ respectively, then we have $S(x,y)-f(x,y)=O(h^2)$ for all $(x,y)\in[x_0,x_{N+1}]\times[y_0,y_{N+1}]$.

Proof. We refer to [8] for a complete proof and the following is a sketch of the proof. Assume $x\in[x_k,x_{k+1})$ and $y\in[y_l,y_{l+1})$ for some k and l. We write $S(x,y)-f(x,y)$ as a sum of five terms; $S(x,y)-S(x_k,y)$, $S(x_k,y)-S(x_k,y_l)$, $S(x_k,y_l)-f(x_k,y_l)$, $f(x_k,y_l)-f(x_k,y)$, and $f(x_k,y)-f(x,y)$. When the first and the fifth are added, we obtain $(x-x_k)\times\ (S_x(x_k,y)-f_x(x_k,y))+O(h^3)$ which is $O(h^3)$. Similarly, the second and the fourth are added to obtain the sum of $O(h^3)$. Since the third is of $O(h^2)$, the sum of the five terms will add to $O(h^2)$. Q.E.D.

Consider the case where $f_{i,j}=0$ for all $(i,j)\neq(2,2)$ and $f_{2,2}=256$, then $S(x,y)=256B_2(x)B_2(y)$ and $S(x_2,y_2)=256B_2(x_2)\ B_2(y_2)=256\times\frac{4}{9}$. Therefore, we have the maximum difference between $S(x,y)$ and $f_{i,j}$ is as large as $256\times\frac{5}{9}$. Note that

$$S(x_k,y_l)=\frac{1}{36}(f_{k-1,l-1}+f_{k-1,l}+f_{k-1,l+1}+ 4f_{k,l-1}+16f_{k,l}+4f_{k,l+1}+f_{k+1,l-1}+4f_{k+1,l}+f_{k+1,l+1})$$

and hence the process of computing $S(x_k,y_l)$ from the image $f_{i,j}$ is equivalent to taking the convolution with 3×3 kernel

$$\frac{1}{36}\begin{bmatrix}1&4&1\\4&16&4\\1&4&1\end{bmatrix}$$

which may be considered as a smoothing operator.

In case the image is a digitized image, i.e. $f_{i,j}\in\{0,1\}$ for all (i,j), then we have the following.

Theorem 2. Let $\{f_{i,j}|i,j=-1,0,1,\cdots N+1\}$ be a digitized image, i.e. $f_{i,j}\in\{0,1\}$ for all i,j and let $S(x,y)=\sum_{i,j=-1}^{N+1} f_{i,j}B_i(x)B_j(y)$ be the continuous cubic spline representation of the image. If we define $g_{k,l}=[S(x_k,y_l)+0.5]$, then $g_{k,l}=f_{k,l}$ for all $k,l=-1,0,1,\cdots N+1$.

Proof. For a complete proof, we refer to [8]. A sketch of the proof is as follows. From the inequality relation $-\frac{4}{9}\leq S(x_k,y_l)-f_{k,l}\leq\frac{4}{9}$, we have $f_{k,l}+\frac{05}{9}\leq S(x_k,y_l)+0.5\leq f_{k,l}+\frac{8.5}{9}$ and so we have $[f_{k,l}+\frac{0.5}{9}]\leq[S(x_k,y_l)+0.5]\leq[f_{k,l}+\frac{8.5}{9}]$. Note that $[f_{k,l}+\frac{0.5}{9}]=[f_{k,l}+\frac{8.5}{9}]=f_{k,l}$ and hence we have the conclusion. Q.E.D.

Next, we consider a fuzzy system representation of the gray scale images. To represent a function $z=f(x,y)$ of two independent variables, we need to define fuzzy sets for the fuzzification of input variables x and y, fuzzy combination rules to define the relations between the dependent variable z and the two independent variables x,y, and the output fuzzy sets to defuzzify the output fuzzy sets to obtain crisp values for the variable z. In the following, we will consider a gray scale image $\{f_{i,j}|i,j=1,2,\cdots N\}$ as a discrete function of two variables.

The input fuzzy set for the value at pixel (k,l) is defined by $B_k(x)\times B_l(y)$ where $B_k(x)$ is the cubic B-spline defined on $[k-2,k+2]$ with $h=1$ in (1) and similarly for $B_l(y)$. Note that we have $B_k(x)\times B_l(y)=B(x-k)\times B(y-l)$ which we write as $S_{k,l}(x,y)$. For the fuzzy combination rules, we simply take the image $\{f_{i,j}\}$ as the rule table. Next, we sort the the numbers $f_{i,j}$ appearing in the rule table and form triangular sets to form output fuzzy sets. Recall that we assume $f_{i,j}\in\{0,1,2,\cdots,255\}$ and hence that there are at most 256 output fuzzy sets. The following theorem proves that the fuzzy system formed by the above method is an exact representation of the cubic spline function $S(x,y)=\sum_{i,j=-1}^{N+1} f_{i,j}B_i(x)B_j(y)$. For a proof of the theorem, we refer to our earlier proof in [7].

2. Gradient and Laplacian Operators by Cubic Spline Function Representation

In this section, we consider how the gradient vector and the Laplacian of the bicubic spline function $S(x,y) = \sum_{i,j=-1}^{N+1} f_{i,j} B_i(x) B_j(y)$ can be evaluated, where $\{f_{i,j} | i,j = 1,2,\cdots,N\}$ is a gray scale image. Recall that $B_i(t)$ defined in (1) is a piecewise cubic ploynomial which is twice continuously differentiable and hence the gradient vector $\nabla S(x,y) = (\frac{\partial S}{\partial x}, \frac{\partial S}{\partial y})$ and the Laplacian $\nabla^2 S(x,y) = \frac{\partial^2 S}{\partial^2 x} + \frac{\partial^2 S}{\partial^2 y}$ not only exist but also they are continuous.

First, we consider one term of $S(x,y)$, i.e. $S_{k,l}(x,y) = B_k(x) \times B_l(y)$. When the gradient and the Laplacian of $S_{k,l}(x,y)$ are comuted and evaluated at the grid points, we obtain the values shown in Table 1 and Table 2. It is routine to check that the vallues at all other grid points are zeros.

Table 1. The Values of $12 \times \nabla S_{k,l}(i,j)$

	j=l-1	j=l	j=l+1
i=k-1	(1, 1)	(4, 0)	(1,-1)
i=k	(0, 4)	(0, 0)	(0,-4)
i=k+1	(-1, 1)	(-4, 0)	(-1,-1)

Table 2. The Values of $\nabla^2 S_{k,l}(i,j)$

	j=l-1	j=l	j=l+1
i=k-1	1	1	1
i=k	1	-8	1
i=k+1	1	1	1

Now, we consider the first components of $\nabla S(i,j)$. It can be computed by $\frac{\partial S}{\partial x}(i,j) =$

$$-f_{i-1,j-1} - 4f_{i-1,j} - f_{i-1,j+1} + f_{i+1,j-1} + 4f_{i+1,j} + f_{i+1,j+1} \tag{2}$$

where the coefficients come from the first components of the vectors in Table 1. Note that the coefficients for corresponding to $k=i$ are all zeros and they do not appear in (2). Similarly the second component of $\nabla S(i,j)$ can be evaluated by $\frac{\partial S}{\partial y}(i,j) =$

$$-f_{i-1,j-1} + f_{i-1,j+1} - 4f_{i,j-1} + 4f_{i,j+1} - f_{i+1,j-1} + f_{i+1,j+1} \tag{3}$$

and the Laplacian $\nabla^2 S(i,j)$ is

$$f_{i-1,j-1} + f_{i-1,j} + f_{i-1,j+1} + f_{i,j-1} - 8f_{i,j} + f_{i,j+1} + f_{i+1,j-1} + f_{i+1,j} + f_{i+1,j+1} \tag{4}$$

By using the convolution formula, the above relations (2), (3), and (4) can conveniently be written as $\frac{\partial S}{\partial x} = S * p$ and $\frac{\partial S}{\partial y} = S * q$ where p and q are the matrices formed by the first and second entries in Table 1 respectively. Similarly, the Laplacian can also be written as $\nabla^2 S = S * h$ where h is the 3×3 matrix in Table 2. Our gradient operator compares with other known gradient operators such as Prewitt, Sobel or Isotropic operators as shown in Table 3a, 3b, 4a and 4b. Note that our gradient operator shown in Table 5a and Table 5b come from relations (2) and (3).

Table 3a. Prewitt-horizontal

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Table 3b. Vertical

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Table 4a. Sobel-horizontal

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Table 4b. Vertical

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Table 5a. Ours-horizontal

$$\begin{bmatrix} -1 & 0 & 1 \\ -4 & 0 & 4 \\ -1 & 0 & 1 \end{bmatrix}$$

Table 5b. Vertical

$$\begin{bmatrix} -1 & -4 & -1 \\ 0 & 0 & 0 \\ 1 & 4 & 1 \end{bmatrix}$$

3. An Edge Detection Method Based on Cubic Spline Function

Edge detection is a useful method for detecting object boundaries. For a continuous image $z = f(x,y)$, the magnitude of its gradient will be small at a point where there is no sharp change of the values from any of its neighbor points and will be larger wherever there is a big jump from any of its neighbor points. Therefore, the magnitude of the gradient should be large at points of the object boundaries. By using our fuzzy system representation, we computed the convolution of the image with the 3×3 kernel shown in Table 5a and 5b, and compared the result with the one by Sobel's gradient.

When the magnitude of the Sobel's gradient is computed and cut it off to zero when the magnitude is below 75, we obtain the pictures fig.1a and fig.2a. If our method is applied on the same figures with the magnitude of the gradient multiplied by $128/(1+S(x,y))$ where $S(x,y)$ is the cubic spline representation of the image, and cut it off to zero when the result is less than 128, then we obtain the results shown in fig.1b and fig.2b. One can see from these figures that our method creates an image of edge boundaries which looks cleaner than the one by Sobel's gradient. We did try to create an image by using the magnitude of the Sobel's gradient divided by 1+Image(i,j), followed by a multiplication with various

Fig. 1a. Edges by Sobel's Gradient

Fig. 1b. Edges by Our Gradient

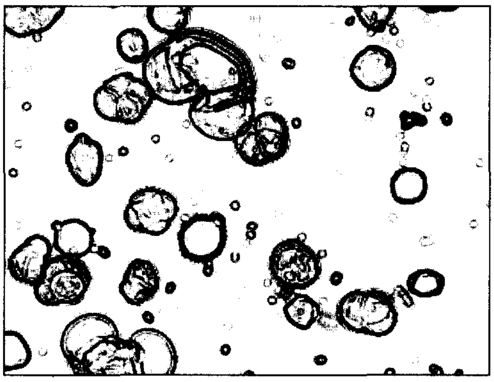


Fig. 2a. Edges by Sobel's Gradient

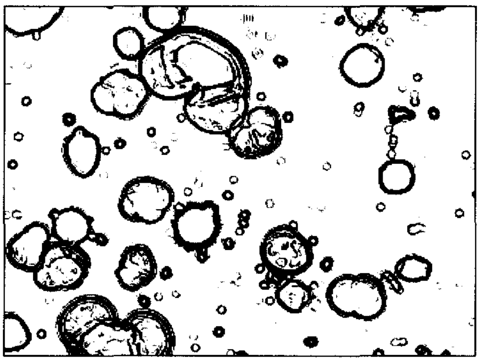


Fig. 2b. Edges by Our Gradient

factors, we found that they are no better than the original Sobel's gradient.

4. Conclusion

We showed that a gray scale image can be represented by a fuzzy system without doing any computation on the image data at all. This representation computes an approximate values at grid points rather than the exact values of the image data. A 3×3 gradient kernel is derived from this representation which is different from other known kernels. Using the magnitude of the gradient obtained from this representation divided by 1 plus the value of the function, we obtain an excellent image for the edge boundaries. We still have to prove numerically, however, that our method produces a better image for the edge boundaries.

Acknowledgements

This work has been carried out under the nuclear research and development program supported by the Ministry of Science and Technology of Korea.

References

[1] Larry S. Davis, "A Survey of Edge Detection Techniques", Computer Graphics and Image Processing, Vol. 4 (1975) 248-270.

[2] Guner S. Robinson, "Edge Detection by Compass Gradient Masks", Computer Graphics and Image Processing, vol. 6 (1977) 49-501.

[3] D. Marr and E. C.Hildreth, "Theory of Edge Detection", Computer Vision IEEE, Los Alamitos, CA (1991) 77-107.

[4] Gregory A. Baxes, *Digital Image Processing, Princples and Applications*, John Wiley & Sons, Inc. (1994) 93-95.

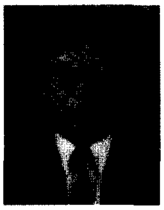
[5] Morton Nadler and Eric P. Smith, *Pattern Recognition Engineering*, John Wiley & Sons Inc. (1993) 107-135.

[6] Milan Sonka, Vaclav Hlavac and Roger Boyle, Image Processing, Analysis, Machine Vision, PWS Publishing (1999), 77-79.

[7] Byung Soo Moon, "A practical algorithm for representing polynomials of two variables by a fuzzy system with accuracy O(h^4)", Fuzzy Sets and Systems, 119(2) (2001) 135-141.

[8] B. S. Moon, D. Y. Lee, H. C. Lee, "Fuzzy System Representation of Digitized Patterns and an Edge Tracking Thinning Algorithm", Proc. Int. Fussy Systems Assoc. (2001) 1466-1471.

저 자 소 개



문병수
1974~ : Univ of Illinois 수학 박사
1974~1978 : Sargent & Lundy Enginners (Chicago) 근무1978-현재 한국원자력연구소 책임연구원

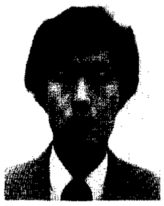
관심분야 : Fuzzy Systems, 인공신경망

Phone : 042-868-2980
fax : 042-868-8916
E-mail : bsmoon@kaeri.re.kr

이현철
1992년 : 한국과학기술원 산업공학과 졸업
1992년~현재 : 한국원자력연구소 선임급

관심분야 : 인간공학, 인간-기계 시스템 설계 및 평가

Phone : 042-868-2942
Fax : 042-868-8916
E-mail : leehc@kaeri.re.kr



김장열
1994년 : 중앙대학교 대학원 전산학박사
1985년~현재 : 한국원자력연구소 책임급

관심분야 : 소프트웨어 검증

Phone : 042-868-2252
Fax : 042-868-8916
E-mail : jykim@kaeri.re.kr