2012 IEEE International Conference on Multimedia and Expo Workshops

Depth Extraction from Monocular Video Using Bidirectional Energy Minimization and Initial Depth Segmentation

Chunyu lin, Jan De Cock, Jürgen Slowack, Peter Lambert and Rik Van de Walle
Multimedia Lab Ghent University IBBT
Gaston Crommenlaan 8 bus 201 9050 Ledeberg-Ghent, Belgium
Email: {Chunyu.Lin, Jan.DeCock, Jurgen.Slowack, Peter.Lambert, Rik.VandeWalleg}@ugent.be

***Abstract*—In this paper, we propose to extract depth information from a monocular video sequence. When estimating the depth of the current frame, the bidirectional energy minimization in our scheme considers both the previous frame and next frame, which promises a much more robust depth map and reduces the problems associated with occlusion to a certain extent. After getting an initial depth map from bidirectional energy minimization, we further refine the depth map using segmentation by assuming similar depth values in one segmented region. Different from other segmentation algorithms, we use initial depth information together with the original color image to get more reliable segmented regions. Finally, detecting the sky region using a dark channel prior is employed to correct some possibly wrong depth values for outdoor video. The experimental results are much more accurate compared with the state-of-the-art algorithms.**

***Keywords*-Depth extraction, depth segmentation, 2D to 3D**

I. Introduction

Depth extraction is a key problem for many research topics, such as robust navigation, scene understanding and 3D reconstruction. Based on the number of views employed, the depth extraction algorithms are generally classified into three groups. That is, depth extraction from monocular video, stereo video and multiview video. Generally the video contents will be created directly into some suitable 3D format. However, the conversion of 2D content is highly interesting because of the large amounts of existing 2D video content, the tremendous production cost and the complicated process for 3D video generation. Hence, extracting depth information from monocular video will be the focus of our paper.

To estimate the depth information from a monocular video or image, a variety of depth cues can be employed. Defocus cues are employed to estimate the blur extent of an image which is then converted to depth information. In [1], the blur extent is estimated by analyzing the image intensity histogram associated with the optical system. In [2], edge defocus is estimated based on wavelet analysis, combined with color segmentation. In [3], the input image is re-blurred using a Gaussian kernel and the amount of defocus blur is obtained from the gradient ratio between the input and re-blurred images. All these three schemes assume that there is defocused information in the image and the obtained depth information will be affected by the original blur in the image, such as motion blur. In addition, it is not easy to distinguish the foreground from the background when the amount of blur is similar [4]. Geometric cues include linear perspective, known size, relative size/height in picture, interposition, and texture gradient. However, the linear perspective is often applied, in which parallel lines converge at infinite distance. The converged point is referred to as the vanishing point and the corresponding lines are denoted as vanishing lines. By detecting the vanishing point and vanishing lines in the image, the depth can be estimated according to the position of the lines and the vanishing point. In [5], the depth map is estimated by image classification combined with vanishing lines and vanishing point detection. In [6], vanishing point and superpixels are combined together to generate a depth map. Atmospheric cues refer to the bluish phenomenon generated by the light rays scattered by the atmosphere. Using this information, it is possible to detect whether objects are located at a close or far distance. In [7], a scheme to remove the atmospheric haze in an image is proposed using a dark channel prior. The interesting thing is that a high-quality depth map can also be obtained as a byproduct.

The above cues are applied to a single image mostly, therefore they are denoted as pictorial cues. Although the depth map of one single image could be good, the above techniques do not consider the smoothness of the depth map between consecutive frames in a video. Different from pictorial cues, another important cue is motion, which exploits information from two or more images. In [8] [9], Structure From Motion (SFM) is used to compute camera parameters and estimate the 3D scene. The performance of this scheme depends on the feature detection process which does not work well for textureless regions such as blue sky. In [10], the motion vector is directly used or modified to approximate the disparity, which provides a good compatibility with the standard. However, the motion-to-disparity technique does not consider the smoothness in the same object. In addition, the consistency of the depth map between different frames is not satisfactory.

In this paper, we propose a depth map extracting scheme using bidirectional energy minimization. Energy minimiza-

978-0-7695-4729-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICMEW.2012.94

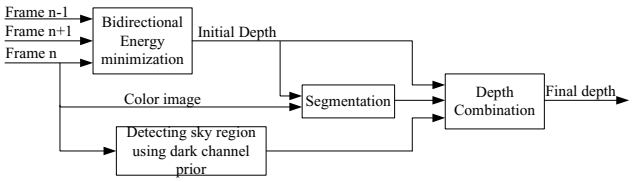


Figure 1. The diagram of the proposed scheme.

tion is employed widely in stereo applications because it allows soft constraints and provides spatial smoothness [11]. By considering both the previous frame and next frame into the energy minimization framework, the proposed scheme promises a much more stable depth map and reduces the occlusion problems to a certain extent. Based on the assumption that there are no large depth changes inside homogeneous color segments, a segmentation algorithm using color information and initial depth information is adopted to refine the depth map. Finally, detecting sky region is employed to correct some possibly incorrect values for outdoor video sequences.

II. The proposed scheme

The proposed scheme is shown in Fig. 1, and includes bidirectional energy minimization, segmentation, detection of sky region and depth refinement. The four steps will be detailed in the following sections.

A. Bidirectional energy minimization

Energy minimization is widely used for stereo matching. Since the motion between two frames can be considered as a form of disparity over time, energy minimization is adopted in our scheme as well. Different from the stereo matching case, here we use the previous and next frame, together with the current frame, to get smooth depth information. It should be noted that the idea of multiple depth maps fusion is also used in [12], where a set of depth maps from neighboring camera positions are combined into a single depth map.

Given one frame n, let I_n denote its intensity value and z_n represent its depth value. The objective is to find the depth value, z_n, of each pixel in frame n with the help of the previous frame $n-1$ and the next frame $n+1$. Generally, the energy minimization function for the stereo case is defined as

$$E(f) = E_{data}(f) + E_{smooth}(f) \tag{1}$$

where f denotes the disparity or depth assignment function. The data term measures the color similarity, which is calculated as the difference in intensity between one pixel and its corresponding pixels. The smoothness term makes the neighboring pixels tend to have similar depth. In the stereo case, the corresponding pixels are generally related by the disparity value between the left and right image. For convenience and later use, the function between disparity and depth is shown in (2), where t is the distance between two cameras and f is the focal length.

$$d = t\frac{f}{z} \tag{2}$$

To get a relative depth for the stereo case, the simple form $d = 1/z$ is generally used. Hence, depth and disparity are used interchangeably. In our case, the previous and next frame are used to get the depth value of the current frame. Hence, there are two pairs of frames (n with $n+1$ and n with $n-1$). for simplicity, we will only consider the situation that the camera moves in the same direction. Hence, the current frame will be taken as right image and left image respectively, relative to the previous frame and next frame. The energy minimization terms in our case will be

$$\begin{aligned} E_{data}(f) = \sum_{\forall p} &(I_n(p) - I_{n+1}(p - t_1 f_1/z(p)))^2 \\ &+(I_n(p) - I_{n-1}(p + t_2 f_2/z(p)))^2 \end{aligned} \tag{3}$$

$$E_{smooth}(f) = \sum_{\forall q \in N(p)} (z(p) - z(q))^2 \tag{4}$$

Here, p denotes the pixel and $N(p)$ represents its neighborhood. Since only the depth of the current frame is estimated here, z is used instead of z_n for simplicity, as well as d instead of d_n. The data item tries to get a depth map that makes the intensity difference between current pixel and its corresponding two pixels as small as possible. The smooth item makes the neighborhood has similar depth map. In reality, the item f is not changed abruptly between two consecutive frames. If we assume the camera moves with a uniform speed, the item t between any two frames should be the same. Otherwise, some techniques could be used to detect t and f as camera parameters[8] [9]. If we assume the value of f and t for the two pairs to be the same, then we can use the disparity and depth interchangeably. For clearness, the energy items for the disparity are shown

$$\begin{aligned} E_{data}(f) = \sum_{\forall p} &(I_n(p) - I_{n+1}(p - d(p)))^2 \\ &+(I_n(p) - I_{n-1}(p + d(p)))^2 \end{aligned} \tag{5}$$

$$E_{smooth}(f) = \sum_{\forall q \in N(p)} (d(p) - d(q))^2 \tag{6}$$

Notice the disparity values between the first and second pair of frames have an opposite sign in the function due to the relative position of the current frame, compared with the previous and next frame.

Such a bidirectional energy minimization has two main advantages. The first is that it provides a more stable depth map. If some disparity values are not detected correctly in the first pair of frames, it could be compensated by considering the second pair of frames. In addition, to get the depth of the current frame, both the previous and next

frames are employed, which means $\frac{2}{3}$ of the information is overlapped in the previous and following depth extraction process. This process promises a more stable depth map for the whole sequence. The second advantage is that occlusions can be reduced. Due to the structure of the scene, some parts of a scene may be visible in only one of two cameras (frames in our case). These pixels are denoted as occluded region or occlusion, which could be seen in Fig.6. Since these regions means that the pixels in one frame cannot find their corresponding ones in another frame, their disparity values are not accurate. By using previous and following frames together, the occluded pixels in the first pair of frames will not be occluded in the second pair of frames with high probability if the scene or camera moves in the same direction. Hence, the occluded region can be compensated to a certain extent, by using two pairs of frames, which can be seen in SectionIII.

To get the occluded region, we need to use (1) for each of the two pairs of frames separately, as shown in the following formulas

$$\begin{aligned} E(f') &= \sum_{\forall p}(I_n(p) - I_{n+1}(p - d'(p)))^2 \\ &\quad + \sum_{\forall q \in N(p)} (d'_n(p) - d'_n(q))^2 \qquad (7) \\ E(f'') &= \sum_{\forall p}(I_n(p) - I_{n-1}(p + d''(p)))^2 \\ &\quad + \sum_{\forall q \in N(p)} (d''(p) - d''(q))^2 \qquad (8) \end{aligned}$$

In stereo matching, it is easy to get the occluded region by cross checking the generation process of the disparity map. The cross checking computes the matches left-to-right and right-to-left, and marks a pixel as occluded if the disparity value from the two disparity maps are not consistent [11]. For our case, the cross checking is implemented on the generation process of d' and d'' to get the two maps of occlusion labels($occ'(p)$ and $occ''(p)$). If occluded regions appear on the objects' left side for the first disparity generation process(d'), then the occluded regions generally lie on the objects' right side for the second disparity generation process(d''), and vice versa. For such occluded regions, we can set the following conditions on the third energy minimization function as,

$$\begin{cases} \sum_{\forall p} 2(I_n(p) - I_{n+1}(p - d(p)))^2 + \sum_{\forall q \in N(p)} (d(p) - d(q))^2, \\ \quad if(occ''(p) == 1 \;\&\&\; occ'(p)! = 1) \\ \sum_{\forall p} 2(I_n(p) - I_{n-1}(p + d(p)))^2 + \sum_{\forall q \in N(p)} (d(p) - d(q))^2, \\ \quad if(occ'(p) == 1 \;\&\&\; occ''(p)! = 1) \\ \sum_{\forall p} (I_n(p) - I_{n+1}(p - d(p)))^2 + (I_n(p) - I_{n-1}(p + d(p)))^2 \\ + \sum_{\forall q \in N(p)} (d(p) - d(q))^2, others \end{cases} \qquad (9)$$

where $occ'(p) == 1$ denotes that pixel p is occluded in the first pair of frames and $occ''(p) == 1$ denotes that pixel p is occluded in the second pair of frames. Take the first case as an example, the data term just uses the first pair of frames to calculate the depth value because this pixel is occluded in the second pair of frames. Only when no occlusion is detected, the average term will be employed. In conclusion, we firstly use (7) and (8) to get two maps of occlusion labels and two depth maps as well. Then the obtained occlusion labels are used conditionally in (9) to get an updated depth map. Hence, the energy minimization procedure needs to be executed three times. In fact, the depth maps obtained after the first two energy minimization steps in (7) and (8) can be used directly, making it possible to eliminate one energy minimization step. The new depth calculation process is obtained as

$$d(p) = \begin{cases} d'(p), if(occ'(p) == 1 \;\&\&\; occ(p)! = 1) \\ d''(p), if(occ(p) == 1 \;\&\&\; occ'(p)! = 1) \\ (d' + d'')/2, others \end{cases} \qquad (10)$$

Hence, using (7) and (8) and (10), the depth map is obtained and it will be used as an initial depth map in the next section. To implement the energy minimization process, the powerful optimization algorithms of graph cut [11] [13] is employed to get a fast approximation.

B. Segmentation with initial depth map

Segment-based methods for stereo matching have attracted a lot of attention [14],[15],[16]. They are based on the assumption that the depth value is similar in one color segmented region. Hence, we want to use segmentation to refine the depth value too. The advantage of the segmentation here is that we will take the initial depth map into consideration.

The typical image or video segmentation methods leverage the color difference between pixels to identify the objects' boundary, which is not accurate when the background and the foreground share similar color. In this case, the depth map could help to improve the segmentation. Although the

color of different objects might be similar, the distance of the object in the foreground and background is obviously different. Hence, we employ the initial depth map obtained from the bidirectional energy minimization to improve the segmentation.

Our segmentation employs the framework of the graph-based algorithm in [17]. This algorithm takes pixels as vertices and two connected pixels as an edge. Each edge uses the intensity difference of the two pixels as its weight. Hence, the segmented region is composed of the edges with a certain rule. The rule is that edges between two pixels in the same component should have relatively low intensity difference, and edges between pixels in different components should have a high intensity difference.

By considering the initial depth map obtained from our bidirectional energy minimization, we change the weight of an edge as follows

$$w(p,q) = 0.5((r(p)-r(q))^2 + (g(p)-g(q))^2 + (b(p)-b(q))^2) + 0.5(z(p)-z(q))^2 \quad (11)$$

where r, g and b denotes the color channel. Here, color image and depth map are considered with equally importance and the obtained segmentation results in Section III show that it can provide much more reliable homogeneous regions. More details about graph-based image segmentation can be found in [17]. In addition, we believe that a much more complex formula between color image and depth image could provide much better segmentation results, which will be part of our future work.

C. Correcting the region with infinite depth

It is obvious that the sky region has infinite depth in a scene. However, the estimation for the depth of sky region is often difficult due to its textureless feature. Hence, we propose to correct the depth of this region by using the dark channel prior [7].

In [7], the dark channel prior is observed and used to remove haze in a hazed picture. However, the dark channel prior is a phenomenon that occurs in a normal picture. Hence, it could be used in our case. The dark chancel prior is that at least one color channel has very low intensity at some pixels in most of the non-sky patches. The following function is used to describe the dark channel prior[7]

$$J^{dark}(p) = \min_{c\in\{r,g,b\}} (\min_{q\in\Omega(p)} (J^c(q))) \quad (12)$$

where J denotes the normal image and $\Omega(p)$ represents a block region surrounding pixel p.

This function can be used in another way. If any of the channels does not have very low intensity, it must be the sky region or the white object. Since the white object is easy to be detected, we will not discuss it here. With the dark channel prior, we can find the sky region and assign it a corrected depth value. This process does not require any training process. Then the depth of other regions can be adjusted correspondingly since the new infinite region is found. For the image without sky region, this process will not affect the performance of the proposed scheme. Hence, there is no need to do a classification on the image.

Table I
DEPTH COMBINATION

Given *initial depth map z, segmented regions s and sky region.*
if pixel p is in *sky region*
    $z(p)$=0;
end
for each segmentations $s(i)$
    $z(s(i))$=average($z(s(i))$)
end

D. Depth combination

After getting the initial depth map, the segmentation and sky region, the final depth map will be obtained with a simple combination process as shown in Table I.

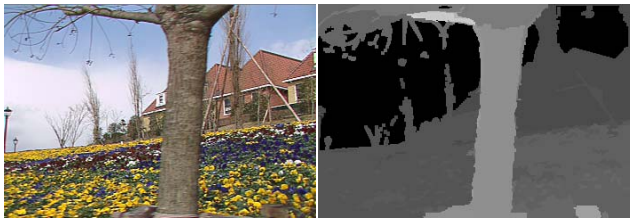
III. EXPERIMENTAL RESULTS

In this section, the proposed scheme is implemented on the monocular video sequences *Flower Garden* and *Parkjoy*. Intermediary results for segmentation with initial depth map are also included. Here, the depth value is obtained by scaling the different disparity values in the range of [0, 255], which are then relative values. Due to the limited resolution and quality of the image on the printed paper, the original pictures and some other results have been made available at our website[1].

In Fig. 2 and Fig. 3, the 6th and 20th frame of *Flower Garden* are shown, as well as their corresponding depth maps. These two frames are selected because they are used in [9], [15] and [16], which permits a subjective comparison. It is fair to compare our results with that of these three papers because all of them use certain optimization method and segmentation information. Moreover, the results of these threee paper are very competive. In [9], a bundle optimization framework is proposed, in which the disparity maps is initialized in the color segmentation process and refined by means of bundle optimization. In [15], a segment-based stereo matching algorithm is proposed, where the energy minimization is applied to different segmented regions. In [16], the belief propogation algorithm and segmentation information are employed to get the depth. From Fig. 2 and Fig. 3, it can be seen that most of the regions in the scene have a relative correct depth, even for the house and sky components, which is better than that of [9], [15] and [16].

In Fig. 4 and Fig. 5, the 2nd and 40th frame from high definition sequence of *Parkjoy* are shown, together with their corresponding depth maps. These two frames

[1] http://multimedialab.elis.ugent.be/users/chlin/Depth_map_results



(a) the 6th frame (b) the depth of the 6th frame

Figure 2. The 6th frame of *Flower Garden* and its depth map.

(a) the 20th frame (b) the depth of the 20th frame

Figure 3. The frame of *Flower Garden* and its depth map.

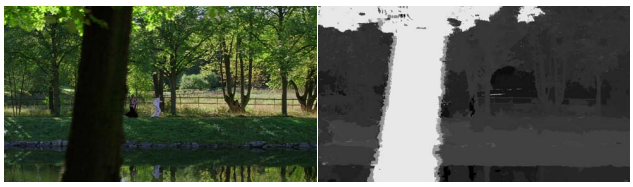
represent two different scenes of the sequence. It can be seen that most of the regions can get a reliable depth value. For the regions containing the people and water, the depth is not completely correct. The water region is difficult for the disparity calculation due to its similar color value in the whole region. In addition, there is shadow in our case, which further complicates the depth estimation. The persons in the scene are independently moving objects, hence their disparity should not be translated into depth directly. To make the system works for the independently moving objects, we need to detect the independently moving objects and correct them according to their motion vectors. This is part of our future work. For a good evaluation of our scheme, more results are provided on our website.

In Fig. 6(a), the occlusion indicated in red color is shown for the first pair of frames, in which the current frame is taken as a left image. Hence, it is reasonable to see that the left side of the objects is occluded occasionally. On the contrary, the right side of the object in the scene is occluded in the second pair of frames, shown in Fig. 6(b). This explains the advantage of our bidirectional energy minimization scheme that could reduce the occlusion to a certain extent by considering two pairs of frames together.

In Fig. 7(a), the segmentation results calculated from the color image combined with initial depth maps is shown. For comparison, the segmentation result for color image alone, which use the graph cut algorithm [11], is also provided in Fig. 7(b). It can be seen that the results of segmentation with initial depth map is more reliable, especially for the tree. This is because the tree as an object has almost the same disparity/depth value in the initial depth map which helps the segmentation to differentiate the real boundary of the object.

(a) the 2nd frame (b) the depth of the 2nd frame

Figure 4. The 2nd frame of *Parkjoy* and its depth map.



(a) the 40th frame (b) the depth of the 40th frame

Figure 5. The 40th frame of *Parkjoy* and its depth map.

IV. Discussion

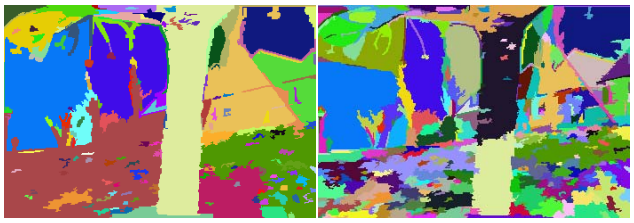
It should be noted that our algorithm can only handle the video with sufficient camera movement, while the independent object motion in the scene cannot get a correct depth. However, with some independent motion detection, the depth of this type of object can be corrected by modifying its disparity value with its motion vector. The case such as static frames probably could be processed with depth propagation or more advanced algorithm, for example combined with pictorial cues, which will be our future work.

V. Conclusion

This paper proposes an effective scheme to estimate the depth map for monocular video sequences. It employs bidirectional energy minimization and initial depth segmentation, as well as sky region detection for outdoor video. Our method provides a more reliable depth map with a few advantages. Firstly, it considers the previous and following frames when extracting the depth map of current frame. Hence, the obtained depth map is more stable and occlusion problems are reduced to a certain degree. Secondly, the segmentation algorithm in our scheme employs both the color image and initial depth map, which provides more meaningful segmented regions. Lastly, detection of the sky region based on dark channel prior enables to correct the depth of sky region without any complex training. Even though the sky region is not always existed in the video, it will not take too much time to check or affect the performance of the video sequence without sky region. The experimental results demonstrate the effectiveness of the proposed scheme.

(a) Left occlusion (b) Right occlusion

Figure 6. The occlusion between the 6th and 7th frame, 6th and 5th frame of *Flower Garden*(Red color parts denote the occluded regions).



(a) Color image+ initial depth map (b) Color image

Figure 7. Segmentation comparison between color image and color image with initial depth map.

ACKNOWLEDGEMENT

This research described in this paper was carried out as part of the IBBT/IWT 3DTV2.0 project and was funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

REFERENCES

[1] H.-Y. Lin and K.-D. Gu, "Depth recovery using defocus blur at infinity," in *Proc. 19th Int. Conf. Pattern Recognition (ICPR)*, 2008, pp. 1–4.

[2] G. Guo, N. Zhang, L. Huo, and W. Gao, "2D to 3D convertion based on edge defocus and segmentation," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 2181–2184.

[3] S. Zhuo and T. Sim, "On the recovery of depth from a single defocused image," in *Proceedings of the 13th International Conference on Computer Analysis of Images and Patterns*, ser. CAIP '09, 2009, pp. 889–897.

[4] L. Zhang, C. Vazquez, and S. Knorr, "3D-TV content creation: Automatic 2D-to-3D video conversion," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 372–383, 2011.

[5] S. Battiato, S. Curti, M. L. Cascia, M. Tortora, and E. Scordato, "Depth map generation by image classification," B. D. Corner, P. Li, and R. P. Pargas, Eds., vol. 5302, no. 1. SPIE, 2004, pp. 95–104.

[6] K. Han and K. Hong, "Geometric and texture cue based depth-map estimation for 2D to 3D image conversion," in *Proc. IEEE Int Consumer Electronics (ICCE) Conf*, 2011, pp. 651–652.

[7] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 33, no. 12, pp. 2341–2353, 2011.

[8] P. Li, D. Farin, R. K. Gunnewiek, and P. H. N. de With, "On Creating Depth Maps from Monoscopic Video using Structure from Motion," in *Proc. IEEE Workshop on Content Generation and Coding for 3D-television*, 2006, pp. 85–92.

[9] G. Zhang, J. Jia, T.-T. Wong, and H. Bao, "Consistent depth maps recovery from a video sequence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, pp. 974–988, June 2009.

[10] M. T. Pourazad, P. Nasiopoulos, and R. K. Ward, "Generating the depth map from the motion information of H.264-encoded 2D video sequence," *J. Image Video Process.*, vol. 2010, pp. 4:1–4:13, January 2010.

[11] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *18th IEEE International Conference on Computer Vision*, vol. 2, 2001, pp. 508–515.

[12] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nister, and M. Pollefeys, "Real-time visibility-based fusion of depth maps," in *IEEE 11th International Conference on Computer Vision*, Oct. 2007, pp. 1 –8.

[13] V. Kolmogorov and R. Zabih, "Multi-camera scene reconstruction via graph cuts," in *Proceedings of the 7th European Conference on Computer Vision-Part III*, ser. ECCV '02, 2002, pp. 82–96.

[14] A. Klaus, M. Sormann, and K. Karner, "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure," in *18th International Conference on Pattern Recognition, ICPR 2006.*, vol. 3, 2006, pp. 15–18.

[15] L. Hong and G. Chen, "Segment-based stereo matching using graph cuts," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2004, pp. 74–81.

[16] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 787 – 800, July 2003.

[17] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vision*, vol. 59, pp. 167–181, September 2004.