韓國컴퓨터情報學會 論文誌
第14卷 第6號, 2009. 6.

2009-14-6-2-3

Improvement of Edge Detection Using Mean Shift Algorithm

Shin Seong-Yoon *, Lee Chang-Woo *, Rhee Yang-Won *

Mean Shift 알고리즘을 활용한 경계선 검출의 향상

신 성 윤*, 이 창 우*, 이 양 원*

Abstract

Edge detection always influenced by the noise of original image, therefore need some methods to eliminate them in advance, and the Mean Shift algorithm has the smooth function which suit for this purpose, so adopt it to fade out the unimportant information and the sensitive noise portions. Above all, we use the Canny algorithm to pick up the contour of the objects we focus on. And, take tests and get better result than the former sole Canny algorithm. This combination method of Mean Shift algorithm and Canny algorithm is suitable for the edge detection processing.

요 약

경계선 검출은 항상 원 이미지의 노이즈에 의해 영향을 받으므로 사전에 노이즈들을 제거하는 방법들이 필요하다. 그리고 Mean Shift 알고리즘은 이러한 목적에 알맞은 smooth 함수를 가지고 있고, 그래서 중요하지 않은 정보와 노이즈에 민감한 부분들을 점차 제거하는 방법들을 택하고 있다. 우선 Canny 알고리즘을 사용하여 객체의 윤곽선을 추출하는데 초점을 맞추었다. 그리고 알고리즘을 테스트 하고 이전의 단독 Canny 알고리즘보다 우수한 결과를 얻었다. 따라서 Mean Shift 알고리즘과 Canny 알고리즘이 조합된 방법은 경계선 검출 처리에 적당함을 말한다.

▸ Keyword : 경계선 검출(edge detection), Mean Shift 알고리즘(Mean Shift Algorithm), Canny 알고리즘(Canny algorithm), Smooth 함수(Smooth function)

• 제1저자 : 신성윤 교신저자 : 이양원
• 투고일 : 2009. 06. 09, 심사일 : 2009. 06. 08, 게재확정일 : 2009. 06. 28.
* 군산대학교 컴퓨터정보공학과 교수

I. Introduction

The definition of edge is the boundary between an object and the background, and indicates the boundary between overlapping objects. This means if the edges in an image can be identified accurately, all of the objects can be located and basic properties can be measured further, so the edge detection is a very important task in image processing and model recognize domains.

Hitherto there are so many methods of edge detection, such as statistic method, differential method, and curve fitting and so on. Regular detection algorithm considers the contour is the lightness or brightness discontinuous result. Majority edge detection algorithms adopt the rectangle window data weight average in order to assume the gradient vector of discrete region. The approach mentioned in this paper base on region density estimation, and regards the different value of density as the metrics of edge detection. However, if just do like this will involve some noise points, and increase useless calculation work. So before the Canny algorithm processing the input image, use Mean Shift cluster method to smooth the image ahead of time. This additional procedure can not only find out the precise edge information effectively and efficiently, but also eliminate the not important information and noise influent.

II. Related Work

Edge detection is a tool in image processing aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. Majority the methods detect edges by first computing a measure of edge strength, usually a first order derivative expression such as the gradient magnitude [1], and then searching for local directional maxima of the gradient magnitude using a computed estimate of the local orientation of the edge, usually the gradient direction. All these alleged gradient based algorithms have kernel operators that calculate the strength of the slope in directions which are orthogonal to each other, commonly vertical and horizontal. Later, the contributions of the different components of the slopes are combined to give the total value of the edge strength. Meanwhile, before it there is a pro-processing step to edge detection, a smoothing stage, typically Gaussian smoothing, is almost always applied because of the all of the operators are usually sensitive to the noise points.

There are many different methods for edge detection. Some of them will look at in the following way.

Fuzzy classifier detects classes of image pixels corresponding to gray level variation in the various directions. It uses an extended Epanechnikov function as a fuzzy set membership function(FSMF) for each class where the class assigned to each pixel is the one with the greatest fuzzy truth of membership. This classification is done first, after which a competition is run as a second step to thin the edges. Like the Canny edge detector, the edge sensitivity of our competitive fuzzy edge detector (CFED) can be set from low to high by the user[2].

[3] introduces the discrete singular convolution(DSC) algorithm for edge detection. Two classes of new edge detectors, DSC edge detector(DSCED) and DSC anti-noise edge detector(DSCANED), are proposed for the detection of multiscale edges.

[4] formulate edge detection as statistical inference. This statistical edge detection is data driven, unlike standard methods for edge detection which are model based. For any set of edge detection filters (implementing local edge cues) we use pre-segmented images to learn the probability distributions of filter responses conditioned on whether they are evaluated on or off an edge.

[5] presents new evaluation methodology and a framework in which edge detection is evaluated through boundary detection, that is, the likelihood of retrieving the full object boundaries from this edge-detection

output.

Depth edges play a very important role in many computer vision problems because they represent object contours. [6] strategically project structured light and exploit distortion of light pattern in the structured light image along depth discontinuities to reliably detect depth edges.

III. Enhancement of Canny and Mean Shift

3.1 Enhancement of Canny Algorithm

The Canny algorithm uses an optimal edge detector based on a set of criteria which include finding the most edges by minimizing the error rate, marking edges as closely as possible to the actual edges to maximize localization, and marking edges only once when a single edge exists for minimal response. According to Canny, the optimal filter that meets all three criteria above can be efficiently approximated using the first derivative of a Gaussian function.

$$G(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$\frac{\partial G(x,y)}{\partial x}axe^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$\frac{\partial G(x,y)}{\partial y}aye^{-\frac{x^2+y^2}{2\sigma^2}}$$

The first stage involves smoothing the image by convolving with a Gaussian filter. This is followed by finding the gradient of the image by feeding the smoothed image through a convolution operation with the derivative of the Gaussian in both the vertical and horizontal directions. The 2-D convolution operation is described in the following equation.

$$I'(x,y)=g(k,l)\otimes I(x,y)=\sum_{k=-N}^{N}\sum_{l=-N}^{N}g(k,l)I(x-k,y-l)$$

Where: g(k, l) = convolutional kernel

I(x, y) = original image

I′(x, y) = filtered image

2N + 1 = size of convolutional kernel

Both the Gaussian mask and its derivative are separable, allowing the 2-D convolution operation to be simplified. The optimization is not limited to software implementation only, but applies to hardware implementation as well.

The non-maximal suppression stage finds the local maxima in the direction of the gradient, and suppresses all others, minimizing false edges. The local maxima found by comparing the pixel with its neighbors along the direction of the gradient. This helps to maintain the single pixel in edges before the final thresholding stage.

Instead of using a single static threshold value for the entire image, the Canny algorithm introduced hysteresis thresholding, which has some adaptivity to the local content of the image. There are two threshold levels, th , high and tl, low where th 〉 tl. Pixel values above the th value are immediately classified as edges. By tracing the edge contour, neighboring pixels with gradient magnitude values less than th can still be marked as edges as long as they are above tl. This process alleviates problems associated with edged is continuities by identifying strong edges, and preserving the relevant weak edges, in addition to maintaining some level of noise suppression. While the results are desirable, the hysteresis stages lows the overall algorithm down considerably.

3.2 Enhancement of Mean Shift Algorithm

Generally speaking, the Mean Shift algorithm is an iteration process which can find out cluster centers. Its basic idea like this, move a shift window on the gradient direction of the feature space, starting at a randomly selected point. The convergence point of the shift window center is a cluster center; hence its kernel that the significant features of the image represent high density regions in the features space of the image and the highest density regions correspond to cluster centers. In recent years, this method is widely used in computer vision field.

Suppose x is d-dimensional Euclidean metric, {xi, 1≤i≤n} is the isolated identically distributed sample set, k(x)is kernel function, h is radius bandwidth, then the kernel density estimator definition of x becomes the well-known expression

$$f(x)=\frac{1}{nh^{d}}\sum_{i=1}^{n}K(\frac{x-x_i}{h})$$

, and the kernel function usually used is Epanechnikov kernel;

$$K_{\varepsilon}(x)=\begin{cases}(2c_d)^{-1}(d+2)(1+x^{T}x) & \left\|x^{T}x\right\|\le 1\\ 0 & otherwise\end{cases}$$

, and the Gaussian kernel function

$$K_G(x)=(2\pi)^{-d/2}\exp(-\frac{1}{2}\left\|x\right\|^{2})$$

. After a succession of calculation, get the kernel density assumption

$$\hat{\nabla} f_{h,k}(x)=\frac{2c_{k,d}}{nh^{d+2}}[\sum_{i=1}^{n}g(\left\|\frac{x-x_i}{h}\right\|^{2})]\bullet\left[\frac{\sum_{i-1}^{n}x_i g(\left\|\frac{x-x_i}{h}\right\|^{2})}{\sum_{i=1}^{n}g\left\|\frac{x-x_i}{h}\right\|^{2}}-x\right]$$

, and the vector of MeanShiftis

$$M_h(x)=\frac{\sum_{i=1}^{n}x_i g(\left\|\frac{x-x_i}{h}\right\|^{2})}{\sum_{i=1}^{n}g(\left\|\frac{x-x_i}{h}\right\|^{2})}-x$$

,and from it can make out the MeanShift iteration functionis

$$y_{j+1}=\frac{\sum_{i-1}^{n}x_i g(\left\|\frac{x-x_i}{h}\right\|^{2})}{\sum_{i=1}^{n}g(\left\|\frac{x-x_i}{h}\right\|^{2})}$$

From the above description of relationship between Mean Shift and probability density function, we make it clear that the Mean Shift algorithm is an adaptive search method that can find peak along the rise of gradient, illustrate in the followed Fig. 1:

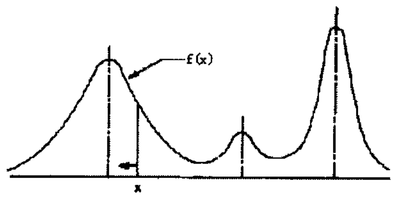


그림 1. Mean Shift 함수
Fig. 1. Mean Shift Function

, if the data set {xi, 1, ..., n} obey the probability density function, given a original point, Mean Shift algorithm will move step by step, finally converge at the first peak.

This method can be used for smoothing image, define one image as p dimensional vector on two-dimensional lattice, every point of lattice stand for a pixel, p = 1 indicate this image is a gray image, p = 3 indicate it is chromatic image, p 〉 3 indicate it is a more spectrum image, the coordinate of point in the lattice denote the space information of image. We consider all the space and color information together, combine p + 2 dimensional vector x=(xs, xr),where xs denote the coordinate of point in the lattice, xr denote the p dimensional vector features. We use kernel function K_{h_s,h_r} assume the distribution of x, K_{h_s,h_r} has the following format,

$$K_{h_s,h_r}=\frac{C}{h_s^{2}h_r^{p}}k\left(\left\|\frac{x^{s}}{h_s}\right\|^{2}\right)k\left(\left\|\frac{x^{r}}{h_r}\right\|^{2}\right)$$

, where hs, hr decide the smooth definition, and c is a normalized constant.

We use xi and zi , i=1,2, ..., n indicate the original and undergone processed image respectively. The steps of Mean Shift algorithm application are as follow:

Firstly, initialized j = 1, and make yi,1 = xi

Secondly, use Mean Shift algorithm to calculate yi,j+1 till convergence. Make the result as yi,c.

Thirdly, evaluate $z_i = \left(x_i^s, y_{i,c}^r\right)$

Ⅳ. Experimental Result

Take experiment to compare the original Canny edge detector with the improved Canny which has Mean Shift procedure in advance, the different is obvious:

그림 2. 원래의 이미지
Fig. 2. The Original Image

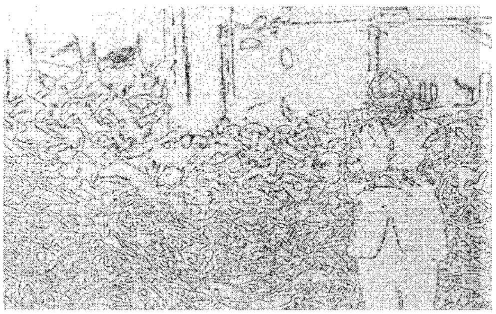


그림 3. σ=0.4, Lratio=0.4, Hratio=0.8일때의 Canny 알고리즘의 결과
Fig. 3. The Result Just use Canny Algorithm, where σ=0.4, Lratio=0.4, Hratio=0.8.

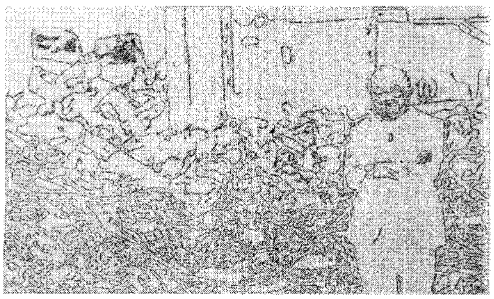


그림 4. hs=32, hr=16일 때 Mean Shift smooth와 Canny 경계선 검출기를 사용한 결과
Fig. 4. The Result use Mean Shift smooth and Canny Edge Detector, where hs=32, hr=16.

The improved Canny algorithm is better than the original one to depict the contour of main objects, and fade out the effect of noise points. However, because of the same defect of both Canny and Mean Shift that their parameters had to be personal set depending on the experience, the result somewhat irrational for the details of main objects especially the inner information. Meanwhile, the trait of Mean Shift is blurring the image boundary, removing not only the noise but also salient information as the compensation of smoothing process. So while the make out image get clearer, some inner details outline of main object disappear. If view it from general perspective, it is deserved. The following Table. 1 shows the parallel compare of merits and demerits between the traditional and improved methods:

표 1. 제안 방법의 장점과 단점
Table. 1 Merits and Demerits of Proposed Method

	time cost	contour resolution	details obtain
Canny	0.24μs	low	more
Mean Shift + Canny	601.04μs	high	less

(The Mean Shift didn't accelerate.)

Ⅴ. Conclusion

This paper presented a method can improve the original Canny algorithm for edge detection. The Mean Shift algorithm can find the cluster centers through iteration process depend on the gradient density, so add this ahead of the Canny procedure can eliminate the noise points who will influent the edge detection. In contrast, this smooth step will fade out some significant inner information more or less as the compensation of blurring image, which is the trait of Mean Shift algorithm.

Reference

[1] E. Smirni, and G. Ciardo, "Workload-Aware Load Balancing for Cluster Web Servers," IEEE Trans. on Parallel and Distributed Systems, Vol. 16, No. 3, pp. 219-232, March 2005.

[2] G. Economou, "Detecting edges using density value", Electronics Letters, 2004.

[3] L.R. Liang, C.G. Looney, "Competitive fuzzy edge detection," Applied Soft Computing 3, pp. 123-137, 2003.

[4] Z.J. Hou, G.W. Wei, "A new approach to edge detection," Pattern Recognition 35 pp. 1559-1570, 2002.

[5] S. Konishi and A. L. Yuille and James M. Coughlan and Song Chun Zhu, "Statistical edge detection: learning and evaluating edge cues," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, pp. 57-74, 2003.

[6] SongWang, Feng Ge, and Tiecheng Liu, "Evaluating Edge Detection through Boundary Detection," EURASIP Journal on Applied Signal Processing Volume 2006, Article ID 76278, Pages 1-15, 2006.

[7] Cheolhwon Kim, Jiyoung Park, Juneho Yi, "Structured Light Based Depth Edge Detection for Object Shape Recovery," Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005.

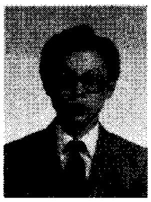
저 자 소 개

신 성 윤

2003년 2월 군산대학교
컴퓨터과학과 이학박사
2006년~현재 군산대학교
컴퓨터정보과학과 교수
관심분야 : 비디오 인덱싱,
비디오요약, 멀티미디어,
색채공학

이 창 우

1996년 경일대학교 컴퓨터공학과
(공학사)
1998년 경북대학교 컴퓨터공학과
(공학석사)
1998년~2001년 포항1대학
전임강사
2004년 경북대학교
컴퓨터공학과(공학박사)
2004년~현재 : 군산대학교 조교수
관심분야 : 텔레매틱스, 패턴인식,
컴퓨터비전, 상황인식



이 양 원

1994년 8월 숭실대학교
전자계산학과 공학박사
1986년~현재 군산대학교
컴퓨터정보과학과 교수
관심분야 : 모바일 프로그래밍,
텔레매틱스, 가상현실