Watershed and Random Walks based Depth Estimation for Semi-Automatic 2D to 3D Image Conversion

Xuyuan Xu, Lai-Man Po, Kwok-Wai Cheung[#], Ka-Ho Ng, Ka-Man Wong, Chi-Wang Ting

Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR, China
[#]Department of Computer Science, Chu Hai College of Higher Education, Hong Kong SAR, China

ABSTRACT

Depth map estimation from a single image is the key problem for the 2D to 3D image conversion. Many 2D to 3D converting processes, either automatic or semi-automatic, are proposed before. Quality of the depth map from automatic methods is low and there exists wrong depth values due to errors estimation in depth cue extraction. The semi-automatic approaches can generate a better quality of depth map based on the user-defined labels, which indicate a rough estimation of depth values in the scene, to generate the rest of depth value and reconstruct the stereoscopic image. However, they require complexity system and are very computational intensive. A simplified approach is to combine the depth maps from Graph Cuts and Random Walks to persevering the sharp boundary and fine detail inside the objects. The drawback is the time consuming of the energy minimization in the Graph Cuts. In this paper, a fast Watershed segmentation based on the priority queue, which indicates the neighbor distance relationship, is used to replace the Graph Cuts to generate the hard constraints depth map. It is appended to the neighbor cost in the Random Walks to generate the final depth map with hard constraints in the objects boundaries regions and fine detail inside objects. The Watershed and Random Walks are low computational intensive and can achieve approximate real time estimation which results in a fast stereoscopic conversion process. Experimental results demonstrate that it can produce good quality stereoscopic image in very short time.

Keywords-component; Depth map estimation; 2D to 3D; Semi-automatic 2D to 3D

I. INTRODUCTION

The 3D image signal processing has become an active topic in the visual processing field [1]. As 3D display technology becomes matures such as 3DTV and auto-stereoscopic display, human aspires to experience the 3D content. Especially after the invention of 3D mobile, human are much easier to view the 3D content. However, the lack of 3D content production is a dilemma for 3D industry. Therefore, a 2D-to-3D converting process has large demand in the 3D industry, particularly for the widely used of 3D mobile in the future. Two popular approaches to produce 3D content are directly capturing the content with multiple cameras and converting the existing 2D content into 3D [2]. The best solution is direct capture. However, direct capture requires special equipment and a complex post production pipeline. In some situation, like converting existing 2D content into 3D, stereo conversion is preferred. The main challenge is how to estimate the depth information from the 2D content that lost in the capture process.

Many 2D-to-3D converting processes have been proposed to tackle this problem, including automatic and semi-automatic. They focus on recovery of the depth information and based on the Depth Image Based Rendering (DIBR) to generate the stereoscopic images. Po bases on motion parallax to estimate the depth map automatically from videos [3] while Schnyder extracts the depth information from sports content video by exploiting context-specific priors [4]. For the automatic 2D image conversion, Jung uses the gradient and linear perspective for depth estimation [5] while Cheng estimates the depth based on scene categories [6]. Zhang uses the fusion of multiple depth cues [7] to produce a better quality of depth map. However, the automatic processes have drawbacks of either computation is very intensive or wrong depth values due to errors in extracting the depth cue. To improve the quality of depth map, user specified labels for the semi-automatic conversion process was proposed to tackle drawbacks of the automatic process. In [8], Guttmann proposes a semi-automatic approach. However, the system is very complicated and requires many processes to generate the final depth map. Later, Phan uses the Graph Cuts segmentation as the prior for depth estimation in Random Walks [9]. The conversion system is simplified but the energy minimization process of Graph Cuts is still computational intensive. In our proposed method, the Watershed segmentation is used instead of Graph Cuts that can additional simplified the system while retain the similar estimated depth quality. The reducing computational intensive conversion is very suitable for developing mobile application of 3D smart phone.

This paper is organized as follows. Section 2 has a review of depth map from Graph Cuts. Section 3 describes the depth map generated from Watershed and Random Walks. The fusion of depth map from Watershed and Random Walks is also presented in section 3. The experimental results are provided in section 4. Finally, a conclusion is drawn in section 5.

II. REVIEW OF DEPTH MAP GENERATED FROM GRAPH CUTS SEGMENTATION

Graph Cuts aims at minimizing the energy function by solving the Maximum-A-Posteriori Markov Random Field (MAP-MRF) labeling problem. It provides the hard segmentation and the energy function is defined [10]:

$$\mathrm{E(P)} = \sum_{p \epsilon P} D_p(f_p) + \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) \quad (1)$$

E(p) is the energy of pixel p. $D_p(f_p)$ is the data cost assigned the label f_p to pixel p. N is the set consisting of adjacent pixels. $V_{p,a}(f_p, f_a)$ is the smoothness cost assigned two different labels to two neighborhood of N. The solution of minimization of energy E(P) can be found by min-cut/max-flow of the graph. Thus best described label value for each pixel can be found out.

The data cost is defined according to the user-defined labels. Beside the weights of k^{th} sink link and n^{th} ($n \neq k$) source link for label k are 0, others weights of link are K. K is a constant value defined in section 2.5 of [10]. The smoothness cost [2] is as follow:

$$N_{(i,j)} = \gamma \left(\frac{2}{1 + \exp(\beta [d(\vec{c_i}, \vec{c_j})]^{\alpha}} \right) \quad (2)$$

Where $d(\vec{c_i}, \vec{c_j})$ is the Euclidean distance of the CIE L*a*b components between pixels i and j. The constant is chosen experimentally as $\alpha = 2$, $\beta = 1$ and $\gamma = 255$.

For regions with user-defined labels, data cost set to zero. While for other regions, data costs are assigned to constant K that is greater than the maximal weight of smooth cost. This can enforce the Graph Cuts segmentation only based on the smooth energy in the minimization process. Only color distant is considered in the label classification and L*a*b color Euclidean distance is used.

Graph Cuts segmentation can only provide a binary segmentation. For multiple labels, the energy minimization for each label needs to be processed separately and combined together at the final step. To minimize the energy of each label, each pixels need to find a best label value that describes its clustering characteristic. This process is very computational intensive and the total of computation time is proportional to the number of user-defined labels.

III. Watershed and Random Walks based Depth Estimation

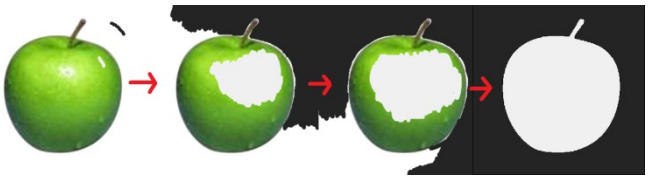
Stereoscopic conversion from monoscopic content is not a process to reconstruct the real depth information from the scene. It creates only the relative depth information. Based on this information and Depth Image Based Rendering, the stereo content can be reconstructed. Depth map from Watershed and Random Walks has the characteristic of sharp object boundary and fine detail inside the object respectively. The Watershed can preserve the sharp boundary information of depth map while the depth values are constant inside one object. On the other hand, depth map from Random Walks is smooth without sharp boundary information. Unlike [9], our approach generates the depth value by taking the average depth labels value using the probabilities distribution from Random Walks instead of using the probabilities as the depth value in [9]. The fusion of these two depth map can result in a better depth map that persevere the sharp boundary and has smooth depth value inside the objects. The final generated stereoscopic content can be more natural.

A. Depth map generation based on Watershed

Conventional watershed segmentation involves two processes: seed selection (minimum region selection) and flooding. It consists of placing a water source in each of regional minimum to flood the relief from sources and barriers are built when different sources are met [6]. The minimum regions are defined by the user-defined labels. Each of the user-defined labels represents one of the minimum regions. In flooding process, it bases on the user-defined labels to find out the most similarity region and assigns the corresponding labels. The similarity is measured as the maximum distance of the RGB component as follow:

$$D_{(i,j)} = \max\left(\left(R_i - R_j\right), \left(G_i - G_j\right), \left(B_i - B_j\right)\right) \quad (3)$$

Where R_k, G_k and B_k are the red, green and blue component of pixels k. i and j are two neighborhood pixels. Here four-connected is used to describe the relationship of neighborhood. The fast flooding process of Meyer [11] is used to perform the segmentation. Before flooding, pixels are classified as two set, L_l (pixels with user-defined label) and L_u (pixels without label). Each time, put the neighborhood pixels of L_l in L_u into the priority queue if any (the priority is equal to the $D_{(i,j)}$) , pop one pixels with least priority value in the priority queue, assign the corresponding label and put it into set L_l. This process repeats until all the pixels got the labels. The final segmentation image is generated after flooding process. A lookup table is used to convert the labels value of segmentation map into depth value.



(a) (b) (c) (d)

Fig. 1: (a) input image with two user-defined labels, the white label on the apple and the black label at top right corner of background. (b) Segmentation map after several flooding from (a), (c) Segmentation map after several flooding from (b), (d) final segmentation map

The flooding process is very efficient. The distant values are integer and are used for the construction of priority queue. The priority queue contains the neighboring pixels of the labeled pixels and the least priority pixel is the next pixel for flooding. The neighbor pixels with higher similarity or less color distant are assigned the same label values after each of flooding process, as shown in Fig. 1. For the object with the color very difference from other objects, one user-defined label is enough to estimate the depth value for the whole object, like Fig. 1 (a). If the object has more than one color and some color is very similar to others, each color region is better to contain the same user-defined label. Otherwise, wrong label value could be assigned in the flooding process, like Fig. 2 (a) and (b). In Fig. 2(a), the background color is similar to that of apple's branch. With the same user-defined label as Fig. 1 (a), the segmentation of apple cannot contain the branch region, shown in Fig. 2 (b). If one more additional label is added to the branch of apple, shown in Fig. 2 (c) to indicate this region is belong to apple, the result segmentation map can become more correct as in Fig. 2(d). In addition, label for the same object can have several distinct connected components as long as they share the same label value, like the label for apple in Fig. 2 (c).

(a) (b) (c) (d)

Fig. 2: (a) input image with two user-defined labels, the white label on the apple and the black label at top right corner of background. (b) Segmentation map of (a), (c) add an additional marker on the branch of apple from (a), (d) Segmentation Map of (c)

Graph Cuts and Watershed both base on the color similarity in the spatial domain to classify the labels of image. The energy minimization process of Graph Cuts can find a global solution of segmentation while the flooding process of Watershed can only get a suboptimal solution for the segmentation. Both use the user-defined label as the starting region to expand the label region based on the color distance of the neighborhood pixels. However, finding out the global solution requires more time in the energy minimization process. Although some fast algorithms are proposed before, the speed is still slow especially for multi-label classification. At the other hand, the flooding process is much more efficient and can achieve real time classification with the use of prior queue. Their performances are similar. Both of the methods have the hard constrain characteristic.

B. Segmentation based on Random Walks

In [12], Grady proposed a multi-labels Random Walks segmentation. The system is aiming at finding the minimum Dirichlet integral by solving a linear equation to get the probability distribution for the un-label pixels

$$L\vec{v} = \vec{b} \quad (4)$$

where $\vec{v}$ is the vector of pixel probabilities indicating a random walker starting from that pixel first reaches a marked pixel. Matrix L is a Laplacian matrix describing the graph connections. $\vec{b}$ is the boundary vector and represents the boundary of the unmark region. For the un-label regions, the corresponding probabilities are found by solving the linear equation (4). However the system is sensitive to the noise. To tackle this problem, Scale-Space Random Walks proposed by Richard [13] is used to reduce the influence of the noise. The only difference is that arithmetic average is used to replace the geometric average to generate the final probabilities. The first purpose is to keep the sum of probabilities for difference labels of one pixel is equal to one, denoted by:

$$1 = \sum_{k=1}^{N} P_k(i,j) \qquad (5)$$

where $P_k(i,j)$ denotes the probability of pixel *C(i,j)* reaches pixel with k label value. N is the number of distinct labels. The other purpose of using arithmetic average is to reduce the computation. For N distinct labels, only N-1 linear equations are need to be solved. The final label probability can be found by one minus the sum of probability of the other N-1 labels since the sum of probabilities for difference labels of one pixel is equal to one.

The Random Walks can be interpreted as a circuit problem as the edge can be simulated as the resistors. For each of labels, its corresponding depth label value L_k, translated from a lookup tables, is used as the voltage source. Therefore, the voltage for each pixels of depth label L_k is $P_k(i,j)L_k$, which is the depth value of pixels *C(i,j)* for label L_k. The final depth value is equal to the sum of voltage from all of the distinct depth labels as following:

$$D(i,j) = \sum_{k=1}^{N} P_k(i,j)L_k \qquad (6)$$

The depth map from the Scale-Space Random Walks is smoother than Random Walks inside the object. Although more computation is need in the Scale-Space Random Walks, the probabilities can be found by solving the linearly equations and this process is very efficient.

C. *Final depth map generation*

Depth Prior [9] is used to generate the final depth map. The Depth map from Watershed is appended to the edge weight of the Random Walks. The edges are calculated by equation (2) with the same parameters, but except $\gamma = 1$. The edge distance is combined with the information of depth from Watershed by using the following equation:

$$d(\vec{c_i}, \vec{c_j}, d_i, d_j|\alpha) = \sqrt{d(\vec{c_i}, \vec{c_j})^2 + (\alpha(d_i - d_j))^2} \qquad (7)$$

Where $d(\vec{c_i}, \vec{c_j})$ are defined in section II. d_i and d_j are the normalized depth labels of pixels i and j from Watershed. α is chosen to equal to 0.5. The final depth map is generated by Scale-Space Random Walks. By adding the depth map of Watershed as the edge weighting in the Scale-Space Random Walks, the edges values are increased only at the object boundary region. Thus the probabilities walk through the edge will decrease and the edges of depth can be preserved more clearly after the Random Walks.

IV. EXPERIMENTAL RESULTS

The proposed method is compared with the approach of Phan [9]. The depth maps generated from Graph Cuts and Watershed are shown in Fig. 3 and Fig. 5. Both of them can preserve the sharp boundaries in depth maps. The differences are caused by the usage of difference edge cost equations. For the region has very obviously color difference, they have the relative similar performance, like the region of sky and tower in Fig. 3, while the differences of depth map exist in the region with high similarities, like the grass in Fig. 3 (a). To generate the stereo image, input image is used as the output right-eye view and only left-eye view is generated. It is based on the 3D video quality and eye dominance characteristic of human perception [14] that around 70% people are right eye dominance. From the subjective evaluation, the final 3D perception quality are very similar, as shown in Fig.3 (i) and (j) while the computation loading of Graph Cuts is much higher than that of the Watershed. To generate the depth map of Fig. 3 (c), 186979ms is used while to generate Fig. 3 (f), it cost only 98ms for the machine with Intel(R) i7 CPU 293GHz and 4GB RAM.

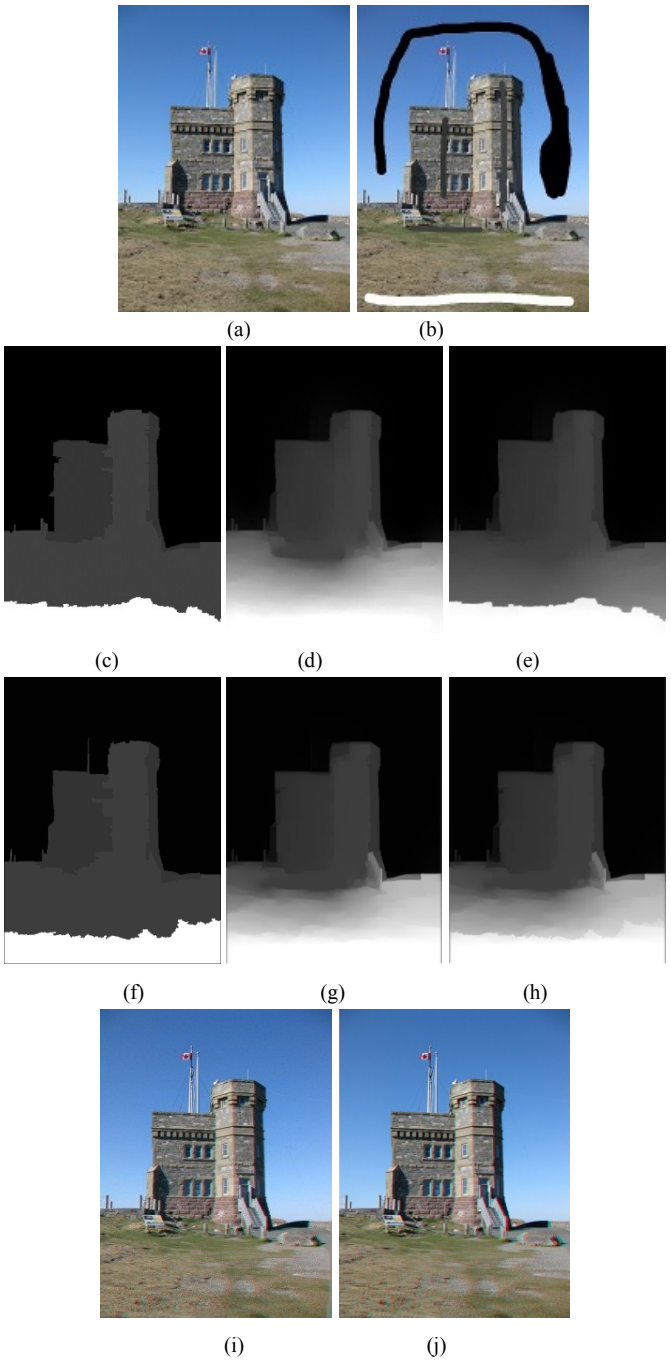


Fig. 3: Comparison of Phan's approach and proposed (a) Input color image, (b) Labeled Image, (c) Depth map from Graph Cuts, (d) Random Walks of Phan's method, (e) Final depth map of Phan's, (f) Depth map from Watershed, (g) Depth map from Random Walks of proposed, (h) Final depth map of proposed, (i) Final anaglyph Image of Phan's method and (j) Final anaglyph Image of proposed.

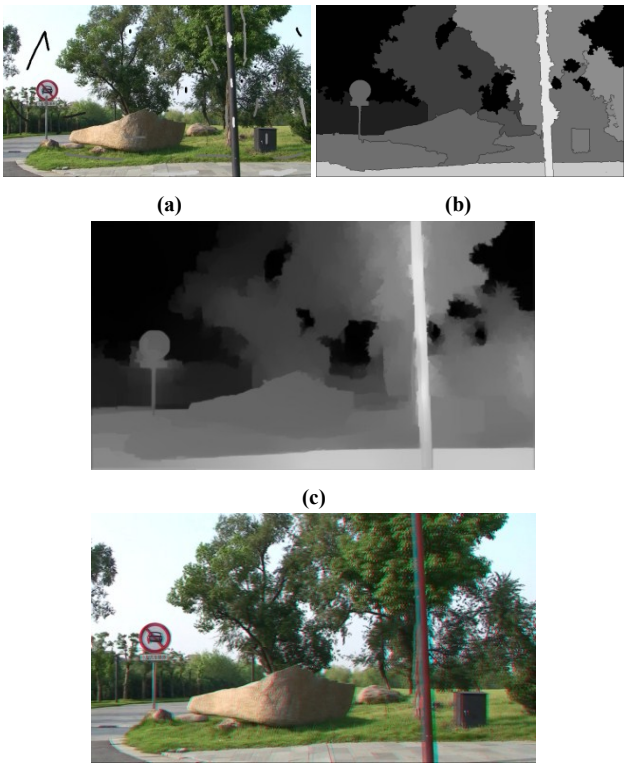


Fig. 4: Results from road image by proposed method : (a) Input image with label (b) depth map from Watershed (c) Generated final depth map (d) Anaglyph image

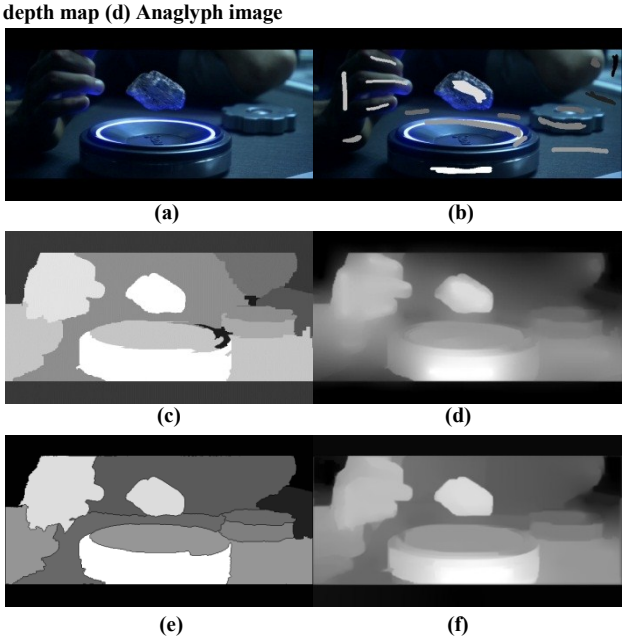


Fig. 5: Comparison of Phan's approach and proposed (a) Input color image, (b) Labelled Image, (c) Depth map from Graph Cuts, (d) Final depth map of Phan's (e) Depth map from Watershed, (h) Final depth map of proposed

For some complicated image, like road sequence [15], the results are shown in Fig. 4. More user-defined labels are required to generate a better depth map, as the result is presented in Fig. 4 (c). From the results of Fig. 4 (b), the boundary edge of telegraph pole in the depth map does not match the color edge as the color information is very similar around the telegraph pole. Beside this, the other depth boundaries match the color edges well. The final depth map generated from the Random Walks can eliminate most of the mismatch of depth edges and the color edges from the Watershed.

Finally, the main problem for the semi-automatic 2D to 3D approach is that different initial labels may result various output depth maps. For complicated image, more labels are required to generate better results. The investigation of simplifying the labeling process and decreasing the impaction of initial label are our next research stages.

V. CONCLUSION

A fast semi-automatic 2D-to-3D conversion is presented in this paper. By using Watershed Segmentation based on priority queue and Random Walks, the final depth map with sharp depth boundaries and fine detail inside the objects is generated. Since the computation of Watershed and Random Walks are not intensive, the conversion process can be very fast and generate a good quality depth map. Experimental results demonstrate this approach can generate the similar results compared with other computational intensive approach and it is suitable for the development of 3D application of the smart phone devices.

REFERENCES

[1] C.C. Cheng; C.T. Li; L.G. Chen; , "A 2D-to-3D conversion system using edge information," Consumer Electronics (ICCE), 2010 Digest of Technical Papers International Conference on , vol., no., pp.377-378, 9-13 Jan. 2010

[2] R. Phan, R. Rzeszutek,D. Androutsos , "Semi-automatic 2D to 3D image conversion using a hybrid Random Walks and graph cuts based approach,"Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on , vol., no., pp.897-900, 22-27 May 2011

[3] L.M. Po; X. Xu; Y. Zhu; S. Zhang; K.W. Cheung; C.W. Ting; , "Automatic 2D-to-3D video conversion technique based on depth-from-motion and color segmentation," Signal Processing (ICSP), 2010 IEEE 10th International Conference on , vol., no., pp.1000-1003, 24-28 Oct. 2010.

[4] L. Schnyder; O. Wang, A. Smolic; "2D to 3D Conversion Of Soports Content Using Panoramas", International Conference on Image Processing (ICIP), 2011 IEEE, vol. no., pp. 2001-2004, 11-14 Sept. 2011

[5] J.I. Jung; Y.S. Ho; , "Depth map estimation from single-view image using object classification based on Bayesian learning," 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2010 , vol., no., pp.1-4, 7-9 June 2010

[6] F.H. Cheng and Y.H. Liang, "Depth map generation based on scene categories", J. Electron. Imaging 18, 043006, Nov 16 2009

[7] Z. Zhang; Y. Wang; T. Jiang and W. Gao, "Visual Pertinent 2D-To-3D Video Conversion By Multi-cue Fusion", International Conference on Image Processing (ICIP), 2011 IEEE, vol. no., pp. 925-928, 11-14 Sept. 2011

[8] M. Guttmann, L. Wolf, and D. Cohen-Or, "Semi-automatic Stereo Extraction from Video Footage," Proc. IEEE ICCV,2009

[9] R. Phan, R. Rzeszutek, D. Androutsos, "Semi-Automatic 2D to 3D Image Conversion Using Scale-Space Random Walks and a Graph Cuts based Depth Prior," ICIP2011, pp. 881-884, Sep 2011.

[10] Y. Boykov and G. Funka-Lea, "Graph Cuts and Efecient N-D Image Segmentation," Intl. Jnl. of Comp. Vis., vol. 2, no. 70,pp. 109–131, 2006

[11] F. Meyer, "Color image segmentation," Image Processing and its Applications, 1992., International Conference on , vol., no., pp. 303- 306

[12] L. Grady, "Random Walks for Image Segmentation," Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.28, no.11, pp.1768-1783, Nov. 2006.

[13] R. Rzeszutek, T. El-Maraghi, and D. Androutsos, "Image Segmentation using Scale-Space Random Walks," Intl. Conf. on Digital Signal Processing (DSP), pp. 1-4, July 2009

[14] H. Kalva, L. Christodoulou, L. M. Mayron, O. Marques, and B. Furht, "Design and evaluation of a 3D video system based on H.264 view coding," Proceeding of the 2006 International Workshop on Network and Operating System Support for Digital Audio and Video, Newport, Rhode Ishland, Nov., 2006.

[15] G. Zhang, J. Jia, T.T. Wong and H. Bao. Consistent Depth Maps Recovery from a Video Sequence. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 31(6):974-988, 200