Optical Engineering

OpticalEngineering.SPIEDigitalLibrary.org

Near-real-time stereo matching method using temporal and spatial propagation of reliable disparity

Sungil Kang
Hyunki Hong

SPIE.

Near-real-time stereo matching method using temporal and spatial propagation of reliable disparity

Sungil Kang[a] and Hyunki Hong[b,*]
[a]Chung-Ang University, Graduate School of Advanced Imaging Science and Arts, 221 Huksuk-dong, Dongjak-ku Seoul, 156-756, Republic of Korea
[b]Chung-Ang University, School of Integrative Engineering, 221 Huksuk-dong, Dongjak-ku Seoul, 156-756, Republic of Korea

Abstract. A stereo approach to resolve the occlusion problem in stereo video sequence is introduced. We define a measure to evaluate the reliability of an initial disparity in combination with a left-right consistency check. An initial matching cost volume is computed with an absolute difference-census measure. In the spatial propagation stage, the outlier with a low reliability value is replaced/updated with the reliable disparity information in the support region. Because previous methods establish correspondence on a per-frame basis, they cannot obtain temporally coherent disparity results over a stereo sequence. In order to overcome the occlusion problem in a dynamic situation, we employ the modified codebook with color, disparity, reliability, array of the matching cost, and final access time in a temporal propagation procedure. Experimental results show that the proposed algorithm with general-purpose computing on graphics processing units (GPGPU) provides better performance when applied to disparity maps of real-time indoor/outdoor scenes. [DOI: 10.1117/1.OE.53.6.063107]

Keywords: stereo matching; correspondence; disparity map; reliability; compute unified device architecture programming.

Paper 140274 received Feb. 17, 2014; revised manuscript received Apr. 30, 2014; accepted for publication May 23, 2014; published online Jun. 24, 2014.

1 Introduction

Dense stereo matching is one of the most extensively studied topics in computer vision.[1–18] It is an effective three-dimensional reconstruction method, since it can usually recover a dense disparity map from a stereo view.

Kinect sensor using an infrared band captures precise range information, but it is only for indoor use and its operation range is substantially limited. Stereo systems are useful in both indoor/outdoor applications, such as robot navigation and autonomous vehicle control.

Stereo matching algorithms are classified widely into local and global matching methods, In addition, stereo algorithms are described in more detail according to four individual components in stereo matching, matching cost computation, cost aggregation, disparity computation, and disparity refinement.[1] Most global stereo methods are computationally expensive and involve many parameters, while local stereo methods are generally efficient and easy to implement.

In the local approaches, most common pixel-based matching costs include absolute difference (AD), normalized cross-correlation (NCC), and Birchfield and Tomasi's measure (BT) to examine the matching cost in a search window. The central problem of local window-based methods is how to determine the size and shape of the aggregation window. Hirschmüller and Scharstein proposed a census method that is robust to brightness changes,[2] but it causes matching ambiguity in image regions with repetitive or similar local structures. Yoon and Kweon assigned different support weights to pixels in the window by using the photometric and geometric relationship with the pixel under consideration, but many problems, including textureless regions, repeated similar patterns, and occlusions, still remain unsolved.[3]

In the global matching approaches, an energy function is used to find the optimal solution in terms of matching cost. It consists of a data term and a smoothness term. The data term represents the degree of image difference between the left and right stereo images according to the disparity level. The smoothness term represents the compensation level of the discontinuity in neighboring pixels. The algorithms make an explicit smoothness assumption, but the search step to find a global solution minimizing the energy function incurs a heavy computational load. The popular energy minimization frameworks, such as graph cuts,[4] belief propagation,[5] and dynamic programming,[6] have attracted attention due to their good performance. Hirschmüller suggested the semiglobal method, which substitutes global and two-dimensional smoothness constraints by the combined one-dimensional constraint in different aggregation directions for pixel-wise matching.[7]

Researchers also developed image segmentation and plane-fitting methods.[8–10] Segmentation methods are based on the assumption that scene structure can be approximated by a set of nonoverlapping planes in the disparity space and that each plane is coincident with at least one homogeneous color segment in the reference image. While segmentation information is generally useful for accurate disparity results, these methods require a large number of computationally demanding iterations. When the color distribution of the foreground object is similar to that of the background element, it is difficult for us to estimate a precise result. The disparity plane-fitting based stereo methods model the scene structure using a set of planar surface patches. These methods estimate an individual plane at each pixel onto which the support region is projected. However, they have a lot of difficulties in finding one of minimum aggregated matching costs among all the candidate planes.

*Address all correspondence to: Hyunki Hong, E-mail: honghk@cau.ac.kr

Blyer et al. and Richardt et al. built a disparity map using temporal propagation,[10,11] but the occlusion problem in various dynamic situations still remain unsolved.

In order to reduce the heavy computational load in the dense matching of stereo views, graphics processing unit (GPU)-based methods were proposed.[11–16] The traditional sum of square difference was used to independently aggregate matching costs in GPU and embedded stereo systems. The GPU-based adaptive window approach can change the shapes of cost aggregation windows according to the content of the local image area, taking into account edges and corners.[13] In Ref. 14, the belief propagation based method is implemented to run at real-time on a GPU. Specifically, compute unified device architecture (CUDA) has been one of the most popular high-performance computing engines to implement real-time stereo matching methods.[15,16]

Some recently proposed methods are suggested to improve both matching accuracy and processing efficiency on a GPU.[17,18] In addition, the stereo video process has different challenges from that in stereo image: the application of techniques on a per-frame basis is not enough to achieve flicker-free and temporally coherent disparity maps. Generally, a video sequence is temporally and spatially correlated with scene elements, such as a human being or objects in an interested scene. However, most of the previous stereo matching methods dealt with correspondence problem on a per-frame basis, so they cannot obtain temporally coherent disparity maps over a stereo video sequence.

The proposed method obtains a more accurate disparity map by using temporal and spatial propagation of reliable disparity information over the stereo sequence. The contribution of this paper consists of three parts. First, we define a measure to evaluate the reliability of an initial disparity map and combine this measure with a left-right consistency (LRC) check. Second, we propose a spatial propagation of the reliable disparity to remove the outliers. Third, we introduce a temporal propagation based on a codebook. Figure 1 shows a flow chart of the proposed algorithm.

To tackle half-occluded (objects scene in one image and not in other) regions in a dynamic situation, we consider background information that is occluded by foreground elements. Several methods have been used for foreground/background segmentation.[19–21] In the generalized mixture of Gaussians, backgrounds with fast variations are not easily modeled with just a few Gaussians. In addition, it is difficult to determine an optimal learning rate to accurately adapt to background changes.[19] The nonparametric technique computes the probability density function at each pixel from many samples using a kernel density estimation.[20] When sampling the background for a long time period, however, this method has a memory constraints problem. The previous codebook with the quantized background values at each pixel was designed to obtain sample values over a long video sequence without making parametric assumptions.[21] We modify the codebook to resolve the occlusion problem in stereo matching by using a temporal correlation over the stereo sequence. We store and use color, reliability, array of the matching cost, and final access time of the scene elements, including background and foreground objects. Our proposed codebook contains temporally coherent information of scene elements over the stereo sequence.

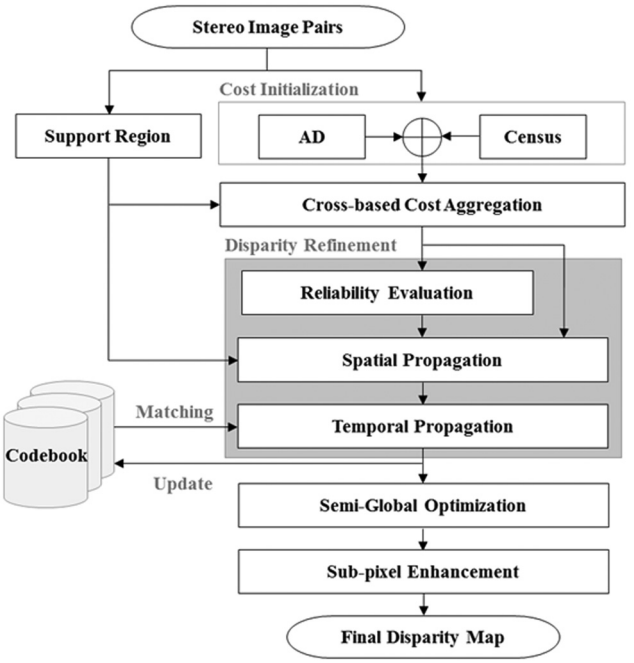


Fig. 1 Proposed flow chart.

2 Proposed Method

2.1 *Initial Matching Cost Computation*

The initial matching cost volume at each pixel and each disparity level is computed using AD-census in parallel, which combines the AD measure and census transform.[17] Because the AD measure examines only the pixel intensity, it is substantially affected by lighting changes. The census transform encodes local image structures with relative orderings of the pixel intensities rather than the intensity value itself to tolerate outliers caused by radiometric changes and image noise.

In the stereo view, the brightness distribution of the left image is different from that of the right image because of different illumination conditions and surrounding environments. A longer baseline length allows us to handle a larger space, but the difference between the two views will increase substantially. So many outlier regions occur in an initial cost volume obtained by the AD-census. To reduce the outlier regions, we aggregate each pixel's matching cost throughout the support region on the assumption that neighboring pixels with similar colors should have similar disparities.[3,17,22]

For each anchor pixel $\mathbf{p}$, an upright cross skeleton of the support region is adaptively constructed with four varying arm lengths determined by color similarity and connectivity constraints. When local cross results are given, a shape-adaptive full support region $U(\mathbf{p})$ can be dynamically built by the process of merging horizontal segments of the crosses in the vertical neighborhood.[22] When a pair of hypothetical correspondences is established [$\mathbf{p} = (x, y)$ in the left image and $\mathbf{p}' = (x', y')$ in the right], we can measure the matching cost between $\mathbf{p}$ and $\mathbf{p}'$ by aggregating the initial cost C_0 in the local support region. The coordinates of $\mathbf{p}$ and $\mathbf{p}'$ are correlated with a disparity hypothesis d:$x' = x - d$ and $y' = y$.

Figure 2 shows an example for a cross skeleton construction of the support region in the Teddy stereo image. The pixel-wise adaptive crosses define the cross skeleton for $\mathbf{p}$ and shape-adaptive support regions are dynamically constructed.

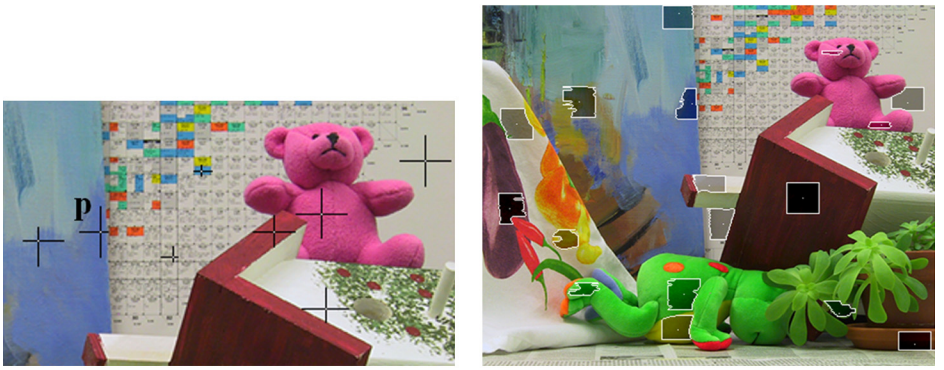


Fig. 2 Construction of cross skeleton and local support region on Teddy image: (a) pixel-wise adaptive cross skeleton at pixel **p** and (b) sample shape-adpative support regions.

In Fig. 2(b), the shaded regions are sample shape-adaptive support regions. In order to symmetrically consider both the left local support region $U(\mathbf{p})$ and the right region $U'(\mathbf{p}')$, we combine two local regions and compute the normalized matching cost C_1 as follows:

$$C_1(\mathbf{p},d)=\frac{1}{\|U_d(\mathbf{p})\|}\sum_{\mathbf{q}\notin U_d(\mathbf{p})}C_0(\mathbf{q},d),$$
$$U_d(\mathbf{p})=\{(x,y)|(x,y)\in U(\mathbf{p}),(x-d,y)\in U'(\mathbf{p}')\}, \quad (1)$$

where $U_d(\mathbf{p})$ is the combined local support region that contains the valid pixels between the support regions only, and $\|U_d(\mathbf{p})\|$ is the number of pixels to normalize the initial cost.

2.2 *Disparity Refinement*

2.2.1 *Disparity reliability evaluation*

Even after the above-described aggregation process, the following factors still cause many disparity errors: difference of illuminations in two views, repeated similar patterns, and occlusion by the foreground. Figure 3 shows typical matching cost distributions in aggregated regions. There is a single minimum matching cost within the disparity level in Fig. 3(a), so we can obtain a precise disparity. In Fig. 3(b), we cannot determine the correct disparity level among several candidates as to an image region with repeated pattern. Figure 3(c) shows matching cost distribution of the textureless region. We cannot determine the precise disparity level because many similar matching costs exist.

The matching cost for the disparity level at each pixel is examined to determine whether it is significantly smaller than any other competitors. In Figs. 3(b) and 3(c), however, the matching ambiguities cannot be completely overcome. The confidence map of the support region describing the reliability of the obtained disparity is computed to improve the matching performance.[23]

At the pixel **p** in the support region, initial disparity maps for the left image $D_0^L(\mathbf{p})$ and the right image $D_0^R(\mathbf{p})$ are computed using a winner-takes-all (WTA) strategy as provided in Eq. (2).[1] Here $d_{\max}$ represents the maximum disparity level. $R(\mathbf{p})$ in Eq. (3) means the reliability degree of the disparity at **p**. When the reliability degree $R(\mathbf{p})$ approaches 1, the obtained disparity value becomes more precise.

$$D_0(\mathbf{p})=\underset{d}{\operatorname{argmin}}\,C_1(\mathbf{p},d), \qquad d\in[0,d_{\max}], \quad (2)$$

$$R(\mathbf{p})=\begin{cases}0, & \|D_0^L(\mathbf{p})-D_0^R(\mathbf{p}')\|>0\\ R_1(\mathbf{p}), & \text{otherwise}\end{cases}, \quad (3)$$

$$R_1(\mathbf{p})=\left(\min\left\{\frac{\min_{d\notin D_0(\mathbf{p})\wedge d\neq D_0(\mathbf{p})\pm1}C_1(\mathbf{p},d)}{C_1[\mathbf{p},D_0(\mathbf{p})]},\tau_{R\text{trunc}}\right\}-1\right)/(\tau_{R\text{trunc}}-1). \quad (4)$$

An LRC check is used to see if the existence of false matching caused environmental lighting changes,

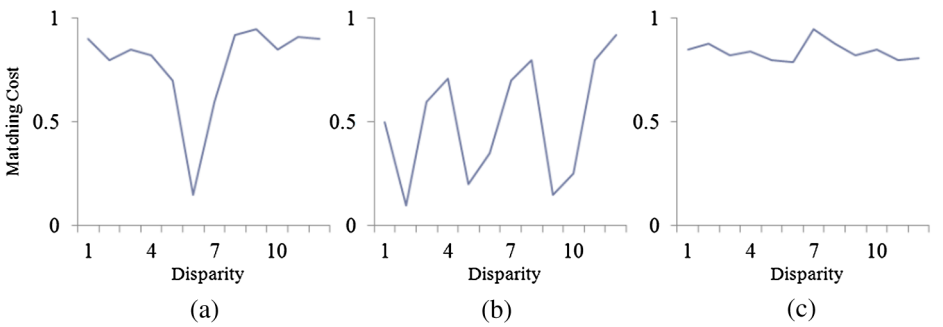


Fig. 3 Typical three matching cost distributions on disparity level: (a) distinguished feature region; (b) repeated pattern region; and (c) textureless region.

background effects, and occlusions. It is performed by taking the computed disparity value in one image and reprojecting it into the other image. We employ the LRC check to consider the unreliable disparity at half-occluded pixels in the final disparity map. The top portion of Eq. (3) shows that LRC check fails at pixel $\mathbf{p}$ and its disparity is unreliable. We put the reliability $R(\mathbf{p}) = 0$ to remove unreliable disparity in the temporal and spatial propagation process. The bottom portion of Eq. (3) shows that the disparity is reliable when the LRC check passes, and then the reliability $R(\mathbf{p})$ becomes $R_1(\mathbf{p})$.

Equation (4) computes the reliability of the first cost space. Here, we examine every depth level excluding both the depth by Eq. (2) and the next/previous depth $[D_0(\mathbf{p}) \pm 1]$, to find the disparity level with the minimum matching error among the matching cost C_1. A truncation constant value $\tau_{R\text{trunc}}$ is used to make the reliability $R_1(\mathbf{p})$ between 0 and 1.

If the difference between the smallest cost and the second smallest cost is large enough as in Fig. 3(a), the precise matching disparity can be obtained. On the contrary, when repeated patterns are present as in Fig. 3(b), the difference becomes a relatively small value. Since the reliability should include a confidence degree of the obtained disparity, we examine more various depth levels along the scan line, except the neighboring levels around the initially obtained depth.

Figure 4 shows the initial disparity map for the Teddy image by WTA and its reliability map. The dark region with relatively unreliable disparity can be refined further using both temporal and spatial propagation.

2.2.2 *Spatial propagation*

After the LRC check to detect the outliers, the outlier is filled with the neighboring reliable disparity in the segmented or the support region by the iterative region voting.[17,24] This means the disparity of the outlier is replaced with that of the highest bin value (most votes) in the support region when neighboring pixels with similar colors have similar disparities. However, when the outlier region is too large or the depth of the foreground is much different from that of the neighboring area in spite of its color similarity, the previous iterative voting would be unsuccessful.

We propose a spatial propagation approach to overcome the outlier problem by using reliable disparity rather than simply filling outliers with the disparity value of the highest bin value. Table 1 shows the spatial propagation of reliable disparity and updating of the codewords. The proposed method builds a histogram $\varphi_{\mathbf{p}}$ of only the reliable disparity in the support region $U(\mathbf{p})$. In II(ii) of Table 1, we obtain the most reliable disparity $d_{\mathbf{p}}{}^*$ and replace the outlier disparity at $\mathbf{p}$ by $d_{\mathbf{p}}{}^*$.

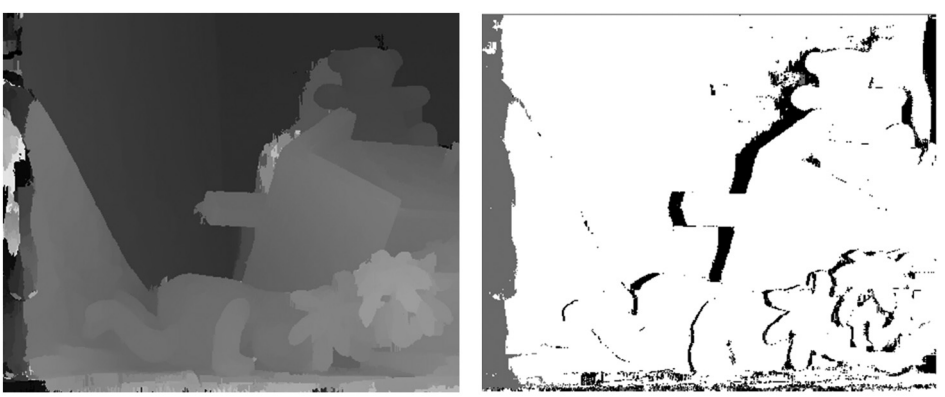
Because there may be many points with $d_{\mathbf{p}}{}^*$ in the support region, we determine the specific pixel position $s_{\mathbf{p}}{}^*$ to update the codewords (the reliability and the matching cost space) of $\mathbf{p}$ for further disparity refinement procedures. In III of Table 1, the subset $\mathbf{Sp}$ satisfying $|D_0(s_{\mathbf{p}}) - d_{\mathbf{p}}{}^*| < \varepsilon$ is determined from any element $s_{\mathbf{p}}$ of the entire set $U(\mathbf{p})$ and ε is the threshold value. Then we compute the color distance between $\mathbf{p}$ and $s_{\mathbf{p}}$, and the pixel with the smallest color distance is determined as the final position. Here, $D_c(\cdot)$ is L-1 color distance measure between two pixels in the RGB space. Both the matching cost $C_1(\mathbf{p}, d)$ and the reliability $R(\mathbf{p})$ at pixel $\mathbf{p}$ are updated as III(iii) in Table 1. $\rho(c, \lambda)$ is a robust function on variable c and it is used to control the influence of the color similarity between two pixels with the control parameter λ. If there is no pixel satisfying the above condition, the reliability $R(\mathbf{p})$ is updated to 0.

In Fig. 5(b), the disparity space image of the Teddy stereo image shows the matching error at a position on the scan line (green line) relative to the disparity level $[0 \sim d_{\max}]$. A more precise disparity map can be obtained at the position with a lower intensity value (matching error). Many undistinguished disparities happen in **A** region because of the matching ambiguity problem. Much more unreliable disparities happen in **B** region because of many repeated patterns. As shown in Fig. 5, the spatial propagation method improves the reliability of the disparity in the invalid areas (**A** and **B**). For further details, the proposed method fills **A** region with more reliable neighboring disparity and reduces the unwanted staircase effects caused by the repeated pattern in **B** region.

The enhanced matching cost and reliability information obtained from the spatial propagation are used to overcome the occlusion problem by the foreground objects in the temporal propagation process.

2.2.3 *Temporal propagation using codebook*

In order to overcome the occlusion and the depth discontinuity of an object, we propose a temporal propagation



(a) (b)

Fig. 4 (a) Initial disparity map and (b) its reliability evaluation result.

Table 1 Spatial propagation of reliable disparity and updating codewords.

I. For the outlier pixel $\mathbf{p}$, build a histogram of an initial disparity $D_0(\mathbf{p})$ with $d_{\max}+1$ bins.

II. (i) Obtain the histogram $\varphi_{\mathbf{p}}$ of only the disparity with a high reliability $[R(\mathbf{p}) \geq \tau]$ in $U(\mathbf{p})$.

(ii) Find the most frequent disparity $d_{\mathbf{p}}{}^*$ with the highest bin value from $\varphi_{\mathbf{p}}$.

(a) Examine if the total number of reliable disparities and the number of $d_{\mathbf{p}}{}^*$ are more than the threshold values.

(b) When the above conditions are satisfied, replace the outlier disparity at $\mathbf{p}$ by the reliable disparity $d_{\mathbf{p}}{}^*$.

III. Determine the specific pixel position $s_{\mathbf{p}}{}^*$ to update the codewords (the reliability and the matching cost space) of $\mathbf{p}$.

(i) Determine the subset S$\mathbf{p}$ satisfying $|D_0(s_{\mathbf{p}}) - d_{\mathbf{p}}{}^*| < \varepsilon$ in $U(\mathbf{p})$.

(ii) $s_{\mathbf{p}}{}^* = \text{argmin}_{S\mathbf{p}\in S\mathbf{p}} D_c(\mathbf{p}, s_{\mathbf{p}})$.

(iii) Update the matching cost space $C_1(\mathbf{p}, d)$ and the reliability $R(\mathbf{p})$ at $\mathbf{p}$.

(a) $C_1(\mathbf{p}, d) = C_1(s_{\mathbf{p}}{}^*, d)$, $d \in [0, d_{\max}]$.

(b) $R(\mathbf{p}) = R(s_{\mathbf{p}}{}^*) - \rho[D_c(\mathbf{p}, s_{\mathbf{p}}{}^*), \lambda_c]$, where $\rho(c, \lambda) = 1 - \exp(-c/\lambda)$.

process using color, reliability, matching cost set, and final access time values as codeword m in the modified codebook.

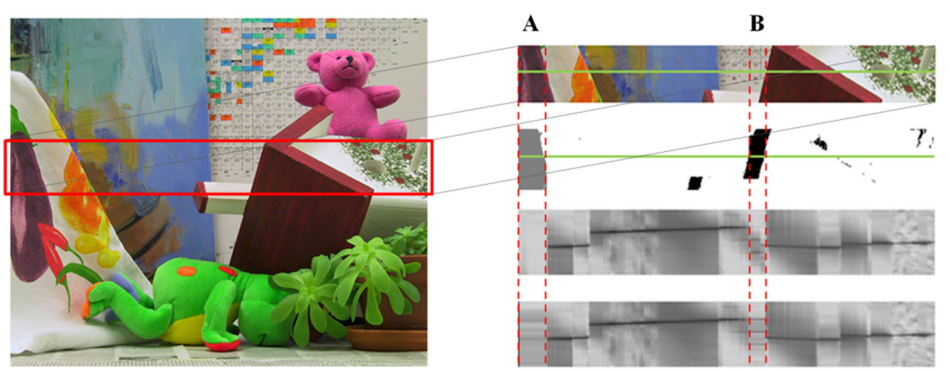
In the conventional codebook approach, the background region is modeled and parameterized only with the minimum and the maximum color values, which are updated at a regular interval to account for the effects of object movement and illumination change.[21] The process is not good enough to overcome the occlusion problem in various situations because it stores only the color information before the occlusion. Bleyer et al. proposed a temporal propagation using the slanted planes over successive frames for a stereo image sequence.[10] It does not sufficiently consider the update frequency of the prior information about the scene.

This proposed codebook $M(\mathbf{p})$ consists of color value $m_{\mathbf{x}}$, reliability m_R, array of the matching cost m_C, and final access time m_t at pixel $\mathbf{p}$. The matching cost and the reliability of the codebook are updated as described in Table 2.

For the pixel $\mathbf{p}$, we find the codeword m satisfying the condition II(i)(a) in Table 2 and update both the matching cost space and the reliability. In Table 2, ω_i represents the relative weight of the previous codewords $C_1(\mathbf{p}, d)$ and $R(\mathbf{p})$, and the current passed information $m_C(d)$ and m_R at pixel $\mathbf{p}$. $D_c(\cdot)$ and $\rho(\cdot)$ represent the color distance measure and the robust function in spatial propagation as in Sec. 2.2.2. In II(ii)(a) of Table 2, the matching cost $C_1(\mathbf{p}, d)$ is updated with the weighted sum of the previous cost at $\mathbf{p}$ and that of the chosen position. In the same way, the reliability $R(\mathbf{p})$ is replaced using II(ii)(b) in Table 2. Here, the color similarity between two pixels is considered as in the spatial propagation process.

The codeword of a codebook is updated using the matched codeword as in Table 3. The codebook $M(\mathbf{p})$ is an empty set at an initial time ($t = 0$). For the reliable pixel $\mathbf{p}$, the codeword satisfying the condition II(i) in Table 2 is used to update the codebook as II(i) in Table 3. If there is no match, a new codeword m', including color, reliability, matching cost, and frame number, is generated in the codebook $M(\mathbf{p})$.

When the codeword is not matched for a while ($\tau_t = 100$), our method concludes that the codeword insufficiently reflects the current image information due to the scene element changes, such as object movement. As shown in Table 4, after examining the effectiveness of the codeword, the unused codeword is removed to improve memory usage efficiency.



(a) (b)

Fig. 5 (a) Teddy image. (b) Matching cost space enhancement by spatial propagation: example area, reliability map, disparity space image (DSI) on scan line, and enhanced DSI (from up to down).

Table 2 Updating both matching cost and reliability.

I. For the pixel $\mathbf{p}$ at a current frame t, with color information $\mathbf{x} = (R, G, B)$.

II. (i) Find the codeword m satisfying condition (a) as well as minimizing the color distance.

(a) $D_c(\mathbf{p}, m_{\mathbf{x}}) < \tau_c$.

(ii) For m satisfying (i), update both the cost space and the reliability as follows:

(a) $C_1(\mathbf{p}, d) = \omega_0 C_1(\mathbf{p}, d) + \omega_1 m_C(d),\ d \in [0, d_{\max}]$.

(b) $R(\mathbf{p}) = \omega_0 R(\mathbf{p}) + \omega_1 (m_R) - \rho[D_c(\mathbf{p}, m_{\mathbf{x}}), \lambda_c]$,

where $\omega_0 = R(\mathbf{p})/[R(\mathbf{p}) + m_R]$, $\omega_1 = m_R/[R(\mathbf{p}) + m_R]$.

To further alleviate the matching ambiguity in the disparity map, an optimizer with smoothness constraints and moderate parallelism should be adopted. We employ a multidirection scan line optimizer based on Hirschmüller's semiglobal method.[7,17] Four scan line optimization processes are performed independently: two along horizontal directions and two along vertical directions. We examine the matching cost distribution of neighboring pixels along the scan line direction, including the appropriate penalty for discontinuities (a threshold value τ_{SO} for color difference).

After computing the matching space with the smoothness in four scan directions, a subpixel enhancement based on quadratic polynomial interpolation is employed to reduce the errors caused by disparity levels.[24] The interpolated disparity is computed with three discrete depth candidates: the depth (d) with the minimal cost and its two neighboring depth levels ($d - 1$ and $d + 1$). The final disparity is obtained by smoothing the interpolated results with a 3×3 median filter.

Table 3 Updating codeword and generating new codeword.

I. For the pixel $\mathbf{p}$ at a current frame t,

II. When condition $R(\mathbf{p}) \geq \tau_R$ is satisfied,

(i) m satisfying condition II (i) in Table 1 is updated as follows:

(a) $m_{\mathbf{x}i} = (m_{\mathbf{x}i} + \mathbf{x}_i)/2,\ i \in \{R, G, B\}$.

(b) $m_R = R(\mathbf{p})$.

(c) $m_C(d) = C_1(\mathbf{p}, d),\ d \in [0, d_{\max}]$.

(d) $m_t = t$.

(ii) Otherwise, a new codeword m' is generated in the codebook as follows:

(a) $m'_{\mathbf{x}i} = \mathbf{x}_i,\ i \in \{R, G, B\}$.

(b) $m'_R = R(\mathbf{p})$.

(c) $m'_C(d) = C_1(\mathbf{p}, d),\ d \in [0, d_{\max}]$.

(d) $m'_t = t$.

(e) $M(\mathbf{p}) = M(\mathbf{p}) \cup m'$

Table 4 Evaluating effective codeword.

I. For a pixel $\mathbf{p}$ at a current frame t,

II. Remove m satisfying condition (i).

(i) $t - m_t > \tau_t$, where $m \in M(\mathbf{p})$.

3 Experimental Results

The following computational equipment is used for the experiment: a PC with Intel Core i7 3.4 GHz CPU and 4 GB RAM with Nvidia GTX680 graphics card. The proposed system is tested with the Middlebury benchmark[25] and the stereo images (320×240) captured by a Bumblebee 3 from Point Grey Inc. in Canada at 15 frames per second. The proposed method is implemented on a GPGPU with CUDA to handle the heavy computational loads of both the stereo matching and the cost refinement.[26]

Figure 6 shows Tsukuba, Venus, Teddy, and Cones stereo datasets and the disparity maps from this method. Table 5 shows the quantitative evaluation results by stereo matching algorithms with a near-real-time computation performance for the Middlebury database set. Here, the performances are evaluated only in the non-occluded region "non-occ," all (including half-occluded) regions "all," and regions near depth discontinuities "disc," respectively. Our method produces the best results on the Venus image pair because the simple scene element would be suitable for spatial propagation. In comparison, the proposed method provides better results than any other method except AD-census.[17] Additionally, the proposed temporal propagation with codebook is useful for improving the matching performance in a stereo image sequence as in Fig. 6.

In Table 6, the computation performances of the modules on a stereo image (400×300) with the maximum depth of 40 are compared on CPU and GPU implementation. The proposed temporal propagation employs codewords with background and stereo matching information. As shown in Tables 2 and 3, the codebook matching step requires much more memory access and codeword comparisons than any other module. Table 6 shows that the codebook matching step provides relatively less performance improvement in spite of its GPU implementation. Thus, GPU implementation is suitable for single instruction multiple data processing, such as initial cost volume process.

The proposed method is designed for dynamic situations, such as mobile robots. When the camera is moved, it is difficult for us to precisely model the background information with the codebook in real-time. For this reason, we do not include the temporal propagation step based on the codebook in a dynamic environment with camera movement. In a dynamic situation with camera movement, the proposed method with spatial propagation requires only 81.71 ms.

As shown in Tables 5 and 7, even though AD-census[17] provides a better accuracy performance, it requires more processing time and it cannot be used for real-time systems, such as mobile robots and natural user interface. The proposed method is suitable for near-real-time application

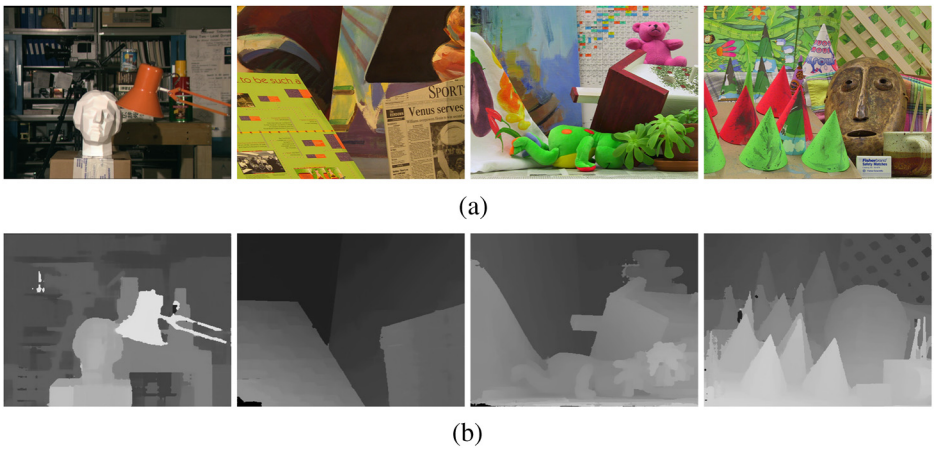


Fig. 6 (a) Tsukuba, Venus, Teddy, and Cones stereo datasets (from left to right). (b) Disparity maps by the proposed algorithm.

because it provides improved matching accuracy and processing efficiency.

The accuracy of the disparity map can be evaluated quantitatively by using the reference depth map of Middlebury database sets. Table 7 provides comparison of the average squared errors by stereo matching methods. The AW method[3] is usually classified as a non-real-time stereo algorithm. The system also helps us to overcome various occlusions by using temporal propagation based on the codebook.

In an indoor environment, the scene background is initially modeled for 5 to 10 frames and the codebook is updated at regular intervals to reduce the unwanted effects of background and lighting changes.

Figure 7 shows an input stereo image (left view) and the results by successive procedures. The minimum averaged squared error on each image is highlighted in boldface. Figures 7(b) to 7(d) show the disparity between the AD-census, initial disparity map by cost aggregation, and the reliability map, respectively. In the reliability map, there are the dark regions with relatively unreliable disparity around people according to their movements. Figures 7(e) and 7(f) show the disparity map for only the spatial propagation, and that for both spatial and temporal propagation, respectively. These unreliable disparity areas in Figs. 7(c) and 7(d) are refined further using spatial and temporal propagation. For example, the reliability map [Fig. 7(d)] shows the ceiling area with a relatively unreliable disparity.

Figures 7(g) and 7(h) show the final disparity map by optimization/subpixel enhancement of Fig. 7(e) and that of Fig. 7(f), respectively. The final disparity results show these regions are much enhanced through semiglobal optimization and subpixel enhancement. In final disparity maps [Figs. 6(g) and 6(h)], we obtain a more accurate disparity map by using both spatial and temporal propagation based on the codebook over the stereo sequence.

In the comparison of matching performances in the outdoor scene (Fig. 8), the proposed algorithm produces better disparity map than the dual-cross-bilateral grid (DCB) grid, adaptive weight method, and cross-based matching.[3,11,29] According to two important threshold parameters, the color difference ($\tau_{SO} = 5$ to 63) in scan line optimization and the reliability ($\tau_R = 0.00$ to 0.87), the matching performances of the proposed method in all (including half-occluded) regions for the Cones and Teddy images are shown in Fig. 9.

Table 5 Quantitative evaluation results for Middlebury database set belief propagation (BP), bitwise fast voting (BFV), Adaptive support-weight (AW), and dual-cross-bilateral grid (DCB).

	Tshkuba			Venus			Teddy			Cones			Aver.
	Non-occ	All	Disc	Non-occ	All	Disc	Non-occ	All	Disc	Non-occ	All	Disc	
AD-census[17]	1.07	1.48	5.73	0.09	0.25	1.15	4.10	6.22	10.90	2.42	7.25	6.95	3.97
Proposed method	1.71	2.46	7.54	0.15	0.51	1.73	4.43	10.3	12.70	2.80	8.81	7.88	5.09
PlaneFitBP[27]	0.97	1.83	5.26	0.17	0.51	1.71	6.65	12.10	14.7	4.17	10.70	10.60	5.78
AW[3]	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.3	18.6	3.97	9.79	8.26	6.67
Real-time BFV[12]	1.71	2.22	6.74	0.55	0.87	2.88	9.90	15.00	19.5	6.66	12.60	13.40	7.65
Real-time GPU[28]	2.05	4.22	10.6	1.92	2.98	20.30	7.23	14.40	17.60	6.41	13.70	16.50	9.82
DCB grid[11]	5.90	7.26	21.00	1.35	1.91	11.20	10.50	17.20	22.20	5.34	11.90	14.90	10.90

occ, occluded; AD, absolute difference.

Table 6 Computation time (millisecond) of modules.

	Preparing step	Initial cost volume	Refinement	Codebook matching	Semiglobal optimization	Total
CPU	343.62	1022.58	98.67	38.21	1729.83	3232.91
GPU	3.75	17.34	14.19	36.30	46.43	118.01

Table 7 Comparison of averaged squared errors belief propagation (BP), bitwise fast voting (BFV), Adaptive support-weight (AW), and dual-cross-bilateral grid (DCB).

	Tshkuba	Venus	Teddy	Cones	Average
AD-census[17]	0.3248	0.2439	**0.6426**	**0.6508**	**0.4655**
AW[3]	0.2488	0.3212	0.8120	0.7678	0.5375
Proposed method	0.3585	**0.2081**	0.7744	0.8259	0.5417
PlaneFitBP[27]	**0.2218**	0.3177	0.7399	1.2084	0.6220
Real-time BFV[12]	0.3216	0.3274	1.0886	1.1188	0.7141
Real-time GPU[18]	0.3968	0.3803	0.8814	1.0841	0.6857
DCB grid[11]	1.0408	0.2374	1.0613	0.9344	0.8185

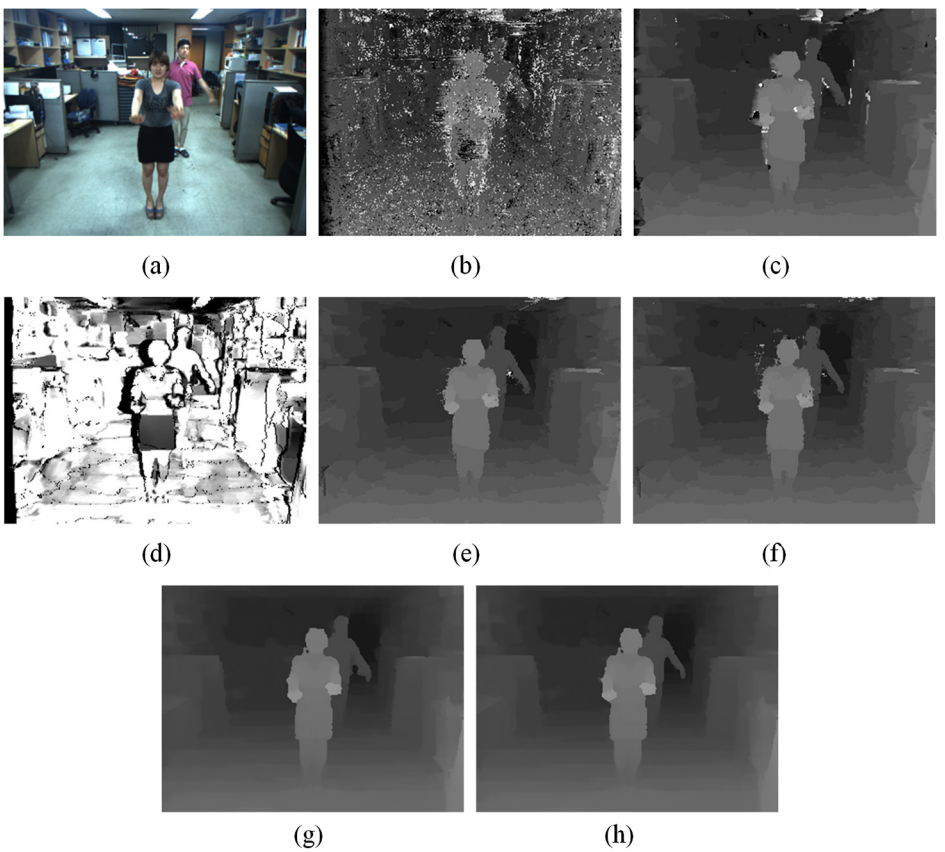


Fig. 7 (a) Stereo image (left view). (b) Disparity by absolute difference-census. (c) Initial disparity map by cost aggregation. (d) Reliability map. Disparity map (e) by spatial propagation and (f) by both spatial and temporal propagations. (g) Final disparity map by optimization/subpixel enhancement of (e). (h) Final disparity map by optimization/subpixel enhancement of (f).

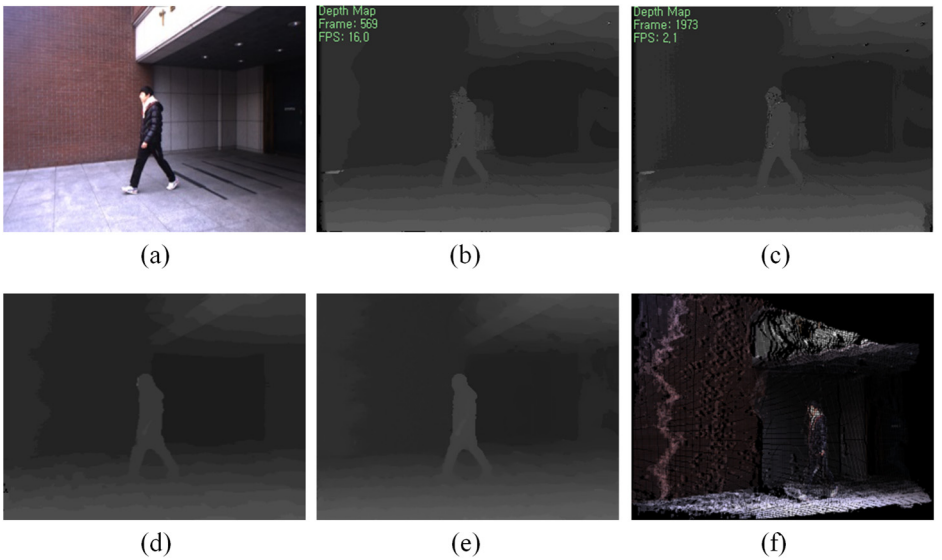


Fig. 8 (a) Outdoor scene image. Disparity map by (b) DCB grid;[11] (c) adaptive weight;[3] (d) cross-based matching;[19] and (e) proposed method. (f) Three-dimensional (3-D) reconstruction view of (e).

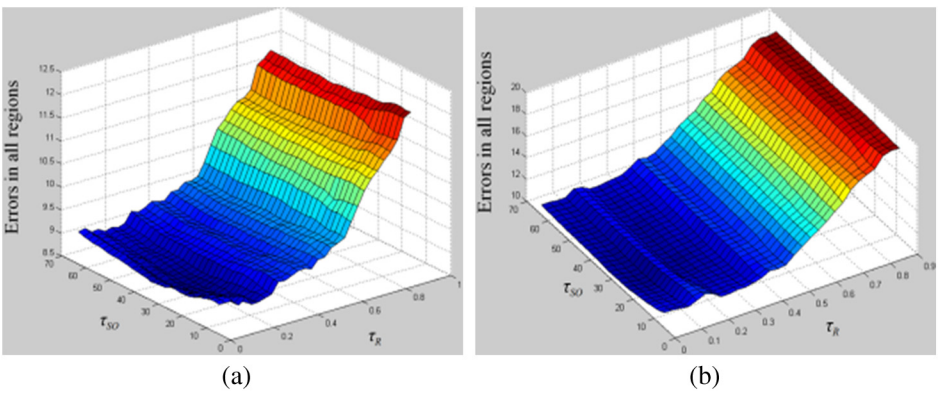


Fig. 9 Performance according to threshold parameters (τ_R is reliability and τ_{SO} is color difference in scanline optimization) on (a) Cones image and (b) Teddy image set.

Fig. 10 Snapshots of Ilkay stereo sequence from Microsoft i2i database: (a) reference frame. Disparity result by (b) aggregation with enhanced segment tree[30] (b) proposed method; and (f) 3-D reconstruction view of (e).

By analyzing matching error distributions of the Middlebury images, we can determine two threshold values for the minimum matching error: τ_R and τ_{SO} are set to 0.172 and 27.552. In addition, λ_c, ε, and τ_C are set to 2, 1, and 15, respectively.

In order to show the effectiveness in qualitative characteristics, we compare the disparity map for non-real-time algorithm[30] with that for the proposed method on a publicly available, real-world stereo video set: an Ilkay sequence from Microsoft i2i database. The disparity result [Fig. 10(b)] by ST-2 has more accurate depth borders and less noise. On the contrary, the proposed method has a near-real-time processing performance and obtains more accurate disparity in textureless areas, such as wall and ceiling.

Even though the kinect sensor captures precise depth information about the scene element, it can be operated within a substantially limited operation range and only in an indoor environment. It is also greatly affected by the reflection properties of the environmental elements, such

as the monitor. Our proposed stereo system provides an extended usage range and a precise depth map result. Thus, it can be used for real-time indoor/outdoor applications.

In the codebook update process, we average the previously stored codeword and the new computed one to reflect the new codeword value as in Table 2. This may lead to runaway codewords if some misclassifications occur, so we will employ another weighted update method for the codebook. The proposed method considers much important information for stereo matching over the stereo video sequence in addition to the color value. However, because we just observe the pixel-wise data, there may be some errors in the disparity map for the proposed method. In order to improve the matching performance, we extend the pixel-wise codebook method into the patch- or segment-based method with spatial correlation.

4 Conclusion

The proposed method improves matching performance by using temporal and spatial propagation of reliable disparity over a stereo sequence. First, we compute a reliability map of an initial matching cost. After examining the LRC to detect the outliers created by occlusion, the proposed spatial propagation fills the outliers with the neighboring reliable disparity information in the support region. In order to overcome the occlusion problem, we employ a codebook including color value, reliability, array of the matching cost, and final access time. The proposed method is implemented on a GPGPU for real-time application. Experiments show that the proposed matching method obtains a more precise depth map of indoor/outdoor scenes with extended usage range.

Acknowledgments

This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (No. 2013R1A1A2008953), and by the IT R&D program of MOTIE/KEIT (10045289, Development of virtual camera system with real-like contents).

References

1. D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.* **47** (1), 7–42 (2002).
2. H. Hirschmüller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(9), 582–1599 (2009).
3. K. Yoon and I. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(4), 650–656 (2006).
4. Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.* **23** (11), 1222–1239 (2001).
5. J. Sun et al., "Symmetric stereo matching for occlusion handling," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, pp. 399–406 (2005).
6. G. Van Meerbergen et al., "A hierarchical symmetric stereo algorithm using dynamic programming," *Int. J. Comput. Vis.* **47**(1), 275–285 (2002).
7. H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.* **30** (2), 328–341 (2008).
8. L. Hong and G. Chen, "Segment-based stereo matching using graph-cuts," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 74–81 (2004).
9. M. Gong, Y. Zhang, and Y. Yang, "Near-real-time stereo matching with slated surface modeling and sub-pixel accuracy," *Pattern Recognit.* **44**(10/11), 2701–2710 (2011).
10. M. Bleyer, C. Rhemann, and C. Rother, "PatchMatch stereo—stereo matching with slanted support windows," in *Proc. of MBVA British Machine Vision Conf.*, pp. 1–11 (2011).
11. C. Richardt et al., "Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid," in *Proc. of European Conf. on Computer Vision*, pp. 6311–6316 (2010).
12. K. Zhang et al., "Real-time accurate stereo with bitwise fast voting on CUDA," in *Proc. of IEEE Int. Conf. on Computer Vision Workshops*, pp. 794–800 (2009).
13. R. Yang, M. Pllefeys, and S. Li "Improved real-time stereo on commodity graphics hardware," in *Proc. of Conf. on Computer Vision and Pattern Recognition Workshop on Real-Time 3D Sensors and Their Use*, Vol. 3, pp. 36–42 (2004).
14. Q. Yang et al., "Real-time global stereo matching using hierachical belief propagation," in *Proc. of MBVA British Machine Vision Conf.*, pp. 989–998 (2006).
15. Y. Zhao and G. Taubin, "Real-time stereo on GPGPU using progressive multi-resolution adaptive windows," *Image Vison Comput.* **29**(6), 420–432 (2011).
16. J. Kowalczuk, E. T. Psota, and L. C. Perez, "Real-time stereo matching on CUDA using an iterative refinement method for adaptive support-weight correspondences," *IEEE Trans. Circuits Syst. Video Technol.* **23**(1), 94–104 (2013).
17. X. Mei et al., "On building an accurate stereo matching system on graphics hardware," in *Proc. of IEEE Int. Conf. on Computer Vision Workshops on GPUs for Computer Vision*, pp. 467–474 (2011).
18. J. Kowalczuk, E. T. Psota, and L. C. Perez, "Real-time stereo matching on CUDA using an iterative refinement method for adaptive support-weight correspondences," *IEEE Trans. Cirsuits Syst. Video Technol.* **23**(1), 94–104 (2013).
19. C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, pp. 246–252 (1999).
20. A. Mittal and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, pp. 302–309 (2004).
21. K. Kim et al., "Real-time foreground-background segmentation using codebook model," *Real-Time Imaging* **11**(3), 167–256 (2005).
22. K. Zhang, J. Lu, and G. Lafruit, "Cross-based local stereo matching using orthogonal integral images," *IEEE Trans. Cirsuits Syst. Video Technol.* **19**(7), 1073–1079 (2009).
23. C. Shi et al., "Stereo matching using local plane fitting in confidence-based support window," *IEICE Trans. Inf. Syst.* **E95-D**(2), 699–702 (2012).
24. Q. Yang et al., "Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling," *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(3), 492–504 (2009).
25. "Middlebury stereo vision," http://vision.middlebury.edu/stereo/eval/.
26. S. Cook, *NVIDIA GPU Programming*, John Wiley & Sons Inc., Hoboken, New Jersey (2012).
27. Q. Yang, C. Engels, and A. Akbarzadeh, "Near real-time stereo for weakly-textured scenes," in *Proc. of MBVA British Machine Vision Conf.*, pp. 72.1–72.10 (2008).
28. L. Wang et al., "High quality real-time stereo using adaptive cost aggregation and dynamic programming," in *Proc. of Int. Symp. on 3D Data Processing, Visualization and Transmission*, pp. 798–805 (2006).
29. K. Zhang, J. Lu, and G. Lafruit, "Cross-based local stereo matching orthogonal integral images," *IEEE Trans. Circuits Syst. Video Technol.* **19**(7), 1073–1079 (2009).
30. X. Mei et al., "Segment-tree based cost aggregation for stereo matching," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 23–28 (2013).

Sungil Kang received his BS degree in computer science and engineering from Chung-Ang University in 2011. He received his MS degree from the Department of Imaging Science and Arts, GSAIM, Chung-Ang University in 2013. He is currently working for LG Electronics, Inc., Republic of Korea. His research interests include computer vision and image processing.

Hyunki Hong received his BS, MS, and PhD degrees in electronic engineering from Chung-Ang University, Seoul, Republic of Korea, in 1993, 1995, and 1998, respectively. From 2000 to 2014, he was a professor in Department of Imaging Science and Arts, GSAIM at Chung-Ang University. Since 2014, he has been a professor in the School of Integrative Engineering, Chung-Ang University. His research interests include computer vision, augmented reality, and multimedia application.