信学技報 MVE2009-57, Vol.109, No.215, pp.131-136 (2009)

社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report

ウォークスルーシミュレータに適した写実的アバタ生成のための コードブックに基づく背景差分

朴 鴈振[†]　大隈 隆史[†]　蔵田 武志[†]

† 産業技術総合研究所サービス工学研究センター
〒305-8568 つくば市梅園 1-1-1 中央第二
E-mail: †{anjin.park,takashi-okuma,t.kurata}@aist.go.jp

あらまし　筆者らは、仮想環境と共に遠隔の人物の写実的アバタを提示することで、移動・作業と会話を含むタスクに関する評価実験や遠隔作業支援を行うことのできるウォークスルーシミュレータ（WTS）を開発している。本稿では、写実的アバタ生成のためのカメラ映像からの前景（人物）抽出手法として、１）実時間性の高さ、２）消費メモリの少なさ、３）ノイズの少なさ、４）前景境界の正確さ、の４つの指標を考慮した、コードブックに基づくマルコフランダム場（MRF）モデルを用いた背景差分法を提案する。本手法では、実時間性の高い実装と消費メモリの抑制のためにコードブックに基づくアプローチを、ノイズ低減や正確な前景輪郭獲得のためにエッジを保存する MRF モデルを採用する。MRF モデルは尤度項を推定するために確率的な観測を必要とするが、提案手法では、オンライン混合正規分布を用いてコードブック中の各コードワードの確率を推定し、グラフカットを用いた MAP-MRF(MAP: Maximum A-Posterior)アプローチにより背景差分を行う。

キーワード　ウォークスルーシミュレータ，写実的アバタ，背景差分，マルコフランダム場，グラフカット

Codebook-based Background Subtraction to Generate Photorealistic Avatars in a Walkthrough Simulator

Anjin PARK, Takashi OKUMA, and Takeshi KURATA

Center for Service Research, AIST
E-mail: {anjin.park,takashi-okuma,t.kurata}@aist.go.jp

Abstract　Foregrounds extracted from the background, which are intended to be used as photorealistic avatars for simulators in a variety of virtual worlds, should satisfy the following four requirements: 1) real-time implementation, 2) memory minimization, 3) reduced noise, and 4) clean boundaries. Accordingly, the present paper proposes a codebook-based Markov Random Field (MRF) model for background subtraction that satisfies these requirements. In the proposed method, a codebook-based approach is used for real-time implementation and memory minimization, and an edge-preserving MRF model is used to eliminate noises and clarify boundaries. The MRF model requires probabilistic measurements to estimate the likelihood term, but the codebook does not use any probabilities to subtract the background. Therefore, the proposed method estimates the probabilities of each codeword in the codebook using an online mixture of Gaussian (MoG), and then MAP-MRF (MRF: Maximum A-Posteriori) approaches using a graph-cuts method are used to subtract the background. In experiments, the proposed method showed better performance than MoG-based and codebook-based methods on the Microsoft DataSet and was found to be suitable for generating photorealistic avatars.

Keyword　Walkthrough Simulator, Photorealistic Avatar, Background Subtraction, Markov Random Field, Graph Cuts

1. Introduction

Since the integration of user representations into the virtual world can provide cues for communications in the virtual world, extracting users from given data, such as images captured by a camera, and representing the extracted users in the virtual world has become a popular topic in the field of virtual reality (VR) [1-3]. For example, Gross et al. presented an immersive and 3D video acquisition environment for virtual design, and used stereo cameras and IR-lights to extract users from a simulated environment [1]. Rhee et al. proposed an infrared-based image segmentation to enable concurrent acquisition and projection in an immersive environment, and used two color cameras and two additional b/w cameras in the near IR spectrum [2]. Yasuda et al. introduced a thermal vision camera in order to realize a human segmentation based on natural heat emissions of the body [3]. However, they used expensive equipment to extract users that were suitable for their simulated environments.

To extract users from captured images, the present paper describes the use of background subtraction, where the objective is to separate the new objects from a background [4-7,10]. The present paper proposes a background subtraction method suitable for generating photorealistic avatars for display in the virtual world. The simulated environment of the present paper, referred to as the *Walkthrough Simulator* (WTS), and the requirements for extracting images to be integrated into the virtual world are described in Section 1.1, and the proposed background subtraction method is briefly introduced in Section 1.2.

1.1. Walkthrough Simulator

Before constructing or renovating houses or buildings, details such as the suitability of the floor layout and navigation signs and whether users will feel comfortable in the buildings should be considered. VR techniques are used to investigate virtual structures in detail. However, since information is generally displayed through monitors and a keyboard or mouse is used to navigate the structure, it is difficult to evaluate the relationship between the details of the structure and the sense of absolute direction of the user. Therefore, we are developing the WTS in order to enable subjects to navigate virtual constructions from the perspective of the customer. The final goal is to create a 360-degree virtual world displayed around the subject as they physically walk through the virtual world. The hexahedral-shaped device shown in the center of Fig. 1 is the WTS. The virtual building is displayed inside the device using multi-projectors as shown on the right-hand side of Fig. 1. The subject stands in the middle of the WTS and move his or her feet to navigate the virtual buildings.

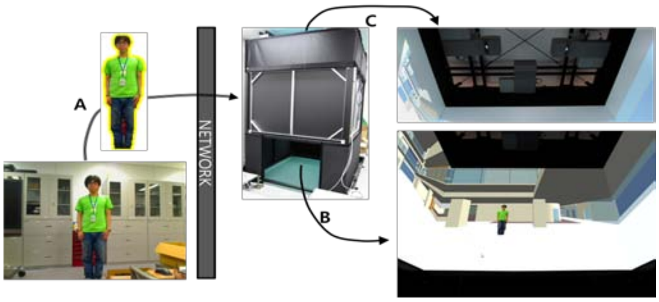


Fig. 1. Schematic diagram of the WTS: (A) photorealistic avatar extracted from the modeled background, (B) photorealistic avatar integrated into the virtual building, and (C) multi-projectors.

In some buildings, such as public institutions, guides provide instructions to customers or visitors to help them reach their destination. In the virtual building, guides are displayed as avatars. In the present study, we use a *Photorealistic Avatar*, in which the appearance of an actual person is used as CG texture, as a guide. The present paper focuses on displaying the photorealistic avatar in virtual buildings. The image of the person that is used to create the photorealistic avatar is extracted by the camera in front of the simulator, as shown in Fig. 1(A), and the photorealistic avatar is displayed in a fixed location in the virtual world inside the simulator, as show in Fig. 1(B).

In the present paper, it is assumed that the moving foreground in front of the fixed camera outside the simulator is an individual whose image will be used to generate the photorealistic avatar. Therefore, background subtraction is used to extract the appearance of the guide from images captured by a camera. There are four requirements for the background subtraction: 1) extraction must be performed in real-time, 2) memory consumption must be limited, 3) the image must be extracted with little noise, and 4) the boundaries of the avatar must be clear.

1.2. Proposed Background Modeling

The proposed method integrates a codebook-based approach, which helps to perform extraction in real time and reduces the required memory, and an edge-preserving MRF model, which can eliminate noise and generate clear boundaries. Although the codebook-based algorithm [10] can model an adaptive and compact background over a long period of time with limited memory, it cannot be used as likelihood term in the edge-preserving MRF, because the similarity (rather than the probability) is used to compare input pixels with the modeled background. Therefore, online mixture of Gaussians (MoG) is used to estimate the probabilities for all codewords in the codebook. In addition, the proposed method estimates the smoothness term using the codebook-based method in order to substantially reduce extraction errors caused by high-contrast edges in cluttered backgrounds, thereby reducing errors on the boundaries of extracted foregrounds.

2. MRF Modeling for Background Subtraction

2.1. Related Research

The simplest background model assumes that pixel values can be modeled by a single Gaussian distribution [4]. However, this basic model cannot handle multiple backgrounds, such as trees moving in the wind. The MoG has been used to model non-static backgrounds [5]. However, it is difficult to detect sudden changes in the background when the learning rate is low, and slowly moving foreground pixels will be absorbed in the background model when the learning rate is high [10]. Sheikh and Shah [7] proposed a MAP-MRF framework, which results in clear boundaries without noise by enforcing spatial context in the process, but this technique [7] cannot be used when long-periods of time are needed to sufficiently model the background, primarily due to memory constraints, because they used a kernel density estimation technique [10]. In order to address the memory constraint problem, Kim et al [10] proposed a codebook-based background subtraction algorithm intended to model pixel values over long periods of time. However, since this algorithm did not evaluate probabilities, but only calculated the distance from the cluster means, it is hard to extent this algorithm to the MAP-MRF framework.

2.2. Energy Function

Many tasks in computer vision involve assigning a label to each pixel p in an input image P, i.e. pixels are assigned labels in some finite set $\mathcal{L}$. A common constraint is that the labels should vary smooth almost everywhere while preserving sharp discontinuities that may exist, e.g. at object boundaries. The goal is to label labeling **F** that assigns each pixel $p \in \mathbf{P}$ a label $f_p \in \mathcal{L}$, where n is the number of the pixels. These tasks can be naturally formulated in terms of energy minimization, and this framework seeks the labeling **F** that minimizes the energy function as follows:

$$P(\mathbf{f}|obs) = \mathrm{argmax}_{\mathbf{f}\in\mathbf{F}}\left(\sum_{s\in\mathcal{S}} D_s(f_s) + \sum_{\{s,s'\}\in N} V_{s,s'}(f_s, f_{s'})\right), \quad (1)$$

where N is all pairs of neighboring pixels.

In the present, $D_s(f_s)$ in Eq. 1 is referred to as the likelihood term derived from the modeled background, which reflects how each pixel fits into the modeled data given for each label, and $V_{s,s'}(f_s, f_{s'})$ is referred to as the smoothness term that encourages spatial coherence by penalizing discontinuities between neighboring pixels s and s'. In addition, $V_{s,s'}(f_s, f_{s'})$ is replaced by $V_{s,s'} \cdot \delta(f_s, f_{s'})$, where $\delta(f_s, f_{s'})$ denotes the delta function defined by 1 if $f_s \neq f_{s'}$, and otherwise denotes the delta function defined by 0. Thus, this is penalty term when two pixels are assigned different labels. The energy function is derived in context of MRF: a minimum of E corresponds to a maximum a posteriori (MAP) of labeling **F** [x].

2.3. Graph Cuts

A graph-cuts method [11] is used to minimize the energy function (Eq. 1), because this method showed the best performance among the conventional energy minimization algorithms [12]. The procedure for energy minimization using the graph-cuts method includes building a graph, wherein each cut defines a single configuration, and the cost of a cut is equal to the energy of its corresponding configuration [12].

For the graph-cuts method, a graph $G = \langle \nu, \varepsilon \rangle$ is first constructed with vertices corresponding to the sites. Two vertices, namely *source* (Src) and *sink* (Sin), also referred to as terminals, are needed in order to represent two labels, and each vertex has two additional edges, $\{s, Src\}$ and $\{s, Sin\}$. Therefore, the set of vertices ν and edges ε are $\nu = \mathcal{S} \cup \{Src, Sin\}$ and $\varepsilon = N \bigcup_{s\in\mathcal{S}}\{\{s, Src\}, \{s, Sin\}\}$, where N are referred to as *n-links* (neighboring links) and $\{s, Src\}$ and $\{s, Sin\}$ are referred to as *t-links* (terminal links). The weights of the graph are set for both *n-links* and *t-links*, where the *t-links* connecting each terminal and each vertex correspond to the likelihood term and *n-links* connecting neighboring vertices correspond to the smoothness term.

Note that the background subtraction problem can be solved by finding the least energy consuming configuration of the MRF among the possible assignments of the random variables **F**. Minimizing the energy function defined in Eq. 1 is equivalent to finding the cut with lowest cost, because the costs of two terms are assigned to the weights of the graph. Specific labels are then assigned to two disjointed sets connected by *Src* and *Sin* by finding the cut with the lowest cost in the graph. The minimum-cost cut of the graph can be computed through a faster version of max-flow algorithm, proposed in [13].

3. Proposed Energy Function

3.1. Likelihood Term

The likelihood term is derived from the modeled background data to measure the cost of assigning the label f_p to the pixel p, and $D_p(f_p)$ is defined as follows:

$$\begin{cases} D_p(f_p = \text{fore}) = 1, & D_p(f_p = \text{back}) = 0, & \text{if } P(p) < T_f, \\ D_p(f_p = \text{fore}) = 0, & D_p(f_p = \text{back}) = 1, & \text{if } P(p) > T_b, \\ D_p(f_p = \text{fore}) = T_b^p, & D_p(f_p = \text{back}) = P(p), & \text{otherwise}, \end{cases}$$

where T_f and T_b are thresholds for hard constraints [14] in constructing graphs, T_b^p is a threshold to extract moving objects from the background, and $P(p)$ is the probability that a pixel p is included in the background. In the present paper, the codebook-based algorithm and MoG are used to estimate the probabilities for the background.

The codebook algorithm is used to construct a background model from long input sequences, and adopts a quantization technique to minimize the required memories. For each pixel, the codebook algorithm builds a codebook consisting of one or more codewords. Samples at each pixel are quantized into a set of codewords based on color and brightness information. The background is then encoded on a pixel-by-pixel basis. For more information on how to construct the codebook from input background images, please refer to the paper [10].

In the case of using codebook-based algorithms, it is difficult to use an MRF because it does not evaluate probabilities, but rather calculates the distance from the RGB vectors and the brightness of the codewords.

To evaluate the probabilities from the codebooks, a mixture of K Gaussian distributions proposed by Stauffer and Grimson [5] is chosen to model the recent history of each pixel, which is included in the same codewords. The probability of observing the current pixel value $\mathbf{x}_\mathrm{t}$ is $P(\mathbf{x}_\mathrm{t}) = \sum_{i=1}^{K} w_{i,t} * \eta(\mathbf{x}_t, \boldsymbol{\mu}_{i,t}, \boldsymbol{\Sigma}_{i,t})$, where K is the number of distributions, $\boldsymbol{\mu}_{i,t}$ is an estimate of the weight of the ith Gaussian in the mixture at time t, $\boldsymbol{\mu}_{i,t}$ and $\boldsymbol{\Sigma}_{i,t}$ are the mean value and covariance matrix, respectively, of the ith Gaussian in the mixture at time t, and η is a Gaussian probability density function. In the experiments, K is determined by the number of frames used for background modeling, and the covariance matrix is assumed to be of the following form: $\boldsymbol{\Sigma}_{k,t} = \sigma_k^2 \mathbf{I}$.

3.2. Smoothness Term

Since a common constraint is that the labels should vary smoothly almost everywhere while preserving sharp discontinuities that may exist, e.g., at boundaries [11], the costs of the smoothness term are assigned for discontinuity preservation between two neighboring pixels, and we use generalized Potts model [11]. As such, $V_{p,p'}$ is defined as follows:

$$V_{p,p'} = dis(p,p')^{-1} e^{(-\beta \cdot \|p-p'\|^2)}, \tag{2}$$

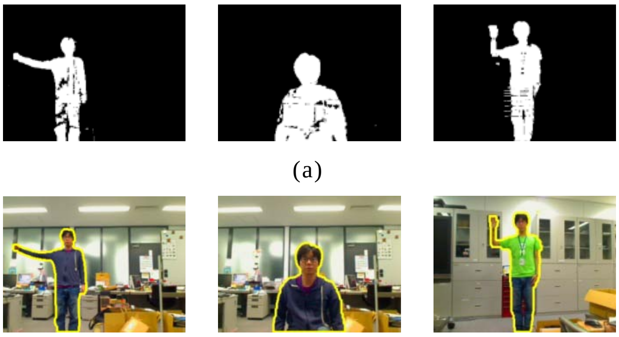
where the contrast term $\|p - p'\|^2$ denotes the dissimilarity between two pixels p and p', and $dis(\cdot)$ is the Euclidean distance between neighboring pixels in the image domain. When $\beta = 0$, the smoothness term is simply the well-known Ising model, which promotes smoothness everywhere. However, it has been shown that it is more effective to set $\beta > 0$, because this relaxes the tendency to smooth regions of high contrast. The constant β is chosen to be $\beta = (\langle \|p - p'\|^2 \rangle)^{-1}$, where $\langle \cdot \rangle$ denotes the expectation over an image. This choice of β ensures that the exponential term in Eq. 2 switches appropriately between high and low constants [14].

However, when the scene contains a cluttered background, notable segmentation errors often occur around the boundary, which generates flickering artifacts in the final results displayed in the virtual world [9]. These errors occur because the MRF model contains two terms for color and contrast. A straightforward idea is to subtract the contrast of the background image from the current image [9]. However, since only one background image is used for this approach, the nonstationary background motion that is ubiquitous in the real world cannot be modeled.

To overcome this problem, the contrast of the background is modeled using the codebook-based algorithm. The difference is that the codebook for the smoothness terms uses $V_{p,p'}$ instead of brightness values as input and does not use color information. This means modeling contrasts between adjacent pixels. This approach helps not only to eliminate the flickering artifacts but also facilitates the use of the generalized Potts model.

4. Experimental Results

Background subtraction was used to generate a photorealistic avatar in the virtual world for the WTS. Section 4.1 presents the resultant images displayed in the WTS, and Section 4.2 presents a quantitative evaluation to verify the effectiveness of the proposed method. All of the experiments were carried out on a 2.40-GHz Pentium 4 CPU.



(a)

(b)

Figure 2. Resultant images of (a) the codebook-based method [10] and (b) proposed method.

4.1. Simulated Environment

The proposed method was based on the codebook-based method [10]. Images resulting from use of the codebook-based method and the proposed method are shown in Fig. 2(b) and (b), respectively. As shown in Fig. 2, the results of the codebook-based method include some noise and holes in extracted regions. However, by applying an edge-preserving MRF framework, the proposed method includes no noise or holes and has clean boundaries. The photorealistic avatar, based on the resultant images presented in Fig. 2(b), was integrated into the WTS as shown in Fig. 3. In addition, MoG-based [5] and codebook-based [10] methods were compared to the proposed method, as shown in Fig. 4.

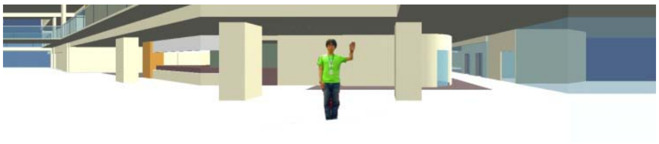
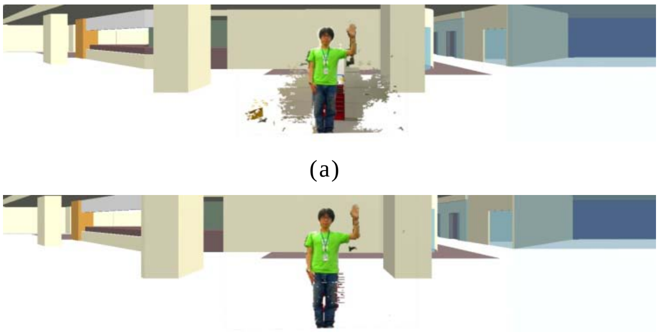


Figure 3. Photorealistic avatar integrated into a virtual building.



(a)

(b)

Figure 4. Guide representation in a virtual building using (a) MoG [5] and (b) codebook-based [10] methods.

4.2. Qualitative Analysis

We tested four data sets described in [6]: Waving Trees, Camouflage, Time of Day, and Moved Object. We chose these four sets because the background images to be modeled might include nonstationary background motion, as in the Waving Trees and Camouflage sets, and because the sequential background images might change gradually as a result of changing light conditions throughout the day, as in the case of the Time of Day set. In the present study, in contrast to [6], moving objects are considered to be in the foreground, because the photorealistic avatar can use objects to express information to the subject. In the experiments, codebook-based [10] and MoG-based [5] results were compared with the results of the proposed method, and the results of these tests are shown in Figs. 5–8.

The accuracy rates were evaluated by two criteria: the number of false positives and the number of false negatives. The number of false positives is the number of foreground pixels that were misidentified, and the number of false negatives is the number of background pixels that were identified as foreground pixels. As shown in Table 1, where M represents the MoG-based method, C represents the codebook-based method, MC represents the graph-cuts method using MoG, and P represents the proposed method, the proposed method had the best performance, except in the case of the Time of Day data set, as shown in Fig. 7. Since brightness values were used to deal with shadows, the proposed approach worked poorly in dark areas of images. Therefore, the leg regions of the human were not extracted by the proposed method. On the other hand, since shadow regions are included in the results shown in Fig. 6, the proposed method had better performance than the MoG-based methods. The processing times for each step of the proposed method are presented in Table 2. Approximately nine frames per second could be extracted using the proposed method.

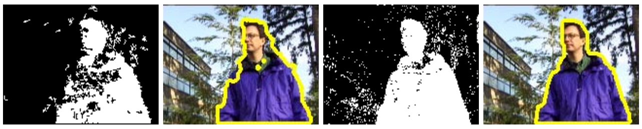


Figure 5. Sample results for Waving Threes obtained using (a) the MoG-based method, (b) the graph-cuts method using MoG, (c) the codebook-based method, and (d) the proposed method.

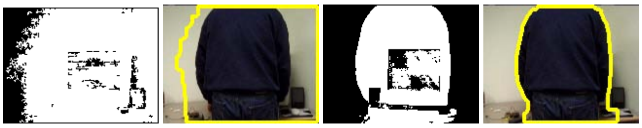


Figure 6. Sample results for Camouflage obtained using (a) the MoG-based method, (b) the graph-cuts method using MoG, (c) the codebook-based method, and (d) the proposed method.

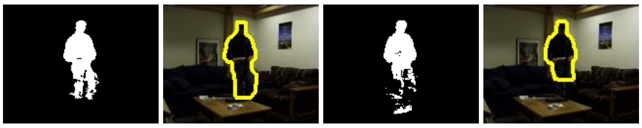


Figure 7. Sample results for Time of Day obtained using (a) the MoG-based method, (b) the graph-cuts method using MoG, (c) the codebook-based method, and (d) the proposed method.

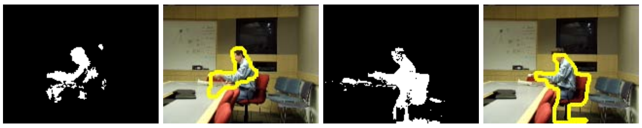


Figure 8. Sample results for Moved Object obtained using (a) the MoG-based method, (b) the graph-cuts method using MoG, (c) the codebook-based method, and (d) the proposed method.

Table 1. False positives and false negatives (%)

	Fig. 5		Fig. 6		Fig. 7		Fig. 8	
	F.P	F.N	F.P	F.N	F.P	F.N	F.P	F.N
M	6.9	2.4	16.1	38.1	5.2	1.4	11.1	9.8
C	3.2	2.6	9.1	1.2	9.8	0.5	12.3	8.3
MC	3.1	0.2	5.9	19.3	2.2	0.9	5.1	9.3
P	**0.02**	**0.04**	**0**	**0.98**	**11.2**	**0.1**	**1.6**	**8.1**

Table 2. Processing times for each step of the proposed method (msec)

Resolution	Codebook Construction	MoG	Graph construction	Graph cuts
160x120	8	5	7	25
320x240	20	10	20	60

5. Conclusion

In the present paper, we proposed a codebook-based MRF model for background subtraction to generate a photorealistic avatar displayed in the virtual world. Although an edge-preserving MRF can eliminate the noise and generate suitable object boundaries, the MRF depends on how the likelihood terms in the energy function are estimated. The proposed method uses a codebook-based method to estimate the likelihood term, which reduces the required memory and enables real-time implementation. Moreover, the proposed method used online MoG to estimate the probability for each codeword, which resulted in minimization of the required memory, reduced noise, and clean boundaries. In addition, the proposed method enabled the photorealistic avatar to be displayed clearly in the virtual world, as compared with previously proposed methods, such as codebook and MoG.

However, the proposed method was not able to extract the foreground in dark regions because brightness values were used to handle shadows. Therefore, in future studies, we intend to investigate how to extract the foreground in the dark regions more effectively. Moreover, we intend to extend the proposed method to extract the foreground inside a WTS that contains non-static backgrounds due to virtual world displayed inside the WTS.

Acknowledgement: This research was entrusted by the Ministry of Economy, Trade and Industry (METI).

References

[1] M. Gross, S. Wuermlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. K-Meier, T. Svoboda, L. Gool, S. Lang, K. Strehlke, A.V. Moere, O. Staadt. Blue-C: a Spatially Immersive Display and 3D Video Portal for Telepresence, ACM Trans. Graph., vol. 22, no. 3, pp. 819-827, 2003.

[2] S-M, Rhee, R. Ziegler, J. Park, M. Naef, M. Gross, M-H. Kim. Low-Cost Telepresence for Collaborative Virtual Environments, IEEE Trans. Vis. Comput. Graph., vol. 13, no. 1, pp. 156-166, 2007.

[3] K. Yasuda, T. Naemura, and H. Harashima. Human Region Segmentation from Video, IEEE Comput. Graph. Appl., vol. 24, no. 1, pp. 26-30, 2004.

[4] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Real-Time Tracking of the Human Body, IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 7, pp. 780-785, 1997..

[5] C. Stauffer and W. Grimson. Learning Patterns of Activity using Real-Time Tracking, IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 8, pp. 747-757, 2000.

[6] K. Toyama, J. Krumm, B. Brumitt, B. Meyers. Wallflower: Principles and Practice of Background Maintenance, Proc. IEEE Conf. Comp. Vis., pp. 255-261, 1999.

[7] Y. Sheikh and M. Shah. Bayesian Modeling of Dynamic Scenes for Object Detection, IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 11, pp. 1778-1792, 2005.

[8] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Image, IEEE Trans. Pattern Anal. Mach. Intell., vol. 6, no. 6, pp. 721-741, 1984.

[9] J. Sun, W. Zhang, X. Tang, H-Y. Shum. Background Cut, Proc. European Conf. on Comput. Vis., Part II, pp. 628-641, 2006

[10] K. Kim, T.H. Chalidabhongse, D. Harwood, and L. Davis. Real-Time Foreground-Background Segmentation using Codebook Model, Real-Time Imaging, vol. 11, no. 3, pp. 172-185, 205.

[11] Y. Boykov, O. Veksler, and R. Zabih. Fast Approximation Energy Minimization via Graph Cuts, IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 11, pp. 1222-1239, 2001.

[12] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A Comparative Study of Energy Minimization Methods for Markov Random Fields, IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 6, pp. 1068-1080, 2008.

[13] Y. Boykov and V. Kolmogorov. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision, IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 9, pp. 1124-1137.

[14] Y. Boykov and G. Funka-Lea. Graph Cuts and Efficient N-D Image Segmentation, Int. J. Comput. Vis., vol. 70, no. 2, pp. 109-131, 2006.

[15] S.Z. Li. Markov Random Field Modeling in Computer Vision, Springer, 2001.