Bayesian Approach to Perceptual Edge Preservation in Computer Vision

Ren-Jie Huang¹, Jung-Hua Wang^{1,*}, Chun-Shun Tseng²

¹Department of Electrical Engineering, National Taiwan Ocean University, Taiwan

²Ship and Ocean Industries R&D Center (SOIC), Taiwan

Copyright©2017 by authors, all rights reserved. Authors agree that this article remains permanently open access under the terms of the Creative Commons Attribution License 4.0 International License

Abstract This paper presents a novel approach for preserving perceptual edges representing boundaries of objects as perceived by human eyes. First, a subset of pixels (pixels of interest, POI) in an input image is selected by a pre-process of removing background and noise. One by one as a target pixel, each of POI is subjected to a Bayesian decision. This approach is characterized by iteratively employing a shape-variable mask to sample gradient orientations of pixels for measuring the directivity of a target pixel, the mask shape is updated after each iteration. We show that a converged mask covers pixels that best fit the orientation similarity with the target pixel, which in effect fulfills the similarity and proximity principles in Gestalt theory. Subsequently, a Bayesian rule is applied to the converged directivity to determine whether the target pixel belongs to a perceptual edge. Instead of using state-of-the-art edge detectors such as Canny detector [1], a pre-process combining Gaussian Mixture Model (GMM) [2] and Difference of Gaussian (DoG) [3] is devised to select POI, wherein GMM is responsible for removing the background of an input image (first screening), whereas DoG for filtering noisy or false contours (second screening). Experimental results indicate that a great amount of computational load can be saved, in comparison with the use of Canny detector in our previous work [4]. Since the perceptual edges are useful for forming a complete object contour corresponding to the human visual perception, the results of this paper can be potentially cooperated with other more advanced object detection methods such as the deep learning-based SSD [5] to achieve the same effect as the human visual system in dealing with obscured or corrupted input images, whereby even if a target object is occluded by other objects or corrupted by rainy water, it can be still identified correctly, this feature should greatly enhance the operational safety of unmanned vehicles, unmanned aircraft and other autonomous systems.

Keywords Perceptual Edges, Gestalt, Bayesian, GMM, Deep Learning

task to computer or machine vision researchers, human vision not only pays special attention to edges of an object of interest, but also to the visual range of information that can be organized into formation of a self-sufficient and balanced entity [6]. This special property is formally explained by the so called *Gestalt theory*. Human visual system has a peculiar ability to deal with obscured or corrupted input images. For example, human visual system can quickly to compile complex scene information into simple object contours and perceive objects such as the woman body in Fig.1 (a), with its corresponding ground truth shown in Fig.1(c). Therefore, it is desirable to embody this capability of depicting perceptual edges on a computing device, so as to enable a computing machine to “perceive” objects such as the woman body in Fig.1(c). In computer vision, perceptual edges form the basis of constructing a complete contour which corresponds to the human visual perception.

To understand our motive, the result of applying the Canny detector [1] to Fig.1(a) is shown in Fig.1(b), one can see overly fragmented or minute edges are undesirably preserved, making virtually any linking or grouping algorithms infeasible to construct the woman body contour behind the water screen. Clearly, using Canny detector is not very effective in preserving features essential for our purpose. This paper aims to provide a novel approach for preserving perceptual edges representing boundaries of objects as perceived by human eyes.

In our previous work of P-EdGE [4] the Canny detector was employed as a pre-process to select Pixels of Interest (POI). However, the Canny detector tends to generate too many candidate points, a large portion of them are not actually the perceptual edges, which leads to a lengthy computation time and may incur unfavorable effects on subsequent processes. In order to solve this problem, we conjecture that, based on an observation that a static natural image often contains two main parts: foreground objects and background, it should be useful to perform *Background Subtraction* and divide image into foreground and background before identifying the perceptual edges. Dong this also avoids complex textures and chaotic background included in the input image which cause unnecessary computations, and allows us to focus on the task of perceptual edge detection.

1. Introduction

How human eyes classify an object presents an intriguing

Figure 1. (a) A complex image (b) Result of applying Canny detector to a (c) ground truth of (a) [4].

Figure 2. Flowchart of the proposed method

Our implementation of background subtraction consists of three stages: (1) applying Gaussian Mixture Model (GMM) to eliminate the background on the basis of histogram of color information of pixels (2) applying Difference of Gaussians (DoG) to remove textures, details and noises contained in the target object (3) subjecting the result of DoG to a final step of edge thinning. In other words, these three steps together replace the Canny edge detection in selecting POI.

2. Methodology

The flowchart of the proposed method is shown in Fig. 2, which is divided into three stages as aforesaid. The proposed method is characterized in that GMM, DoG and edge thinning are incorporated to remove background and eliminate textures and noises contained in the target object, thereby improving computation efficiency, detection sensitivity and accuracy, as well as reducing false alarm.

A. Remove the Background by GMM (Gaussian Mixture Model)

The idea of GMM is rooted from the observation that a seemingly unique-color block may actually contain pixels of different intensity values, this phenomenon in color distribution is in line with the characteristics of Gaussian distribution, thus an arbitrary input image should consist of

a certain number of Gaussian distributions. Most filtering methods are derived on a pixel-by-pixel basis, in contrast, Javed et al. [7] proposed a hierarchical background filtering method, which uses color and gradient information to establish a GMM background model, their method is effective when the color difference between the background and objects is obvious. Thus, we will apply GMM in this work to filter out the background portion of the input image. We start by letting the three color components R、G and B representing the feature vector to establish a GMM for each pixel, the k Gaussian distribution of a pixel (i, j) at time t is given by Eq.(1):

$$g(x_{i,j}^t | m_{i,j}^{t,k}, \sum_{i,j}^{t,k}) = \frac{1}{(2\pi)^{n/2} |\sum_{i,j}^{t,k}|^{1/2}} \exp \left[-\frac{1}{2} (x_{i,j}^t - m_{i,j}^{t,k})^T (\sum_{i,j}^{t,k})^{-1} (x_{i,j}^t - m_{i,j}^{t,k}) \right] \quad (1)$$

$x_{i,j}^t$ is the color vector of a pixel (i, j) , $m_{i,j}^{t,k}$ and $\sum_{i,j}^{t,k}$ are the averaged vector and the variation matrix, where the three colors are independent of each other. Therefore, $\sum_{i,j}^{t,k} = \text{diag}[(\sigma_{i,j}^{t,k,R})^2, (\sigma_{i,j}^{t,k,G})^2, (\sigma_{i,j}^{t,k,B})^2]$. In our work, since a static image is only a special case of Javed et al. [7], namely, t is simply set as fixed parameter. Fig. 3 shows an example of the background removal effect.

Figure 3. (a) input image, (b)Canny edge detection of (a), (c) remove the background by GMM and use Canny edge detection of (a).

B. Difference of Gaussians (DoG)

The idea of using DoG in this work is rooted from the Scale Invariant Feature Transform (SIFT) first proposed by Lowe *et al.* [3, 8]. Basically SIFT uses DoG to calculate extremum in scale space, and extracts a *key point* as the local feature of a pattern, and the gradient values of pixels in a region surrounding the *key point* are combined with *key point* to form a unique "feature descriptor".

Gaussian functions with various different standard deviations (σ) are respectively convolved with the input image to produce plural layers (each corresponds to an output image) as shown on the left part of Fig. 4, and then from top to down every two adjacent output images are subjected to a subtraction operation for producing a difference of Gaussian image. The resultant effect as a whole is equivalent to a band-pass filter, which can delete the frequency components other than the frequency components to be reserved. Most edge detectors enhance both high-frequency signals and noises. The characteristic of DoG is that it can effectively remove noises of high frequency, which meets our requirement of filtering noises while preserving edges.

The zero-mean Gaussian function $G(x, y, \sigma)$ is described by Eq.(2). The larger of σ value, the smoother Gaussian convolution result. In Eq.(3), $L(x, y, \sigma)$ is the convolution of $G(x, y, \sigma)$ with the input image $I(x, y)$, which is the first layer of the pyramid in the bottom of the left half of Fig. 4. The parameter σ is multiplied by a larger ratio for layers from the second layer to the fifth layer to produce higher degrees of Gaussian blur image. For example, the second, third layer, fourth, and the fifth layer are multiplied by k, k^2, k^3, k^4 . Each layer in the right half (Gaussian difference scale space) of the pyramid in Fig. 4 is obtained in order by subtracting two adjacent layers at the left half, wherein the first DoG layer is prescribed in Eq.(4). All simulation results in this paper use the first DoG layer only.

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

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (3)$$

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (4)$$

Figure 4. The left half of the Gaussian scale space and the right half of the Gaussian difference scale space [3].

Figure 5. (a) The Gaussian blur of the parameter σ , (b) The Gaussian blur of the parameter $k\sigma$, (c) Gaussian difference results from (a) and (b).

Figure 6. (a) applying GMM and DoG to Fig. 3(a), (b) result of (a) after edge thinning.

Fig.5 (a) shows the Gaussian blur result of scale space $L(x, y, \sigma)$, Fig.5 (b) shows Gaussian blur result of scale space $L(x, y, k\sigma)$, Fig.5 (c) shows the Gaussian difference of (a) and (b).

C. Edge Thinning

The effect of performing edge thinning is shown in Fig. 6, where the result of applying GMM and DoG to Fig. 3(a) is shown in Fig. 6(a), one can see that wider lines are preserved. Since our purpose is to detect perceptual edges, subjecting these wider lines to an edge thinning process would reduce the total number of POI (see Fig. 6(b)) and hence the computational efficiency for the subsequent P-Edge.

3. Experimental Results

Experimental results are provided in this section to show the feasibility and effectiveness of our algorithm. First, the proposed method and Canny detector are used as the POI selection method. Table 1 and 2 list ROC (receiver operating

characteristic) comparison results for both POI selection methods.

Comparison (1): as shown in Fig.7(c), Canny edge detection produces an excessive number of edge lines. The corresponding result by applying P-EdGE to Fig. 7(c) is shown in Fig.7(d), we can see there are still too much edges left that do not match the ground truth in Fig.7(b). Fig. 7(e) is the result of applying our POI selection method to Fig.7(a), and the corresponding result by applying P-EdGE to Fig.7(e) is shown in Fig.7(f), in which a better preservation of perceptual edges than that of Fig. 7 (d), regardless within or outside the target object.

Comparison (2): Using Intel® Xeon 3.40GHz CPU and 8G RAM under Window 10, the total computation time of P-EdGE with Canny selected POI is 136.2s (0.9s for pre-processing). The total computation time of P-EdGE with our method selected POI is only 23.3s (10.9s for pre-processing). Thus, the number of pixels selected and hence the computation load of P-EdGE can be significantly reduced by using our method, even though our method is more time-consuming than Canny detector.

Figure 7. (a) input image, (b) the ground truth of (a), (c) applying Canny edge detection to (a), (d) applying P-EdGE to (c), (e) applying POI selection method to (a), (f) applying P-EdGE to (e).

Comparison (3): in order to show the consistency between experimental results and ground truth, the ROC (Receiver Operating Characteristic) is used to compare the performances, wherein TP, FP, TN, FN, represent True Positive, False Positive, True Negative, and False Negative, respectively. As shown in Table 1, the much smaller FN values in Fig.7(f) than in Fig.7(d) indicate that our proposed selection method has a better ability to filter out many cluttered edges or non-perceptual edges.

We further calculate the following three performance indicators, and show the result in Table 2.

- True positive rate (TPR), also call *hit rate*, as prescribed by Eq.(5). It represents the rate of all edge points of P-Edge matching with the ground truth.

- False positive rate (FPR), also call *false alarm rate*, as prescribed by Eq.(6).
- Accuracy (ACC), as prescribed by Eq.(7).

$$TPR = TP / P = TP / (TP + FN) \quad (5)$$

$$FPR = FP / N = FP / (FP + TN) \quad (6)$$

$$ACC = (TP + TN) / (P + N) \quad (7)$$

Our method has a much higher TPR than Canny detector, which means our method is able to pick out POI that are more similar to ground truth. Both selection methods have almost the same *false alarm* performance. Another noteworthy point is that the TP value of Fig.7(f) slightly

smaller than that of Fig.7(d) (1502 vs. 1759), this is because the Canny detector produces more complete edge lines (both for background and noise) than our method does. In addition, Difference of Gaussians tends to generate disconnect edges, which in part explains why the result is not perfectly consistent with the ground truth. Finally, our method performs slightly better in terms of ACC, which is mainly attributed to the difference in FN values (192 vs. 2850), meaning our method is less likely to miss true edge points. It is concluded that our method, as a POI selection method, is more efficient than Canny detector.

To further verify effectiveness of our method, we also tested the popular benchmark dataset of PASCAL[9]. Since the dataset does not provide the ground truth corresponding to the human visual perception, herein we purposely used the Segmentation Object dataset in PASCAL as the contour ground truth. In addition, due to the various restrictions (such as multiple objects, color or size dissimilarity between background and objects not significant enough) that may deteriorate the performance of GMM in removing the background pixels, we selected from the PASCAL database 141 pictures (out of the total 1024 pictures of the 2010 & 2011 PASCAL) which contain no such restrictions, the test result is shown in Table 3. As can be seen, our method gives a much higher TPR value, which again verifies our method is more suitable than the Canny detector for use as a POI selection scheme.

Table 1. ROC performance comparison of our method and [4]

Performance evaluation of Fig. 7 (d) by comparing with Fig. 7 (b)	
TP=1759	FP=462
FN=2850	TN=149330
Performance evaluation of Fig. 7 (f) by comparing with Fig. 7 (b)	
TP=1502	FP=719
FN=192	TN=151988

Table 2. ROC and computing efficiency comparison

	using Canny detector[4]	Our method
TPR	38.1%	88.6%
FPR (False Alarm)	0.3%	0.4%
ACC	97.8%	99.4%
Total computation time	136.2s	23.3s

Table 3. Comparison results of PASCAL

	using Canny detector[4]	Our method
TPR	38.7%	74.3%
FPR (False Alarm)	0.00002%	0.00009%
ACC	97.6%	98.8%
Total computation time	142.6s	31.4s

4. Conclusions

In summary, conventional edge detection methods such as Canny detector do not distinguish foreground objects from the background. Thus, instead of using Canny detector the POI pre-process for the P-Edge, this work combine Gaussian Mixture Model and Difference of Gaussian to select POI. GMM is responsible for removing the background of an input image, whereas DoG for filtering noisy or false contours. Experimental results have shown that a great amount of computational load can be saved, in comparison with the use of Canny detector in our previous work [4].

Although GMM is quick in removing the background, but when the input image contains complex interlaced color features, using color distribution only is not enough to achieve good background removal effect. In the future we will try other removal background methods. For example, if ample training data and expensive GPU computing hardware are available, using deep learning to learn how to divide foreground and background could be more general and robust. We also note that all test results presented in this paper are conducted on still images, in video applications where the background or object division can be implemented simply by subtracting the current from the preceding frames, in this case it is believed that background detection will be easier and more accurate than the case of static images.

As to the edge thinning, in some cases we cannot obtain the desirable edges. As shown in Fig. 6(b), the edge points may not exactly fit the ground truth pixel positions due to the wider lines generated by DoG, which causes a dislocation problem in some edge's position, and solving this problem presents another worthy research direction in the future.

REFERENCES

- [1] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, pp. 679-698, 1986.
- [2] Javed, O., Shafique, K., and Shah, M., "A hierarchical approach to robust background subtraction using color and gradient information," *Motion and Video Computing, 2002. Proceedings. Workshop on. IEEE*, pp. 22-27, 2002.
- [3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, Vol. 60, no. 2, pp. 91-110, 2004.
- [4] C. S. Tseng, and J. H. Wang, "Perceptual edge detection via entropy-driven gradient evaluation," *IET Computer Vision*, Vol. 6, no. 2, pp. 163-171, 2016 (SCI).
- [5] W. Liu, et al, "SSD: Single Shot MultiBox Detector," *European Conference on Computer Vision*. Springer International Publishing, 2016.

- [6] R. Arnheim, "The power of the center: A study of composition in the visual arts," University of California Pr, 1983.
- [7] O. Javed, K. Shafique, and M. Shah, "A hierarchical approach to robust background subtraction using color and gradient information," *Motion and Video Computing, 2002. Proceedings. Workshop on*, pp. 22-27, 2002.
- [8] D. G. Lowe, "Object recognition from local scale-invariant features," *The proceedings of the seventh IEEE international conf. on Computer vision*, Vol. 2, pp. 1150-1157, 1999.
- [9] M. Everingham, L. Van Gool, C. K. Williams, J. Winnm, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International journal of computer vision*, Vol. 88, no. 2, pp. 303-338, 2010.