Local Disparity Estimation With Three-Moded Cross Census and Advanced Support Weight

Zucheul Lee, *Student Member, IEEE*, Jason Juang, and Truong Q. Nguyen, *Fellow, IEEE*

***Abstract*—The classical local disparity methods use simple and efficient structure to reduce the computation complexity. To increase the accuracy of the disparity map, new local methods utilize additional processing steps such as iteration, segmentation, calibration and propagation, similar to global methods. In this paper, we present an efficient one-pass local method with no iteration. The proposed method is also extended to video disparity estimation by using motion information as well as imposing spatial temporal consistency. In local method, the accuracy of stereo matching depends on precise similarity measure and proper support window. For the accuracy of similarity measure, we propose a novel three-moded cross census transform with a noise buffer, which increases the robustness to image noise in flat areas. The proposed similarity measure can be used in the same form in both stereo images and videos. We further improve the reliability of the aggregation by adopting the advanced support weight and incorporating motion flow to achieve better depth map near moving edges in video scene. The experimental results show that the proposed method is the best performing local method on the Middlebury stereo benchmark test and outperforms the other state-of-the-art methods on video disparity evaluation.**

***Index Terms*—Census transform, disparity estimation, motion flow, spatial temporal consistency.**

I. Introduction

THE resurgence of interest in 3D films and television has launched a new era in visual media consumption and research. Given a pair of stereoscopic views, disparity estimation is a crucial step for depth-based processing and communications. The numerous advanced algorithms for disparity estimation may be generally categorized as either local or global methods. The global methods compute all disparities of the image simultaneously by optimizing the global energy function. They produce accurate disparity map but they are usually complex and computationally expensive. On the other hand, the local methods compute the disparity of the pixel based on the support window cost aggregation. They have simple structure and are more efficient in terms of computational complexity compared to the global methods. However, it is difficult to find the correct matching point in flat areas because all pixels in the window contains similar structure and texture. Two main concerns in local methods are the accuracy of the similarity measure and the proper support window on which the matching accuracy depends.

Common similarity measures are sum of absolute difference (SAD), sum of squared difference (SSD), normalized cross correlation (NCC), and non-parametric transforms such as rank and census. The rank and census transforms are more robust to radiometric distortion because they yield relative ordering of the pixel intensity rather than the intensity values themselves. Therefore, for image regions with similar colors, non-parametric transforms may cope with the matching ambiguities well, while for image regions with similar local structures, the color differences (SAD and SSD) may cope with the matching ambiguities well. According to the evaluation of similarity measures [1], census transform achieves the best overall performance throughout all experiments with simulated and real radiometric differences, except in the presence of strong image noise.

Another important research topic in local method is how to select the proper support window for each pixel. The simple fixed size rectangular window is used to find corresponding pixels in a pair of left and right images in early local approaches. However, this results in the foreground smearing problem near depth discontinuities due to the assumption that all pixels in the window have the same disparity. To solve this problem, the adaptive-window method [2] finds an optimal window based on the local variation of intensity and disparity. This method uses a rectangular window, which is not suitable for arbitrarily shaped depth discontinuities. The multiple-window method [3] calculates the correlation with nine pre-defined windows and selects the disparity with the smallest matching cost. This method also has the limitation of window shape. To obtain more accurate results at depth discontinuities, the locally adaptive support weight approach (LASW) [4] adjusts the support weights of the pixels in the window by using the photometric and geometric distance with respect to the center pixel. This method deals with the pixels near depth discontinuities more effectively than the two methods mentioned above. Segment-support [5] improves the reliability of adaptive support aggregation by adding additional segmentation process. Disparity calibration [6] increases the matching process to two steps by adding disparity calibration while the traditional local methods use one-step process. PatchMatch [7], the best local method among all other local methods on the Middlebury benchmark [8] uses

Manuscript received November 16, 2012; revised February 14, 2013; accepted April 11, 2013. Date of publication June 20, 2013; date of current version November 13, 2013. This work was supported in part by NSF grant CCF-1065305, by Intel/CISCO under the VAWN program and by the Technology Development Program for Commercializing System Semiconductor funded by the Ministry of Knowledge Economy (MKE, Korea). (No. 10041126, Title: International Collaborative R&BD Project for System Semiconductor). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Feng Wu.

The authors are with the Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093 USA (e-mail: z1lee@ucsd.edu; jajuang@ucsd.edu; tqn001@ucsd.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2013.2270456

additional processes such as iteration, slanted plane and propagation scheme to obtain better results. These three methods are computationally expensive. Cost-filter [9] obtains consistent edge-preserving results by using the guided filter and it is one of the best local methods for the Middlebury dataset. It is worth noting that LASW and Cost-Filter do not use any iteration and additional step, which could make the algorithm more complex. LASW and Cost-filter are good edge-preserving methods but they do not provide a reliable solution for disparity estimation in textureless (flat) areas, which have different characteristics from edges.

Stereo video disparity estimation is at an early stage while stereo image disparity estimation is mature. This may be the consequence of two factors. First, it is due to lack of stereo video datasets with ground-truth disparity maps. Second, it is due to temporal inconsistency problems, such as flickering resulting from simply applying current state-of-the-art image-based algorithms to video. To reduce this artifact, [10] uses median filtering along flow vectors computed by the method of Horn and Schunck [11]. However, the results are of moderate quality. Total Variation (TV) method [12] shows impressive results by treating the video disparity as a spatio-temporal volume to improve spatial and temporal consistency and it presents the possibility for directly extending current image-based disparity algorithms to the video domain.

In this paper, we propose a three-moded census transform with a noise buffer to be more tolerant of image noise in flat areas and a cross-square census to increase the reliability of census measure. We suggest the effective combination of three cost measures (census, color and gradient), which have different characteristics on stereo matching, to obtain more accurate cost measure in a variety of image regions. These three new ideas can be utilized in both stereo images and videos in the same form for similarity measure. In video processing, motion is a crucial factor and generally moving objects tend to have a higher degree of saliency. However, most disparity methods may have difficulty dealing with fast moving edges in video scenes. To solve this problem, we incorporate optical flow for support weight computation within the localized window. This approach is new and helps to determine the spatial ambiguities by utilizing temporally consistent information. To further improve the support weight, we define the conditional relation between similarity and proximity and the correlated relation between similarity and motion by analyzing each gestalt principle. We demonstrate that the proposed local method is the best performing local method on both the Middlebury stereo benchmark test and the video disparity evaluation on 5 synthetic stereo video datasets. In particular, these meaningful results are achieved with adding no additional process and keeping the classical framework of local methods. We enforce temporal consistency by refining our disparity estimates with the spatio-temporal consistency algorithm described in [12].

This paper is an extension of our conference paper [13]. The rest of this paper is organized as follows. The system structure and new similarity measure are presented in Section II. We present the details of the advanced support weight for stereo image and video in Section III. The disparity computation and occlusion filling method are present in Section IV. Section V shows experimental results and discusses their significance. Section VI concludes with some remarks.

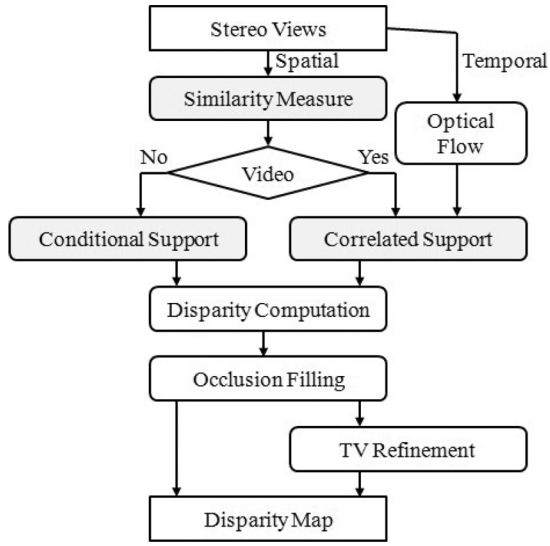


Fig. 1. Block diagram of the overall system.

II. System Structure and Similarity Measure

A. Structure of the Proposed Local Method

The proposed method is an efficient one-pass local method applicable to both stereo images and videos with no iteration. The block diagram of the overall system is shown in Fig. 1. It consists of four main components: similarity measure, support weight, disparity computation, and occlusion filling. For video disparity estimation, optical flow and TV refinement algorithms will be incorporated. The core blocks for the accuracy of disparity estimation are similarity measure and support weight block which will be discussed in details.

B. Three-Moded Cross Census and Combination of Similarity Measures

Census transform encodes the pixel value into bit-stream representing the relative ordering of the neighboring pixels. To achieve more precise census similarity measure, we need to obtain larger spatial structure by increasing the size of census window. However, the error probability might increase as the window size increases over the certain value. The larger the census window size is, the more occluded pixels would be included in the transformed bitstream. There is a trade-off between more spatial information and accuracy of the estimate. Fig. 2 illustrates that the big square census window in Fig. 2(b) is more likely to be affected by the occlusion area (gray color area) than the window in Fig. 2(a), and therefore its transformed information will be severely distorted. To alleviate this problem, we propose the cross-square shape of census window which can contain more spatial information while being less exposed to the occlusion area as shown in Fig. 2(c).

As discussed in Section I, the census transform is robust to radiometric distortions and achieves the best overall performance

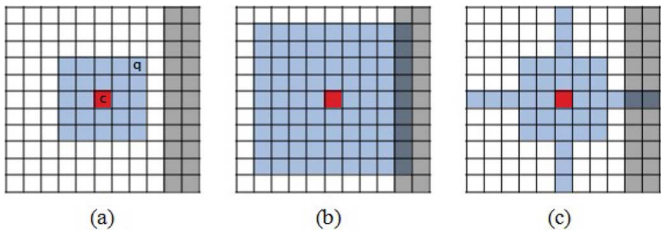


Fig. 2. Three different census windows. (a) 5 × 5. (b) 9 × 9. (c) Cross-square.

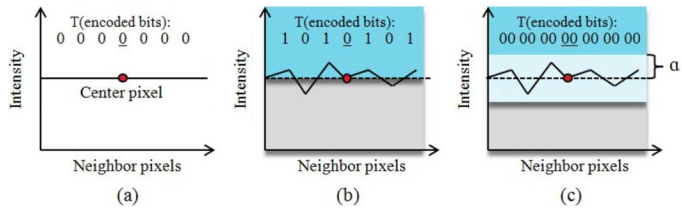


Fig. 3. Comparison of the original census and the three-moded census in flat areas. (a) Original census without noise. (b) Original census with noise. (c) Three-moded census with noise.

in both local and global methods. However, it experiences difficulties in finding the correct correspondences in flat areas, as most methods do. This difficulty is due to the fact that the census matching cost is extremely sensitive to even small image noise in flat area since all pixels have similar intensity value, and then the left and right census can be encoded differently due to camera noises. Most of stereo images are distorted due to camera noises except for synthetic stereo images. To reduce the mismatch due to the distortion from left and right camera, we propose a three-moded census transform with a noise buffer. The original census has two modes where a bit is set to 1 if the neighboring pixel in the census window has higher intensity than the center pixel and 0 otherwise. On the other hand, our three-moded census uses two bits to implement three modes and it is defined as

$$T = \bigotimes_{q \in W} \xi(I_c, I_q)$$
$$\xi(x, y) = \begin{cases} 10 & \text{if } y > x + \alpha \\ 01 & \text{if } y < x - \alpha \\ 00 & \text{otherwise} \end{cases} \quad (1)$$

where $\bigotimes$ denotes concatenation and W represents the census window. I_c represents the intensity at center pixel c and α is noise buffer threshold. Camera noise is intensity-dependent and the noise variance is proportional to intensity [14], [15]. The noise buffer should be increased to get consistent results as the noise variance increases. Therefore, we can define α as a function of intensity:

$$\alpha = \left[\frac{I_c}{\beta}\right] \quad (2)$$

where $[\cdot]$ denotes nearest integer operator and, empirically, a reasonable value for β is 500 and 50 for synthetic images and real-world images, respectively. Fig. 3 shows how three-moded census works under noisy environment. In flat areas, the neighboring pixels show the same intensity as shown in Fig. 3(a). Under noisy environment, the original census transform yields very different bit-stream from the noiseless case as shown in Fig. 3(b), while the three-moded census transform produces a consistent bit-stream as shown in Fig. 3(c). Additionally, we don't define the census transform at the center pixel because it is always zero.

Fig. 4 shows example of left and right bitstreams resulting from the three-moded cross census transform, which are used in the calculation of Hamming distance (ΔH). To further improve the matching accuracy, we incorporate the color distance (ΔI) and gradient distance (ΔG) between two center pixels as shown

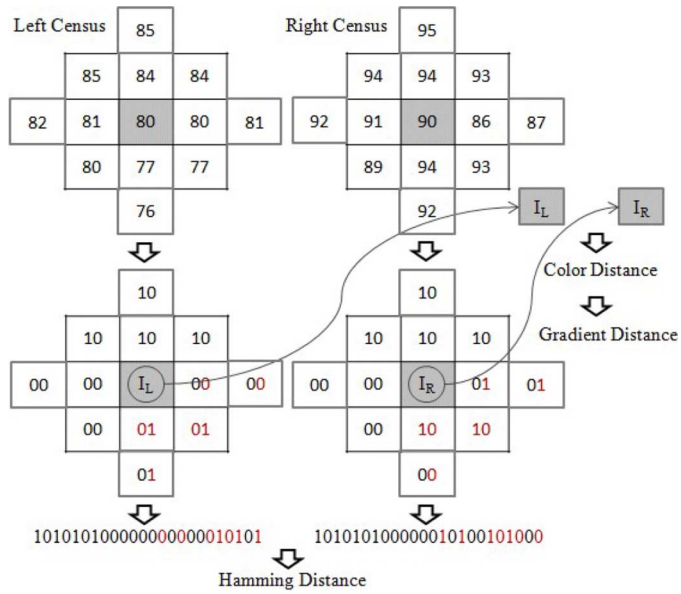


Fig. 4. Example of three-moded census transform with $\alpha = 2$ and three similarity measures.

in Fig. 4. In other words, we use the census transform to compare the spatial structure of two census windows, while we use the color and gradient distance to compare two center pixels. The Hamming distance of two census transforms is defined as

$$\Delta H = d(T_L, T_R) = T_L \oplus T_R \quad (3)$$

where T_L represents the left transformed bit-stream and $\oplus$ denotes the bitwise XOR operation. The color distance between $I_L = (I_L^r, I_L^g, I_L^b)$ and $I_R = (I_R^r, I_R^g, I_R^b)$ in RGB vector space is defined as

$$\Delta I = d(I_L, I_R) = \sqrt{\sum_{j=r,g,b} (I_L^j - I_R^j)^2}. \quad (4)$$

The gradient $G = (G_x, G_y)$ is composed of two components, which are partial derivatives along x-axis and y-axis. The partial derivative G_x can be expressed as (G_x^r, G_x^g, G_x^b) in RGB space. The gradient distance between $G_L = (G_{Lx}, G_{Ly})$ and $G_R = (G_{Rx}, G_{Ry})$ is defined as

$$\Delta G = d(G_L, G_R) = \sqrt{d(G_{Lx}, G_{Rx})^2 + d(G_{Ly}, G_{Ry})^2}$$
$$d(G_{Lx}, G_{Rx}) = \sqrt{\sum_{j=r,g,b} \left(G_{Lx}^j - G_{Rx}^j\right)^2} \quad (5)$$

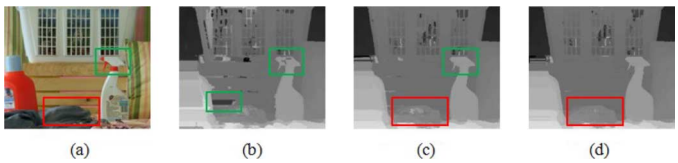


Fig. 5. Disparity maps on "Laundry" computed by different similarity measures. (a) Left image. (b) Color. (c) Combination of color and census. (d) Combination of color, census, and gradient.

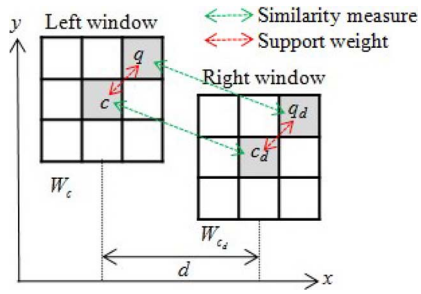


Fig. 6. Left support window and right support window.

where G^j_{Lx} represents the partial derivative along x-axis in the j color domain of the left image.

We propose the combination of three distances, which is simple but very effective by yielding more reliable similarity measure by compensating one another. Fig. 5 illustrates how each similarity measure improves the accuracy of disparity estimation. Fig. 5(b) is computed by using color distance, which is commonly used and it shows many errors in the similar color area (green box). In Fig. 5(c), some errors are recovered by combining census Hamming distance. However, wrong matches in densely textured region with high frequency condition (red box) are not recovered. In Fig. 5(d), different types of errors are recovered and the best overall disparity map is produced by using the combination of three measures. Note that there exist many similarity measures showing different characteristics and it is important how to choose proper measures and integrate them for better matching performance. For integrated similarity matching cost, we use a robust cost function including three distances:

$$C_0(q,q_d) = 3 - \exp\left(-\frac{\Delta H_{qq_d}}{\gamma_H}\right) - \exp\left(-\frac{\Delta I_{qq_d}}{\gamma_I}\right) - \exp\left(-\frac{\Delta G_{qq_d}}{\gamma_G}\right) \quad (6)$$

where ΔH_{qq_d}, ΔI_{qq_d} and ΔG_{qq_d} are Hamming distance, color distance and gradient distance, respectively, between pixel q and pixel q_d as shown in Fig. 6. γ_H, γ_I and γ_G are empirical parameters.

III. Advanced Adaptive Support Weight

A. Gestalt Grouping

According to Gestalt principles, human observers are able to group visual objects that share certain common characteristics [16]. The best-known grouping laws are proximity (objects that are close to each other are grouped together), similarity (objects that have similar color are grouped together), and common fate (objects that move at the same speed in the same direction are grouped together) [17]. Common fate is closely related to motion flow, which will be denoted as "motion" for simplicity. Whenever objects have characteristics in common, they get grouped and formed a larger visual object, known as a gestalt [16].

From these observations, we can assume that human observers group pixels in a scene based on how close two pixels are spatially, how similar their colors are, and how similar their velocities are. Thus, we can use the strength of grouping when computing the support weight of a pixel, which should be proportional to the probability that the two pixels have the same disparity. The closer two pixels are in proximity and color, the larger their support weight. The same can be said about the motion flows of two pixels. These three observations may be treated in an integrated manner to obtain a reasonable grouping. Each grouping law can compensate for the others when they fail in specific cases. For instance, the motion cue helps viewers distinguish figures when the object color or outlines are not clear. Therefore, it would be beneficial to model the human visual system and segment objects by using support weights based on Gestalt principles. We analyze each principle and their relationship to find effective integration method for stereo image and video.

B. Support Weight on Stereo Images

The ideal support window is an arbitrarily shaped window which consists of only pixels at the same depth. It is very difficult to determine accurately which pixels belong to the same object. We consider the adaptive support window on the basis of two gestalt grouping laws (color similarity and proximity), which can be used together to group objects as in [4]. To obtain an advanced support weight, we analyze two issues, color space and relationship between similarity and proximity to decide how to integrate. First, the previous works use the Euclidean distance in CIELab color space. The CIELab color space is perceptually uniform and its Euclidean distance corresponds to the perceptual color difference between any two colors. However, the use of the CIELab color space makes the color distance less selective for the pixels, which are close chromatically. Fig. 7 illustrates the comparison of two color spaces in the area where each pixel has chromatically similar color. In Fig. 7(a), the center pixel $I_c = [80\ 80\ 80]$ and the neighboring pixel $I_q = [79\ 79\ 79]$ in RGB space are converted to $[34.029\ 0.002\ -0.004]$ and $[33.603\ 0.002\ -0.004]$ in CIELab space, respectively. The RGB and CIELab color difference are 1.7321 and 0.4260, respectively. The ratio $(\frac{\Delta I_{RGB}}{\Delta I_{CIELab}})$ in similar color region (I_c and I_q) is 4.0657. On the other hand, the ratio in distinct color region (I_c and $I_p = [200\ 80\ 80]$) is 2.1321. It shows higher ratio in similar color region than distinct color region. Fig. 7(b) and (c) show color difference at each pixel with respect to the center pixel in RGB and CIELab space. RGB produces more selective distance than CIELab in similar color region. Additionally, the $L^*a^*b^*$ metrics are particularly sensitive to errors in low RGB signal [18]. The color space should provide good distance metric for area with similar color as well as with distinct colors. To this end, we use RGB space for color similarity. The RGB color difference (Δs_{cq}) between the center pixel and the neighboring

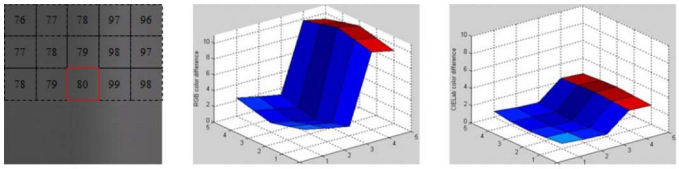


Fig. 7. Comparison of RGB and CIELab color difference. (a) Support window. (b) RGB color difference. (c) CIELab color difference.

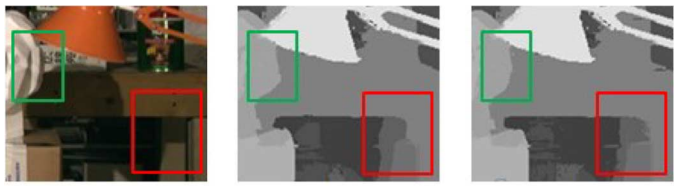


Fig. 8. Comparison of the original support and the proposed conditional support on "Tsukuba". (a) Left image. (b) Original support. (c) Conditional support.

pixel is calculated as in (4). The spatial distance (proximity) is calculated as Euclidean distance.

The adaptive support weight is based on the strength of grouping by similarity and proximity. The strength of grouping by similarity is defined using Laplacian kernel as

$$g_{\gamma_s}(\Delta s_{cq}) = \exp\left(-\frac{\Delta s_{cq}}{\gamma_s}\right) \tag{7}$$

with γ_s being an empirical similarity parameter. The strength of grouping by proximity is defined in the same way as in (7). The weights based on the spatial proximity with respect to the center pixel are constant for every shifted window while the weights based on the color similarity vary for each shifted window. Hence, the spatial fixed kernel might yield negative consequences in the specific area such as the disparity discontinuity area with similar color because it is blindly aggregated according to the distance and it causes wrong matches near disparity discontinuity. To alleviate this problem, we suggest the conditional adaptive support weight as

$$w(c,q) = \begin{cases} g_{\gamma_s}(\Delta s_{cq}) & \text{if } \Delta s_{cq} \le \epsilon \\ g_{\gamma_s}(\Delta s_{cq}) g_{\gamma_p}(\Delta p_{cq}) & \text{otherwise} \end{cases} \tag{8}$$

where Δp_{cq} is the spatial distance between pixel c and pixel q and ϵ is a color difference threshold determining the color similarity between two pixels.

Fig. 8 depicts the process where the conditional support weight improves the disparity map. Fig. 8 shows the left image and two disparity maps; Fig. 8(b) shows the estimate using original support always including proximity, while Fig. 8(c) shows the estimate with conditional support measure. At the border of the disparity discontinuity area with a similar color in the foreground and background (red box), the spatial proximity kernel may produce many wrong disparities due to the blind aggregation by the close spatial distance, as shown in Fig. 8(b). In this case, we exclude the proximity term to avoid the blurring support at the edge of disparity and exploit only the color similarity to determine the correct support according to even slightly different color difference. Therefore, our conditional support recovers a lot of errors as shown in Fig. 8(c). This is precisely the goal of the conditional adaptive support weight in (8).

C. Benefits of a Motion Cue

Although motion is a key factor in video processing, it has not been used for support weight computation within the localized window. Fig. 9 illustrates the benefits of using motion cues. We use the LASW method, in which proximity and similarity are exploited in the independent manner, and extend it to evaluate how the motion term affects the quality of the disparity maps. As the local methods require pixel-based computation, we use classic optical flow method with the weighted non-local term [19], which is one of state-of-the-art optical flow methods. We exploit the motion to compute the independently integrated support weight. The "car" and "skydiving" video frames are processed at a resolution of 480×270 and 480×276, respectively. The parameters used are fixed throughout the experiment. In Fig. 9, the selected left view (Fig. 9(a) and (f)) and its optical flow (Fig. 9(b) and (g)) are shown. Fig. 9(c) and (h) are obtained by using only the proximity term for the support weight, Fig. 9(d) and (i) are obtained by adding the similarity term, and Fig. 9(e) and (j) are obtained by adding the motion term. As shown in Fig. 9(a), it is challenging to discover the outline of the car since it is very ambiguous. In Fig. 9(c), many errors are observed at the edges of the moving car (red circle). In Fig. 9(d), some errors are recovered by using the color cue but edges are not preserved. In Fig. 9(e), incorporating the motion term preserves the edges even though they are visually ambiguous. We believe that this is due to the preserved background flow as shown in Fig. 9(b). Although there is ambiguity in the stereo correspondence, motion between a pair of successive video frames is much more consistent, especially in a localized window in background regions. There exist large forward motions in the "skydiving" video as shown in Fig. 9(g). Generally, moving objects tend to have a higher degree of saliency and viewers will fixate on the skydiver fast landing forward as shown in Fig. 9(f). Therefore, accurate disparity estimation is required at these moving edges. The dotted red line in Fig. 9(f) represents the moving edge of the skydiver. In Fig. 9(h) and (i), large smearing problem is observed at the edges while in Fig. 9(j), the problem is much alleviated by incorporating the motion cue for support weight. Note that the left edge of the foreground is compared and the occlusion appears at the right side of the foreground due to the negative disparity in Fig. 9(f). Disparity is estimating spatial correspondences while motion estimates temporal correspondences, so the additional information promotes disambiguation. Consequently, the results in Fig. 9 imply that the support weight integrating the motion cue yields more accurate disparity estimates, especially near the edges of moving objects.

D. Support Weight on Stereo Videos

The effectiveness of using motion cue for support weight has been verified in previous section and the conditional relation between similarity and proximity has been defined in Section III.B. We analyze the motion flow estimate and the relationship between similarity and motion to verify how the motion term should be integrated. The motion difference between two

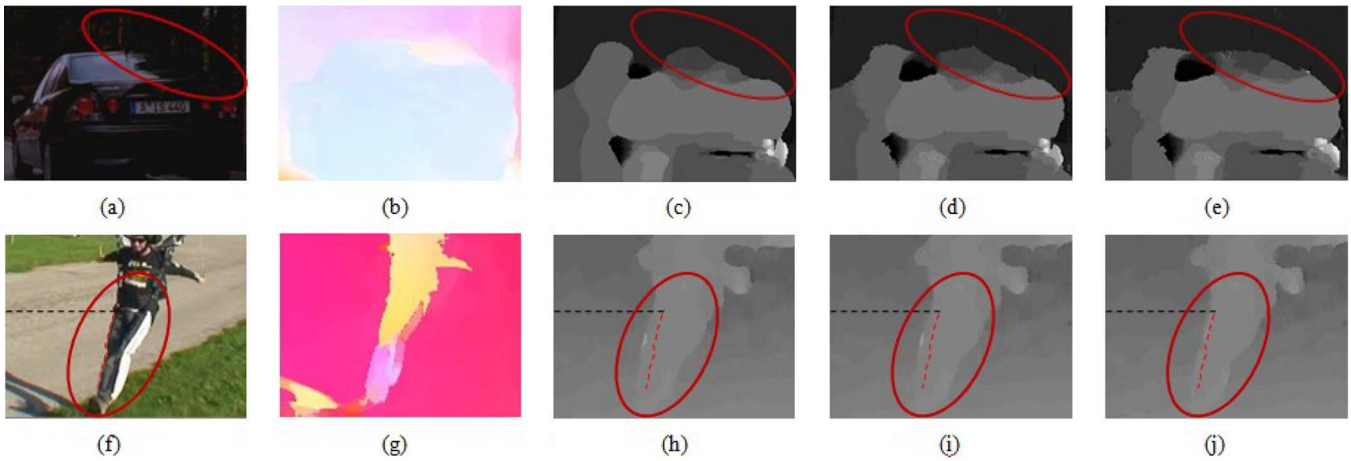


Fig. 9. Disparity maps for "Car" in the upper row and "Skydiving" in the lower row. (a) and (f) Left view. (b) and (g) Optical flow. (c) and (h) using only proximity. (d) and (i) using proximity and similarity. (e) and (j) using proximity, similarity, and motion.

pixels is calculated by using a measure of optical flow. There are two types of motion difference computation: absolute flow endpoint difference (ED) and angular difference (AD) [20]. We use ED because AD penalizes errors in larger flows less than errors in small ones [20], which is undesirable. Let $m_c = (u_c, v_c)$ and $m_q = (u_q, v_q)$ be the flow vectors of pixel c and pixel q, respectively. We suggest the truncated motion difference:

$$\Delta m_{cq} = min\left(\sqrt{(u_c - u_q)^2 + (v_c - v_q)^2}, \tau\right) \quad (9)$$

where τ is a truncation value. Such a model reduces the influence of flow outliers just as the truncated matching cost limits the influence of wrong matches [21]. We must keep in mind that the optical flow is an estimated value and cannot be completely error free. The support weight based on the three gestalt grouping principles should be redefined for video disparity. We suggest a correlated model, in which the conditional property should be inherited for the integrated support weight as shown in (10) at the bottom of the page. This model originates from the intuition that color similarity and motion tend to correlate with each other in general. For example, the center pixel and its neighboring pixel have a high likelihood of having different motion vectors if they also differ significantly in color, as expected near object edges. When this occurs, the correlated model decreases the overall support weight as compared with the independent model, since the Laplacian is raised to a power based on the large color difference. Additionally, the two pixels are likely to have similar motion if they also have the same color, as in the flat areas of an object surface. In this case, we can also expect to find a positive correlation among the two metrics. Therefore, the support weight will increase in reference to the independent model. However, while color is an observed quantity, motion is an estimated value. Therefore, color should take precedence over motion when there is a discrepancy between them and the correlation assumption fails. This is precisely what the model in (10) enforces. For example, if there is a large difference in color but a small difference in motion, then the value for the correlated support weight is decreased. Therefore, the support weight depends more on the color cue than the motion cue. In contrast, the independent model always treats all of the gestalt principles equally. In summary, we define conditional relation between similarity and proximity and correlated relation between similarity and motion.

IV. Disparity Computation and Occlusion Filling

Once the support weights are calculated, the aggregated cost is computed by aggregating the raw similarity measures, scaled by the support weights in the window. If we consider only the left support window, the cost computation may be erroneous since the right support window may have pixels from different disparity levels. To reduce such errors, the aggregated cost is computed by combining the support weights of both support windows as in [4]. The aggregated matching cost between pixel c and pixel c_d in Fig. 6 is given in the weighted mean form:

$$A(c, c_d) = \frac{\sum_{q \in W_c, q_d \in W_{c_d}} w(c, q) w(c_d, q_d) C_0(q, q_d)}{\sum_{q \in W_c, q_d \in W_{c_d}} w(c, q) w(c_d, q_d)} \quad (11)$$

where W_c and W_{c_d} represent the left and right support windows, respectively, and the function $w(c_d, q_d)$ is the support weight of pixel q_d in the right window.

After the aggregated matching costs have been computed within the disparity range, the disparity map is obtained by determining the disparity d_p of each pixel p through the Winner-Takes-All (WTA) algorithm:

$$d_p = \arg\min_{d \in S} A(c, c_d) \quad (12)$$

where S represents the set of all possible disparities.

$$w(c,q) = \begin{cases} g_{\gamma_m}(\Delta m_{cq})^{\frac{\Delta s_{cq}}{\gamma_s}} g_{\gamma_s}(\Delta s_{cq}) & \text{if } \Delta s_{cq} \le \epsilon \\ g_{\gamma_m}(\Delta m_{cq})^{\frac{\Delta s_{cq}}{\gamma_s}} g_{\gamma_s}(\Delta s_{cq}) g_{\gamma_p}(\Delta p_{cq}) & \text{otherwise} \end{cases} \quad (10)$$

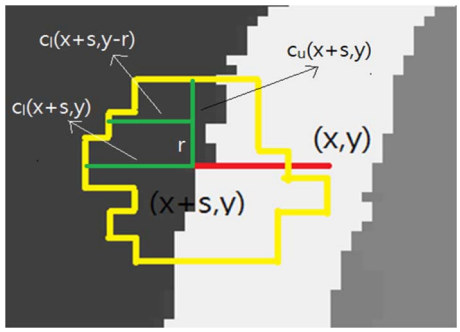


Fig. 10. Illustration of the occlusion filling process.

To make sure that both left and right disparities are spatially consistent, we perform a left-right consistency check to detect unreliable pixels. These unreliable pixels are those having different disparities on the left and right images. Fig. 10 illustrates an example of occlusion handling. In Fig. 10, for each unreliable pixel (x, y), the cross-based aggregation method [22] generates a neighborhood for $(x + s, y)$ as shown for the yellow region in Fig. 10, where $(x+s, y)$ is the left most reliable pixel. The white region indicates unreliable (occluded) region, dark gray region is background, and the light gray region is the foreground. All reliable pixels within the neighborhood vote for the candidate disparity value at (x, y). The unreliable pixel at (x, y) is filled with the majority of the reliable pixel in the voting region. By this method, the center pixel is not automatically selected as the center pixel for occlusion handling. Instead, a first non-occluded pixel is selected to define the neighborhood. In Fig. 10, a left disparity map is used as an example, where occlusion pixels (white) appear at the right side of the background, left side of the foreground if the disparity is positive. (In the right image, occlusion pixels would appear at the left side of the background, right side of the foreground). Only the occlusion pixels are selected and needed to be processed. For an arbitrary occlusion pixel (x, y), the method starts at its left neighbor pixel to determine whether it is an non occluded pixel. If it is occluded, continue to the left. If it is non-occluded, the procedure stops. In Fig. 10, for the pixel at (x, y), the process goes to the left for s pixels. A neighborhood is constructed based on the cross-based aggregation method on pixel $(x + s, y)$. Every non-occluded pixel within that region votes. The majority disparity values in that region is assigned to occlusion pixel (x, y). Fig. 10 presents an ideal situation where the majority is obviously the background, and consequently, the white region will be filled with the background.

Prior window-based voting methods [23] have been based on (x, y) instead of $(x + s, y)$. The number of non occlusion pixels in the window constructed based on (x, y) will be significantly smaller than that in the window constructed based on $(x + s, y)$. Therefore, such methods are much more sensitive to outliers due to fewer votes, and yield inaccurate result.

Other methods such as plane fitting [24] for multiple disparity planes, which is very computationally expensive. It is an iterative process that treats the occlusion pixel as outliers and finds the plane that minimizes the error for non occlusion regions, and fills the occlusion pixel as if it is on the plane. In the other hand, the proposed occlusion method is non-iterative and is thus more efficient.

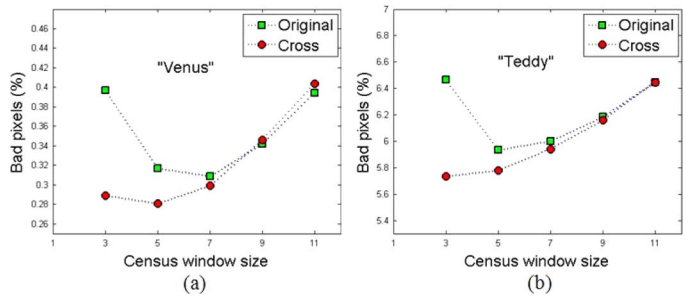


Fig. 11. Errors (Bad pixels) rate versus census window size. (a) "Venus". (b) "Teddy".

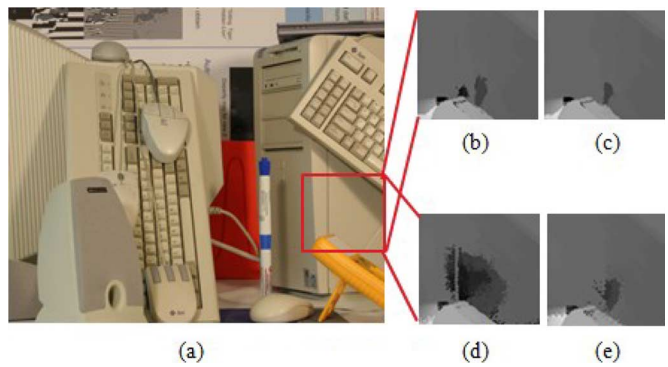


Fig. 12. Comparison of the original census (2 mode) and the three-moded census with a noise buffer on "Computer". (a) Left image. (b) Original census. (c) Three-moded census. (d) Original census on noise added image. (e) Three-moded census on noise added image.

V. Experiments and Results

A. Disparity Estimation Results on Stereo Images

To evaluate how the size of the original and cross-square census window affects the disparity performance, we use two Middlebury datasets ("Venus" and "Teddy"). As shown in Fig. 11(a), the error rate of original census (green) decreases sharply as the window size increases from 3 to 7. That is when the census transformed data contains more spatial structure information, and therefore the similarity measure is more accurate. However, the error rate increases as the window size increases from 7 to 15. It is due to the fact that larger census window would include more pixels from occlusion areas as well as including more noises, which decreases the accuracy of the similarity measure. The cross census window has four wings as shown in Fig. 2, and each wing consists of 3 pixels in the experiment. It is worth noting that the proposed cross-square window outperforms the original square one with even smaller window size as shown in Fig. 11.

We implement the three-moded census transform with a noise buffer for robustness to image noise in flat areas. Fig. 12 illustrates that the three-moded census with a noise buffer performs better than the original census in flat area. To simulate noise in flat areas, we add Gaussian noise, distributed as $\mathcal{N}(0, 10^{-4})$, to the original image. For the noise buffer α, the parameter β is set to be 50. First we perform the experiment on the original image, where the three-moded census reduces some errors in the flat area as shown in Fig. 12(c). Second we perform the experiment on the noise-added image. In this case, the proposed noise buffer is more effective, as it reduces much more errors, shown in Fig. 12(e). We also verify that the three-moded census

TABLE I
LOCAL METHOD PERFORMANCE EVALUATION ON MIDDLEBURY
(BAD PIXEL PERCENTAGE WITH THRESHOLD OF 1)

Methods	Rank	Avg.Err.(%)	Err. non-occluded pixels(%)			
			Tsukuba	Venus	Teddy	Cones
Proposed	13	5.12	2.10	0.12	5.46	2.12
PatchMatch [7]	15	4.59	2.09	0.21	2.99	2.47
CostFilter [9]	20	5.55	1.51	0.20	6.16	2.71
InfoPermeable	21	5.51	1.06	0.32	5.60	2.65
GeoSup [21]	28	5.80	1.45	0.14	6.88	2.94
AdaptDisCalib [6]	37	6.10	1.19	0.23	7.80	3.62
SegmentSupport [5]	53	6.44	1.25	0.25	8.43	3.77
AdaptWeight [4]	67	6.67	1.38	0.71	7.88	3.97

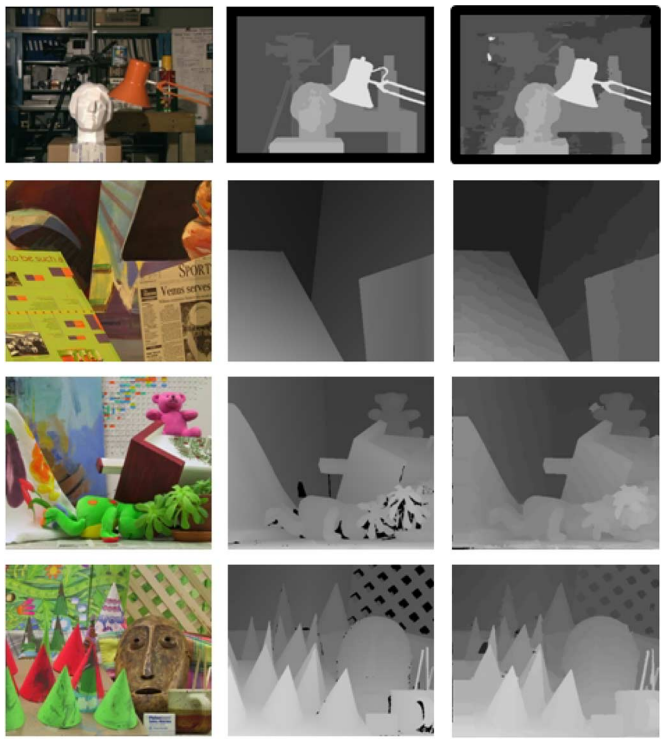


Fig. 13. Disparity maps for the "Tsukuba", "Venus", "Teddy" and "Cones". Centered column shows ground truth disparity map and right-most column shows the disparity map from the proposed algorithm.

TABLE II
PERFORMANCE COMPARISON OF METHODS ON FIVE STEREO VIDEOS
(BAD PIXEL PERCENTAGE WITH THRESHOLD OF 1)

Video/ # of frames	LASW	Cost-filter	Proposed method
Tunnel/ 99	1.435 %	2.157 %	0.997 %
Book/ 40	5.933 %	4.919 %	3.601 %
Temple/ 99	10.145 %	10.700 %	10.362 %
Street/ 99	9.978 %	7.305 %	7.246 %
Tanks/ 99	5.714 %	4.826 %	4.811 %

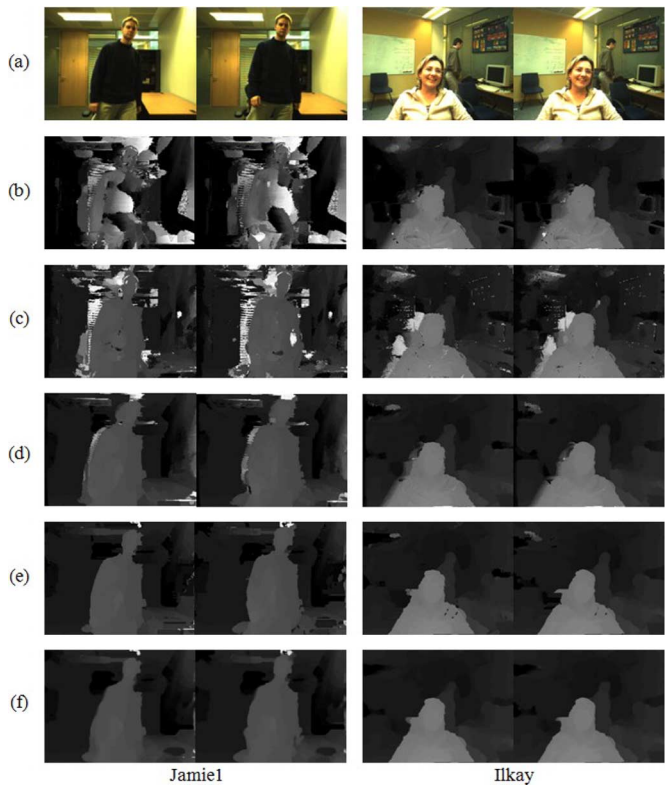


Fig. 14. Disparity map for "Jamie1" and "Ilkay". (a) Left frames. (b) LASW. (c) Cost-filter. (d) Proposed method. (e) After occlusion filling. (f) After TV [12].

shows better overall performance in terms of bad pixel rate than the original census.

The performance evaluation is performed on Middlebury datasets with ground truth disparity maps provided by Middlebury online benchmark [8]. The parameters are set to constant values: $\gamma_s = 33$, $\gamma_p = 20$, $\gamma_H = 29$, $\gamma_I = 45$, $\gamma_G = 14$ and $\epsilon = 3$. The size of the support window is 35×35 (the same size as the LASW [4]) and the size of the cross-square census window is 5×5 for square with 3 pixels for a wing. Table I summarizes the quantitative results taken from the Middlebury database for local methods. The bad pixel (error) rate is expressed as

$$B(\%) = \frac{100}{|\Omega|} \sum_{p \in \Omega} I(|D_p - d_p| > \theta) \tag{13}$$

where $|\Omega|$ represents the number of pixel in whole image and I denotes the indicator function. D_p represents the true disparity at pixel p and θ represents the bad pixel threshold. Our method achieves excellent results ranking 13th out of about 130 methods and is the best performing local method at the time of the submission. Our method is an efficient one-pass method with no iteration or postprocessing. It outperforms the original local method (LASW ranking 67th), using efficient algorithms and structures. Fig. 13 shows left images (in the first column), ground truth disparity maps (in the second column) and our disparity map (in the third column). The proposed method produces accurate dense disparity map as shown in Fig. 13. Our method ranks 1st on "Cones" in the both non-occlusion test and discontinuity test.

In the proposed method, it takes about 12 s to compute the disparity map on "Tsukuba". It has been presented in [25] that the LASW [4] can be adopted into a real-time application by using a Graphics Processing Unit (GPU). Therefore, the same could be done with our work since our method has a similar framework to [4].

B. Disparity Estimation Results on Stereo Videos

To assess the performance of the proposed method quantitatively on stereo videos, we use 5 synthetic stereo videos

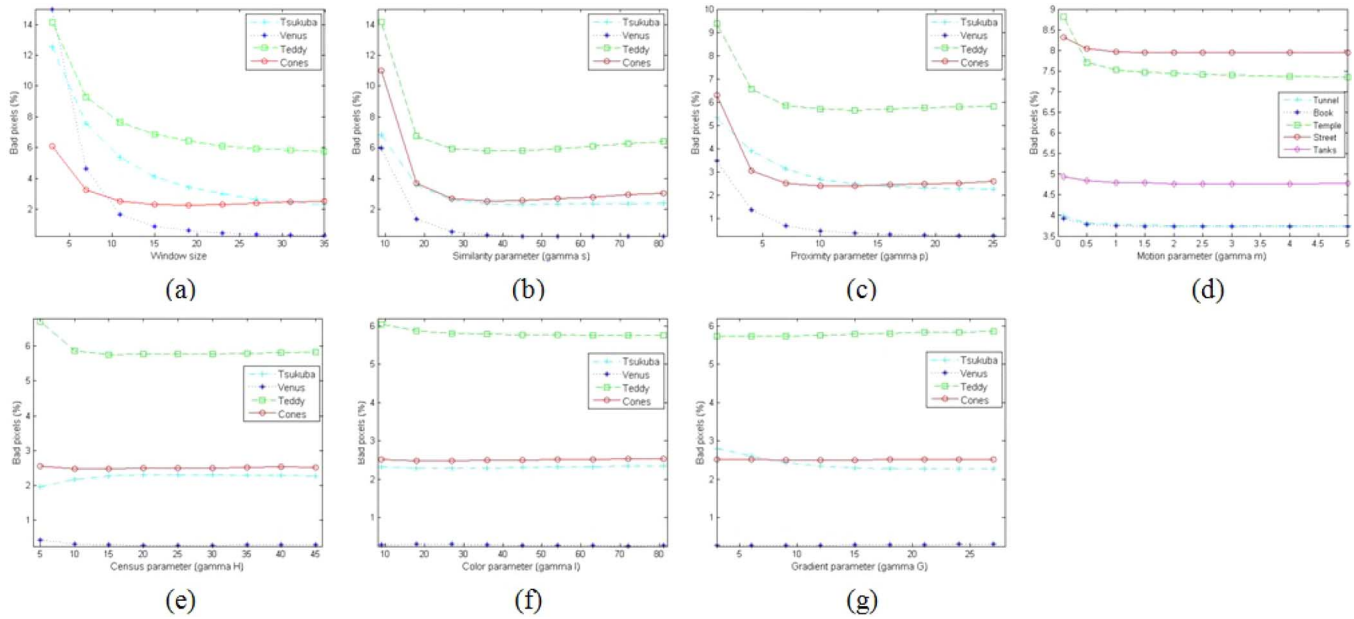


Fig. 15. Performance evaluation according to the window size and 6 parameters on four stereo images and five stereo videos. (a) Changing the window size. (b) Changing γ_s. (c) Changing γ_p. (d) Changing γ_m. (e) Changing γ_H. (f) Changing γ_I. (g) Changing γ_G while the other parameters are kept constant.

(400×300, 64 disparity range) with ground truth disparity [26]. We compare three methods (LASW, Cost-filter, and proposed method) without occlusion filling to compare their pure performance. The LASW method ranks 67th and the Cost-filter, which is one of the best performing local method ranks 20th on the Middlebury benchmark. All of three methods are efficient one-pass local methods having similar structure. Table II shows the average percentage of bad pixels (threshold of 1) over all frames. We ignore borders when computing statistics since they lack correspondences. Table II illustrates that the proposed method, using the motion cue has the best performance.

To assess the performance of the proposed method subjectively, we perform experiments on real-world videos, "Jamie1" and "Ilkay," scenes from the Microsoft i2i database (320×240, 64 disparity range). Jamie1 video is more challenging than Ilkay because it contains large flat areas and repetitive patterns, as shown in Fig. 14. Fig. 14(b) shows the disparity maps produced by LASW, Fig. 14(c) shows the disparity maps produced by Cost-filter, and Fig. 14(d) shows the disparity maps produced by the proposed method. Fig. 14 illustrates that the proposed method obtains the best quality of disparity map. On the other hand, LASW yields the worst quality and Cost-filter produces many errors in flat and repetitive areas. As additional results, Fig. 14(e) shows the disparity maps where the occlusion areas in Fig. 14(d) are filled by valid values and Fig. 14(f) shows the disparity maps refined with TV algorithm [12], which reduces errors such as spatial noise and temporal inconsistencies in the background.

C. Sensitivity to the Parameter Values

The robustness of the proposed method when changing the parameters is examined. Fig. 15(a) shows the performance evaluation for different support window size on four Middlebury stereo images and illustrates that the proposed method is fairly insensitive to the support window size when the size is larger than 15×15. This is due to the fact that the advanced support weight method groups same depth pixels well, and thus outliers do not increase even though the window size increases. Fig. 15(b) and (c) show the performance according to changing the similarity parameter (γ_s) and the proximity parameter (γ_p). They also illustrate that the proposed method is robust to the different parameter setting when they are larger than a certain value. Fig. 15(d) shows the performance for different motion parameter (γ_m) value on five stereo videos where the performance of the proposed method is almost constant to the motion parameter values. As shown in Fig. 15(e), (f), and (g), the performance is also insensitive to the three cost measure parameters $(\gamma_H, \gamma_I, \gamma_G)$. Consequently, the six parameter values are not critical in the performance of the proposed method since they are used in the efficiently integrated form as in (6) and (10).

VI. Conclusion

In the local stereo matching, the accuracy of the disparity map depends on the similarity measure and the support weight. We propose a novel three-moded census with a noise buffer and cross-square census to increase robustness to image noise in flat areas and the accuracy of similarity measure. We show that the combination of three similarity measures produces more reliable cost measure in a variety of image regions. To obtain more precise support weight, conditional and correlated support model are introduced. We consider object motion flow to take advantage of benefits of motion in video disparity estimation. Simulation results verify that the proposed method outperforms the other state-of-the-art local methods on both stereo images and videos. Moreover, the proposed method is not sensitive to the parameter values.

REFERENCES

[1] H. Hirschmuller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 9, pp. 1582–1599, Sep. 2009.

[2] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 9, pp. 920–932, Sep. 1994.

[3] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, Jun. 1997, pp. 858–863.

[4] K.-J. Yoon and I. S. Kweon, "Adaptive support weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 650–656, Apr. 2006.

[5] S. Mattoccia, F. Tombari, and L. D. Stefano, "Segmentation-based adaptive support for accurate stereo correspondence," in *Proc. PSIVT*, 2007, pp. 427–438.

[6] Y. Liu, Z. Gu, X. Xu, and Q. Zhang, "Local stereo matching with adaptive support-weight, rank transform and disparity calibration," in *Pattern Recognit. Lett. 29*, 2008, pp. 1230–1235.

[7] C. Rhemann, M. Bleyer, and C. Rother, "PatchMatch stereo—stereo matching with slanted support windows," in *Proc. BMVC*, 2011.

[8] D. Scharstein and R. Szeliski, Middlebury Stereo Evaluation Version 2, 2010. [Online]. Available: http://vision.middlebury.edu/stereo/eval.

[9] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, Jun. 2011, pp. 3017–3024.

[10] M. Bleyer and M. Gelautz, "Temporally consistent disparity maps from uncalibrated stereo videos," in *Proc. ISPA*, 2009, pp. 383–387.

[11] B. Horn and B. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, pp. 185–203, 1981.

[12] R. Khoshabeh, S. H. Chan, and T. Q. Nguyen, "Spatio-temporal consistency in video disparity estimation," in *Proc. ICASSP*, 2011, pp. 885–888.

[13] Z. Lee, R. Khoshabeh, J. Juang, and T. Q. Nguyen, "Local stereo matching using motion cue and modified census in video disparity estimation," in *Proc. Signal Processing Conf. (EUSIPCO)*, Aug. 2012, pp. 1114–1118.

[14] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang, "Noise estimation from a single image," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, 2006, pp. 901–908.

[15] L. Zhang, S. Vaddadi, H. Jin, and S. K. Nayar, "Multiple view image denoising," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, 2009, pp. 1542–1549.

[16] D. Angens, *From Gestalt Theory to Image Analysis : A Probabilistic Approach*. New York, NY, USA: Springer, 2008.

[17] G. Papari and N. Petkov, "Adaptive pseudo dilation for gestalt edge grouping and contour detection," *IEEE Trans. Image Process.*, vol. 17, pp. 1950–1962, 2008.

[18] C. Connolly and T. Fleiss, "A study of efficiency and accuracy in the transformation from RGB to CIELAB color space," *IEEE Trans. Image Process.*, vol. 6, pp. 1046–1048, 1997.

[19] D. Sun, S. Roth, and M. J. Black, "Secrets of optical flow estimation and their principles," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, 2010, pp. 2432–2439.

[20] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," in *Proc. IEEE Int. Conf. Computer Vision Workshops (ICCV Workshops)*, 2007, pp. 1–8.

[21] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann, "Local stereo matching using geodesic support weights," in *Proc. ICIP*, 2009, pp. 2093–2096.

[22] K. Zhang, J. Lu, and G. Lafruit, "Cross-based local stereo matching using orthogonal integral images," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1073–1079, 2009.

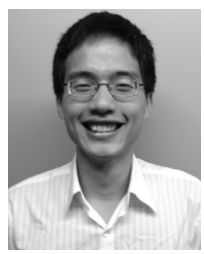
[23] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On building an accurate stereo matching system on graphics hardware," in *Proc. IEEE International Conf. Computer Vision Workshops (ICCV Workshops)*, Nov. 2011, pp. 467–474.

[24] H. Tao and H. S. Sawhney, "Global matching criterion and color segmentation based stereo," in *Proc. Workshop Application of Computer Vision (WACV2000)*, 2000, pp. 246–253.

[25] L. Wang, M. Gong, R. Yang, and M. Gong, "A performance study on different cost aggregation approaches used in real-time stereo matching," in *Proc. IJCV*, 2007, vol. 75, pp. 283–296.

[26] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. A. Dodgson, "Real-time spatio-temporal stereo matching using the dual-cross-bilateral grid," in *Proc. ECCV*, 2010.

Zucheul Lee (S'12) received the B.S. and M.S. degrees in electrical engineering from Yonsei University, Seoul, South Korea, in 1997 and 2003, respectively. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering in the Department of Electrical and Computer Engineering, University of California, San Diego (UCSD), La Jolla. He worked at Korea Telcom in South Korea from 1997 to 2009 as a Wireless Sensor Network and IPTV research engineer. His research interests include stereo image/video disparity and motion estimation for 3-D applications.



Jason Juang is a Ph.D. student at University of California, San Diego (UCSD), in the Electrical and Computer Engineering department. He received his M.S. in electrical and computer engineering in 2012 at UCSD, and he received his B.S. in electrical engineering in 2010 at National Taiwan University. He has specialized in image/video processing and computer vision. He also has a background in hardware programming.

Truong Q. Nguyen (F'05) received the B.S., M.S., and Ph.D. degrees from California Institute of Technology, Pasadena, in 1985, 1986, and 1989, respectively, all in electrical engineering. He is currently a Professor with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla. He is the coauthor (with Prof. G. Strang) of a popular textbook, Wavelets & Filter Banks (Wellesley-Cambridge Press, 1997), and the author of several MATLAB-based toolboxes on image compression, electrocardiogram compression, and filter bank design. He has more than 300 publications. His research interests are video processing algorithms and their efficient implementation. Dr. Nguyen was the recipient of the IEEE Transactions on Signal Processing Paper Award (Image and Multidimensional Processing area) for the paper he co-wrote with Prof. P. P. Vaidyanathan on linear-phase perfect-reconstruction filter banks (1992). He received the NSF Career Award in 1995 and is currently the Series Editor (Digital Signal Processing) for Academic Press. He served as Associate Editor of the IEEE Transactions on Signal Processing in 1994–1996, for the Signal Processing Letters in 2001–2003, for the IEEE Transactions on Circuits and Systems in 1996–1997 and 2001–2004, and for the IEEE Transactions on Image Processing from 2004 to 2005.