Color Image Segmentation Combining Rough Depth Information

Wen Su[1,2], Jing Qian[1,2], Zhiming Pi[1], and Zeng-Fu Wang[1,2(✉)]

[1] University of Science and Technology of China, Hefei, Anhui, China
`zfwang@ustc.edu.cn`
[2] Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, Anhui, China

Abstract. A novel color image segmentation method is presented in this paper. Firstly a *Luv* color histogram based method is used to estimate the color bandwidth, then a mean shift algorithm with adaptive color bandwidth is employed to pre-segment the input image. Next, a boundary detection algorithm based machine learning is used to calculate the probability boundary of objects from both depth and color information. Then, a correction procedure is performed by mapping the depth boundary onto the color image. Finally, Graph cut is used to segment color image based on Gaussian Mixture Model which is built with the above pre-segmentation and correction results. The experimental results show that the segmentation algorithm is an effective one. It can effectively segment an image into some semantic objects.

Keywords: Image segmentation · Depth map · Adaptive mean shift · Gaussian mixture model · Graph cut

1 Introduction

The use of depth information has been the subject of a number of vision-related tasks such as image segmentation and retrieval. For the task of image segmentation, depth information has obvious advantages as compared with color information. Firstly it is invariant to lighting and/or texture variation; secondly it is invariant to camera pose and perspective change. Therefore using depth can potentially enable successful segmentation independent of illumination or view, significantly expanding the range of operation conditions. In this background, the color image segmentation combining depth information has received much attention.

Recently, many efforts have been made. Crabb et al. employed depth map to extract foreground objects in real time[1]. In their segmentation the color image is only used to support a small fraction (1%-2%) of the pixels which are not solved by the depth threshold. Another interesting use of depth is implemented in [2]. Several channels of the depth camera are used and combined with the color channel to assist background subtraction. A new indoor scene dataset, completing with accurate depth maps and dense label coverage, is introduced in

[3] by Nathan Silberman et al. Their model which is evaluated on this dataset is inclined to solve the indoor image segmentation. Meir Johnathan Dahan et al. present a technique [4] which is the most relevant work with ours. However, their results depend on the color image segmentation method greatly.

In this paper we segment image with an improved mean shift algorithm with adaptive color bandwidth. Then we strengthen segmentation results combining color and depth effectively. This algorithm can weaken the illumination changes and object shadow effects on image segmentation. In particular, it can reduce errors of segmentation in the neighboring semantic objects segmentation which have the same color.

The rest of this paper is organized as follows. Section 2 discusses our algorithm. Section 3 gives the experiments and discussions. Finally, we conclude the paper and give remarks in Section 4.

2   Algorithm

2.1   Preprocess

Before segmenting combining the information from depth map, several problems must be solved. Firstly the depth data spatial resolution does not always match the photometric resolution. Secondly in many cases the depth image is imperfect and contains numerous artifacts. As a result, we preprocess the depth map with the cross-bilateral filter [5] and the joint bilateral filter [7] before combining it in the color image segmentation.

2.2   Adaptive Mean Shift Image Segmentation

Since the spatial resolution of color images is often higher compared with the depth map, the color image can provide more accurate information such as the outline of the semantic objects, color, texture and structure. Therefore, the image segmentation algorithm should be a color based one. We select the Mean Shift algorithm [8–11] for the purpose of image segmentation.

Let $C_i : (L_i, u_i, v_i)$ and $C_j : (L_j, u_j, v_j)$ be two points of the CIE LUV [12] color space respectively, then the Euclidean distance of the two points (that is, the two colors) is given by

$$disc_{ij} = \sqrt{(L_i - L_j)^2 + (u_i - u_j)^2 + (v_i - v_j)^2} \tag{1}$$

It is well known that the bandwidth parameters are important parameters that determine the step length and direction in an iterative process of Mean Shift algorithm, and therefore affect the final result of segmentation. The existing methods of bandwidth parameter determination can be found in [13]. In this paper we develop an adaptive bandwidth parameter determination algorithm. The details are as below.

In our application, the dimension of feature space for image segmentation is equal to five: two for spatial dimension, and three for color dimension. For simplicity, the corresponding bandwidth Matrix can be expressed by

$$H = diag(h_x, h_y, h_l, h_u, h_v)$$

where $h_s = h_x = h_y$ is spatial bandwidth, and $h_c = h_l = h_u = h_v$ is color bandwidth. The two spatial bandwidths mainly affect the computation speed of the algorithm, while the three color bandwidths heavily affect the segmentation result. Due to the fact above, we only consider the problem of color bandwidth selection when the two spatial bandwidths are given.

We have found that different images have different optimal color bandwidth. In our adaptive color bandwidth determination algorithm, the color bandwidth h_c can be obtained by the steps as below:

(1) Calculate LUV values of all pixels in image, and obtain the normalized LUV repre-sentation of image shown in Fig.1.
(2) Divide the color cube into $N_l * N_u * N_v$ equal parts (as shown in Fig.1, each of them is a rectangular solid and corresponds to a LUV color cluster or mode), count the number of pixels falling in each rectangular solid, and then create the 3D color histogram of the image according to the statistics. Here N_l, N_u, and N_v are segment numbers along l-axis, u-axis, and v-axis of color space *Luv* respectively.
(3) Find all local peaks as follows: check each rectangular solid mentioned above, if the number of pixels falling in a rectangular solid is bigger than the number of pixels falling in other rectangular solid surrounding it and meanwhile is bigger than the threshold set by system, then consider the rectangular solid to be a local peak (the black points in Fig.2, note and the number of pixels falling in the rectangular solid is called the value of the local peak) and save the corresponding information of location and number of pixels respectively. As a result, the local peaks detected are ranked in descending order.
(4) Compare the values of each local peaks with m threshold values Th_i, $i = 1, 2, m$ set by system, and count the number p_i, $i = 1, 2, m$ of local peaks, whose values are larger than the corresponding threshold values Th_i, $i = 1, 2, m$. These local peaks are called effective peaks. It is obvious that the bigger the threshold value Th_i is, the less the number p_i is. The colors corresponding to the above effective peaks consist of the main color clusters of image.
(5) Determine the color bandwidth h_c as below.

$$h_c = w * disC_{top} + (1 - w) * disC_{val} \qquad (2)$$

where, $disC_{top}$ is the mean Euclidean distance between arbitrary two local peaks within the range of k bigger local peaks that is given by

$$disC_{top} = \frac{1}{N_{top}} \sum_{i<j}^{k} (n_i + n_j) disC_{ij}, N_{top} = k \sum_{i=1}^{k} n_i$$

Here, $disC_{ij}$ is defined by (1), and n_i is the value of the i_{th} biggest peak. Similarly, $disC_{val}$ is the mean Euclidean distance of all effective peaks to their center:

$$disC_{val} = \frac{1}{m}\sum_{i=1}^{m} w_{vi}(\frac{1}{p_i}\sum_{j=1}^{p_i} disC_{ic}), w_{vi} = \frac{p_i}{\sum_{i=1}^{m} p_i}, \sum_{i=1}^{m} w_{vi} = 1$$

where, $disC_{ic}$ is the Euclidean distance of an effective peak corresponding to threshold values Th_i to the center of all effective peaks.
Besides, w is a weight defined by

$$w = \frac{\sum_{i=1}^{k} n_i}{\sum_{i=1}^{m} p_i}$$

Finally, in our experiments, the parameter k is set to be 8.

It can be seen from (2) that the color bandwidth h_c is selected to be dependent upon the values of $disC_{top}$ and $disC_{val}$, and the weight of w in our system.

After selection of color bandwidth h_c, we should select a proper kernel function for the iterative process of Mean Shift algorithm. In our system, we use the Gauss kernel function [14] for the purpose.

In this way, we can obtain adaptive color bandwidth based segmentation result. Fig.4 (c) shows the segmentation result to an example image by improved Mean Shift algorithm of ours based on adaptive color bandwidth determination.
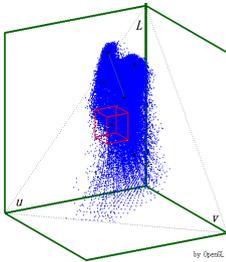
Fig. 1. Adaptive Bandwidth
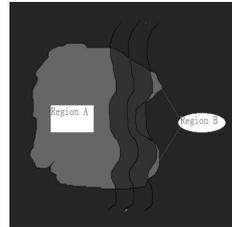
Fig. 2. Probability boundary calculation and dilation

2.3 Probability Boundary with Depth Discontinuities

Adaptive Mean Shift image segmentation algorithm as well as general image segmentation method depend on the color information greatly. Therefore, if the color of semantic object is similar while depth is discontinuities (occlusion) or the depth is continuous while objects have different colors, the accuracy of segmentation algorithm is reduced. Depth information give us insight into solving this problem. In general, the boundary with depth discontinuous correspond to

the boundary of two semantic objects. We need to find reliable boundary with depth discontinuities, According to the boundary we correct segmentation results coming from adaptive Mean Shift image segmentation algorithm.

We apply a robust probability boundary with depth discontinuities algorithm combining depth map and color image. The algorithm is based on the probability boundary detection algorithm [6]. The main steps include: a) Applying of probability boundary detection algorithm to calculate the depth map and color image probability boundary (Fig.3 (c)(d)), here the probability boundary map refers to a map which each pixel in is a probability of being part of edge.). b) Two probability map are fused so as to correcting the probability boundary map of depth map based on the probability boundary map of color image to obtain a more reliable probability boundary map of depth. c) The thresholds are set on the probability boundary map of depth to extract reliable depth jump boundary. d) Obtain the binarization result of probability boundary of depth map.

A probability boundary algorithm combining depth map and color image

---

a) calculating the probability boundaries map of color image $P_{color}(x)$ and the probability boundaries map of depth map $P_{depth}(x)$;
b) fusing the two map $P(x) = P_{color}(x) * P_{depth}(x)$;
c) extracting a more reliable probability boundaries:

$$E_{anchor}(x) = \begin{cases} 1, & p(x) > Th \\ 0, & p(x) \leq Th \end{cases}$$

where Th is the threshold;
d) binarized probability boundaries map of depth:

$$E_{depth}(x) = \begin{cases} 1, & p_{depth}(x) > Th \\ 0, & p_{depth}(x) \leq Th \end{cases}$$

$$E(x) = \begin{cases} 1, & E_{anchor}(x) \bigcup \{E_{depth}(x) \xrightarrow{\text{connect to}} E_{anchor}(x)\} \\ 0, & \text{other} \end{cases}$$

Experimental results show that our algorithm performed better than the algorithm which depends on the depth information exclusively. Because we obtain the probability boundary of depth with the introduction of color, texture, and other information.

2.4   Graph Cut Based on GMM Modeling

Mean Shift algorithm separate different objects into patches according to the color information. However, if a patch has similar color while depth is discontinuous, then the patch is not belong to same object. Or, if there are two patches
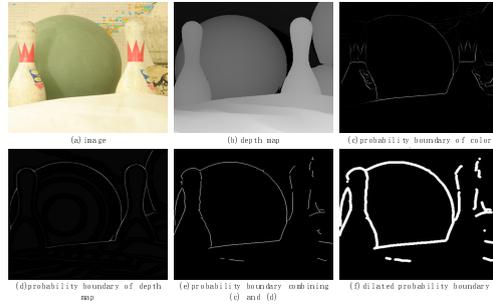
Fig. 3. Adjacency pattern analysis

whose color is different while depth is continuous, then the two patches should belong to the same object. The former should appear depth discontinuity probability boundary in a patch, while the latter should not appear depth discontinuity probability boundary at the location of the two patch adjacenting in the depth map. As a result, we can correct segmentation depending on depth discontinuities probability boundary.

We map reliable depth discontinuities probability boundary to the color image segmentation. Because of the factors such as noises and the error that acquisition equipment cause, depth discontinuous probability boundary may not be coincide with the boundary of segmented regions. We apply depth boundary dilation operation on the probability boundary, and dilate the boundary with δ pixels(we take δ as 3 in the experiment), as shown in Fig.3 (f).

After mapping, different labels is set to both sides of the dilated probability boundary. For the segmented regions which has intersection over the both sides of dilated depth discontinuous probability boundary, the region area which calculated from the difference set is counted, and the other region area which is neighbor to both sides is counted as well. We label the regions through comparing the size of two area. The difference set between the segmented regions and the dilated depth discontinuous probability boundary has three adjacency patterns with the dilated depth discontinuous probability boundary. One is that it is neighbor to one side of dilated depth discontinuous probability boundary only. Then we label the segmented region with the neighboring boundary. Another one is that it has an intersection over both sides of the dilated depth discontinuous probability boundary while the intersection on the same side is discontent (such as B). As shown in Fig.2. Then we count the area of the neighboring regions on the two sides separately and label the region with the bigger area label. The last one is that the difference set is a single connected region. We can not judge the label so that we do not set the label temporarily.

After getting the corrected segmentation which we call as seed labels, as shown in Fig.4 (d), we have translate the segmentation problem to labeling the region which we cannot judge above.It has many possibilities because there are lots of seed labels in the image already. In order to solve the problem, a graph cut

optimization based on Gaussian Mixture Model (GMM) is implemented, similar to [4]. We have a relatively reliable initial solution to the following graph cut optimization. The seed labels defines $|L|$ layers in the depth. These layers are built from objects, parts of objects, and some objects joined together. Mapping the seed labels to the depth map, corresponding depth regions are obtained. Each region is considered to be the seed for a label l in the segmentation. Hence, $|L|$ is the number of labels used in the graph-cut procedure. For each of the $|L|$ segments, a GMM, similar to [14], is fit. We use two Gaussians for each modality: two for the color image $pc^l(\bullet)$ and two for the depth image $pd^l(\bullet)$. These models are used to define the distance between elements in the definition of the energy function of the graph-cut segmentation. We use the regular construction of the graph G: every pixel in the image is a node v in the graph, and we use 4-connectivity for the edges e which are called n-links. Every node v is also connected with edges, which are called t-links, to the terminal nodes t_l.

After the calculation of a minimum cut on the graph, we are left with a labeling assignment for unlabeled pixels, which leads directly to a minimization of the energy function defined (3). The energy function is consisted of two terms. The first one is data term, which is refers to the t-links. The second one is smooth term, which is refers to the n-links.

$$E(X) = \sum_p E_d(x_p) + \sum_{p,q} E_s(x_p, x_q) \tag{3}$$

The two types of energy terms fit the two types of edges in the graph: the smoothness term as the n-link and the data-term as the t-link. We assign the weights of the two types of links as (4) and (5).

$$E_d(x_p) = max(\alpha \cdot L_p^{cl}, L_p^{dl}) \tag{4}$$

$$E_s(x_p, x_q) = (max(\beta \cdot \mu \parallel I_p - I_q \parallel, \varphi \parallel D_p - D_q \parallel))^{-1}) \tag{5}$$

In the data term, α is the weight which is used to adjust the ratio between the color information and depth information. L_p^{cl} and L_p^{dl} are defined as (6) and (7). $pc^l(\bullet)$ and $pd^l(\bullet)$ are the GMM models for each l, for color and depth, respectively. I_p and D_p are the color and depth values at pixel p, respectively. We merge the color and depth channels by using the maximum of their distance to the respective model. This way fitting each pixel gives lower energy for a good match. The actual weights on the t-links are calculated using the negative data term.

$$L_p^{cl} = -ln(pc^l(I_p)) \tag{6}$$

$$L_p^{dl} = -ln(pd^l(D_p)) \tag{7}$$

In the smooth term, we also combine the color information and the depth information. The n-links is refers to the similarity between the two connected node in the graph. We calculate the similarity from the color image and depth map separately and set the weights as the higher one. In this term, the β is the

weight either. $\mu \parallel I_p - I_q \parallel$ and $\varphi \parallel D_p - D_q \parallel$ are expectation operators over the whole color and depth map image, respectively.

By running the minimum cut algorithm [15] [16] [17], each pixels of color image are assigned a label to determine which segmentation the pixel belongs to. Finally we get the optimal segmentation of the color image.

3 Experiments

We used the images with four different scenes to make the experiments of segmentation. The experimental results as shown in Fig.4 indicate that the algorithm presented in this paper has a good performance. The results are influenced by the lighting or texture variation more slightly. We can effectively segment objects from an image, especially in the case of two objects of the same color are mutual supported. This paper uses three segmentation evaluation: PR [18], VI [19] as well as GCE [20]. PR refers to the proportion of consistent pixels between tested segmented image and the ground truth (The Probabilistic Rand Index). The Variation of Information (VI) metric defines the distance between two segmentations as the average conditional entropy of one segmentation given the other, and thus roughly measures the amount of randomness in one segmentation which cannot be explained by the other. The Global Consistency Error (GCE) measures the extent to which one segmentation can be viewed as a refinement of the other. Segmentations which are related in this manner are considered to be consistent, since they could represent the same natural image segmented at different scales. The experiments indicates that our algorithm has improvement on most of three indicators as shown in Tab.1.

(a) Original Image   (b) Depth map   (c) Adaptive Mean Shift   (e) Corrected segmentation   (d) Our method   (e) Ground truth
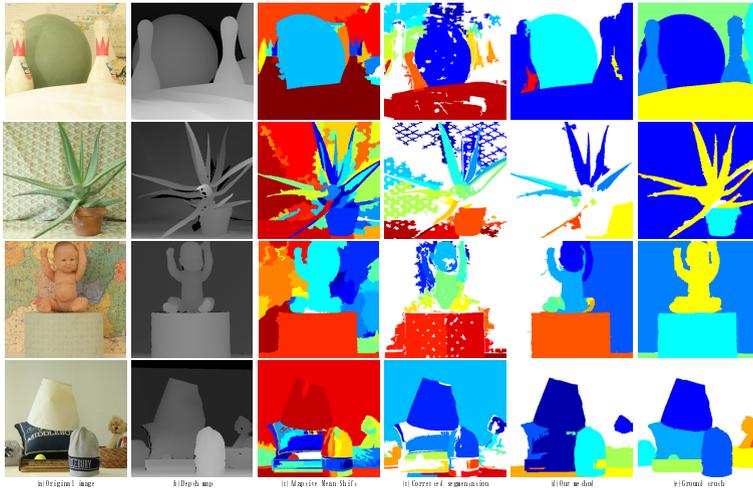
Fig. 4. Experiments results

Table 1. Three kinds of evaluation indexes

PR

	A	B	C	D	mean
Mean Shift	0.6448	**0.7303**	0.7955	0.6401	0.7027
Graph cut	0.7965	0.5686	0.6744	0.6397	0.6631
Adaptive Mean Shift	**0.8899**	0.4641	**0.9190**	0.7060	0.7448
Our Method	0.8118	0.6618	0.8611	**0.8080**	**0.7857**

VI

	A	B	C	D	mean
Mean Shift	1.8496	**1.3086**	2.0168	1.8248	1.7500
Graph cut	1.6316	2.0625	2.4687	1.9013	2.0160
Adaptive Mean Shift	1.5194	4.2297	1.6564	2.3827	2.4471
Our Method	**1.3175**	1.7850	**1.2917**	**1.1823**	**1.3941**

GCE

	A	B	C	D	mean
Mean Shift	0.1222	0.1882	0.2656	0.3786	0.2387
Graph cut	0.2085	0.3094	0.3931	0.4006	0.3279
Adaptive Mean Shift	0.2592	**0.1101**	**0.0668**	0.0619	**0.1245**
Our Method	**0.1089**	0.2784	0.1802	**0.0562**	0.1560

4   Conclusion

This paper presents a novel segmentation algorithm combining both color image and depth map. The experimental results show that the segmentation algorithm is an effective one. It can effectively implement segmentation with less influence from lighting or texture variation. The performance of the algorithm is partly dependent upon the image pre-segmentation result. When the scene have a great change on the lighting or the color and depth are similar in the same time, the algorithm may give a false result. The future work is to address the problem and find the methods of overcoming the difficulties.

References

1. Crabb, R., Tracey, C., Puranik, A., et al.: Real-time foreground segmentation via range and color imaging. In: Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–5 (2008)
2. Leens, J., Piérard, S., Barnich, O., Van Droogenbroeck, M., Wagner, J.-M.: Combining color, depth, and motion for video segmentation. In: Fritz, M., Schiele, B., Piater, J.H. (eds.) ICVS 2009. LNCS, vol. 5815, pp. 104–113. Springer, Heidelberg (2009)
3. Silberman, N., Fergus, R.: Indoor scene segmentation using a structured light sensor. In: proc. of IEEE International Conference on Computer Vision, pp. 601–608 (2011)

4. Dahan, M.J., Chen, N., Shamir, A., et al.: Combining color and depth for enhanced image segmentation and retargeting. The Visual Computer **28**(12), 1181–1193 (2012)
5. Paris, S., Durand, F.: A fast approximation of the bilateral filter using a signal processing approach. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 568–580. Springer, Heidelberg (2006)
6. Martin, D.R., Fowlkes, C.C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(5), 530–549 (2004)
7. Riemens, A.K., Gangwal, O.P., Barenbrug, B., et al.: Multistep joint bilateral depth upsampling. In: International Society for Optics and Photonics, pp. 72570M–72570M-12 (2009)
8. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory **21**(1), 32–40 (1975)
9. Cheng, Y.: Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence **17**(8), 790–799 (1995)
10. Comaniciu, D., Meer, P.: Mean shift analysis and applications. In: Proc. of IEEE International Conference on Computer Vision, vol. 2, pp. 1197–1203 (1999)
11. Georgescu, B., Shimshoni, I., Meer, P.: Mean shift based clustering in high dimensions: a texture classification example. In: Proc. of IEEE International Conference on Computer Vision, pp. 456–463 (2003)
12. Cheng, H.D., Jiang, X.H., Sun, Y., et al.: Color image segmentation: advances and prospects. Pattern Recognition **34**(12), 2259–2281 (2011)
13. Comaniciu, D., Ramesh, V., Meer, P.: The variable bandwidth mean shift and data-driven scale selection. In: Proc. of IEEE International Conference on Computer Vision, vol. 1, pp. 438–445 (2001)
14. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics (TOG) **23**(3), 309–314 (2004)
15. Zabih, R.D., Veksler, O., Boykov, Y.: System and method for fast approximate energy minimization via graph cuts. U.S. Patent No. 6,744,923 (2004)
16. Kolmogorov, V., Zabin, R.: What energy functions can be minimized via graph cuts? IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(2), 147–159 (2004)
17. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(9), 1124–1137 (2004)
18. Pantofaru, C., Hebert, M.: A Comparison of Image Segmentation Algorithms technical report. Robotics Inst., Carnegie Mellon Univ. (2005)
19. Meila, M.: Comparing clusterings: an axiomatic view. In: Proc. of International Conference on Machine Learning, pp. 577–584 (2005)
20. Martin, D., Fowlkes, C., Tal, D., et al.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. of IEEE International Conference on Computer Vision, vol. 2, pp. 416–423 (2001)