A Close-Form Iterative Algorithm for Depth Inferring from a Single Image

Yang Cao, Yan Xia, and Zengfu Wang

Automation Department, University of Science and Technology of China,
Jinzhai Road 96, Hefei, China
forrest@ustc.edu.cn, xiayan@mail.ustc.edu.cn, zfwang@ustc.edu.cn

Abstract. Inferring depth from a single image is a difficult task in computer vision, which needs to utilize adequate monocular cues contained in the image. Inspired by Saxena et al's work, this paper presents a close-form iterative algorithm to process multi-scale image segmentation and depth inferring alternately, which can significantly improve segmentation and depth estimate results. First, an EM-based algorithm is applied to obtain an initial multi-scale image segmentation result. Then, the multi-scale Markov random field (MRF) model, trained by supervised learning, is used to infer both depths and the relations between depths at different image regions. Next, a graph-based region merging algorithm is applied to merge the segmentations at the larger scales by incorporating the inferred depths. At the last, the refined multi-scale image segmentations are used as input of MRF model and the depth are re-inferred. The above processes are iteratively continued until the expected results are achieved. Since there are no changes on the segmentations at the finest scale in the iterative process, it still can capture the detailed 3D structure. Meanwhile, the refined segmentations at the other scales will help obtain more global structure information in the image. The contrastive experimental results verify the validity of our method that it can infer quantitatively better depth estimations for 62.7% of 134 images downloaded from the Saxena's database. Our method can also improve the image segmentation results in the sense of scene interpretation. Moreover, the paper extends the method to estimate the depth of the scene with fore-objects.

Keywords: Depth inferring, monocular cues, image segmentation, Markov random field, scene reconstruction.

1 Introduction

Inferring 3D scene structure from a single image is an extremely challenging topic in computer vision, since it is an ill-posed problem in a mathematical sense and we can never know if the image is a picture of a painting or if it is a picture of an actual 3D environment. However, people have no difficulty to infer the scene structure from a single image. Here people utilize monocular depth cues to infer 3D information, which include some physical phenomenon

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part V, LNCS 6315, pp. 729–742, 2010.

and object characteristics, such as lighting and shading, perspective, occlusion, texture gradient and so on.

In recent works, researchers exploited some of these cues to obtain some 3D information from a single image. Saxena et al. [1,2,3,4,5] presented a Markov random field model for inferring depths from multi-scale monocular image features and applied the monocular depth perception to drive a remote-controlled car autonomously. Hoiem et al. [6,7,8] used texture and perspective cues to build pop-up models under a strong assumption that the scene consists of ground/horizontal planes and vertical walls (and possibly sky). Based on this, Hoiem et al. [9] also presented a closed form framework to integrate surface orientations, occlusion boundaries and objective identifications to develop a 3D scene understanding system. But the methods cannot be applied to the many scenes that are not made up only of vertical surfaces standing on a horizontal floor [10], such as mountains, trees, rooftops and so on.

In this paper, the goal is to propose a close-form iterative algorithm for improving the accuracy of depth inferring. In Hoiem et al's and Saxena et al's work, the depths of nature scene are approximately inferred from an over-segmentation of the image under the assumption that the 3D scene is made up of a number of small planes. This implies that image segmentation and depth inferring are inter-correlated. The image segmentations can help inferring the relations between the depths of different image regions. On the other hand, the depths can also be used as an additional attribute to improve segmentation results. Our algorithm utilizes this inter-correlated property and processes image segmentation and depth inferring alternately.

As mentioned in Saxena et al's work, local image features are insufficient to estimate the depth and multi-scale image features have to be used to capture more global properties. So we apply an EM-based multi-scale image segmentation algorithm to obtain the initial segmentation results. The image feature vectors extracted from the multi-scale segmentations are used to infer the different depth of each pixel in the image. The inferred depths are fed back and integrated with image segmentation into a cognitive loop. It is particularly noted that the depth inferring is regarding the segmentation regions at the finest scale, while the region merging is acting on the regions at the larger scales. This method will not decrease the number of the patches made up of 3D scene structure and can capture the rich detailed 3D scene structure. At the same time, the refined segmentations at the larger scales can offer more global structure information in multiple spatial scales which can be used to improve the accuracy of depth inferring. The above processes are iteratively continued until the expected results are achieved.

By using this close-form iterative framework, our algorithm can significantly improve the depth estimation results. Compared with the exiting methods, our algorithm can provide sharper depthmaps for 62.7% of 134 test images. The 3D flythrough reconstruct results using our algorithm are also a bit more visually pleasing. In additional, our method can improve the image segmentation results in the sense of scene interpretation.

Furthermore, we also consider the problem of depth inferring for scene with fore-objects. Under the assumption that the fore-objects lie vertical on the ground, the fore-objects regions are extracted from the image and the depth inferring of these regions are processed solely. After the other regions have also been processed, the depth estimations are incorporated together.

The remainder of this paper is organized as follows. Related works are reviewed in section 2. The overview of the proposed algorithm is introduced in section 3. The close-form iterative algorithm is described in section 4. Experimental results are shown in section 5. The depth inferring method for scene with fore-object is explained in section 6, before concluding in section 7.

2 Related Works

In some specific settings, monocular cues have been applied to perform the tasks of depth inferring from a single image. A number of researchers have studied the corresponding problems and proposed some effective methods including shape from texture (SFT) [11,12], shape from shading (SFS)[13,14] and tour into picture (TIP)[15]. Different from the geometric methods relied on feature matching and triangulation, such as stereo vision [16] and shape from motion [17], these methods use the cues contained in image to obtain rich 3D information. However, these methods often ignore the additional useful cues and enforce hard assumption that the scene structure is simple and uniform, thus they can only be applied in limited environment. For example, the TIP method can only be used in fully structured environment.

Recently, great progresses have been made in applying monocular cues to obtain 3D information. Based on the assumption that the environment is made of a ground-vertical structure, Delage et al. [18] and Hoiem et al. [6,7], built a simple pop-up 3D model from an image by classifying the image into horizontal/ground and vertical regions (also possibly sky). Delage considered indoor images, while Hoiem considered outdoor scenes. Based on these concepts, Hoiem et al. [10] and Sudderth et al.[19] integrated learning-based object recognition with 3D scene reconstruction; Hedau et al. [8] presented an algorithm to recover the spatial layout of cluttered room. Saxena et al. [1,2,4,5] presented an algorithm for inferring depth from monocular image cues. This algorithm was also successfully applied for improving the performance of stereovision [3] and autonomous navigation of remote-controlled car [20]. Heitz et al. [21] developed cascaded classification models (CCM) that combined a set of related subtasks of scene categorization, object detection and 3d reconstruction and these tasks can be solved in its own level and help each other. Hoiem et al. [9] regarded surface orientations, occlusion boundaries and objective identifications as intrinsic images and presented a closed form framework for interfacing scene analysis processes.

Our work seems like Heitz et al's and Hoiem et al's works for integrating the tasks of image segmentation and depth inferring. However, their works have strong leaning towards image understanding rather than depth inferring, and their algorithms contain many steps include object detection, region labeling and

so on. Moreover, their algorithms are based on iterative training which requires the knowledge of the implementation of each step, while our algorithm does not need retraining and is more flexible to be used in some specific applications such as robot navigation.

3 Overview of Our Algorithm

The overview of our proposed algorithm is illustrated in Fig. 1. There are three main modules, image segmentation, depth inferring, and region merging. Our input data are the multi-scale image segmentations, obtained by an EM-based algorithm at different scales. From these multi-scale segmentations, image feature vectors are first extracted through a template. Subsequently, a multi-scale Markov random field, trained by supervised learning, is used to model the relations between image feature vectors and the different depths of image regions at the finest scale. Then the inferred depths are fed back to incorporate the larger-scale image segmentations that are closed in 3D structure. Combined with the initial segmentation at the finest scale, the refined multi-scale segmentation results are obtained. The above processes are iteratively continued until the expected depth inferring results are achieved.

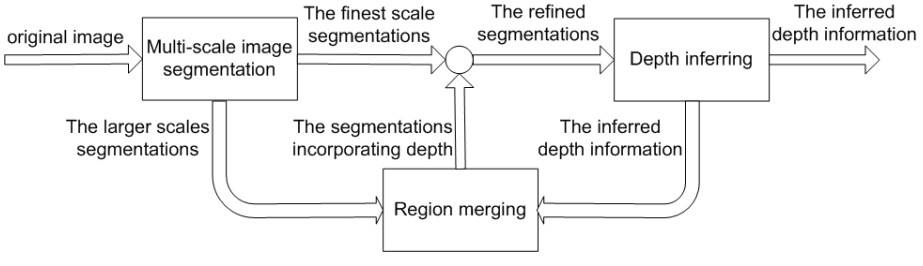

Fig. 1. Overview of our algorithm integrating depth inferring with image segmentation

The three modules are integrated in a cognitive loop. For each image, the region merging module receives initial segmentations and depth information from the other two modules and feeds back the refined multi-scale segmentations. Thus, the modules exchange information that helps compensate for their individual disadvantages and improves overall system performance. The pipeline of our algorithm is detailed in the following sections.

4 The Framework of Our Algorithm

4.1 Multi-scale Image Segmentation

Same as Hoiem et al's and Saxena et al's works, our algorithm also begins by segmenting the image into many such small planar surfaces. In order to capture

the depth cues directly from the local structure of the brightness pattern of a single monocular image, we use an expectation-maximization image segmentation algorithm [22,23] to obtain the initial segmentation results. The algorithm can offer an effective solution to bridge the gap from the low-level image features to surface reconstruction. Due to the extent of the inner-workings of the algorithm, we refrain from explaining in detail of the well-known algorithm, but limit the introduction to our specific application of the algorithm.

Creating multi-scale segmentation of an image involves three steps. (1) Select an appropriate scale for each pixel, and then extract color, texture and position features for that pixel at the selected scale. (2) Group pixels into regions by modeling the distribution of pixel features with a mixture of Gaussian using Expectation-Maximization. (3) Repeat the above two steps at multiple spatial scales.

In this image segmentation algorithm, the pixels are represented by the descriptor consists of eight values: three for color, three for texture and two for position. The three color components are the coordinates of Lab color space, which is approximately perceptually uniform and has the distances to be meaningful. The three texture components are polarity, anisotropy and contrast of each pixel, computed at the selected scale. The anisotropy and polarity are each modulated by the contrast since they are meaningless in regions of low contrast. The position of the pixel in the image, which can describe the spatial distribution, is also included in the feature vector.

Then, the Expectation-Maximization (EM) algorithm is applied to segment the pixels into patches. Since an image can be regarded as points in an eight-dimensional feature space after the process of feature extraction, the segmentation problem are transformed into dividing these points into groups. So the EM algorithm is actually used to determine the maximum likelihood parameters by assuming K Gaussian mixture model in the feature space. In order to avoid under-segmentations, we choose a rather large value of K, where K=256 for a 1024x768 size image in our experiment.

In order to capture more global structure properties from image, the segmentation algorithm are applied at three different scales (image resolutions, which are 1x, 3x and 9x of the original one in our experiments). The segmentations at the larger two scales are to be replaced by the refined ones after the region merging. An example result is shown as Fig.2.

Fig. 2. (a) Original image, (b)-(d) multi-scale image segmentation results

4.2 Depth Inferring

Feature Vector. In our algorithm, we choose the same features with Saxena et al's. There are two types of features: absolute features and relative features, which are used to estimate the absolute depth and relative depths respectively. As described in [2], a 17 dimensional template consists of 9 Laws'masks, 2 color channels and 6 texture gradients are used to compute summary statistics for a patch i at scale s in the image I. For the absolute depth feature, the outputs are incorporated to compute the sum absolute energy and sum squared energy. After including features from itself and its 4 neighbors at 3 scales and its 4 location features, the absolute feature vector x is $19 \times 34 = 646$ dimensional. For the relative depth features, a 10-bin histogram of each of the template output is computed, giving us a total of 170 features y_{is} for each patch i. Then the 170 dimensional relative depth features vector y_{ijs} for two neighboring patches i and j at scale s are computed as $y_{ijs} = y_{is} - y_{js}$.

Multi-scale Markov random model. The monocular depth cues of a particular patch are not only contained in this patch, but also can be captured from the relations between the patches which are adjacent at multiple spatial scales. Similar to Saxena et al's work [2,5], a hierarchical multi-scale Markov Random Field (MRF) is used to model the relationship between the depth of a patch and the depths of its neighboring patches. The model is formulated as,

$$P(d|X;\theta,\sigma)=\frac{1}{z}\prod_{i}^{K}\prod_{p_i}^{P_i=1} f_1(d_{i,p_i}|X_{i,p_i},\theta_r,\sigma_{1r})\prod_{s=1}^{3}\prod_{i}^{K}\prod_{j\in N_s} f_2(d_i(s),d_j(s)|y_{ijs},\sigma_{2r}). \tag{1}$$

where Z is the normalization constant for the model; K is the total number of patches in the image (at the lowest scale); with a total of P_t points in the patch i, $\mathrm{X}_{i,p_i} = \{\in R^{646}, p_i = 1,2,3\cdots P_i\}$ is the absolute depth feature vector for the point p_i in the patch i;$s = \{1,2,3\}$ is the 3 scales of image; $N_s(i)$ are the 4 neighbors of patch i at scale s; $\theta_r, \sigma_{1r}, \sigma_{2r}$ are the parameters of the model. The model consists of two terms, $f_1(\cdot)$ and $f_2(\cdot)$. The first term $f_1(\cdot)$ captures the relations between the depth d_{i,p_i} and the absolute feature X_{i,p_i} and it is formulated as,

$$f_1(\cdot) = \exp(-|d_{i,p_i}(1) - x_{i,p_i}^T\theta_r|\sigma_{1r}) \ . \tag{2}$$

The parameter σ_{1r} is modeled as a linear function of the features, which is $\sigma_{1r} = u_r^T x_{i,p_i}$. The second term $f_2(\cdot)$ captures the relations between the depths of patches which are adjacent at multiple spatial scales and it is formulated as,

$$f_2(\cdot) = \exp(-|d_i(s) - d_j(s)|\sigma_{2r}) \ . \tag{3}$$

In our algorithm, there are two constraints on the depths of patch i at the scale s. The first one is that the depths of patch i are the average of the depths of all of points in patch i.

$$d_i(s) = 1/P_i \sum_{p_i=1}^{P_i} d_{i,p_i}(s) \ . \tag{4}$$

The second one is that the depths at a higher scale are the average of the depths at the lower scale.

$$d_i(s+1) = 1/5 \sum_{j \in N_s(i) \cup i} d_j(s) \ . \tag{5}$$

Similar to the parameter σ_{1r}, σ_{2r} is modeled as $\sigma_{2r} = v_{rs}^T y_{ijs}$. In detail, different parameters (θ_r, u_r, v_r) are used for each row r in the image to learn the different statistical properties of different rows of image. Since the location features are also included in image segmentation and feature extraction, it can improve to detect some specific regions, such as sky and ground. For example, a blue region might represent sky if it is in upper part of image, and a green region might be more likely to be ground if in the lower part of the image.

Parameter Learning and MAP Inference. As described in [2], an approximate parameter learning of the model is made by using Multi-Conditional Learning (MCL). With $u_r \geq 0$ and $v_{rs} \geq 0$, the model parameters are estimated by solving a Linear Program (LP). After learning the parameters, the depth inferring problem is transformed into the MAP inference problem by maximizing (1) in terms of d. It can be seen that the first term in (1) models depth as an exponential function of multi-scale features of the points in the single patch i. The second term places a constraint that depends on the multi-scale relative features y_{ijs} on the depths, which plays a role to improve the accuracy of initial depth estimates. The MAP inference of the depth d_i can also be performed by solving a LP.

4.3 Region Merging

Region merging is the core part of our algorithm. As shown in Fig. 1, the inputs of region merging module are the inferred depth and the initial image segmentation results, and the outputs are the refined segmentations at the two larger scales. With this module, our algorithm can capture the strong interactions between the depths of patches which are not immediate neighbors. For example, consider the patches that lie on a large building, which are to be at similar depths. However, some adjacent patches are difficult to recognize as parts of the same object, since there are discontinuities in feature space (such as a window on the wall of a building). When the depth information is fed back, the adjacent patches tend to be incorporated and the discontinuities are eliminated. Then the depths of the patches will be highly correlated according to the MRF model.

As described above, each segmentation region represents a coherent region in the scene with all the pixels having similar properties. Thus, the 3D scene model is assumed to be made of a set of small planes. For ease of description, the basic unit of representation in the region merging module will be these small planes in the world. The relations between depths and the planar parameters are described as Fig.3. The planar surface on which a segmentation region lies is represented by using a set of plane parameters $\alpha \in R^3$, as described in [5]. The

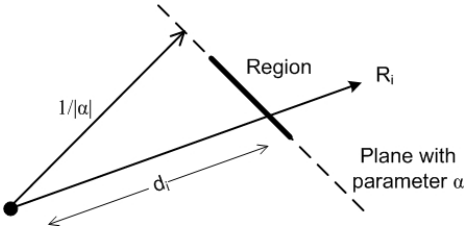

Fig. 3. Illustration of the relations between plane parameter α and the depth d of the point i (Cited from [5])

camera viewpoint is a two-parameter and is assumed to be constant. The value $1/|\alpha|$ is the distance from the camera center to the closest point on the plane, and the normal vector $\widehat{\alpha} = \alpha/|\alpha|$ gives the orientation of the plane. R_i is the unit vector from the camera center to a point i lying on a plane with parameters α. Then the planar parameter α can be computed by least square fitting according to R_i and the estimated depth d_i at the corresponding plane.

Then, the relations between two adjacent regions are weighted by the angle between the two planes on which the two regions lie. The weighted function $W(i,j)$ is defined as,

$$W(i,j) = \begin{cases} \theta_{ij} & if\ the\ region\ i\ is\ adjacent\ to\ region\ j \\ \infty & if\ the\ region\ i\ is\ not\ adjacent\ to\ region\ j \end{cases} . \tag{6}$$

$$\theta_{ij} = |\alpha_i/|\alpha_i| - \alpha_j/|\alpha_j|| . \tag{7}$$

where θ_{ij} is the angle between two planes i and j. A graph-based segmentation algorithm is used to realize region merging. The image is abstracted into an undirected weighted graph $G(V,E)$. V is the vertex set of G with its elements V_i representing the regions. E is the edge set of G with its elements $E_{i,j}$ representing the relations between two vertexes V_i and V_j. Based on the obtained graph $G(V,E)$, the region merging algorithm is performed as described in [24]. An example result is shown as Fig.4.

Fig. 4. (a) Image segmentation region at the largest region, (b) Initial depth reconstruction result,(c) Region merging result. (Best viewed in color)

5 Experiments

In order to verify the validity of our approach, we performed contrastive experiments that compare our algorithm with Saxena et al's [2,5],and Hoiem et al's work [10]. We downloaded 534 images+depthmaps from Saxena's homepage and used 400 for training model. The rest 134 images are used for quantitative comparison and the other 150 internet images are used for qualitative comparison.

We use relative depth error $|d - \widehat{d}|/d$ as the performance metric to decide which algorithm is quantitative better. We also perform a further qualitative comparison experiment that we ask a person to compare the three 3D fly through results, and decide which algorithm is qualitative better. The quantitative and the qualitative comparison results are shown in the Table 1. Since Hoiem et al's work is leaning towards surface reconstruction rather than depth inferring, the average relative depth errors are rather large, but the scene reconstruction results are more visual pleasing. Compared with Saxena et al's work, our method gives better relative depth accuracy for 62.7% of 134 images. Our algorithm also outputs visually better model in 35% of the cases, while Saxena et al's method outputs better model in 21% cases and Hoiem et al's work outputs better model in 35% cases (the rest cases are hard to decide).

Table 1. The quantitative and qualitative comparison results

Algorithm	Quantitative better	Average relative depth error	Qualitative better
Hoiem et al's	0.7%	4.055	35%
Saxena et al's	36.6%	0.400	21%
Our	62.7%	0.312	35%

The inferring depthmap compared with Saxena et al's work and ground truth are shown in Fig.5 and the typical scene reconstruction results are shown in Fig.6. As seen in the 3^{rd} image at the 2^{nd} row in Fig.5 and the 2^{nd} image at the 4^{th} row in Fig.6, the details of the distant region in image are arbitrarily to be reconstructed as a uniform one due to using depth information. Although the region merging is only acted on the regions at the larger scales in order to improve this, this situation still happens sometimes. On the whole, nevertheless, using the close form iterative framework yields better reconstruct results than before.

As a byproduct of our algorithm, the image segmentation results incorporated depth information are also obtained. The typical image segmentation results at the largest scale are shown in Fig.7. From the aspect of scene structure interpretation, the segmentation results get better and better after 1-3 iterations.

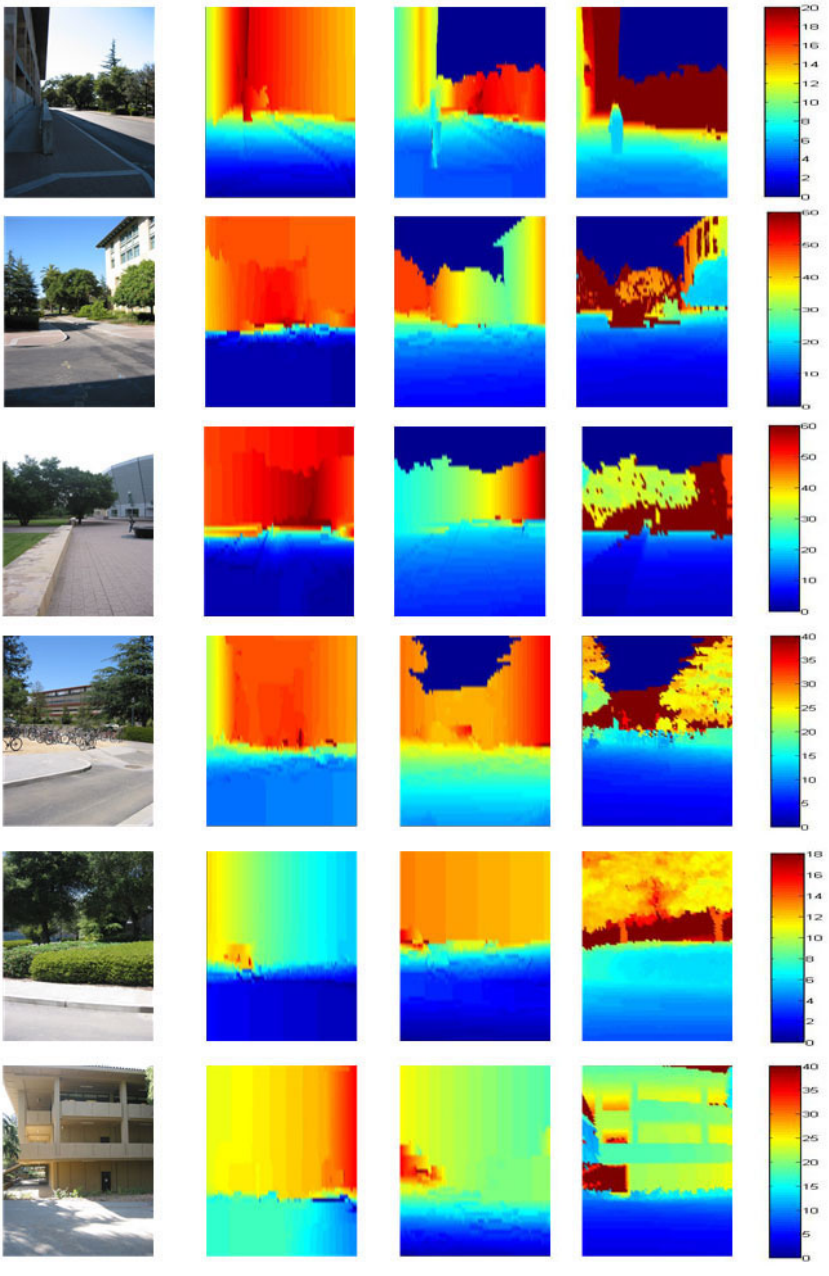

Fig. 5. Results for the predicted depthmap. The column a is original images, the column b is the results of Saxena's methods, the column c is the results of our methods, the column d is the groundtruth and the column e is the depth scale. The depths of sky regions in column c and d are denoted as zero. (Best viewed in color)

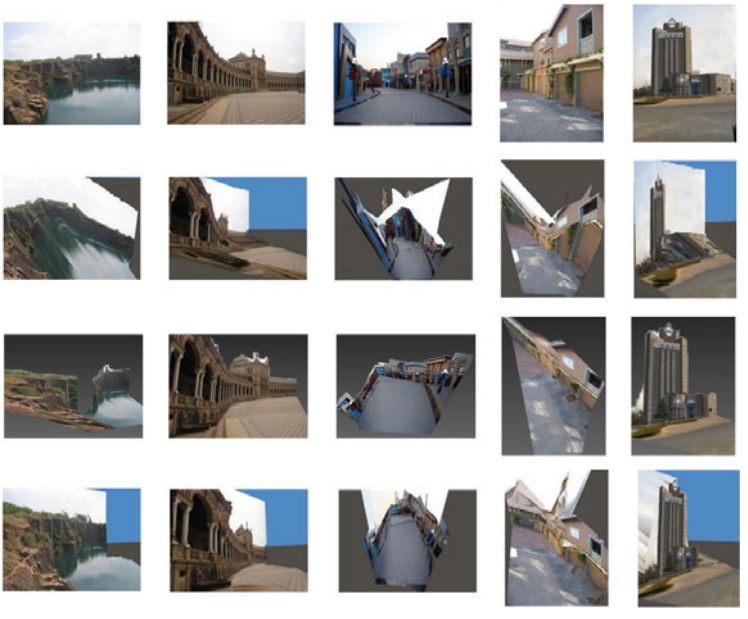

Fig. 6. Results for scene reconstruction. The 1^{st} row is original images, the 2^{nd} row is Saxena's scene reconstruction results, the 3^{rd} row is Hoiem's results and the 4^{th} row is our results.(Best viewed in color)

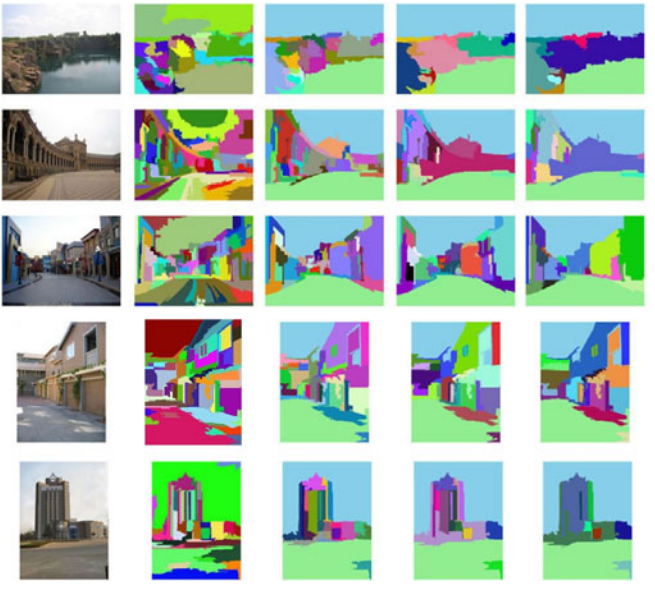

Fig. 7. Results for image segment. The $1st$ column is original image; the $2nd$ column is the initial segmentation under the largest scale;the $3rd$ -$5th$ columns show the obtained segmentation results after 1-3 iterations.(Best viewed in color)

6 Scene Reconstruction with Fore-Object

As mentioned above, each segmentation region with the pixels having similar properties represents a coherent region in the scene. Thus, it will be sometimes failed when there are fore-objects in the scene. The example is shown as Fig.8(a). A stool lies in front of the back wall with the similar color and texture. In the inferred 3D scene shown in Fig.8(b), the stools are conjoint to the back wall, which is obviously wrong.

Under the on-the ground assumption, we propose a method to deal with the above problem. Actually, the fore-objects are most likely to be on the ground, rather than in it, especially at indoor environments. So we firstly find the ground region in an image. According to the initial scene reconstruction results, the edges of the ground region can easily be extracted and denoted as a set of lines $l_1, l_2, \cdots l_n$. Then the pixels surrounded by $l_1, l_2, \cdots l_n$ are marked as ground region. As for the fore-object region in image, it is most likely to be intersected with the ground region, rather than to be included in it. So if a region has only a part of pixels marked as ground region, it can be regarded as fore-object region. The example of extracting the fore-object is shown as Fig. 8(c), black line is the edge of ground region and the red block is the fore-object.

Then the fore-object regions and the rest regions are dealt with respectively. As for the fore-object, it can be assumed to be vertical to ground since there is no more information about it. Based on the assumption, the depth is predicted according to projective geometry. As for the rest regions, the depth can be inferred by the methods described in section 4. Finally the scene reconstruction results are incorporated together. The experimental results are shown as Fig.8(d,e,f).

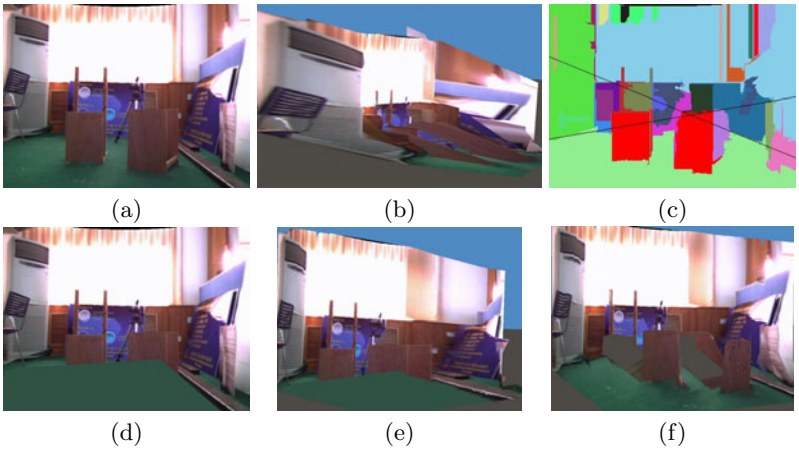

Fig. 8. (a) Original image, (b) Results of scene reconstruction without the detecting of fore-object, (c) Results of detecting the edges of ground region (black lines) and extracting the fore-object (red block), (d, e) Results of scene reconstruction after extracting fore-object, (f) Final scene reconstruction results. (Best viewed in color)

7 Conclusion

Over the last few decades, great progresses have been made on the depth inferring and scene reconstruction form stereo, motion and other "triangulation" cues. However, the vast majority of this work has only used the geometric cues, but neglected the other depth cues contained in the image, such as texture, color, defocus and so on. In contrast, the recent research of monocular depth perception, such as Saxena et al's and Hoiem et al's work, is commendably supplementary to computer vision.

Inspired by these works, this paper presents a close-form iterative algorithm that utilizes the inter-correlated property between image segmentation and depth inferring. The algorithm can significantly improve segmentation and depth inferring by processing them alternately iteratively. Our algorithm firstly obtains the initial segmentation results by an EM-based algorithm. Then, a multi-scale Markov random field, trained by supervised learning, is used to model the relations between feature vectors and different depths. After the depth of each pixel are inferred, it is fed back to refine the segmentation results at the larger scales. This method can offer more global structure information without decreasing the number of the patches made up of 3d scene structure. The above processes are iteratively continued until the expected results are achieved. The experimental results show the validity of our algorithm. Moreover, the paper also extends the method to deal with the problem that infers depth of the scene with fore-objects. We believe that our algorithm can be used for many other applications in vision, such as robot navigation, building 3-d models of urban environments, and object recognition.

Acknowledgments. This research was funded by NSFC(No: 60705015, 60805019). We would like to thank Shuai Fang for her valuable contribution towards this research.

References

1. Saxena, A., Chung, S.H., Ng, A.Y.: Learning depth from single monocular images. In: Neural Information Processing System (NIPS), vol. 18 (2005)
2. Saxena, A., Chung, S.H., Ng, A.Y.: 3-d depth reconstruction from a single still image. International Journal of Computer Vision (IJCV) 76, 53–69 (2007)
3. Saxena, A., Schulte, J., Ng, A.Y.: Depth estimation using monocular and stereo cues. In: International Joint Conference on Artificial Intelligence (IJCAI) (2007)
4. Saxena, A., Sun, M., Ng, A.Y.: Make3d: depth perception from a single still image. In: AAAI Conference on Artificial Intelligence (AAAI) (2008)
5. Saxena, A., Sun, M., Ng, A.Y.: Make3d: Learning 3-d scene structure from a single still image. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 31, 824–840 (2008)
6. Hoiem, D., Efros, A.A., Hebert, M.: Automatic photo pop-up. In: International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH) (2005)
7. Hoiem, D., Efros, A.A., Hebert, M.: Geometric context from a single image. In: International Conference on Computer Vision (ICCV) (2005)

8. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: International Conference on Computer Vision (ICCV) (2009)
9. Hoiem, D., Efros, A.A., Hebert, M.: Closing the loop on scene interpretation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2008)
10. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. International Journal of Computer Vision (IJCV) 80, 3–15 (2008)
11. Malik, J., Rosenholtz, R.: Computing local surface orientation and shape from texture for curved surfaces. International Journal of Computer Vision (IJCV) 23, 149–168 (1997)
12. Malik, J., Perona, P.: Preattentive texture discrimination with early vision mechanisms. Journal of the Optical Society of America A 7, 923–932 (1990)
13. Maki, A., Watanabe, M., Wiles, C.: Geotensity: Combining motion and lighting for 3d surface reconstruction. International Journal of Computer Vision (IJCV) 48, 75–90 (2002)
14. Zhang, R., Tsai, P.S., Cryer, J.E., Shah, M.: Shape from shading: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 21, 690–706 (1999)
15. Horry, Y., Anjyo, K.I., Arai, K.: Tour into the picture: using a spidery mesh interface to makeanimation from a single image. In: International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH) (1997)
16. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision (IJCV) 47, 7–42 (2002)
17. Forsyth, D., Ponce, J.: Computer Vision: A Modern Approach. Prentice Hall Professional Technical Reference (2002)
18. Delage, E., Lee, H., Ng, A.: A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2006)
19. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Depth from familiar objects: A hierarchical model for 3d scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2006)
20. Michels, J., Saxena, A., Ng, A.Y.: High speed obstacle avoidance using monocular vision and reinforcement learning. In: International Conference on Machine Learning (ICML) (2005)
21. Heitz, G., Gould, S., Saxena, A., Koller, D.: Cascaded classification models: Combining models for holistic scene understanding. In: Neural Information Processing Systems (NIPS) (2008)
22. Carson, C., Belongie, S., Greenspan, H., Malik, J.: Blobworld: image segmentation using expectation-maximization and its application to image querying. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 24, 1026–1038 (2002)
23. Garding, J., Lindeberg, T.: Direct computation of shape cues using scale-adapted spatial derivative operators. International Journal of Computer Vision (IJCV) 17, 163–191 (1996)
24. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. International Journal of Computer Vision (IJCV) 59, 167–181 (2004)