Outlier detection and disparity refinement in stereo matching

Qicong Dong, Jieqing Feng*

State Key Lab of CAD&CG, Zhejiang University,
Hangzhou, 310058, China

Abstract

Disparity estimation in ill-posed regions, such as occlusions, repetitive patterns and textureless regions, is a challenging problem in stereo matching. The initial disparities obtained in these regions tend to be regarded as outliers that must be detected and addressed. In this paper, two outlier detection methods are proposed, i.e., the efficient approach and the accurate approach. The efficient approach detects outliers by exploring the disparity map for the left image only and reduces runtime and memory costs. First, the match fixed point jumps (MFPJ) algorithm is proposed as an initial solution to detect outliers. Then, a high-probability outlier detection algorithm is proposed to accomplish denser outlier detection with less noise. The accurate approach first classifies outliers as occlusions or mismatches. Then, 3D label assignment is performed for occlusion outliers and normal-based plane fitting is conducted for mismatch outliers to refine the disparities of the outliers and to achieve an accurate stereo matching result. Evaluations of the Middlebury datasets demonstrate that the proposed methods effectively improve the stereo matching performance.

Keywords: Outlier detection, Stereo matching, Match fixed point jumps, Normal-based plane fitting

1. INTRODUCTION

Stereo matching is a fundamental problem in computer vision that has received increasing attention in the literature in recent decades. Consider two images taken from two horizontal cameras as input. The goal of stereo matching is to identify the disparity d of one point at position (x, y) in the left image such that the corresponding point appears at position $(x - d, y)$ in the right image. Once the disparity d is obtained, the depth z of the pixel can be computed in the 3D scene as follows: $z = fB/d$, where f is the focal length of the camera and B is the baseline length (the distance between the centers of the two cameras). Scharstein and Szeliski [1] describe a taxonomy for stereo matching. According to the taxonomy, a stereo algorithm consists of four steps: matching cost computation, cost aggregation, disparity computation (optimization) and disparity refinement. One challenging problem in stereo matching is to find the corresponding points in ill-posed regions, e.g., occlusions, textureless regions, repetitive patterns and slanted regions. Many attempts have been made to accurately find the corresponding points. These studies mainly focus on the definition of the matching cost function [2, 3, 4] or the subsequent optimization [5, 6, 7] to enhance the performance of stereo matching. When investigating the above stereo matching algorithms, outlier detection and disparity refinement are nonnegligible components that can further improve the accuracy of stereo matching.

This paper studies outlier detection and the disparity refinement of outliers in stereo matching. Two approaches are proposed, i.e., the efficient approach and the accurate approach. The efficient approach detects outliers by exploring the disparity map for the left image only, thereby reducing the runtime and memory costs by approximately one-half. Compared with traditional efficient outlier detection methods, the proposed approach yields dense and accurate outlier detection results. The accurate approach first classifies outliers as occlusions or mismatches, and the classification guides the following disparity refinement step. Two refinement strategies are presented in the accurate approach, i.e., 3D-label assignment for occlusion outliers and normal-based plane fitting for mismatch outliers. As a result, the stereo matching performance is enhanced by the specific disparity refinement strategies.

The remainder of this paper is organized as follows. Closely related work is reviewed in Section 2. The matching cost initialization and disparity optimization are described in Section 3. The efficient and accurate approaches are described in detail in Section 4. The experimental results and discussion are presented in Section 5. Finally, conclusions are drawn and further research directions are discussed in Section 6.

2. RELATED WORK

A large volume of studies on stereo matching have been published; we review the closely related literature.

Kong and Tao [8] first utilize the sum of absolute differences (SAD) or the sum of squared differences (SSD) to compute the matching cost. Zabih and Woodfill [9] use the census transform, which considers not only the information of the current pixel itself but also that of its neighbors, to define the matching cost. Mei et al. [2] combine the absolute differences (AD) and census transform to define a robust matching cost, namely, the AD-census matching cost. The normalized cross-correlation (NCC) [10] is another robust matching cost that is invariant to linear variation in illumination. Žbontar and LeCun [11] compute the matching cost by training a convolutional neural network (CNN), and their data-driven similarity measurement outperforms most traditional hand-crafted metrics. Many recent algorithms [12, 13, 14] utilize this data-driven matching cost and achieve state-of-the-art stereo matching results. In this paper, we also adopt the matching cost trained by a CNN as the input.

In general, the initial matching cost is sensitive and noisy. Thus, the costs are usually aggregated in a support region. Yoon and Kweon [15] aggregate the matching costs via an adaptive weight. The weight is based on color and spatial differences between a neighboring pixel and the center pixel in a fixed window. Zhang et al. [16] propose cross-based cost aggregation in an adaptive support region with a constant weight. He et al. [17] propose a guided image filtering algorithm that is a constant-time and edge-aware filter that can produce accurate stereo matching results. These methods implicitly make the frontalparallel surface assumption that all the points belonging to the support region share the same disparity. However, this assumption is sometimes violated in practice. Bleyer et al. [18] use Patch-Match stereo matching with a slanted support window to overcome this problem. Lu et al. [19] propose a Patch-Match filter algorithm to accelerate the original Patch-Match algorithm. The approach adopts the super-pixel as the unit of computational cost and achieves a time complexity of approximately $O(1)$. Taniai et al. [12] infer per-pixel 3D plane labels on a pairwise Markov random field by using local expansion moves and achieve an accurate stereo matching result.

After the cost aggregation step, the initial disparity map can be computed using a winner-take-all (WTA) strategy or some other disparity optimization strategy. The initial disparity map contains many outliers, and extensive research has been conducted on outlier detection in stereo matching. Outliers in stereo matching are mainly caused by occlusions or mismatches. Five outlier (mainly occlusion) detection methods are summarized in a previous survey [20]: bi-modality (BMD) [21, 22], match goodness jumps (MGJ) [23], ordering (ORD) [24, 25] constraint, occlusion constraint (OCC) [26], and left-right checking (LRC) [27, 28, 29]. BMD, MGJ, ORD and OCC do not require computation of the disparity map for the right image, rendering these methods memory-and-time efficient, especially for large stereo images. However, the outlier detection results of this type of method are usually sparse and noisy and are not suitable for highly accurate stereo matching. In this paper, a high- probability outlier detection algorithm, in which the detection results are denser, less noisy and more accurate, is proposed to overcome this problem. LRC utilizes additional information of the disparity map from the right image. LRC can detect outliers densely, and the outliers can also be classified as occlusions and mismatches. However, the classification is not always accurate because LRC detects outliers only at the pixel level. Some outliers (especially occlusion outliers) tend to appear regionally. The proposed accurate outlier detection first classifies outliers reasonably and accurately and then processes occlusion outliers and mismatch outliers with different disparity refinement algorithms and achieves accurate stereo matching.

Outlier detection is akin to confidence measure detection in stereo vision. One point in the image tends to be an outlier when the confidence value of that point is low. Hu and Mordohai [30] evaluate several confidence measures in stereo vision. Some of the approaches in [30] compute the confidence value of a single point by comparing the minimum cost and the second minimum cost [31, 32, 33], whereas some approaches explore the shape of the cost curve [33]. Other approaches compute the confidence value by treating the value assigned to each potential disparity as a probability for the disparity [34, 35]. Yoon et al. [36] use the distinctiveness of points as a confidence measure. Hu and Mordohai [30] also propose a measure called left-right difference (LRD) that considers both the two smallest minima of the costs and the consistency of the minimum costs across two images to obtain a robust confidence measure.

All these methods detect outliers based on 'abnormal phenomena'. These phenomena are distinguished by the matching cost (e.g., MGJ) or the disparity (e.g., BMD, ORD and the visibility constraint), while LRC identifies outliers by detecting disparities in the left image that are in conflict with the disparities of the corresponding points in the right image. In this paper, the aim is to design a method to detect outliers more efficiently and accurately and to facilitate the subsequent disparity refinement step is designed to produce a more accurate stereo matching result.

3. MATCHING COST COMPUTATION AND DISPARITY OPTIMIZATION

Fig. 1 illustrates the whole algorithm of the proposed method. The details of each part of the proposed method will be described in the following sections.

3.1. Matching cost computation

Let I^L and I^R be the left image and the right image, respectively. The purpose of the proposed method is to estimate the disparities of the left image

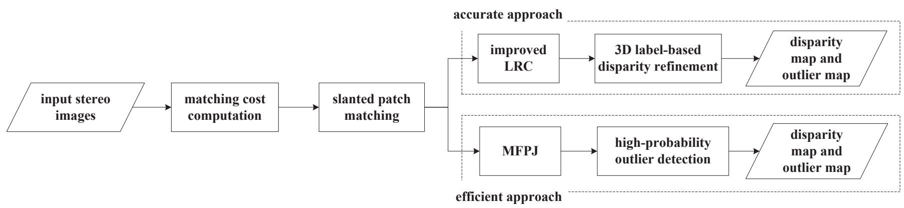


Figure 1: The flowchart of the proposed method

robustly. First, we initialize the matching cost. The position of a point in the image is denoted by a lowercase letter, such as p or q. A point $p(p_x, p_y)$ in the left image matches the point $q(p_x - d, p_y)$ in the right image with disparity d, and the point-wise matching cost is denoted by $C(p, d)$. We adopt the convolutional neural network (CNN) approach in [11] to compute the matching cost:

$$C(p, d) = C_{CNN}(R^L(x, y), R^R(x - d, y)) \tag{1}$$

where $C_{CNN}(\cdot, \cdot)$ is the predicted similarity measure obtained by the trained CNN. The cost is computed between the $11{*}11$ image patch R^L centered at pixel p of I^L and the image patch R^R centered at the corresponding pixel of I^R.

3.2. Slanted patch matching

The initial matching cost is not generally robust. For accurate stereo matching, we aggregate the matching costs in slanted support windows [18]. For each point $p(p_x, p_y)$, the goal is to identify a fitting plane such that the disparity of p can be computed as follows:

$$d_p = a_p \cdot p_x + b_p \cdot p_y + c_p \tag{2}$$

where a_p, b_p and c_p are three parameters of the plane. The tuple (a_p, b_p, c_p) is called a 3D label. We then utilize the local expansion moves proposed in [12] to optimize each pixel's 3D label in both the left and right images and obtain the disparity maps D^L and D^R, respectively.

4. OUTLIER DETECTION AND DISPARITY REFINEMENT APPROACHES

Two approaches are proposed in this section to address the trade-off between the computing time and accuracy. The efficient approach, which does not need to compute the disparity map for the right image, is tuned for runtime and memory consumption and reduces the time and memory expenses by approximately one-half. The accurate approach improves the initial LRC outlier detection map by distinguishing occlusion and mismatch outliers more reasonably. Then, two different disparity refinement approaches, i.e., 3D label assignment for occlusion outliers and normal-based plane fitting for mismatch outliers, are proposed for better stereo matching performance.

4.1. The efficient approach

In this section, we introduce an efficient outlier detection approach. First, match fixed point jumps (MFPJ), which utilizes the cost volume information of the left image to detect initial outliers, is proposed. Then, a high-probability outlier detection algorithm is presented to obtain a denser and less noisy outlier detection result.

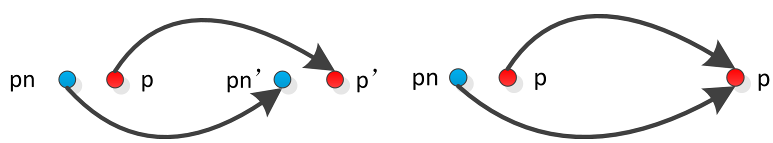
4.1.1. Match fixed point jumps

In this step, the outliers are detected by using the proposed MFPJ. In MFPJ, the confidence of a point p is defined as follows:

$$C_{MFPJ}(p) = \min_{p_n \in \mathcal{N}} \{C(p_n, d_n)\} - C_1(p, d) \tag{3}$$

where $C_1(p, d)$ is the minimum matching cost between point p and its corresponding pixel p', p_n is a neighboring point of p in a window $\mathcal{N}$ centered at p, and $C(p_n, d_n)$ is the matching cost between p_n and p'. The confidence defined in Equation 3 indicates that if the point's minimum cost $C_1(p, d)$ is relatively high, then a high probability exists that the point does not have a good matching point, indicating that the point is likely to be an outlier (the confidence is low). Note that this method differs from LRD in [30], which considers the consistency of the minimum costs across the two images. We utilize only the left cost volume information and do not search the whole disparity range (which is usually greater than 100 in current stereo pairs). Thus, the method is efficient.

Many confidence measures, such as MGJ [20], consider the relative magnitude of a point's minimum costs, for example by comparing the relative magnitude of the current point's minimum cost and the neighboring points' minimum costs. However, the proposed method differs from MGJ to some extent. Fig. 2 illustrates the difference between MFPJ and MGJ. MGJ compares the current point's minimum cost and the neighboring points' minimum costs, whereas MFPJ compares the current point's minimum cost and the costs of matching the neighboring points to a fixed point.



(a) (b)

Figure 2: An illustration of MGJ and MFPJ. p and p' denote corresponding points, as well as p_n and p'_n. (a) MGJ defines the confidence using the minimum costs among the point pair (p, p') and its neighboring pairs (for example, (p_n, p'_n)); (b) MFPJ defines the confidence using the minimal costs among the point pair (p, p') and the costs of the pairs composed of the neighboring point (p_n for instance) and the fixed point p', i.e., (p_n, p').

However, if significant photometric variation exists between the matched features, MGJ tends to detect a false signal [20]. Fig. 3 shows an example. In the circled region, strong photometric variation exists near the edges of the books on the bookshelf. The detection result of MGJ is noisy, while that of the proposed MFPJ method is reasonable.

The reason that MFPJ achieves a better result is because in regions with significant photometric variation, points with different colors may have different magnitudes of minimum costs, which may mislead outlier detection. Thus, MGJ tends to pick up wrong signals. When the MFPJ method determines whether one point is an outlier, the corresponding point in the right image is fixed, and

(a) (b) (c)

Figure 3: Comparison of the outlier detection results of MGJ and MFPJ. (a) Raw image; (b) the outlier detection result of MGJ, where the bright points denote outliers, and (c) the outlier detection result of MFPJ, which has substentially less noise than the MGJ result

only the costs of the neighboring points matched with this point are adopted for comparison. Thus, wrong signals are not picked up in this case.

Let M denote the outlier map detected by MFPJ. For each point, when the confidence is lower than a threshold, the point is regarded as an outlier, as follows:

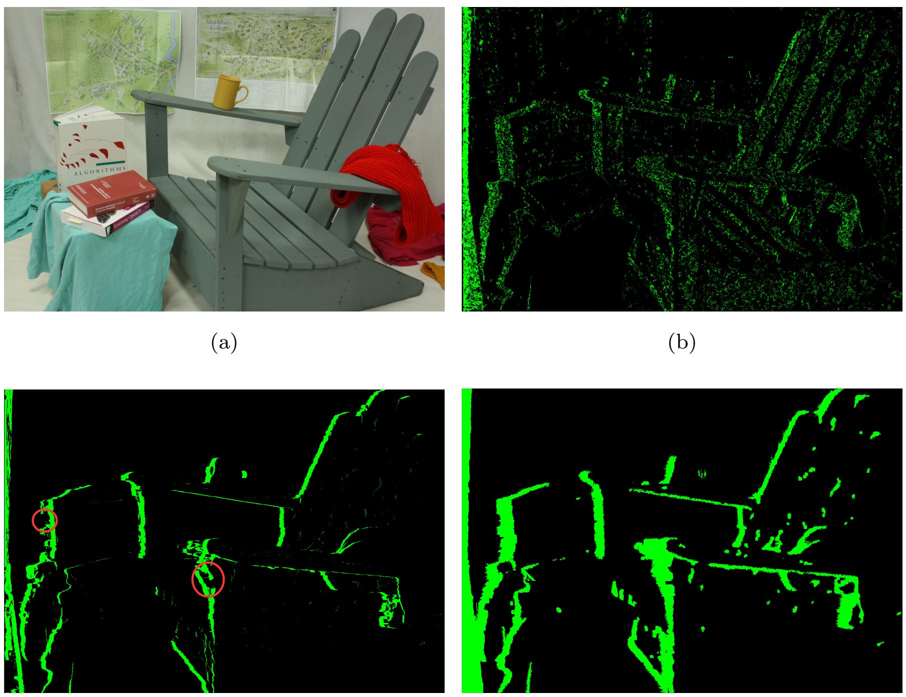
$$M(p) = \mathbb{1}\{C_{MFPJ}(p) < \eta\} \tag{4}$$

where $\mathbb{1}\{\cdot\}$ is the indicator function, and η is a threshold. The choice of all the parameters will be discussed in Section 5.4. In the following sections, we improve this outlier map and achieve more accurate outlier detection results.

4.1.2. High-probability outlier detection

In Section 4.1.1, we demonstrate that the MFPJ strategy has advantages over MGJ. However, the detection result is still relatively noisy and sparse, especially for some large-sized images, as shown in Fig. 4(b). Actually, this is a universal problem with outlier detection methods that explore the disparity map for the left image only. In this section, a high-probability outlier detection method is proposed to address this problem.

Since the detection result of MFPJ is still noisy and sparse, the defect should be remedied. Ordering constraint (ORD) [20] can be adopted to detect the ordering of points in both the left and the right images. The basic idea of this algorithm is that the order of the points in a scanline in the left image should be consistent with that of the corresponding points in the right image. Any point that violates this rule is regarded as an outlier. The ORD detects a point's order from the disparity map D^L computed in Section 3.2. The detection result is shown in Fig. 4(c). For some occlusion pixels (circled), the disparities still obey the ordering constraint and the ORD cannot detect them. However, the outlier map of the ORD has less noise than that of MFPJ; thus, this map can be adopted as a guidance map to improve the result of MFPJ.



(a) (b)

(c) (d)

Figure 4: Outlier detection results. (a) Raw image of Adirondack; (b) the outlier map detected by MFPJ is still noisy and sparse; (c) in the outlier map detected by the ORD, some occlusion outliers shown in the circled regions are missed; and (d) the high-probability outlier detection result. The outlier map is denser than the MFPJ and ORD maps

Let O denote the outlier map detected by ORD. For each outlier $p(p_x, p_y)$ in M, we compute the numbers of outliers in the local window centered at p in M

and in O, respectively, and define the probability of the point p being an outlier as:

$$P(p_x, p_y) = \frac{\alpha \cdot N_M(p_x, p_y) + (1-\alpha) \cdot N_O(p_x, p_y)}{N} \tag{5}$$

where $N_M(p_x, p_y)$ and $N_O(p_x, p_y)$ are the numbers of outliers in M and O centered at (p_x, p_y), N is the size of the local window, and α is a constant value. We define point p as a high-probability outlier when $P(p_x, p_y)$ is greater than a threshold μ. The parameters will be analized in Section 5.4

To achieve a dense outlier detection result, we fill the black points that are surrounded by high-probability outliers since these points also tend to be outliers. For each black point, we search the high-probability outliers along the horizontal and vertical directions in a window centered on the black point. We mark the black point as an outlier if and only if four high-probability outliers are found in the window. Fig. 4(d) illustrates the final outlier detection results by using the proposed efficient approach.

4.2. The accurate approach

In this section, an accurate approach is proposed to detect outliers and refine the outliers' disparities more accurately than with the efficient approach. In this approach, first, the outlier map detected by the LRC is adopted as the input. Then, the outliers are classified as occlusions or mismatches. Finally, a 3D label-based disparity refinement is proposed to achieve accurate stereo matching.

4.2.1. Improved outlier classification

The LRC checks the disparity consistency of corresponding points and is formulated as follows [29]:

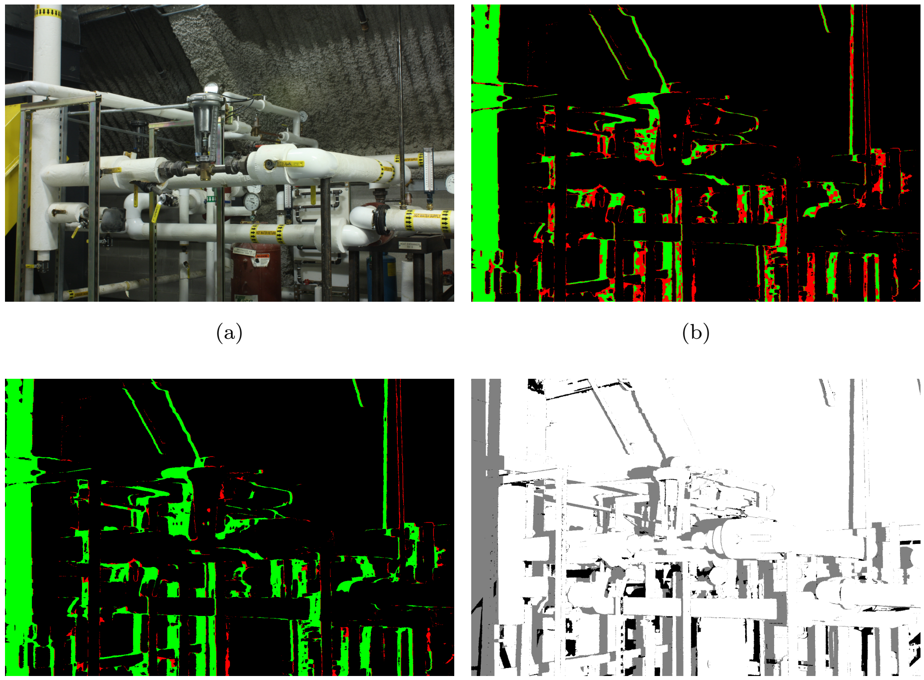
$$\left| D^L(\mathbf{p}) - D^R(\mathbf{p} - D^L(\mathbf{p})) \right| \leq 1 \tag{6}$$

A point that violates Equation 6 is regarded as an outlier. The detected outliers are further classified as occlusions and mismatches. If one point violates Equation 6 for the current disparity but the point has any other candidate disparities that are satisfied with Equation 6, then the point is a mismatch; otherwise, the point is an occlusion [11].

However, the classification of mismatch and occlusion outliers via LRC is not always accurate. Fig. 5(b) illustrates an outlier map detected by LRC. In some occlusion regions, the outliers are classified as mismatches, which will have a negative impact on the subsequent disparity refinement. Through observation, we find that occlusion points tend to appear together rather than in isolation, as shown in Fig. 5(d). Thus, for each mismatch outlier p, we will reclassify it as an occlusion outlier if the ratio of the number of occlusion points in a window centered at p is greater than a threshold κ, which will be discussed and determined in the Section5.4. The improved outlier map obtained via the improved classification is illustrated in Fig. 5(c), which shows better classification than that of LRC.

4.2.2. 3D label assignment for occlusion outliers

In this subsection, each occlusion point is assigned a reliable 3D label to enhance the stereo matching performance. For each occlusion point p, its closest reliable points (not outliers) are searched along the horizontal and vertical directions. One of these reliable points is selected to assign p a reliable 3D label. However, the search should be in the neighborhood of p since points that are far away tend to lie in different planes. Thus, if a discontinuous boundary is met, then the search should stop. The discontinuous boundaries can be determined via both texture edges and disparity edges. First, a texture edge map and a disparity edge map are detected by the Canny operator and Sobel operator respectively, as illustrated in Fig. 6(b) and Fig. 6(c). Clearly, the detected texture edges are redundant for discontinuous boundaries. We alleviate this problem with the help of detected disparity edges. For each texture edge point, we search the corresponding neighboring points in the disparity edge map. If the ratio of the number of neighboring disparity edge points is higher than a threshold ρ, which will be also discussed and determined in Section 5.4, then the texture edge point will be regarded as a real discontinuous boundary point.

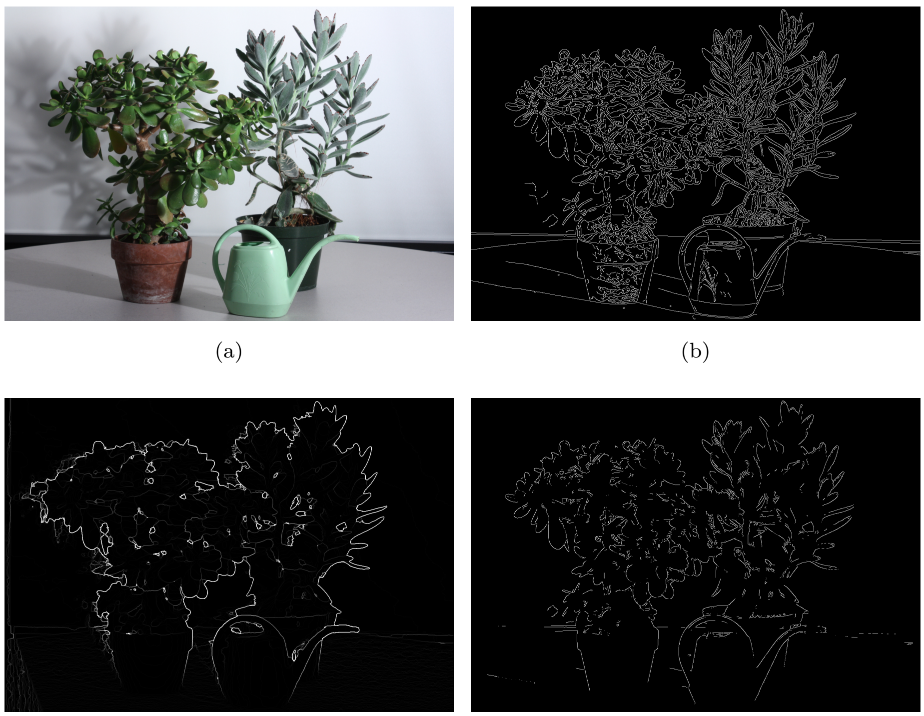


(a) (b) (c) (d)

Figure 5: Outlier map comparison between the LRC and the improved LRC. (a) Raw image of Pipes; (b) the outlier map classified by the LRC, where the green and red points denote occlusion and mismatch outliers, respectively; (c) the outlier map classified by the improved LRC; and (d) the ground truth of the non-occlusion mask, where the gray points denote occlusion points. The improved LRC classifies occlusion outliers more properly. (Best viewed in color)

Fig. 6(d) illustrates the result of discontinuous boundaries.

Considering the detected discontinuous boundaries, the search for reliable 3D labels along the horizontal and vertical directions stops when a reliable 3D label is identified or a discontinuous boundary is met. Then, the disparity of point p is obtained by assigning p a reliable 3D label. Here, the point with the lowest disparity value is adopted as the point p's filled-in disparity since occlusion points tend to occur in the background.



(a) (b)

(c) (d)

Figure 6: An example of discontinuous boundary detection. (a) Raw image of Plants; (b) the texture edge map detected by the Canny operator; (c) the disparity edge map detected by the Sobel operator; and (d) the discontinuous boundaries

4.2.3. Normal-based plane fitting for mismatch outliers

The disparities of mismatch points also need to be refined. Here, a normal-based plane fitting algorithm, which is based on the observation that mismatch points often appear at the slanted planes, is proposed. In some disparity maps, we find that the error pixels are usually at the slanted planes. Thus, refining mismatch outliers with neighboring reliable points using a plane fitting strategy is reasonable. Since we not only have the disparities of the reliable points but also their 3D labels, the information of these 3D labels should be utilized in the plane fitting strategy. For each mismatch point p, we search its closest reliable points along the horizontal and vertical directions, as in Section 4.2.2. Then, $k(k \leqslant 4)$ reliable points and their corresponding 3D labels are obtained. The three components of the reliable point's normal can be computed by $n_z = 1/\sqrt{a_q^2 + b_q^2 + 1}$, $n_x = -a_q \cdot n_z$, $n_y = -b_q \cdot n_z$, where q denotes the reliable point and (a_q, b_q, c_q) is its 3D label. Then, p's normal is defined as the average of the reliable points' normals. A least-squares plane fitting strategy based on p's normal is then implemented for the new 3D label of mismatch point p:

$$min \sum_{i=1}^{n} \frac{[a_p \cdot q_{x_i} + b_p \cdot q_{y_i} - d_{q_i} + c_p]^2}{a_p^2 + b_p^2 + 1} \tag{7}$$

$$s.t.\ \mathbf{n_p} = \sum_{i=1}^{k} \frac{\mathbf{n_{q_i}}}{k} \tag{8}$$

where (q_x, q_y) is the position of the point q, d_q is the disparity of q, (a_p, b_p, c_p) is p's 3D label, and $\mathbf{n_p}$ is p's refined normal. Note that the disparity of p is not refined if we do not find any reliable point around it ($k = 0$). The disparity of p is computed when assigning p a new 3D label. The proposed method considers the information of neighboring reliable points' 3D labels. In the slanted regions, plane fitting with only disparity information may not be sufficient to obtain an accurate result. With the information of the neighboring 3D labels, the plane fitting strategy is more robust for the refinement of disparities. The experimental results demonstrate that the proposed approach can achieve accurate disparity refinement.

4.2.4. Weighted median filter

Finally, the weighted median filter from [37] is adopted to refine the disparity map. Let D^N denote the disparity map obtained after normal-based plane fitting. The final disparity map is refined as follows:

$$D^M(p) = \underset{q\in\Omega}{med}\left\{\omega(p,q)D^N(q)\right\} \tag{9}$$

where $\omega(p,q)$ is a weight factor that depends on the Euclidian distance between p and q in RGB color space, and Ω is the neighborhood region of p. $D^M(p)$ is the final disparity map with our accurate approach.

5. EXPERIMENTAL RESULTS

In this section, the proposed method is evaluated from various perspectives, including the performance of the accurate approach in Section 5.2, the performance of the efficient approach in Section 5.2, the performance of the confidence measure in Section 5.3, and the parameter analysis in Section 5.4. The MC-CNN-acrt matching cost computation [11] is executed on a personal desktop computer equipped with an Nvidia Titan X graphics card. The other components of our stereo algorithm are executed on a personal computer with an Intel(R) Core(TM) i5-4590 CPU with 3.30 GHz and 16 GB of RAM. The Middlebury stereo dataset version 2014 [38] is adopted to evaluate our method since it is a more challenging dataset than the older version of the Middlebury dataset.

5.1. Performance of the accurate approach

Some disparity maps obtained using our accurate approach on the Middlebury stereo dataset are shown in Fig. 7, which can help us to qualitatively evaluate the proposed method. The proposed method can generate accurate disparity maps with sharp edges. Then, the proposed accurate approach is compared with other state-of-the-art methods on the Middlebury stereo benchmark. Here, half-size images are adopted in our test since the memory of GPU limits the MC-CNN-acrt matching cost computation. The criteria bad 1.0 (%), bad 2.0 (%), and bad 4.0 (%) for non-occluded (*nonocc*) regions and the whole image area (*all*) are adopted for the evaluations. The statistical results are given in Table 1. As shown, more reasonable outlier classification and disparity refinement results in more accurate disparity estimation. The proposed method achieves a state-of-the-art result for both *nonocc* and *all* regions. As of May 2nd, 2018, the proposed accurate approach ranks 3rd in the criterion bad 4.0 (%) (nonocc) and 2nd in all the other criteria.

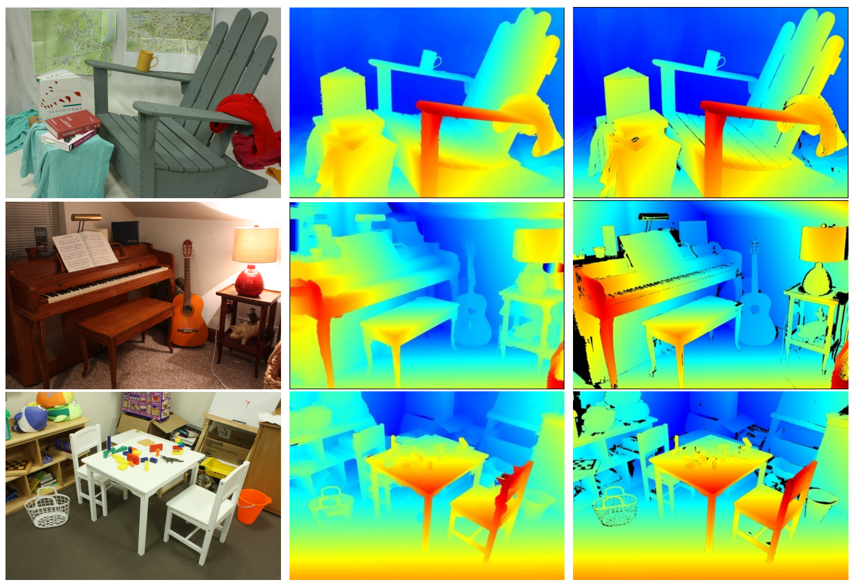


Figure 7: Disparity maps of examples in the Middlebury dataset generated by the proposed accurate approach. From left to right: left images, disparity maps and ground truth

The stereo matching accuracy between the LRC [29] approach and the proposed accurate approach is then compared. The difference is that the former uses LRC to detect outliers while the proposed accurate approach utilizes the improved LRC. The criterion bad 2.0% is used to evaluate their performances. As shown in Fig. 8, the proposed accurate approach achieves a relatively superior result to that of the LRC approach since the improved LRC in the accurate approach considers the regional information more appropriately.

	Average Error						Runtime(s)
	bad 1.0		bad 2.0		bad 4.0		
	nonocc	all	nonocc	all	nonocc	all	
Ours-accurate	13.6	19.8	6.30	12.0	3.83	8.58	867
Ours-efficient	14.1	21.1	7.04	13.8	4.73	10.8	411
NOSS	12.9	19.5	5.82	11.9	3.67	8.73	1545
LocalExp [12]	13.7	19.9	6.52	12.1	4.07	8.69	846
SGM-Forest	14.8	21.9	7.01	13.7	3.73	9.55	64.6
3DMST [13]	15.1	21.5	7.08	12.9	4.43	9.22	167
APAP-Stereo	20.9	27.3	7.53	14.3	4.50	10.8	117
FEN-D2DRR [39]	16.7	23.8	7.89	14.1	3.98	8.53	73.3
PMSC [40]	16.4	23.0	8.20	14.2	5.15	10.0	579
LW-CNN [41]	16.6	25.7	8.31	17.8	4.89	14.1	224
MeshStereoExt [42]	18.4	26.2	9.32	16.8	5.53	12.0	133
MCCNN-Layout	18.0	27.0	9.34	18.6	5.21	14.3	300
OVOD	17.6	24.2	9.65	15.8	5.67	10.8	59.9
NTDE [43]	18.1	25.8	9.94	16.9	6.13	11.8	128
MC-CNN-acrt [11]	18.4	27.7	10.1	19.7	6.34	15.7	106

Table 1: Comparison of the proposed method and the state-of-the-art stereo methods against the Middlebury benchmark 3.0

We also evaluate the effect of normal-based plane fitting in Section 4.2.3. We compare the performances of plane fitting with and without the normal constraint. The statistical results are shown in Fig. 9. Here, the criterion bad 2.0% is adopted. Clearly, plane fitting with the normal constraint performs better for most sequences than trivial plane fitting since the proposed method utilizes more information. A more reasonable 3D label can be obtained for mismatch outliers when using the normal constraint.

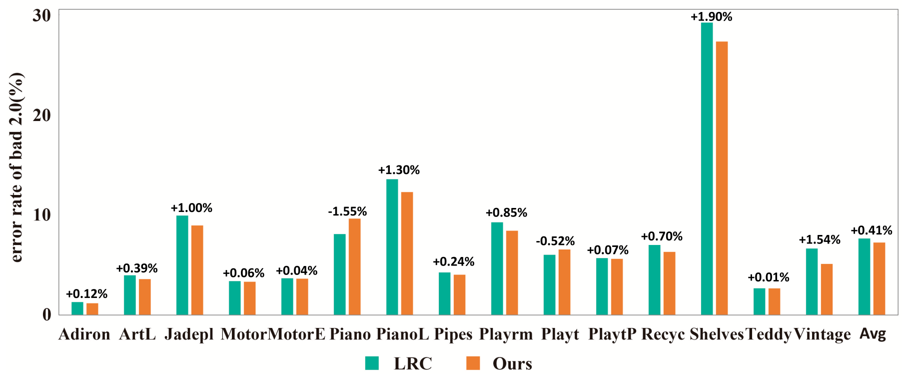


Figure 8: Comparison of the LRC approach and the proposed accurate approach using the criterion bad 2.0%. For each pair of bars, the baseline is "Ours"

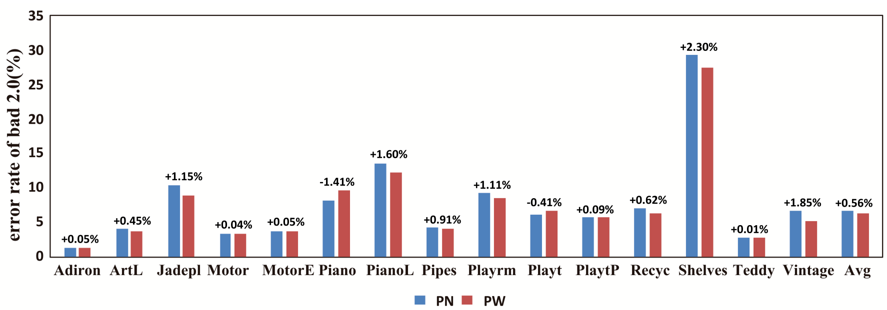


Figure 9: Comparison of plane fitting without the normal constraint (PN) and plane fitting with the normal constraint (PW). The criterion bad 2.0% is adopted, and for each pair of bars, the baseline is "PW"

5.2. Performance of the efficient approach

We first compare the runtimes of the proposed efficient approach and the outlier detection approach that computes the disparity maps for both the left and the right images, e.g., LRC. No refinement is conducted in this experiment since we consider only the runtimes for obtaining the outlier maps. Table 2 compares the runtimes of the proposed efficient approach and LRC and shows that the proposed efficient approach consumes approximately half of the runtime of LRC since the efficient approach does not need to compute the disparity map for the right image.

	Andiron	ArtL	Jadepl	Motor	MotorE	Piano	PianoL	Pipes
LRC	1032.58	344.22	1024.09	1036.02	950.79	829.83	955.22	984.52
Ours	466.78	169.63	493.49	506.67	498.43	504.52	538.34	388.49
	Playrm	**Playt**	**PlaytP**	**Recyc**	**Shelvs**	**Teddy**	**Vintge**	**Average**
LRC	854.91	730.95	849.15	953.54	994.28	361.90	1192.00	876.93
Ours	394.63	373.78	455.35	491.03	476.83	177.6	474.69	410.88

Table 2: Runtime (s) comparison of the LRC and the proposed efficient approach on the Middlebury stereo dataset version 2014

To test how the efficient outlier detection approach influences the accuracy of stereo matching, we refine the disparities of the outliers uniformly using the 3D label assignment algorithm in Section 4.2.2 since the proposed efficient approach does not classify outliers as occlusions and mismatches. The statistical results of the efficient approach are shown in Table 1. The efficient approach performs well in terms of accuracy after the disparity refinement step, and it is also efficient in terms of time and memory.

Note that the size of the Middlebury dataset is relatively large, and some global optimization algorithms [12], which are time-consuming are used to compute the disparity maps here. Subsequently, the runtime difference between LRC and the proposed efficient approach is large. In some real-time embedded multimedia systems, obtaining the disparity map for the right image is relatively cheap, and the runtime overhead is therefore little [44]. Nevertheless, the proposed efficient approach balances the computing time and accuracy well, especially for some time-consuming algorithms.

5.3. Performance of the confidence measure

In this sub-section, the performance of the proposed efficient approach is evaluated further and compared with some other efficient outlier detection approaches in terms of confidence measure. First, some notations are given. Let

$d_1(p) = D^L(p)$ denote the predicted disparity at position p, let $c_1(p) = C_1(p,d)$ denote the smallest matching cost, and let $c_2(p)$ denote the second smallest matching cost.

We compare several confidence measures: (1) the naive version of peak ratio (PKRN) [30], i.e., the ratio between the second smallest matching cost and the smallest cost, $C_{PKRN} = c_2/c_1$; (2) curvature (CUR) [33], $C_{CUR} = -2c(d_1) + c(d_1+1) + c(d_1-1)$; (3) maximum likelihood measure (MLM) [34], $C_{MLM} = e^{-c_1/\sigma^2_{MLM}} / \sum_d e^{-c_d/\sigma^2_{MLM}}$ (4) LRD [30]; and the proposed efficient approach. We use the area under the curve (AUC) measure in [30] to evaluate the performance of these confidence measures. The average AUC of the Middlebury training dataset is shown in Table 3. The proposed method performs better than the other confidence measures. Some detection results are also illustrated in Fig. 10. Clearly, the outlier detection result of the proposed efficient approach has less noise than that of the other approaches and achieves dense detection results.

	PKRN	CUR	MLM	LRD	Ours
AUC	9.35	11.55	14.85	11.19	7.75

Table 3: The average AUC for the Middlebury dataset for different confidence measures. The proposed method outperforms other confidence measures in terms of the AUC

To further evaluate the occlusion detection effectiveness of the proposed efficient approach, two criteria are utilized: the hit rate and the false-positive rate of occlusions [20]. The Middlebury dataset with manually labeled occlusions is adopted here. Table 4 shows the hit rate and false-positive rate of occlusions for various detection methods. The proposed approach achieves the highest hit rate and the lowest false-positive rate.

5.4. Parameter analysis

In this sub-section, the sensitivity of the parameters used in two approaches are analyzed and carefully discussed. First, all the parameter settings are listed in Table 5. Then, the sensitivity of the parameters is evaluated.

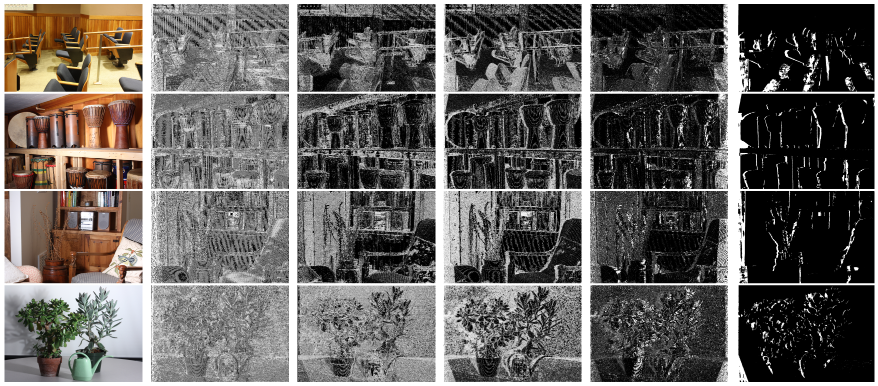


Figure 10: Comparison of various outlier detection approaches. From left to right: raw images, confidence maps of MLM, confidence maps of CURVE, confidence maps of PKRN, confidence maps of LRD, and confidence maps of the proposed method. Bright pixels indicate low confidence, i.e., a tendency to be an outlier. Our confidence maps (outlier detection maps) have less noise and achieve dense detection results

	PKRN	CUR	MLM	LRD	Ours
Hit rate	0.589	0.531	0.331	0.716	0.761
False-positive rate	0.172	0.251	0.280	0.130	0.110

Table 4: The hit rate and false-positive rate of occlusions [20] for the Middlebury dataset for different confidence measures

5.4.1. Parameter analysis of the efficient approach

In the efficient approach, the parameter η in Equation 4 controls the threshold of whether one point is an outlier when using MFPJ. Fig. 11(a) illustrates the sensitivity analysis result for the Middlebury 3.0 dataset. The AUC measure [30] is adopted to evaluate the outlier detection performance of different parameter values. The proposed MFPJ method is insensitive to η in a large

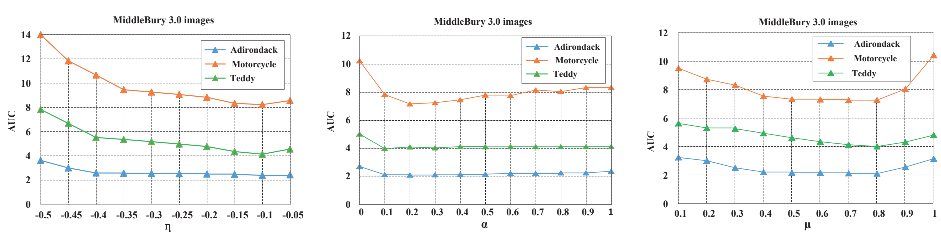
η	α	μ	κ	ρ
-0.1	0.15	0.8	0.6	0.2

Table 5: The parameter settings of the proposed method

parameter setting range. Note that in this experiment, the other parameters are fixed as in Table 5, except for the parameter η. The following experiments for the other parameter analyses are conducted similarly.

When detecting high-probability outliers in Equation 5, the parameter α controls the numbers of outliers detected by MFPJ and the ORD. Fig. 11(b) shows the AUC using different α on Middlebury 3.0 dataset. When using only the ORD method (α equals to 0), the AUC is relatively high since the ORD has its own limitations as stated in Section 4.1.2. When using only the MFPJ method (α equals to 1), the AUC is lower than that of the ORD but is still relatively high. The performance of outlier detection is better when these two methods are combined, and the high-probability outlier detection performance is stable for the parameter α in a large range.

Parameter μ controls the threshold of whether one point is a high-probability outlier. As shown in Fig. 11(c), the proposed high-probability outlier detection method is insensitive when μ varies from 0.5 to 0.8.

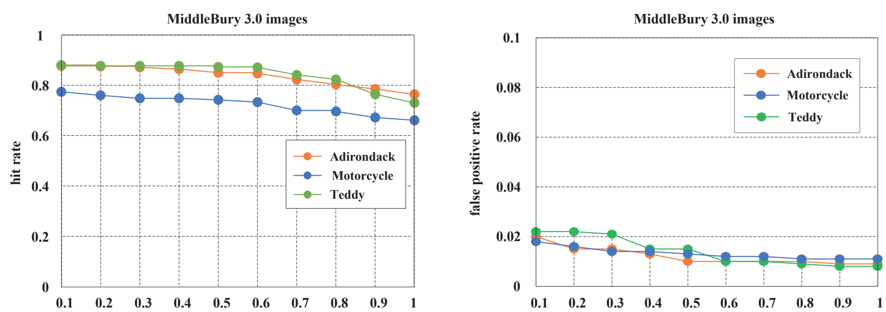


(a) (b) (c)

Figure 11: Sensitivity analysis of the parameters in the efficient approach on the Middlebury 3.0 dataset

5.4.2. Parameter analysis of the accurate approach

In the accurate approach, parameter κ controls the ratio between the occlusions and mismatches in the improved LRC in Section 4.2.1. If one mismatch is surrounded by occlusions, and the ratio of the occlusions is greater than the threshold κ, then the mismatch is reclassified as an occlusion. To evaluate the detection performance of the improved LRC when using different κ, two criteria are used: the hit rate and the false-positive rate of occlusions [20]. Fig. 12 illustrates the hit rate and the false-positive rate of occlusions at various κ. The hit rate of occlusions decreases when κ increases, as shown in Fig. 12(a). The false-positive rate of occlusions also decreases when κ increases, as illustrated in Fig. 12(b). Both the hit rate and the false-positive rate are stable regardless of the variation of κ.



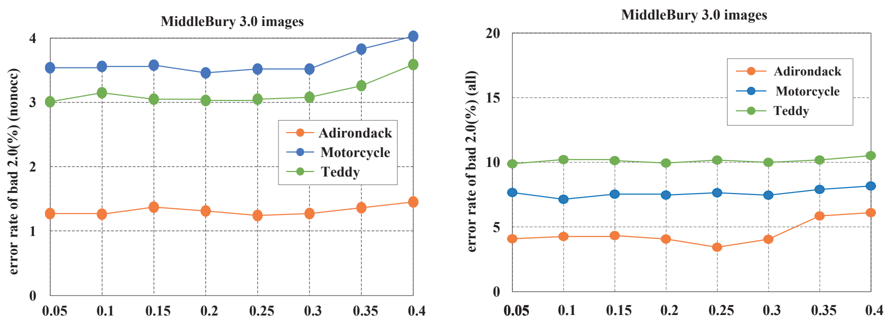
(a) (b)

Figure 12: The hit rate and false-positive rate of occlusions [20] with various κ

Parameter ρ is the threshold of whether one texture edge point is regarded as a real discontinuous boundary point in Section 4.2. Fig. 13 shows the error rate of bad 2.0 (%) in the nonocc regions and in all regions of the Middlebury dataset. The real discontinuous boundary is an auxiliary means to confine the search region of reliable points, and the final stereo matching result is not sensitive to ρ in the range from 0.05 to 0.3. Note that when ρ is too large to detect real discontinuous boundaries, the result will be influenced since the search region is not confined.

6. CONCLUSION

Outlier detection is an important component of stereo matching. Good outlier detection will increase the robustness and accuracy of the stereo matching algorithm. The disparities in outlier regions can be discarded or filled by some



(a) (b)

Figure 13: The error rate of bad 2.0 (%) in the nonocc regions are in the all regions with different ρ

reliable disparities depending on different application conditions. The efficient outlier detection approach detects outliers at a relatively low cost, i.e., approximately half of the time and memory expenses. The accurate outlier detection approach classifies outliers reasonably, providing a better guidance for disparity refinement.

One possible future direction is to investigate how to utilize the outlier detection results to guide other stereo matching steps, such as matching cost computation and cost aggregation, to obtain a more robust stereo matching result.

7. ACKNOWLEDGMENT

The authors would like to thank Qing Ran for her instructive commentary regarding of this paper. This work was supported by the National Natural Science Foundation of China under Grants No. 61732015 and No. 61472349, and Key Research and Development Program of Zhejiang Province under Grants No. 2018C01090.

References

[1] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, International Journal of Computer Vision, 47 (1-3) (2002) 7–42.

[2] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, X. Zhang, On building an accurate stereo matching system on graphics hardware, in: Computer Vision Workshops (ICCV Workshops), 2011, pp. 467–474.

[3] J. Jiao, R. Wang, W. Wang, S. Dong, Z. Wang, W. Gao, Local stereo matching with improved matching cost and disparity refinement, IEEE MultiMedia, 21 (4) (2014) 16–27.

[4] Y. Zhan, Y. Gu, K. Huang, C. Zhang, K. Hu, Accurate image-guided stereo matching with efficient matching cost and disparity refinement, IEEE Transactions on Circuits and Systems for Video Technology, 26 (9) (2016) 1632–1645.

[5] A. Li, D. Chen, Y. Liu, Z. Yuan, Coordinating multiple disparity proposals for stereo computation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4022–4030.

[6] S. N. Sinha, D. Scharstein, R. Szeliski, Efficient high-resolution stereo matching using local plane sweeps, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1582–1589.

[7] S. Hadfield, K. Lebeda, R. Bowden, Stereo reconstruction using top-down cues, Computer Vision and Image Understanding, 157 (2017) 206–222.

[8] D. Kong, H. Tao, A method for learning matching errors for stereo computation, in: British Machine Vision Conference, Vol. 1, 2004, p. 2.

[9] R. Zabih, J. Woodfill, Non-parametric local transforms for computing visual correspondence, in: European Conference on Computer Vision, Springer, 1994, pp. 151–158.

[10] Y. S. Heo, K. M. Lee, S. U. Lee, Robust stereo matching using adaptive normalized cross-correlation., IEEE Transactions on Pattern Analysis and Machine Intelligence, 33 (4) (2011) 807.

[11] J. Zbontar, Y. LeCun, Stereo matching by training a convolutional neural network to compare image patches, Journal of Machine Learning Research, 17 (2016) 1–32.

[12] T. Taniai, Y. Matsushita, Y. Sato, T. Naemura, Continuous 3d label stereo matching using local expansion moves, IEEE Transactions on Pattern Analysis and Machine Intelligence, 40 (11) (2018) 2725–2739.

[13] L. Li, X. Yu, S. Zhang, X. Zhao, L. Zhang, 3d cost aggregation with multiple minimum spanning trees for stereo matching, Applied optics, 56 (12) (2017) 3411–3420.

[14] S. Drouyer, S. Beucher, M. Bilodeau, M. Moreaud, L. Sorbier, Sparse stereo disparity map densification using hierarchical image segmentation, in: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing, 2017, pp. 172–184.

[15] 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.

[16] K. Zhang, J. Lu, G. Lafruit, Cross-based local stereo matching using orthogonal integral images, IEEE Transactions on Circuits and Systems for Video Technology, 19 (7) (2009) 1073–1079.

[17] K. He, J. Sun, X. Tang, Guided image filtering, IEEE Transactions on Pattern Analysis and Machine Intelligence, 35 (6) (2013) 1397–1409.

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

[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: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1854–1861.

[20] G. Egnal, R. P. Wildes, Detecting binocular half-occlusions: Empirical comparisons of four approaches, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (8) (2000) 1127–1133.

[21] J. J. Little, W. E. Gillett, Direct evidence for occlusion in stereo and motion, Image and Vision Computing, 8 (4) (1990) 328–340.

[22] A. Spoerri, The early detection of motion boundaries, Proceedings of the IEEE International Conference on Computer Vision, (1987) 209–218.

[23] B. L. Anderson, K. Nakayama, Toward a general theory of stereopsis: binocular matching, occluding contours, and fusion, Psychological review, 101 (3) (1994) 414.

[24] H. H. Baker, Depth from edge and intensity based stereo, Tech. rep., Stanford Univ Ca Dept of Computer Science (1982).

[25] A. L. Yuille, T. Poggio, A generalized ordering constraint for stereo correspondence, Tech. rep., Massachusetts Inst of Tech Cambridge Artificial Intelligence Lab (1984).

[26] S. S. Intille, A. F. Bobick, Disparity-space images and large occlusion stereo, in: European Conference on Computer Vision, Springer, 1994, pp. 179–186.

[27] C. Chang, S. Chatterjee, P. R. Kube, On an analysis of static occlusion in stereo vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1991, pp. 722–723.

[28] J. Weng, N. Ahuja, T. S. Huang, Two-view matching, in: International Conference on Computer Vision, 1988, pp. 64–73.

[29] S. Mattoccia, F. Tombari, L. D. Stefano, Stereo vision enabling precise border localization within a scanline optimization framework, in: Asian Conference on Computer Vision, 2007, pp. 517–527.

[30] X. Hu, P. Mordohai, A quantitative evaluation of confidence measures for stereo vision, IEEE Transactions on Pattern Analysis and Machine Intelligence, 34 (11) (2012) 2121–33.

[31] D. Scharstein, R. Szeliski, Stereo matching with nonlinear diffusion, International Journal of Computer Vision, 28 (2) (1998) 155–174.

[32] H. Hirschmüller, P. R. Innocent, J. Garibaldi, Real-time correlation-based stereo vision with reduced border errors, International Journal of Computer Vision, 47 (1-3) (2002) 229–246.

[33] G. Egnal, M. Mintz, R. P. Wildes, A stereo confidence metric using single view imagery with comparison to five alternative approaches, Image and Vision Computing, 22 (12) (2004) 943–957.

[34] L. Matthies, Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation, International Journal of Computer Vision, 8 (1) (1992) 71–91.

[35] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, Real-time visibility-based fusion of depth maps, in: IEEE International Conference on Computer Vision, 2007, pp. 1–8.

[36] K.-J. Yoon, I. S. Kweon, Distinctive similarity measure for stereo matching under point ambiguity, Computer Vision and Image Understanding, 112 (2) (2008) 173–183.

[37] M. Mozerov, X. Roca, J. J. Villanueva, Trinocular stereo matching with composite disparity space image, in: IEEE International Conference on Image Processing, 2009, pp. 2065–2068.

[38] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, P. Westling, High-resolution stereo datasets with subpixel-accurate ground truth, in: German Conference on Pattern Recognition, 2014, pp. 31–42.

[39] X. Ye, J. Li, H. Wang, H. Huang, X. Zhang, Efficient stereo matching leveraging deep local and context information, IEEE Access, 5 (2017) 18745–18755.

[40] L. Li, S. Zhang, X. Yu, L. Zhang, Pmsc: Patchmatch-based superpixel cut for accurate stereo matching, IEEE Transactions on Circuits and Systems for Video Technology, 28 (3) (2018) 679–692.

[41] H. Park, K. M. Lee, Look wider to match image patches with convolutional neural networks, IEEE Signal Processing Letters, 24 (12) (2017) 1788–1792.

[42] C. Zhang, Z. Li, Y. Cheng, R. Cai, Meshstereo: A global stereo model with mesh alignment regularization for view interpolation, in: IEEE International Conference on Computer Vision, 2015, pp. 2057–2065.

[43] K. R. Kim, C. S. Kim, Adaptive smoothness constraints for efficient stereo matching using texture and edge information, in: IEEE International Conference on Image Processing, 2016, pp. 3429–3433.

[44] K. R. Bae, B. Moon, An accurate and cost-effective stereo matching algorithm and processor for real-time embedded multimedia systems, Multimedia Tools and Applications, 76 (17) (2016) 1–16.