Orientation-Guided Geodesic Weighting for PatchMatch-based Stereo Matching

Yutong Jiang[a,b], Changming Sun[b,*], Xiao Tan[c], Li Yang[a,*]

[a] *School of Mechatronical Engineering, Beijing Institute of Technology, 5 South Zhongguancun Street, Haidian District, Beijing 100081, China.*
[b] *CSIRO Digital Productivity, PO Box 52, North Ryde, NSW 1670, Australia.*
[c] *Computer Science Department, The University of Hong Kong, Pokfulam Road, Hong Kong, China.*

Abstract

Recently, PatchMatch-based methods for local stereo matching are experiencing great progress with the use of compact and over-segmented regions that have similar intensities or colors. Using patches as support regions, this paper proposes an orientation-guided geodesic weighting (OGGW) strategy to search for an approximate shortest path from a support pixel in the patch to a current pixel of interest along a guided orientation. The OGGW is computed by accumulating intensity differences or color dissimilarities between connected pixels along the path. After obtaining matching cost updates by model fitting, the OGGW is used for

*Corresponding authors.
Email addresses: Changming.Sun@csiro.au (Changming Sun), yanglibit@bit.edu.cn (Li Yang)

weighted averaging on the updated costs to obtain a filtered cost volume. In addition, a new filter that combines the PatchMatch filter with curved surface fitting (PMF-CS) is presented in this paper. Curved surface fitting along with outliers removal is carried out to seek for a reliable regression model for estimating the disparities on a patch to achieve a disparity map with sub-pixel accuracy. We conduct a number of experiments to evaluate the performances of OGGW and PMF-CS on cost volume filtering and disparity estimation. Experimental results show that our algorithm produces accurate stereo matching results and outperforms the current state-of-the-art PatchMatch-based methods.
Keywords: 3D/stereo scene analysis, Stereo matching, PatchMatch-based filter, Orientation-guided geodesic weighting, Curved surface fitting.

1. Introduction

Dense two-frame stereo matching, a crucial issue in the area of computer vision, has been studied for decades and employed in many applications such as navigation [23], scene reconstruction [40], object recognition [12], and 3D recovery [11]. A vision task requires a high-accuracy disparity map to be obtained from dense matching. However, a produced disparity map may not be reliable sometimes due to noise, distortion, or occlusion. To

handle these problems, researchers have been making efforts in developing robust stereo matching algorithms under various conditions. According to a taxonomy and evaluation of dense two-frame stereo correspondence algorithms proposed by Scharstein and Szeliski [30], most stereo matching algorithms can be categorized into global and local methods. Generally speaking, global methods usually formulate the stereo matching problem as the minimization over an energy function defined on the Markov random field model where a smoothness assumption is made to achieve better results but with a higher computational cost [34]. This is because global energy minimization algorithms using either graph cuts [6, 16], maximum surface [31], or belief propagation [33] is often time-consuming. On the other hand, local methods firstly calculate the stereo matching costs between two frames; then cost aggregation is carried out with adaptive support weighting in local regions based on the hypothesis that neighboring pixels are likely to be on the same object and thus have similar disparities; and finally a disparity map is extracted from the aggregated costs by a winner-take-all (WTA) strategy.

The proposed method is a local method with an efficient implementation for achieving fast edge-aware aggregation. Two

sub-problems are highlighted: one focuses on adaptive support weighting (ASW) to evaluate the contribution from neighboring pixels in the same local region to the current pixel of interest which is referred as an "anchor pixel" in this paper; the other focuses on local regions such as shiftable windows [24] and adaptive windows [15] to preserve depth discontinuity.

An extensive evaluation study gives the summary on the existing ASW methods on support weighting for local stereo matching [13]. Recently, edge-aware filtering on a cost volume has enabled local matching algorithms to compete with global algorithms in terms of results quality and computational speed. Yoon and Kweon [41] introduced the bilateral filter (BF) [37] to stereo matching to achieve an edge-aware disparity calculation. As BF is computationally expensive, an increasing number of approximation methods are emerged to improve the computational speed such as fast BF [25], O(1) BF [26], and real-time BF [38]. However, these are quantization-based methods that do not perform well as summarized in [28]. The guided filter (GF) proposed by He et al. [10] exhibited its superiority over BF for the high-quality results at real-time processing speeds. It is also applied to smooth the cost volume in [27], and outperforms most local methods on the Middlebury benchmark [29] for

both speed and accuracy. However, GF does not always perform well due to its predefined square window shape for the lack of spatial adaptivity. Furthermore, adaptive support regions are introduced in the cross-based local multipoint filtering (CLMF) method [18], which outperforms both BF and GF to avoid covering unrelated regions with fixed windows. However, the CLMF method utilizes adaptive support weightings with intensity or color similarity only, neglecting the spatial information between pixels which may be represented by a geodesic distance. No doubt, the geodesic distance can directly influence the weighting from a support pixel to the anchor pixel in a local region for a weighted averaging process. Hosni et al. [14] proposed a geodesic support weighting that is inversely proportional to the geodesic distance. This geodesic distance between a support pixel and the anchor pixel in a square window is defined as the cost along a shortest path that connects the two pixels, which is calculated as the minimum sum of a sequence of color differences between neighboring points with 8-connectivity. This strategy cannot be implemented efficiently because the shortest path is selected from the set of all possible paths between the support pixel and the anchor pixel in a square support window.

Compared with the methods that use a segmentation-based approach to

obtain the support regions for stereo matching [20, 39, 4, 36], the PatchMatch-based methods [2, 3, 19] with over-segmentation have superior performances. The image patches, which include pixels with similar intensities or colors and have compact and regular shapes [7], are used as the support regions to ensure that no large disparity jumps occur within one region. This is why patch-based methods have advantages over segmentation-based methods for local stereo matching.

In this paper, an initial cost volume is constructed with cost calculations between candidate matching pixels on the reference and target images as shown in Fig. 1(a). Tan et al. proposed a multipoint filtering method with local polynomial approximation and range guidance (multipoint LPA) on the disparity map [35]. This method estimates the disparities by incorporating information on intensities or colors and the spatial positions of pixels, and avoids incorrect smoothing caused by incomplete information. However, it may be too late for an algorithm to update the disparity map if the correct disparities are excluded at the WTA step. Considering the cost volume which includes more comprehensive information on corresponding pixels, two steps are carried out on the cost volume. Firstly, we adopt the multipoint LPA method on the initial cost

volume to obtain all the cost updates by model fitting. Secondly, we propose an orientation-guided geodesic weighting (OGGW) strategy for weighted averaging on the updated costs. The normalization process is carried out on the costs of all the points in the initial cost volume. The points with normalized costs equalling 0.5 are selected as an example to illustrate the process of cost volume filtering, and the effects of each step during the process are given in Fig. 1. The selected points with initial costs are given in Fig. 1(b). Then these costs are smoothed using the multipoint LPA models for each slice independently to obtain the cost updates as shown in Fig. 1(c), and the updated cost are finally edge-aware filtered using a weighted averaging process with our OGGW strategy as shown in Fig. 1(d). In the process of weighted averaging on cost updates, we choose the simple linear iterative clustering (SLIC) algorithm [1] to obtain patches as the support regions for our OGGW strategy. There are two observations indicating that the SLIC method is superior to other state-of-the-art methods [21, 22, 17]. One is that the patches obtained by SLIC preserve the boundaries of objects in the image well, and the other is that the SLIC method takes a shorter runtime compared with others. The patches obtained from the reference image are not only used as the support regions

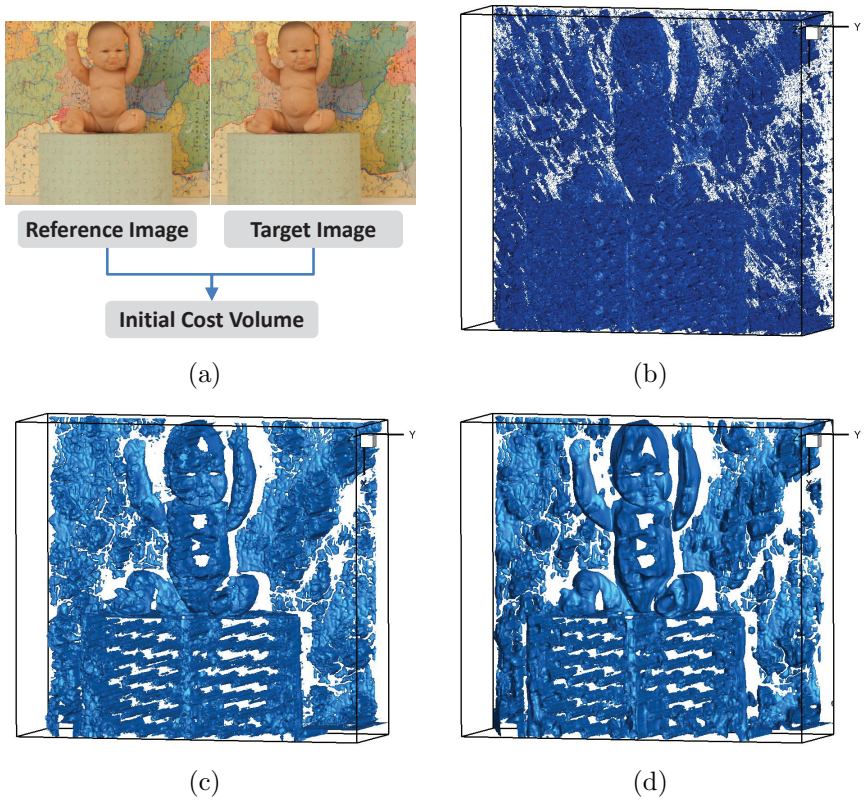


Figure 1: Illustration of the cost volume filtering process. (a) The process of obtaining the initial cost volume. (b) The points with normalized costs equalling 0.5 as an example are selected from the initial cost volume. (c) The cost updates of (b) using multipoint LPA models. (d) The weighted averaging of (c) using our OGGW strategy.

for weighted averaging on the updated costs, but also applied to the process of disparity estimation. Because the pixels from the same patch are more likely to share the same curved surface of objects, the disparity map for each patch is obtained with a curved surface model which provides with sub-pixel accuracy. This method is called PatchMatch filter with curved surface fitting (PMF-CS). The PMF-CS method achieves higher accuracy than the PatchMatch filter with constant (PMF-C) or slanted plane fitting (PMF-S) on the disparity map [19].

This paper presents a new local stereo matching method with two main contributions: (1) Proposing an OGGW strategy for weighted averaging on the updated costs for patches to filter the cost volume; (2) Proposing a PMF-CS method via curved surface fitting with outliers removal to obtain a reliable model on each patch in order to obtain the disparity map with sub-pixel accuracy. Experimental results demonstrate the superiority of our method.

2. Outline of the Proposed Method

Firstly, the preprocessing component of our method consists of two steps: (1) a required number of patches on the reference image is obtained using the

SLIC method [1]; and (2) initialization of the cost volume is carried out with rectified stereo images as input [32]. Secondly, a cost volume filtering process is carried out in two steps: (1) in order to suppress noises on the cost volume, the multipoint LPA method [35] is adopted to obtain the cost updates on the initial costs by model fitting; and (2) for weighted averaging on the updated costs, an OGGW strategy is proposed to obtain the weightings from support pixels in the patch to the anchor pixel by searching for approximate shortest paths along guided orientations. After that, a filtered cost volume with edge preservation is obtained. Thirdly, an initial disparity map is obtained by a WTA step to select the non-occluded pixels with a left-right cross checking procedure. In order to improve the accuracy of the disparity map, a PMF-CS method is proposed for carrying out reliable curved surface fitting for each patch on the initial disparity map. Finally, we obtain a disparity map with sub-pixel accuracy. The framework of our method is shown in Fig. 2.

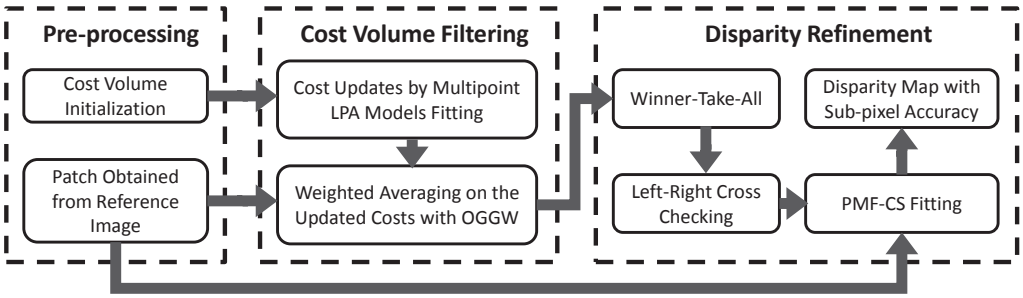


Figure 2: The framework of the proposed method.

3. Cost Volume Filtering

The proposed cost volume filtering method includes two procedures. One is the cost updates using the multipoint LPA method. This method incorporates spatial position and intensity value or color components into the LPA model to update the initial cost $C(p, p_d)$, which is calculated with pixel p on the reference image I_R and its corresponding pixel p_d on the target image I_T with a shift d. The other procedure is the weighted averaging on the updated costs using the OGGW strategy on patches. The reference image I_R is decomposed into K patches with $P = \{P(k) \,|\, \bigcup_{k=1}^{K} P(k) = I_R, \forall k \neq l, P(k) \bigcap P(l) = \emptyset\}$. Here K depends on the required number of patches, and pixel p on the patch with label k is noted as $p \in P(k)$, as shown in Fig. 3.

According to [35], the LPA model is built based on the matching costs in a square window Ω_{q_i} centered on pixel q_i in the reference image, and the anchor pixel p is one of the pixels in Ω_{q_i} as $p \in \Omega_{q_i}$. The matching costs in the square window are fitted using a linear regression model with coefficients $\boldsymbol{A}_{q_i}$ and B_{q_i} by taking advantage of its simplicity and high computational efficiency. The linear multipoint LPA model is given as

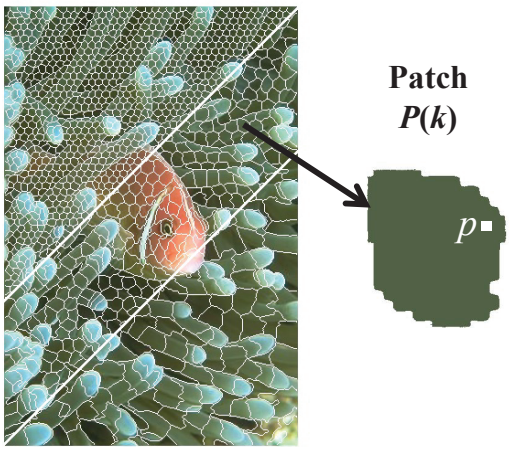


Figure 3: Left: Patch obtained using the SLIC method with approximate patch sizes of 50, 150, and 500 pixels; Right: Patch $P\big(k\big)$ extracted from the reference image with pixel p on $P\left(k\right)$.

$$C_{q_i}\left(p, p_d\right) = \boldsymbol{A}_{q_i}\boldsymbol{S}_p + B_{q_i} \tag{1}$$

with

$$\boldsymbol{A}_{q_i} = \left[\alpha_{q_i}, \beta_{q_i}, \boldsymbol{\gamma}_{q_i}\right]$$

$$\boldsymbol{S}_p = \left[x_p, y_p, \boldsymbol{I}\left(p\right)\right]^{\mathrm{T}}$$

Here the output $C_{q_i}\left(p, p_d\right)$ is obtained from the multipoint LPA model on a square window Ω_{q_i} centered on pixel q_i as a cost update of the initial cost $C\left(p, p_d\right)$. The input $\boldsymbol{S}_p$ represents a vector consisting of the spatial coordinate $\big(x_p, y_p\big)$ and the intensity value or color components $\boldsymbol{I}\left(p\right)$ of pixel

p on the reference image. The vector $\boldsymbol{A}_{q_i}$ with elements α_{q_i}, β_{q_i}, and $\boldsymbol{\gamma}_{q_i}$ are the corresponding coefficients of x_p, y_p, and $\boldsymbol{I}(p)$ respectively. B_{q_i} is a constant. The least square method is employed to estimate the coefficients $\boldsymbol{A}_{q_i}$ and B_{q_i} of the regression model [35]. An anchor pixel p in Ω_{q_i} is included in multiple square windows centered on pixel $q_i (i = a, b, c, \cdots)$ as shown in Fig. 4. It is assumed that the anchor pixel p is covered by multiple LPA models for overlapping square windows Ω_{q_i}. Thus, the final cost estimation $C'(p, p_d)$ for the anchor pixel p is obtained using weighted averaging on the updated costs from various LPA models covering the anchor pixel p by

$$C'(p, p_d) = \frac{\sum\limits_{q_i : p \in \Omega_{q_i}} w_{pq_i} C_{q_i}(p, p_d)}{\sum\limits_{q_i : p \in \Omega_{q_i}} w_{pq_i}} \tag{2}$$

where w_{pq_i} is the weighting related to the anchor pixel p and its neighboring pixel q_i in the same patch, and this weighting is to be introduced in Section 4. According to Eq. (2), Ω_{q_i} is the square window for the multipoint LPA model, and the condition for the summation of weighted cost updates is $q_i : p \in \Omega_{q_i}$, which is equivalent to $q_i \in \Omega_p$. The region Ω_p requires pixel q_i within it to have a similar intensity or color with pixel p and also have the spatial proximity with pixel p, and the patches we have already obtained from the

reference image can satisfy the requirements of Ω_p. Assuming that the anchor pixel p with patch label k is included in patch $P(k)$ $\big(p \in P(k)\big)$, the patch $P(k)$ can replace region Ω_p for weighted averaging because of the similar intensities or colors and the proximate spatial distances of pixels in the same patch. Then Eq. (2) can be rewritten as

$$C'(p, p_d) = \frac{\sum\limits_{q_i \in P(k)} w_{pq_i} C_{q_i}(p, p_d)}{\sum\limits_{q_i \in P(k)} w_{pq_i}} \tag{3}$$

The final $C'(p, p_d)$ is estimated by weighted averaging on the updated costs $C_{q_i}(p, p_d)$, which are obtained from multiple LPA models centered on each pixel q_i lying in patch $P(k)$ with the related weighting to the anchor pixel p.

The selection on the size of the square window Ω_{q_i} for the cost update by multipoint LPA model fitting and patch $P(k)$ for weighted averaging on the updated costs are to be decided. A box filter that has a fixed computational complexity with respect to the size of the square window is used to update the costs by the multipoint LPA model. The final cost estimate for the anchor pixel p is obtained by weighted averaging on the updated costs for pixel p, which has already been fitted with the multiple LPA models centered on every pixel in patch $P(k)$. To make the pixels on the boundaries of a patch

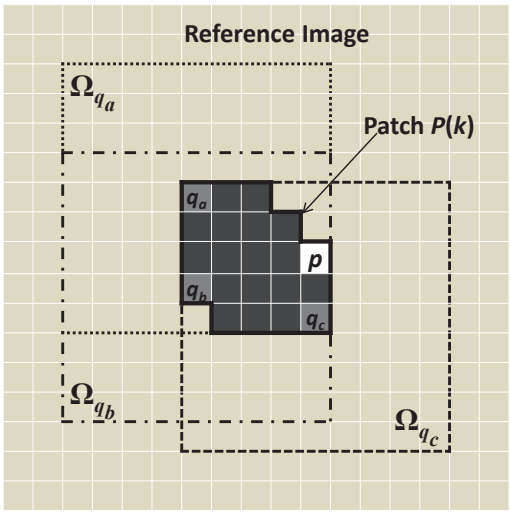


Figure 4: Illustration on support regions Ω_{q_a}, Ω_{q_b}, and Ω_{q_c} for multipoint LPA cost volume filtering models centered on pixels q_a, q_b, and q_c respectively which are pixels in the same patch $P(k)$.

covered by the multipoint LPA model as shown in Fig. 4, the size of the square window for multipoint LPA models is defined as $\left(2r_w - 1\right) \times \left(2r_h - 1\right)$, which is determined by the maximal width r_w and height r_h of the patch. The shape of patch $P(k)$ is arbitrary and depends on the over-segmentation of the reference image. Meanwhile, the size of patch $P(k)$ is determined by the number of pixels within it, which is noted as $|P(k)|$.

As shown in Fig. 1, a more reliable cost volume filtering is obtained from a process of weighted averaging on the updated costs. The boundaries of objects in the filtered cost volume are not over-smoothed with the use of

patches as support regions during the weighted averaging process.

4. Orientation-Guided Geodesic Weighting (OGGW)

In this section, we will discuss the weighting w_{pq_i} between the anchor pixel p and a support pixel q_i in the process of weighted averaging on cost updates. Even though many ASW methods have been proposed for local stereo matching, none of them is developed especially for patch computation. The following two aspects provide the reasons why patch weighting is different from the current ASW merhods: (1) the pixels in the same patch with similar intensities or colors and proximate spatial distances will make the weighting from support pixels to the anchor pixel on the same patch more reliable; and (2) the patch is a fixed support region for all pixels inside, thus we do not need to search for each support region for each pixel. To obtain the geodesic support weighting, Hosni et al. [14] utilized the method of [5] to calculate the approximations of the geodesic distances from the support pixels to the anchor pixel at the center of a square support window. In this process, a mask can be used to compute the color differences of neighboring points, which traverses the whole square support window in a row major order through three iterations of forward and backward passes at least. For an efficient alternative, we focus on an

orientation-guided search from each support pixel to the anchor pixel in a patch to calculate an approximate shortest path, instead of forward and backward searches with several iterations in a row major order from all the pixels in the square support window. Thus, a novel OGGW strategy applied to the patch is presented in this paper. The weighting from a support pixel to the anchor pixel is calculated by firstly searching for an approximate shortest path along a guided orientation, and then computing the summation over a sequence of distances between neighboring points along this path as an approximate geodesic distance. This distance between two neighboring points is defined based on the intensity difference or color dissimilarity of two pixels.

According to the final cost estimation obtained by Eq. (3), the w_{pq_i} is used as the weighting for the support from pixel q_i in patch $P(k)$ to the anchor pixel p. The weighting is computed with the information in the reference image, and the weighting w_{pq_i} between pixel p and q_i is given by

$$w_{pq_i} = \exp\big(-\frac{D(p,q_i)}{\gamma}\big) \tag{4}$$

Here γ is a user-defined parameter to control the weighting. $D(p,q_i)$ is the

approximate geodesic distance between the support pixel q_i and the anchor pixel p on the same patch, and is calculated on an approximate shortest path by searching for a sequence of pixels along the guided orientation with almost constant intensity or color. As shown in Fig. 5(a), $q_{i_n}\,(n = 1, 2, 3, \cdots, 8)$ represents the 8-connectivity of pixel q_i. The intensity or color difference between pixel q_i and one of its 8-connectivity neighbor q_{i_n} is calculated by

$$d\left(q_i, q_{i_n}\right) = \left\|\boldsymbol{I}\left(q_i\right) - \boldsymbol{I}\left(q_{i_n}\right)\right\|_2 \tag{5}$$

Here $\boldsymbol{I}\left(q_i\right)$ and $\boldsymbol{I}\left(q_{i_n}\right)$ represent the intensity values for gray scale images or the RGB channels for color images at pixel q_i and pixel q_{i_n} in the reference image respectively. Then the approximate geodesic distance $D\left(p, q_i\right)$ is defined by

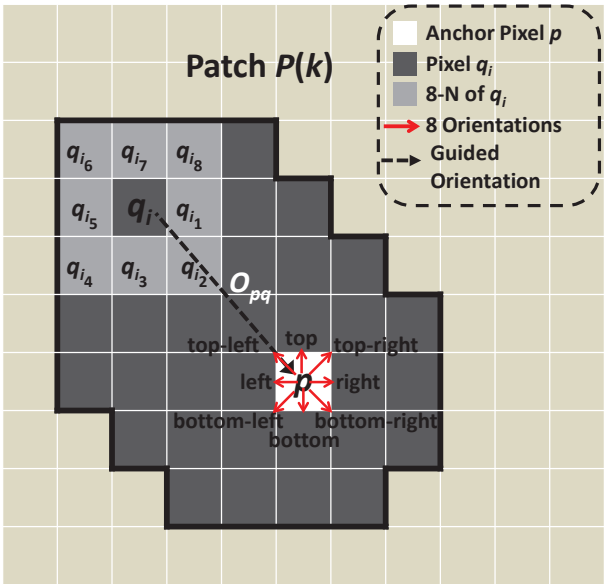
$$D\left(p, q_i\right) = \min_{q_{i_n} \in O_{pq_i}} \left(d\left(q_i, q_{i_n}\right) + D\left(p, q_{i_n}\right)\right) \tag{6}$$

Here O_{pq_i} is the orientation-guided subset decided by the guided orientation from pixel q_i to pixel p. As shown in Fig. 5(a), pixel q_i is possibly located on eight relative orientations [8] of the anchor pixel p, among which the orientations left, right, top, and bottom are four principal orientations. If pixel q_i is located in the region between the extended directions of the top

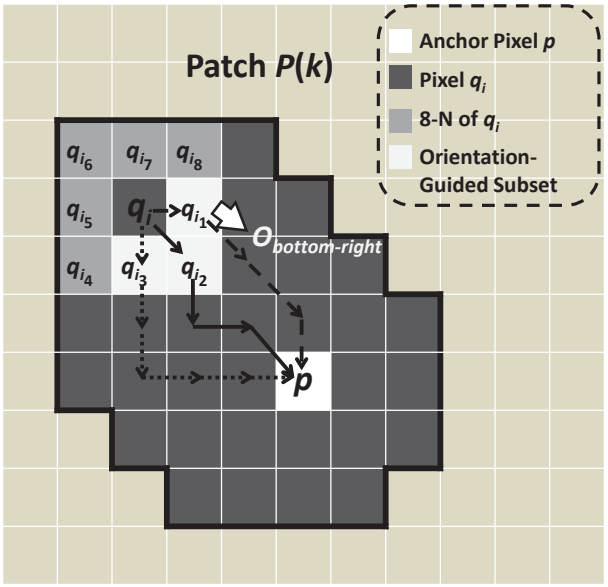
and the left of pixel p, it is defined as located on the orientation top-left of pixel p. The orientations top-right, bottom-left, and bottom-right are defined in a similar manner. According to the spatial locations between pixels q_i and p as shown in Fig. 5(a), pixel q_i is located on the top-left of pixel p, and pixel p is located on the bottom-right of pixel q_i. Then the orientation bottom-right is defined as the guided orientation from pixel q_i to the anchor pixel p for calculating the approximate geodesic distance. When the guided orientation is determined, the orientation-guided subset O_{pq_i} is then selected from the eight possible subsets $\{O_{left}, O_{right}, O_{top}, O_{bottom}, O_{top-left}, O_{top-right}, O_{bottom-left}, O_{bottom-right}\}$ corresponding to the eight guided orientations respectively. The 8-connectivity of pixel q_i including eight pixels $q_{i_n}\,(n = 1, 2, 3, \cdots, 8)$ is divided into eight orientation-guided subsets as $O_{left} = \{q_{i_4}, q_{i_5}, q_{i_6}\}$, $O_{right} = \{q_{i_1}, q_{i_2}, q_{i_8}\}$, $O_{top} = \{q_{i_6}, q_{i_7}, q_{i_8}\}$, $O_{bottom} = \{q_{i_2}, q_{i_3}, q_{i_4}\}$, $O_{top-left} = \{q_{i_5}, q_{i_6}, q_{i_7}\}$, $O_{top-right} = \{q_{i_1}, q_{i_7}, q_{i_8}\}$, $O_{bottom-left} = \{q_{i_3}, q_{i_4}, q_{i_5}\}$, and $O_{bottom-right} = \{q_{i_1}, q_{i_2}, q_{i_3}\}$. Each of them has three pixels to ensure that all the candidate pixels along the guided orientation from pixel q_i to p are used in the comparison of accumulated differences. Each step in obtaining the approximate shortest path from

pixel q_i to the anchor pixel p is decided by a selection within its orientation-guided subset.

According to Eq. (6), the approximate geodesic distance $D(p, q_i)$ is obtained by minimizing the sum of two components: one is the difference $d(q_i, q_{i_n})$ for q_i and its orientation-guided pixels q_{i_n}, and the other is their respective previous approximate geodesic distance $D(p, q_{i_n})$ to the anchor pixel p as shown in Fig. 5(b). Therefore, the selected smallest distance is converted into a larger support weighting with Eq. (4). Fortunately, the approximate geodesic distance for the current support pixel to the anchor pixel can be obtained by accumulating the approximate geodesic distances for the pixels in the orientation-guided subset. As shown in Fig. 5(b), the pixels q_{i_n} in the orientation-guided subset are always located closer than pixel q_i to the anchor pixel in terms of spatial distance, and these closer pixels have the priority for the calculations of the approximate geodesic distances to the anchor pixel. Therefore, this OGGW strategy can reduce the computational cost for geodesic distance calculation, and enables us to compute the weighting on a patch efficiently. The algorithm steps for our



(a)



(b)

Figure 5: (a) Illustration for patch $P(k)$ from the reference image. Pixel q_i can be one of the eight orientations: left, right, top, bottom, top-left, top-right, bottom-left, and bottom-right of the anchor pixel p. (b) The candidate approximate shortest paths from a support pixel q_i to the anchor pixel p are obtained from adding the differences between pixel q_i and pixels q_{i_1}, q_{i_2}, and q_{i_3} to the approximate shortest paths of q_{i_1}, q_{i_2}, and q_{i_3} to p (represented by dash, solid, and dotted lines) respectively, which are the three pixels in the orientation-guided subset $O_{bottom-right}$ of pixel q_i to p.

Algorithm 1 Steps for our OGGW strategy.

Input: The intensity values or color components of all the pixels on the reference image I_R; The patches $P(k)$ $(k = 1, 2, 3, \cdots, K)$ obtained from I_R

Output: The weightings for each pixel on I_R are obtained with our OGGW strategy

repeat
  for each patch $P(k)$ **do**
    for each anchor pixel p on $P(k)$ **do**
      The other pixels q_i on $P(k)$ are the support pixels to p;
      Sort q_i by the spatial distance to p, the nearest comes first;
      for each sorted support pixel q_i around the anchor pixel p **do**
        Determine the guided orientation from q_i to p;
        Select the orientation-guided subset O_{pq_i};
        Obtain $D(p, q_i) = \min_{q_{i_n} \in O_{pq_i}} \big(d(q_i, q_{i_n}) + D(p, q_{i_n})\big)$;
        and $w_{pq_i} = \exp\big(-\frac{D(p,q_i)}{\gamma}\big)$
      end for
    end for
  end for
until the weightings for all the pixels on I_R have been calculated.

OGGW strategy is summarized in Algorithm 1.

5. PatchMatch Filter with Curved Surface Fitting (PMF-CS)

Through the process of cost volume filtering, the initial cost volume is updated by the multiple LPA models, and the cost updates are carried out through weighted averaging with the OGGW strategy. Then a filtered cost volume is obtained. After that, a coarse disparity map is extracted with a WTA scheme from the filtered cost volume, and the occluded pixels are detected using left-right cross checking. Finally, the initial disparities

between corresponding pixels on the stereo images with pixel accuracy can be obtained.

To model the disparity about a scene, a PatchMatch filter (PMF) was developed to approximate constant or slanted planes on patches using PMF-C and PMF-S methods [19]. However, most surfaces of objects in a scene are not flat or slanted planes, and the estimated constant or slanted planes may not represent the true disparities for objects surfaces. In order to achieve a more accurate disparity map, we propose a curved surface fitting method named PMF-CS, which is used to fit the initial disparity map based on the PatchMatch filter.

Our proposed PMF-CS method involves a two-step random sample consensus (RANSAC) algorithm [9], consisting of a possible fitting model construction step and an outliers removal step. Firstly, the initial disparity map is overlaid with patches obtained from the reference image, and a possible curved surface fitting model is built for pixels lying in the same patch. For each patch $P(k)$, we use a quadratic function as the curved surface model to fit the disparities of pixels in this patch

$$d_q(x_q, y_q) = a_k x_q^2 + b_k y_q^2 + c_k x_q y_q + d_k x_q + e_k y_q + f_k \tag{7}$$

Here q represents the pixel in patch $P(k)$ with coordinate (x_q, y_q); a_k, b_k, c_k, d_k, e_k, and f_k are the parameters of the curved surface computed using the least square method. Secondly, in order to identify outliers, we calculate the residuals between the initial disparities after the WTA step and the estimated disparities with curved surface fitting. If the residual of one disparity is larger than 95% of all other disparity residuals, this disparity is regarded as an outlier and will be removed in the fitting process. After outliers are removed, the procedure of obtaining the curved surface models with only inliers in the patch is carried out. Therefore, a more reliable curved surface model for each patch is constructed and the disparity map with sub-pixel accuracy is obtained.

This method obtains surfaces for scenes with higher accuracy than PMF-C and PMF-S as shown in Fig. 6. The patch boundaries obtained from the reference image "Bowling" are overlaid on its disparity map. Then the disparity values of pixels on patch $P(k)$ are fitted by the constant, slanted plane, and curved surface respectively. The fitted results obtained with the curved surface method on the patch are more accurate compared with those

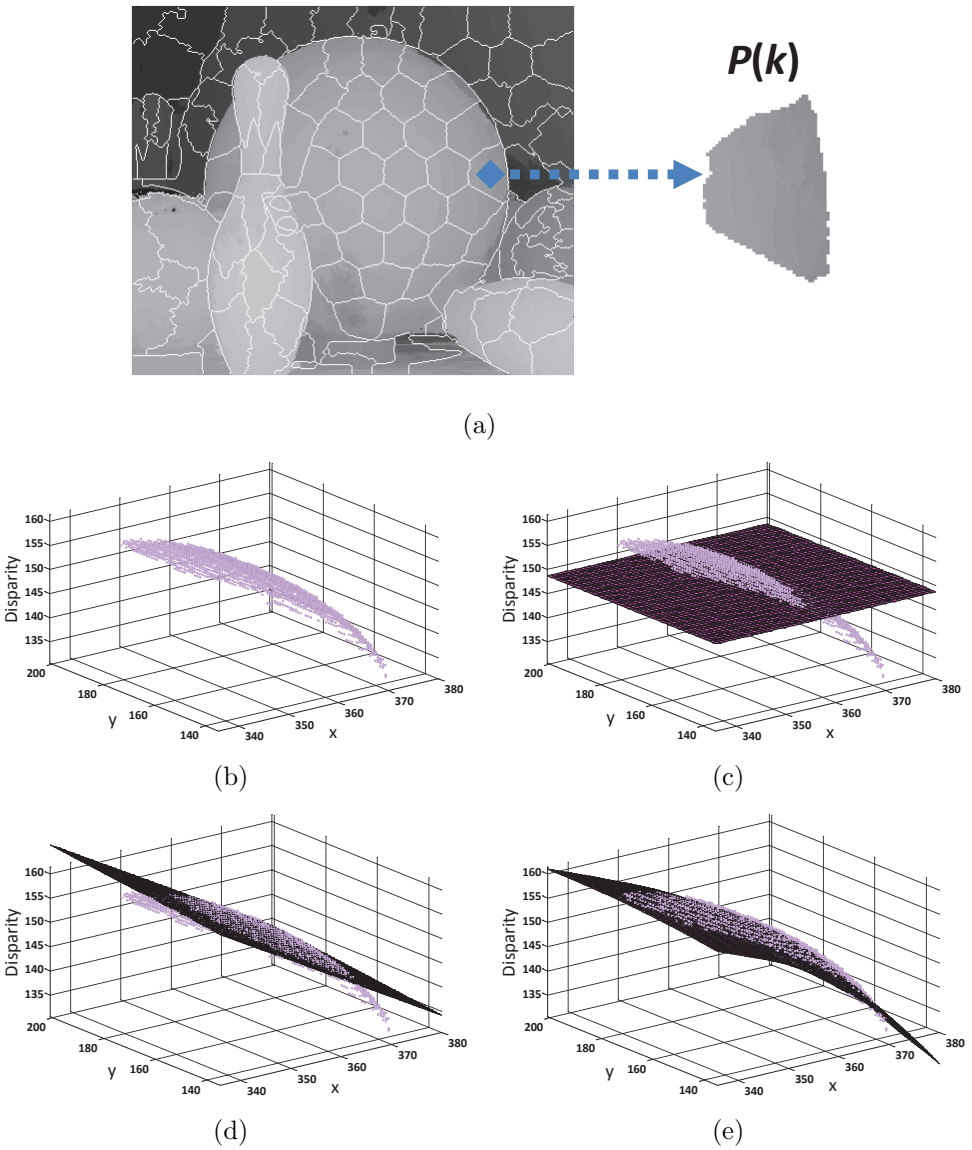


Figure 6: (a) Patches obtained from the dataset "Bowling" and overlaid on the disparity map and one example patch $P(k)$. (b) Disparity values of pixels in $P(k)$. (c)-(e) Using PMF-C, PMF-S, and PMF-CS methods to fit disparities in (b) respectively.

obtained with the constant or slanted plane fitting methods.

6. Experimental Results and Discussion

The performances of our OGGW and PMF-CS strategies proposed in this paper are evaluated comparing with current state-of-the-art algorithms on the Middlebury website [29].

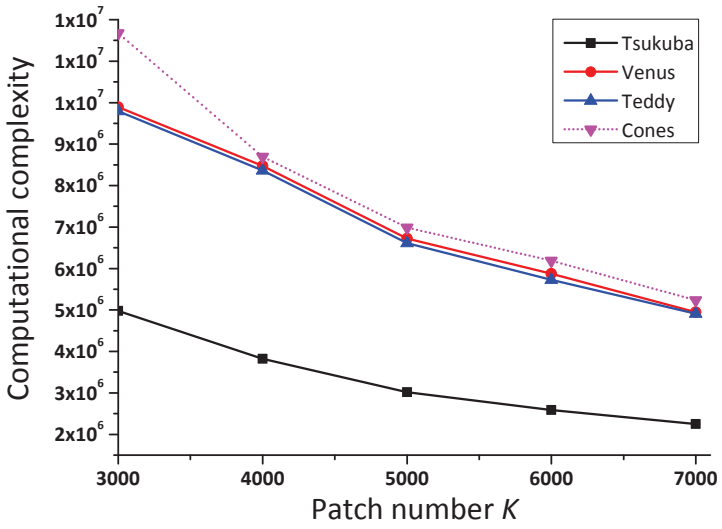
6.1. Parameter Selection

We use the same parameters on all image pairs. The parameter K is the number of patches being used in the preprocessing step, which influences the performance of the OGGW strategy in terms of the accuracy and the computational complexity. In this paper, the computational complexity measured by the number of arithmetic operations for OGGW is mainly related to the calculations of the approximate geodesic distance given in Eq. (6). Assuming the number of patches on the reference image is K, patch $P(k)$ $(k = 1, 2, 3, \cdots, K)$ includes all the pixels with label k, and the computation complexity of OGGW for all the pixels is $\sum_{k=1}^{K} |P(k)|^2$, where $|P(k)|$ is the size of patch $P(k)$. The results of our algorithm on four standard Middlebury datasets [29] with different patch numbers are shown in Fig. 7. It is observed that the computational complexity decreases with the increase of the number of patches as shown in Fig. 7(a). However,

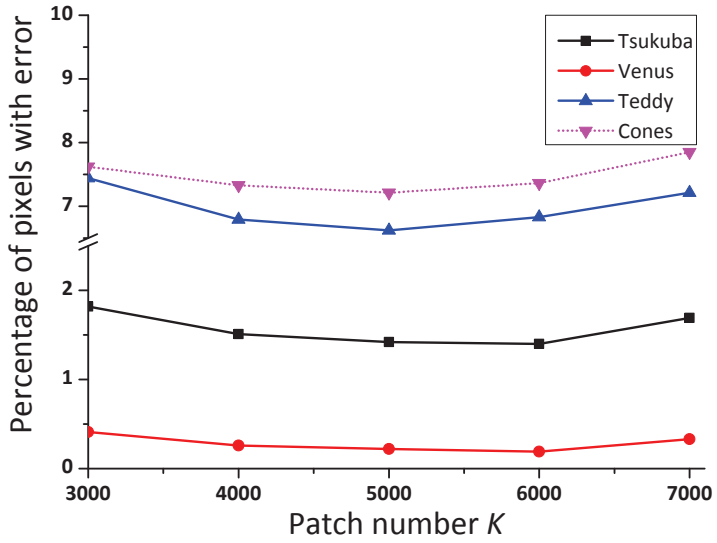
smaller patch sizes will reduce the reliable computation of weighted averaging on the updated costs, and will result in a higher percentage of error pixels on the disparity map as shown in Fig. 7(b). The lowest percentage of pixels with error on the disparity maps for datasets "Teddy" and "Cones" is achieved with 5000 patches, while it is achieved with 6000 patches on the datasets "Tsukuba" and "Venus". Another parameter γ for calculating the weightings in our OGGW strategy also has influence on the process of weighted averaging on cost updats. Hence, a two-parameter test between K and γ is conducted to check the performance of our algorithm as shown in Fig. 8. Based on the results, a trade-off between the two parameters for better performances is observed, and the parameters are set at $K = 6000$ and $\gamma = 10$.

6.2. Results Evaluation

Firstly, the comparisons between our OGGW strategy and that in GEO [14] on obtaining approximate geodesic distance are conducted using square windows as the support regions as used in [14]. We also conducted tests using patches as support regions for our OGGW strategy. We calculate the support weighting for the cost aggregation step using the GEO strategy with square windows, the OGGW strategy with square



(a)



(b)

Figure 7: (a) The computational complexity with respect to the number of patches on the "Tsukuba", "Venus", "Teddy", and "Cones" datasets. (b) The percentage of error pixels with respect to the quantity of patches on the "Tsukuba", "Venus", "Teddy", and "Cones" datasets (error $>$ 1 pixel).

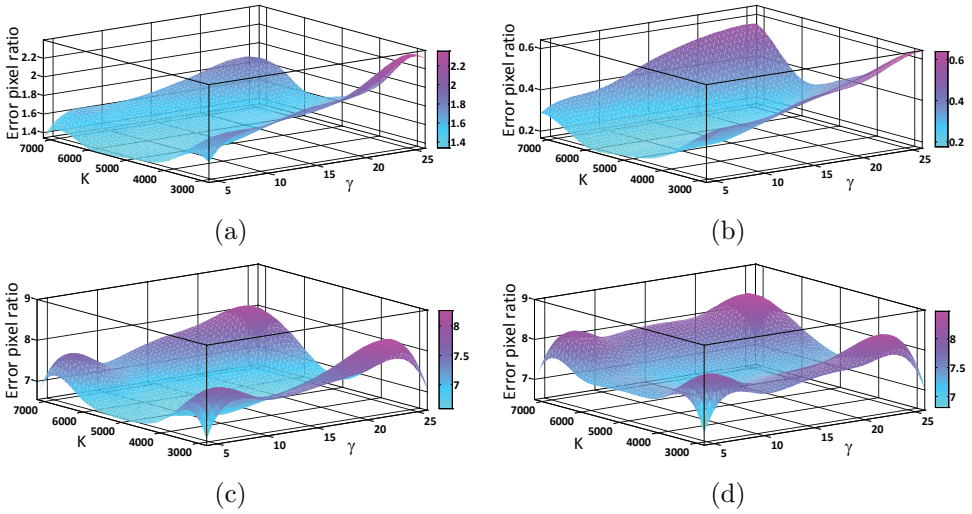
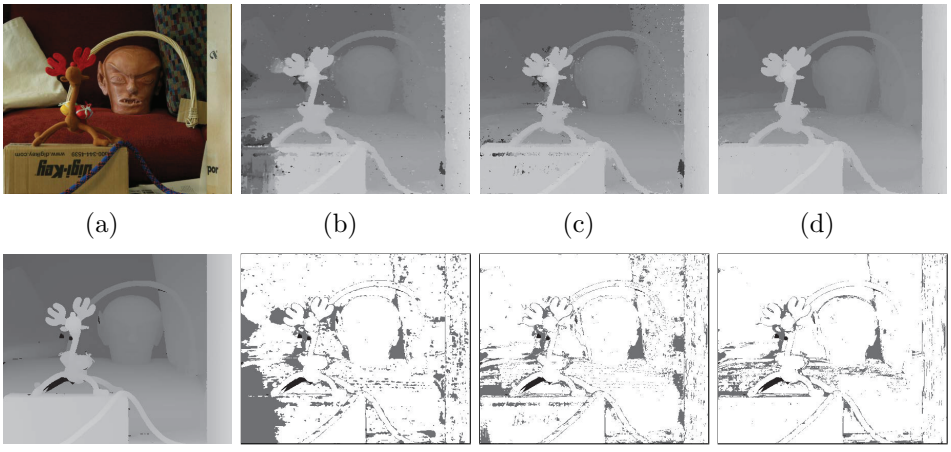


Figure 8: The performance of our method with respect to K and γ on the "Tsukuba" dataset (a), on the "Venus" dataset (b), on the "Teddy" dataset (c), and on the "Cones" dataset (d).

windows, and the OGGW strategy with patches respectively. Other steps for local stereo matching are the same as those in the GEO strategy. The computational complexity of the most efficient weighted averaging method GF [27] for local stereo matching is $O(1)$ for each pixel, which is independent of the size of the support region. However, the computational complexities of the GEO and our OGGW strategies depend on the size of their support regions. The GEO strategy is performed in $O(|W|)$ time, where $|W|$ is the number of pixels in the square window. Our OGGW strategy is performed in $O(|P(k)|)$ time, where $|P(k)|$ is the number of pixels in the patch. The experiments on running time which can give a

good indication of the computational complexity are carried out on the Middlebury dataset "Reindeer" [29] with a size of 447×370 pixels. The dataset is rectified and radial distortion has been removed so that the image pair has purely horizontal disparities. Moreover, the comparisons on running time tested on a standard PC with a 3.0 GHz CPU and 4 GB RAM using MATLAB R2012b are shown in Table 1. The running time of our OGGW strategy with a square support window is longer than that of GEO, because our OGGW strategy takes time for determining the guided orientation. However, our OGGW strategy using patches as the fixed support regions runs much faster than the GEO strategy with moving support regions. This indicates that our OGGW strategy using patches is more efficient for the calculation of support weighting. As shown in Fig. 9, the reliability of the disparity map is improved using our OGGW strategy with patches for support weighting, and our OGGW strategy outperforms that of the GEO strategy with a higher running speed and a lower percentage of pixel errors. More comparison experiments on the Middlebury datasets [29] are given in Fig. 15.

Secondly, we compared the performance of the proposed weighting strategy using OGGW with those of the state-of-the-art adaptive support



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

(e) (f) Error: 18.13% (g) Error: 11.19% (h) Error: 8.49%

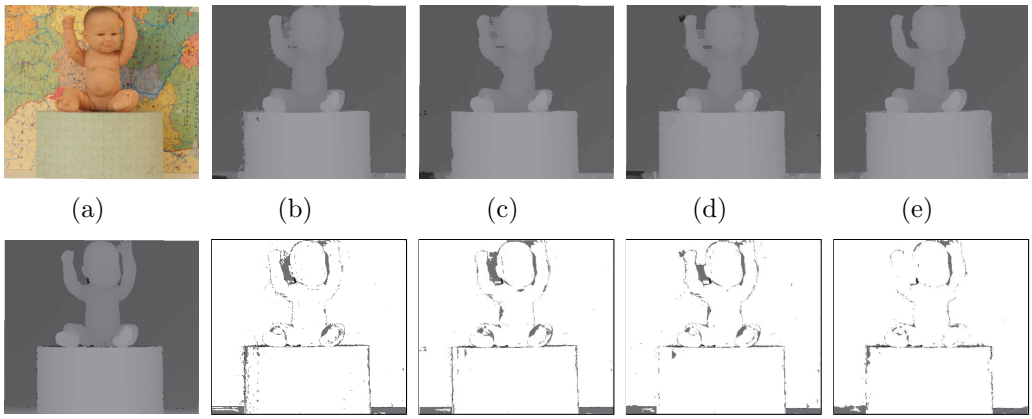
Figure 9: (a) The reference image of the "Reindeer" dataset. (b)-(d) Result from GEO [14], result from our OGGW with square support windows, and result from our OGGW with patches. (e) Ground truth of (a). (f)-(h) The corresponding error maps for results of (b)-(d) (error > 1 pixel). In the error maps, black pixels are bad pixels in occluded regions and grey pixels are error pixels in non-occluded regions.

Table 1: Running time of support weighting calculation on the dataset "Reindeer".

Methods	Support region	Running time (s)
GEO	square window	313.95
OGGW	square window	437.79
OGGW	patches	85.36

weighting strategies, including GEO [14], GF [27], and CLMF [18]. One example image is the Middlebury dataset "Baby" [29] with the size of 413×370 pixels as shown in Fig. 10. For these algorithms, the parameter settings are the same as those in the original references. The results indicate that our OGGW strategy based on patches achieves the best performance, especially in providing better disparity estimation and preserving boundaries. To better understand why our proposed OGGW strategy is superior to the conventional adaptive support weighting strategies on weighted averaging for filtering, an example with a close-up part of the dataset "Teddy" is shown in Fig. 11. The OGGW strategy accumulates geodesic distances by searching for approximate shortest paths along guided orientations, which makes an important connection for pixels with similar intensity or color and proximate spatial distance. Furthermore, we introduce patches as support regions for the OGGW strategy to avoid regions crossing depth discontinuities. Thus more reliable weightings from support pixels in the patch to the anchor pixel are obtained. The OGGW is the key strategy to filter the cost volume with boundary preservation.

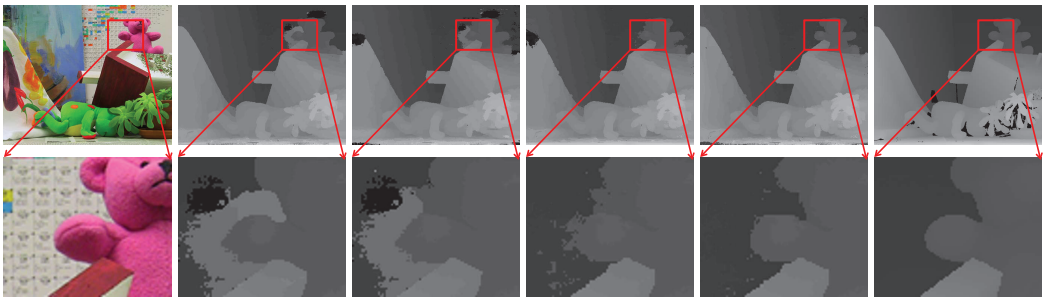
Thirdly, we compare the performance of the proposed PMF-CS method with those of PMF-C and PMF-S [19] methods. These methods fit the



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

(f) (g) Error: 4.93% (h) Error: 4.84% (i) Error: 4.21% (j) Error: 3.59%

Figure 10: (a) The reference image of the "Baby" dataset. (b)-(e) Result from GEO [14], result from GF [27], result from CLMF [18], and result from our OGGW strategy. (f) Ground truth of (a). (g)-(j) The corresponding error maps for results of (b)-(e) (error > 1 pixel). In the error maps, black pixels are bad pixels in occluded regions and grey pixels are error pixels in non-occluded regions.

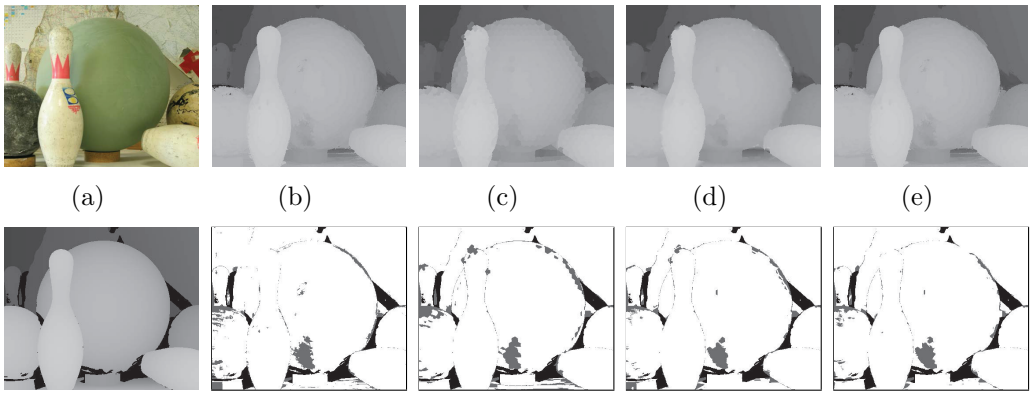


(a) (b) (c) (d) (e) (f)

Figure 11: Results of weighted averaging for filtering on the "Teddy" dataset. (a) From top to bottom: Reference image of the "Teddy" and close-up of a part of the "Teddy" dataset. (b)-(e) The corresponding disparity maps and their close-up parts of (a) obtained using GEO [14], GI [10], cross-based ASW [18], and our OGGW strategy with the WTA scheme. (f) The ground truth of (a).

disparities in the patch with the curved surface model, the frontal plane model, and the slanted plane model respectively. One example of the Middlebury dataset "Bowling" (with 443 × 370 pixels) [29] with just occlusion detection and filling and three kinds of PatchMatch filters are shown in Fig. 12. The PMF-CS method performs better than PMF-C and PMF-S in reconstructing surfaces of objects with different depths. To understand why PMF-CS performs better than PMF-C and PMF-S on disparity estimation, an example with a close-up part of the dataset "Cones" is shown in Fig. 13. Our PMF-CS performs best on the tops and edges of cones among the three PatchMatch-based methods. The main advantage of our PMF-CS method over PMF-C and PMF-S [19] is that the curved surface fitting model rather than the constant or the slanted plane is employed on the patch. The plane models are not always suitable in representing the surfaces of objects, while curved surface models can well represent multiple types of surfaces including frontal, slanted, and curved surfaces. However, the computational complexity of curved surface fitting increases as the order of the fitting functions becomes higher.

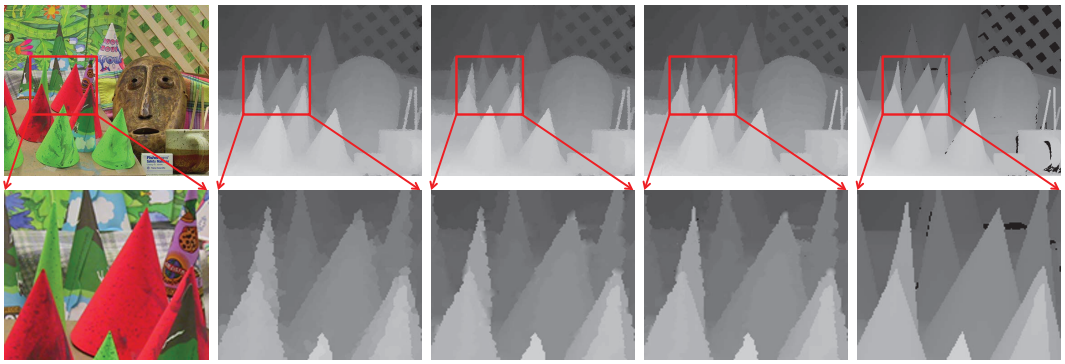
Finally, we run our local stereo matching algorithm on four standard Middlebury datasets [29] "Tsukuba" (384 × 288 pixels), "Venus" (434 × 383



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

(f) (g) Error: 9.57% (h) Error: 10.6% (i) Error: 9.24% (j) Error: 8.69%

Figure 12: (a) The reference image of the "Bowling" dataset. (b)-(e) Result from occlusion detection and filling [27], result from PMF-C [19], result from PMF-S [19], and result from our PMF-CS. (f) Ground truth of (a). (g)-(j) The corresponding error maps for results of (b)-(e) (error > 1 pixel). In the error maps, black pixels are bad pixels in occluded regions and grey pixels are error pixels in non-occluded regions.



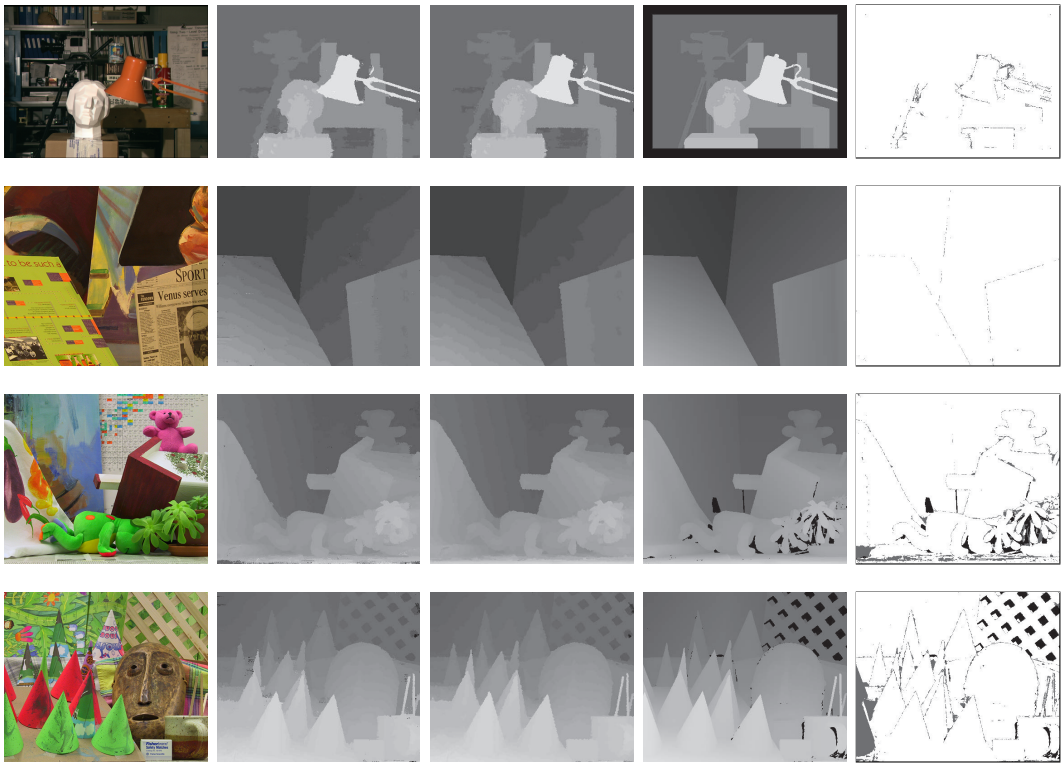
(a) (b) (c) (d) (e)

Figure 13: Results of PatchMatch-based filtering on the "Cones" dataset. (a) From top to bottom: Reference image of the "Cones" and close-up of a part of the "Cones" dataset. (b)-(d) The corresponding disparity maps and their close-up parts of (a) obtaind from PMF-C [19], PMF-S [19], and our PMF-CS algorithm. (e) The ground truth of (a).

pixels), "Teddy" (450 × 375 pixels), and "Cones" (450 × 375 pixels) to test the performance with the combination of our OGGW strategy in the process of cost volume filtering and our PMF-CS method in the process of disparity estimation. Results are shown in Fig. 14. The performances of our algorithm on the "Tsukuba", "Venus", "Teddy", and "Cones" datasets are evaluated on the Middlebury website (Version 2), and ranked based on the percentage of error pixels among all pixels (all), non-occluded pixels (vis), and visible pixels near depth discontinuous (disc). Table 2 shows the results of our algorithm and those of other local methods such as PMF [19], PatchMatch [3], CLMF [18], GF [27], and GEO [14]. Our algorithm ranks on the top of the list. The comparison experiments are also carried out on other datasets [29] as shown in Fig. 15, and the results demonstrate the superiority of our method.

Table 2: Comparison results on datasets on the Middlebury stereo evaluation website (Version 2).

Methods	Total rank	Avg rank	Avg error	Tsukuba			Venus			Teddy			Cones		
				vis	all	disc	vis	all	disc	vis	all	disc	vis	all	disc
OurMethod	**13**	**29.2**	4.51%	**1.14**	**1.42**	**4.63**	**0.14**	**0.22**	1.91	5.42	6.62	14.5	2.81	7.21	8.10
PMF [19]	32	44.5	**4.06%**	1.74	2.04	8.07	0.33	0.49	4.16	**2.52**	**5.87**	**8.30**	**2.13**	**6.80**	**6.32**
PatchMatch [3]	39	47.7	4.59%	2.09	2.33	9.31	0.21	0.39	2.62	2.99	8.16	9.62	2.47	7.80	7.11
CLMF [18]	46	51.5	5.13%	2.46	2.78	6.26	0.27	0.38	2.15	5.50	10.6	14.2	2.34	7.82	6.80
GF [27]	51	56.2	5.55%	1.51	1.85	7.61	0.20	0.39	2.42	6.16	11.8	16.0	2.71	8.24	7.66
GEO [14]	61	61.0	5.80%	1.45	1.83	7.71	0.14	0.26	**1.90**	6.88	13.2	16.1	2.94	8.89	8.32



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

Figure 14: Results on four standard Middlebury stereo image datasets. (a) Left images of the datasets. (b) Disparity maps after using our OGGW strategy and the WTA scheme. (c) Disparity maps after using our OGGW strategy and our PMF-CS method. (d) The ground truth of each dataset. (e) The corresponding error maps for results of (c) (error $>$ 1 pixel).

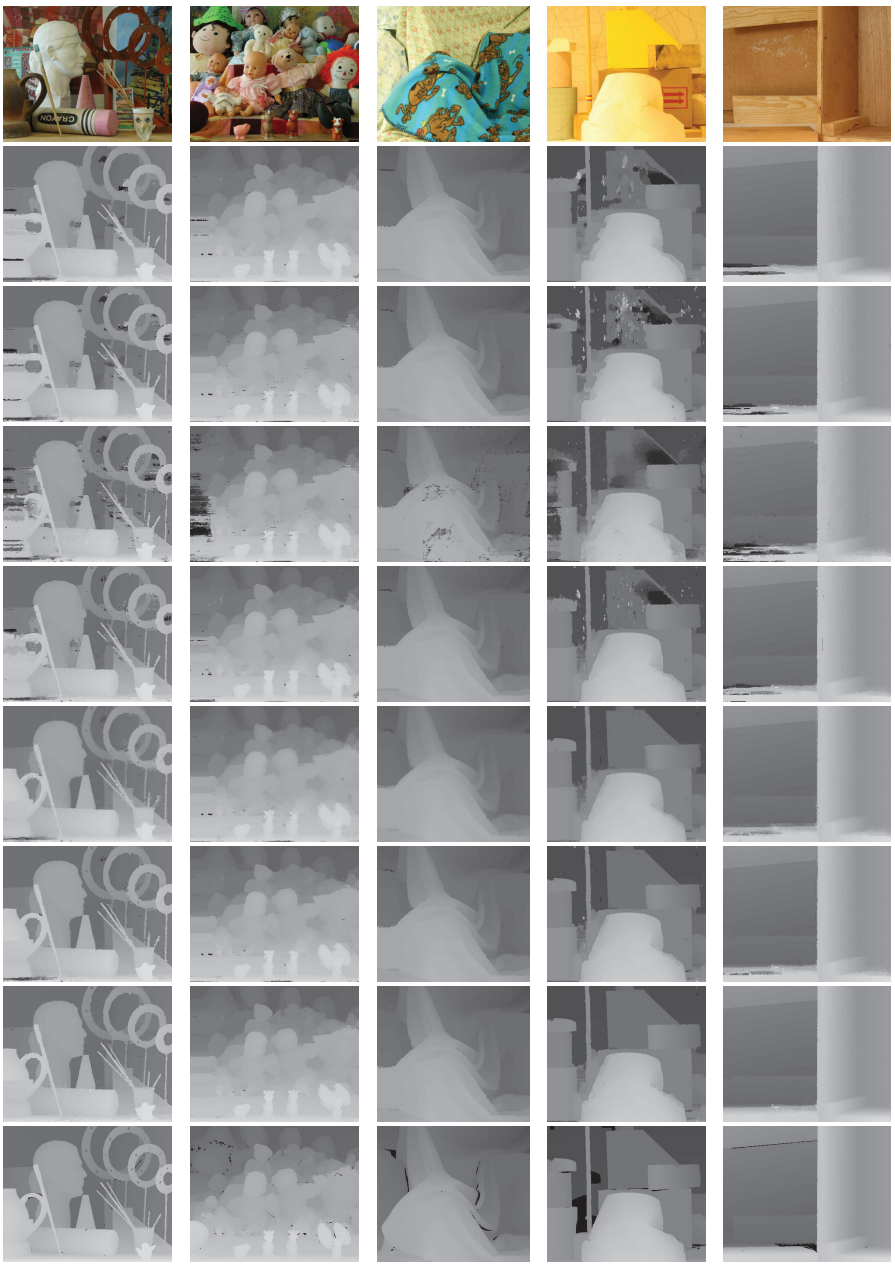


Figure 15: First row: Left images of the datasets with the size of 463×370, 463×370, 417×370, 433×370, 457×370 pixels from left to right. Second row: Results from GF [27]. Third row: Results from CLMF [18]. Fourth row: Results from GEO [14]. Fifth row: Results from our OGGW strategy with square support windows. Sixth row: Results from PMF [19]. Seventh row: Results from PatchMatch [3]. Eighth row: Results obtained from our method. Last row: The ground truths of the first row.

7. Conclusion

This paper presents a novel local stereo matching method with weighted averaging on the updated costs using our OGGW strategy and disparity estimation with PMF-CS. There are two main contributions in this paper. The proposed OGGW strategy provides a reasonable support weighting by searching for an approximate shortest path along a guided orientation from a support pixel to the anchor pixel in the same patch, which gives the desired property of preserving boundary discontinuity. We also propose a PMF-CS method using curved surface fitting along with outliers removal to estimate the disparity map with sub-pixel accuracy. A number of experiments are carried out to verify the effectiveness of the proposed method and show its unique property over the current state-of-the-art methods. The comparisons with the state-of-the-art methods also indicate that our OGGW and PMF-CS strategies are competitive on current popular patch-based applications. Our further work will focus on reducing the computational cost of weight calculation and improving on regression models based on patch.

Acknowledgments

Jiang is supported by the China Scholarship Council. We thank Dadong Wang and Xixian Xiong of CSIRO for their comments on this paper. We also thank the anonymous reviewers for their helpful and constructive comments.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Susstrunk, SLIC superpixels compared to state-of-the-art superpixel methods, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (11) (2012) 2274–2282.

[2] C. Barnes, E. Shechtman, D. B. Goldman, A. Finkelstein, The generalized PatchMatch correspondence algorithm, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 29–43.

[3] M. Bleyer, C. Rhemann, C. Rother, PatchMatch stereo – stereo matching with slanted support windows, in: The British Machine Vision Conference, 2011, pp. 14.1–14.11.

[4] M. Bleyer, C. Rother, P. Kohli, Surface stereo with soft segmentation, in:

IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 1570–1577.

[5] G. Borgefors, Distance transformations in digital images, Computer vision, graphics, and image processing 34 (3) (1986) 344–371.

[6] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (11) (2001) 1222–1239.

[7] F. Cheng, H. Zhang, M. Sun, D. Yuan, Cross-trees, edge and superpixel priors-based cost aggregation for stereo matching, Pattern Recognition (2015) 7–42.

[8] C. Cigla, A. A. Alatan, Efficient edge-preserving stereo matching, in: IEEE International Conference on Computer Vision Workshops, 2011, pp. 696–699.

[9] M. A. Fischler, R. C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Communications of the ACM 24 (6) (1981) 381–395.

[10] K. He, J. Sun, X. Tang, Guided image filtering, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 1–14.

[11] C. Hong, J. Yu, D. Tao, M. Wang, Image-based 3D human pose recovery by multi-view locality sensitive sparse retrieval, IEEE Transactions on Industrial Electronics 62 (6) (2015) 3742–3751.

[12] C. Hong, J. Yu, J. You, X. Chen, D. Tao, Multi-view ensemble manifold regularization for 3D object recognition, Information Sciences 320 (2015) 395–405.

[13] A. Hosni, M. Bleyer, M. Gelautz, Secrets of adaptive support weight techniques for local stereo matching, Computer Vision and Image Understanding 117 (6) (2013) 620–632.

[14] A. Hosni, M. Bleyer, M. Gelautz, C. Rhemann, Local stereo matching using geodesic support weights, in: IEEE Conference on Image Processing, 2009, pp. 2093–2096.

[15] T. Kanade, M. Okutomi, A stereo matching algorithm with an adaptive window: Theory and experiment, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (9) (1994) 920–932.

[16] V. Kolmogorov, R. Zabih, Computing visual correspondence with occlusions using graph cuts, in: Eighth IEEE International Conference on Computer Vision, vol. 2, 2005, pp. 399–407.

[17] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, K. Siddiqi, Turbopixels: Fast superpixels using geometric flows, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (12) (2009) 2290–2297.

[18] J. Lu, K. Shi, D. Min, L. Lin, M. N. Do, Cross-based local multipoint filtering, in: IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 430–437.

[19] J. Lu, H. Yang, D. Min, M. N. Do, Patch Match filter: Efficient edge-aware filtering meets randomized search for fast correspondence field estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1854–1861.

[20] G. Medioni, R. Nevatia, Segment-based stereo matching, Computer Vision, Graphics, and Image Processing 31 (1) (1985) 2–18.

[21] A. P. Moore, S. Prince, J. Warrell, U. Mohammed, G. Jones, Superpixel

lattices, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[22] A. P. Moore, S. J. Prince, J. Warrell, U. Mohammed, G. Jones, Scene shape priors for superpixel segmentation, in: 12th International Conference on Computer Vision, IEEE, 2009, pp. 771–778.

[23] D. Murray, J. J. Little, Using real-time stereo vision for mobile robot navigation, Autonomous Robots 8 (2) (2000) 161–171.

[24] M. Okutomi, Y. Katayama, S. Oka, A simple stereo algorithm to recover precise object boundaries and smooth surfaces, International Journal of Computer Vision 47 (1-3) (2002) 261–273.

[25] S. Paris, F. Durand, A fast approximation of the bilateral filter using a signal processing approach, International Journal of Computer Vision 81 (1) (2009) 24–52.

[26] F. Porikli, Constant time O(1) bilateral filtering, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[27] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, M. Gelautz, Fast cost-volume filtering for visual correspondence and beyond, in: IEEE

Conference on Computer Vision and Pattern Recognition, 2011, pp. 3017–3024.

[28] C. Richardt, D. Orr, I. Davies, A. Criminisi, N. A. Dodgson, Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 510–523.

[29] D. Scharstein, R. Szeliski, http://www.vision.middlebury.edu/stereo/, 2011.

[30] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, International Journal of Computer Vision 47 (1) (2002) 7–42.

[31] C. Sun, Fast stereo matching using rectangular subregioning and 3D maximum-surface techniques, International Journal of Computer Vision 47 (1-3) (2002) 99–117.

[32] C. Sun, Closed-form stereo image rectification, in: Proceedings of the 27th Conference on Image and Vision Computing New Zealand, ACM, 2012, pp. 132–137.

[33] J. Sun, N.-N. Zheng, H.-Y. Shum, Stereo matching using belief

propagation, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (7) (2003) 787–800.

[34] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, C. Rother, A comparative study of energy minimization methods for Markov random fields with smoothness-based priors, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (6) (2008) 1068–1080.

[35] X. Tan, C. Sun, T. D. Pham, Multipoint filtering with local polynomial approximation and range guidance, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2941–2948.

[36] X. Tan, C. Sun, X. Sirault, R. Furbank, T. D. Pham, Stereo matching using cost volume watershed and region merging, Signal Processing: Image Communication 29 (10) (2014) 1232–1244.

[37] C. Tomasi, R. Manduchi, Bilateral filtering for gray and color images, in: Sixth International Conference on Computer Vision, IEEE, 1998, pp. 839–846.

[38] Q. Yang, K.-H. Tan, N. Ahuja, Real-time O(1) bilateral filtering, in:

IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 557–564.

[39] Q. Yang, L. Wang, R. Yang, H. Stewénius, D. Nistér, Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (3) (2009) 492–504.

[40] Y. Yang, X. Wang, Q. Liu, M. Xu, L. Yu, A bundled-optimization model of multiview dense depth map synthesis for dynamic scene reconstruction, Information Sciences 320 (2014) 306–319.

[41] K.-J. Yoon, I. S. Kweon, Adaptive support-weight approach for correspondence search, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (4) (2006) 650–656.