Continuous Symmetric Stereo with Adaptive Outlier Handling

Chen Li[1*] Lap-Fai Yu[2] Zhichao Lu[3] Yasuyuki Matsushita[4] Kun Zhou[1] Stephen Lin[5]
[1]State Key Lab of CAD&CG, Zhejiang University [2]University of Massachusetts Boston
[3]Peking University [4]Osaka University [5]Microsoft Research

Abstract

We present a method for symmetric stereo matching in which outliers from occlusions, texture-less regions, and repeated patterns are handled in a soft and adaptive manner. Rather than making binary outlier decisions, our model incorporates continuous-valued confidence weights that account for outlier likelihood, to promote robustness in disparity estimation. In contrast to previous outlier labeling techniques that fix the labels at the start of optimization, our method iteratively updates our outlier confidence weights as the matching results are gradually refined. By doing this, errors in an initial labeling can be rectified in the matching process. Our model is optimized in an Expectation-Maximization framework that efficiently produces continuous disparity estimates. This approach provides a good combination of accuracy and speed. Experiments show that our method compares favorably to prior outlier labeling techniques on the Middlebury benchmark, and that it can generate high-quality reconstruction for outdoor images with much more complex occlusions.

1. Introduction

Stereo matching is an ill-posed problem with ambiguities due to occlusion, color homogeneity and repeated textures. These matching ambiguities, also referred to as outliers, are commonly dealt with using some form of smoothing or filtering which masks their detrimental effects. For example, global optimization approaches incorporate smoothness regularization into their objective functions to obtain a better approximation of disparities in such cases, while local approaches employ cost aggregation on pixel-based stereo matching costs. Though these schemes can help to reduce the impact of this problem to some degree, outliers nevertheless are a major source of error, and addressing them effectively is essential for high-quality stereo matching.

*This work was done while Chen Li was an intern at Microsoft Research.

Figure 1. 3D scenes reconstructed by our method.

The issue of outlier handling has been explicitly examined in some recent works. Most of these methods identify pixels that are half-occluded (*i.e.*, whose counterpart in the other image is occluded), and then tag these pixels with the intent to limit their influence in global optimization [26, 17, 25]. In [28], the notion of outlier confidence is introduced, where a likelihood measure of a pixel half-occlusion (which we will refer to simply as occlusion) is used instead of a binary label. By softly incorporating this information into their matching model, more robust disparity estimation is achieved. These previous uses of outlier labeling have led to improvements in estimation accuracy, but are limited in that the labels are fixed at the beginning of the optimization process. As a result, incorrectly labeled pixels, such as occlusions that were mistakenly overlooked, cannot be rectified in the latter process. Also, pixels initially considered as outliers are omitted from the optimization, even if there exists a correct match for it. The inclusion of (or high confidence in) actual outliers, combined with the exclusion of (or low confidence in) pixels that can provide useful matching data, can lead to degradation of stereo matching results.

To address this problem, we propose a novel formulation that handles outliers in an adaptive manner, where soft outlier confidence weights are progressively updated as the disparity solution is gradually refined. With improvements in the disparity map, confidence values can evolve, since disparity inconsistencies between the two images may be

resolved, and initially missed outliers may become more apparent. We model this soft and adaptive outlier confidence in terms of left-right disparity consistency and ordering constraints. In addition, the adaptive confidence weights are incorporated in manner that allows for an efficient solution in a continuous disparity space, unlike the discrete models of previous outlier handling works [26, 17, 25, 28].

Optimization of the proposed matching formulation presents a challenge, since the inclusion of left-right disparity consistency constraints would require a higher order model in the MRF framework, and solving the matching problem in a continuous and multi-dimensional disparity space makes discrete search methods such as graph cuts [6] and belief propagation [11] unsuitable. To address this optimization problem, we iteratively update the adaptive confidence weights and estimate the disparity values in an Expectation-Maximization (EM) framework. In the maximization step where we estimate disparity, we approximate the optimization problem as a linear system from which a continuous solution can be generated.

With this technique, continuous and symmetric stereo disparities for the left and right views can be more efficiently and accurately estimated than in previous works that employ outlier labeling [31, 26, 28]. We demonstrate this experimentally with good performance on the Middlebury benchmark [23, 24] and on outdoor images without substantial computation time. Examples of 3D point clouds reconstructed by our method are displayed in Fig. 1.

2. Related Work

Stereo matching is a well-studied area in which many techniques are compared on the Middlebury website [1]. We review methods most closely related to our work.

Recently, outlier and occlusion handling has been addressed by many researchers. Some methods estimate which pixels are outliers only at the start of disparity optimization. In [29], pixels are labeled as occluded if they fail a left-right disparity consistency check, and the remaining pixels are labeled as stable or unstable according to the distinctiveness of their optimal matching energy, as this indicates the stability of the winner-take-all scheme. In [31], binary confidence weights are computed for the purpose of excluding overly-texture-less and overly-repetitive pixels. The method of [28] computes a continuous-valued outlier confidence map for soft handling of occluded pixels. Since these methods do not re-evaluate the outlier status of pixels during disparity optimization, any errors in this initialization are baked into the result.

There are other occlusion-handling methods that update their outlier estimates as part of the optimization process. In [3, 2], a definition of half-occluded regions was introduced, and simple equations were derived for determining these regions from the disparity function. The method of [26] solves for an occlusion map and a disparity map in alternation, where the occlusion map is computed from a visibility constraint which only requires that an occluded pixel have no match while a non-occluded pixel have at least one match. The weakness of this visibility constraint arises in part from its discrete representation of disparity, for which left-right disparity consistency does not necessarily hold [26]. In [4], a continuous disparity method is presented using the Mumford-Shah framework [19]. Occlusion pixels are determined iteratively based on left-right disparity consistency. In contrast, our method additionally accounts for ordering constraints and accommodates continuous outlier weights for non-deterministic soft handling.

A generative method for multi-view stereo is presented in [25], in which visibility and depth are jointly estimated in an iterative manner. Within a probabilistic framework, outliers are detected if they cannot be explained by the majority of images. This approach, while effective for multi-view stereo, is less suited to the case of only two images, which provides limited data for statistical inference. On the Middlebury benchmark, occlusion regions are shown not to be effectively handled.

3. Formulation

Our problem setting is similar to that of [27, 31]. Given a rectified stereo image pair $I = \{I_L, I_R\}$, where I_L, I_R are the left and right images, our goal is to estimate dense disparity maps $D = \{D_L, D_R\}$. We use a conventional stereo formulation by following previous approaches [23] and define an objective function $E(D)$, which consists of a data term E_d and smoothness term E_s weighted by the smoothness weight λ_s as

$$E(D) = E_d(D) + \lambda_s E_s(D). \tag{1}$$

These two energy terms are described in Secs. 3.1 and 3.2.

3.1. Data Term

Our data term E_d encodes the photo-consistency of pixel correspondence for the hypothesized disparity as

$$E_d(D; I) = \sum_{i \in I} \omega_i E_d(i, d_i; I), \tag{2}$$

where ω_i is the adaptive weight, which represents the confidence of the disparity estimation d_i for pixel i.

We select the pixel-based matching cost to be a truncated absolute difference of the color and the gradient at matching points, which has been shown to be robust to illumination changes [21, 7]:

$$C(i, d; I) = (1 - \alpha) \min\left(\|I_{i'} - I_i\|, \tau_c\right) + \alpha \min\left(\|\nabla_x I_{i'} - \nabla_x I_i\|, \tau_g\right). \tag{3}$$

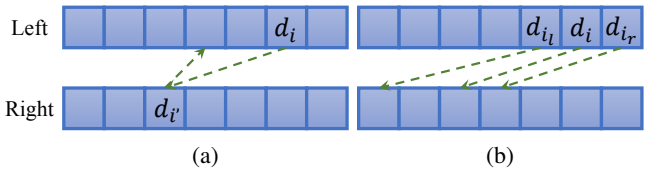


Figure 2. Factors for adaptive confidence weights. (a) Left-right consistency, where matched pixels are expected to have consistent disparity estimates. (b) Ordering, where the horizontal order of matches is expected to be consistent between the images.

In the above, $I_i \in \mathbb{R}^3$ is the RGB color vector for pixel i, and i' is the matched pixel of i in the other view with disparity d. $\nabla_x I$ is the gray-scale gradient in the x-direction, α balances the color and gradient terms, and τ_c, τ_g are truncation values for greater robustness.

We then filter the pixel-wise cost of Eq. (3) using guided image filtering [14, 21] and define the photo-consistency term as

$$E_d(i, d_i; I) = \sum_j W_{i,j} C(j, d_i; I), \tag{4}$$

where $W_{i,j}$ is the filter weight defined in [14].

Adaptive confidence weight ω_i Our adaptive confidence weight is inspired by recent occlusion handling methods [26] and previous uses of fixed confidence weights for the data term [28, 31]. In formulating the confidence weight for a pixel i, we account for two factors that are indicative of matching accuracy, namely left-right consistency and ordering of matches, as illustrated in Figure 2. We define the confidence weight ω_i as the product of their two measures:

$$\omega_i = \omega_d \omega_o, \tag{5}$$

where ω_d is the left-right consistency weight, and ω_o is the ordering weight. Though the values of ω_d and ω_o typically vary among pixels, we omit subscript i to simplify notation.

We define the left-right consistency weight ω_d as a Gaussian function of the disparity difference between matched pixels:

$$\omega_d = \exp\left(-\frac{|d_i - d_{i'}|^2}{{\sigma_d}^2}\right), \tag{6}$$

where σ_d controls the sensitivity to the disparity difference.

The ordering weight ω_o is defined as follows. Consider three adjacent pixels i_l, i, and i_r in the left image with disparities d_{i_l}, d_i, d_{i_r}. If none of these points are occluded in the other view, the three matched pixels ${i_l}'$, i', and ${i_r}'$ in the other image should preserve the left-to-right order [13]. This ordering implies a constraint on their disparities: $d_{i_l} + 1 \geq d_i \geq d_{i_r} - 1$. Based on this observation, we define the ordering weight for the left image as

$$\omega_o = m_l m_r, \tag{7}$$

where

$$m_l = \begin{cases} 1.0 & \text{if } d_{i_l} + 1 \geq d_i \\ T_s & \text{otherwise} \end{cases} \tag{8}$$

$$m_r = \begin{cases} 1.0 & \text{if } d_i \geq d_{i_r} - 1 \\ T_s & \text{otherwise.} \end{cases} \tag{9}$$

The value of T_s is set to be small, such that inconsistent orderings are penalized. The ordering weight for the right image is similarly computed by flipping the inequalities.

3.2. Smoothness Term

Like previous approaches, our method uses smoothness regularization in disparity estimation. Our smoothness term consists of local and regional smoothness terms, E_l and E_r, respectively:

$$E_s(D) = E_l(D) + \lambda_r E_r(D), \tag{10}$$

where λ_r balances the two smoothness terms.

Local smoothness term E_l This smoothness term is inspired by the use of bilateral filtering in adaptive support-weight stereo cost aggregation [30] and also in the recent depth upsampling method of [20]. Our local smoothness term E_l is defined as

$$E_l(D) = \sum_{i \in I} \sum_{j \in \mathcal{N}(i)} \omega_c \omega_p (d_i - d_j)^2, \tag{11}$$

where $\mathcal{N}(i)$ is the set of 4-connected neighbors of i. The factors ω_c and ω_p are the weights of the bilateral filter, representing color similarity and spatial distance in the image coordinates, respectively. These are defined as

$$\begin{cases} \omega_c = \exp\left(-\frac{\|I_i - I_j\|^2}{{\sigma_c}^2}\right), \\ \omega_p = \exp\left(-\frac{\|p_i - p_j\|^2}{{\sigma_p}^2}\right), \end{cases} \tag{12}$$

where p_i and p_j are the image locations of pixels i and j, and σ_c and σ_p adjust the sensitivity to color similarity and spatial distance.

Regional smoothness term E_r Recent segmentation-based stereo approaches have demonstrated high accuracy in stereo matching. Our method incorporates a segmentation-based soft constraint in a manner similar to [26] by favoring disparity values within a segment to fit closely to a 3D plane. We define the regional smoothness term E_r with respect to a local plane $d = ax + by + c$ in the disparity domain as

$$E_r(D) = \sum_{i \in I} \left(d_i - (a_i x_i + b_i y_i + c_i)\right)^2, \tag{13}$$

where $\{a_i, b_i, c_i\}$ are the 3D plane parameters fit to the disparity values in the segment containing pixel i with image location $p_i(x_i, y_i)$. For segmentation, we use mean-shift color segmentation [8] with an appearance range resolution of $h_c = 8.0$, spatial resolution of $h_s = 3.0$, and smallest segment size of $M = 500$. The plane for a segment S is fitted by weighted least squares using our confidence weight ω_i of Eq. (5):

$$\{a, b, c\} = \underset{a,b,c}{\operatorname{argmin}} \sum_{i \in S} \omega_i (\hat{d}_i - (ax_i + by_i + c))^2, \quad (14)$$

where $\hat{d}$ is the disparity estimate of pixel i.

4. Solution method

Our problem described in Sec. 3 is difficult to optimize using a standard discrete optimization technique, such as graph cuts or belief propagation, because our disparity space is continuous, and more importantly because our left-right consistency and ordering constraints would turn the MRF into a higher-order model.

To efficiently solve this problem, we take an iterative optimization approach in the EM framework. Given a stereo pair I, we estimate unknown disparities D with a confidence map O (equal to 0 for unsure matchings, and 1 for confident matchings) regarded as hidden data:

$$\begin{aligned} D^* &= \underset{D}{\operatorname{argmax}} \log P(I, D) \\ &= \underset{D}{\operatorname{argmax}} \log \sum_{O \in \psi} P(D, O, I), \end{aligned} \quad (15)$$

where ψ represents the solution space of the hidden data O.

The unknown parameters D are initialized by fast local approaches, such as [21, 12]. We then iteratively perform the expectation (in Sec. 4.1) and maximization (in Sec. 4.2) steps to simultaneously estimate the disparity maps D of both the left and right views.

4.1. Expectation step

In the expectation step, we assign a probability to each pixel of being an outlier given the disparity estimates $D^{(n)}$ at the n-th iteration. Unlike [26], which directly estimates binary occlusion maps, we use the adaptive confidence weight ω_i in Sec. 3.1 to determine the confidence map probabilities $P(o_i = \{0, 1\}|D^{(n)})$ of pixel i:

$$\begin{cases} P(o_i = 0|D^{(n)}) = 1 - \omega_i^{(n)}, \\ P(o_i = 1|D^{(n)}) = \omega_i^{(n)}. \end{cases}$$

The confidence map probabilities inherit the properties of our adaptive confidence weights, and have the effect of disregarding matches that do not satisfy left-right disparity consistency or disparity orderings.

4.2. Maximization step

In the maximization step, we maximize Eq. (15) with respect to the parameter D given the observation I:

$$\begin{aligned} D^{(n+1)} &= \underset{D}{\operatorname{argmax}} \sum_{O \in \psi} P(O|D^{(n)}) \log P(O, I, D) \quad (16) \\ &= \underset{D}{\operatorname{argmax}} \sum_{O \in \psi} P(O|D^{(n)}) \log \left(P(O, I|D)P(D)\right) \\ &= \underset{D}{\operatorname{argmin}} \sum_{O \in \psi} P(O|D^{(n)})(L(O, I|D) + L(D)), \end{aligned}$$

where L is the negative log likelihood of P, *i.e.*, $L = -\log P$.

For unreliable matchings, we set $L(O = 0, I|D)$ to 0 to disregard their effects. Also, because $L(D)$ is independent of the hidden data O, it can be moved out of the summation. As a result, Eq. (16) can be rewritten as

$$D^{(n+1)} = \underset{D}{\operatorname{argmin}} \left\{ L(D) + P(O = 1|D^{(n)}) L(O, I|D) \right\}. \quad (17)$$

Since $L(D)$ represents prior knowledge about the disparity map, we can relate the term with our smoothness term E_s. In addition, the second term $P(O = 1|D^{(n)})L(O, I|D)$ has a correspondence with our data term E_d as it represents the likelihood of photo-consistency given a disparity, weighted by the confidence. From this, we can cast our stereo matching problem of Eq. (1) into that of Eq. (17) with the following relationships:

$$\begin{cases} L(D) = E_s(D), \\ P(O = 1|D^{(n)})L(O, I|D) = E_d(D). \end{cases}$$

Since the photo-consistency data term E_d defined in Eq. (4) is highly non-convex and difficult to optimize, we use the progressive convex hull filtering and parabola fitting of [31] to approximate E_d at each maximization step:

$$E_d(i, d_i; I) \approx a_i^{(n)} \left(d_i - \hat{d}_i^{(n)}\right)^2 + b_i^{(n)} \left(d_i - \hat{d}_i^{(n)}\right) \quad (18)$$

where $a_i^{(n)}$ and $b_i^{(n)}$ are the curvature and tangent of the fitted parabola in [31], and $\hat{d}_i^{(n)}$ is the estimated disparity of pixel i in $D^{(n)}$.

As a result, the sub-problem in the maximization step becomes an over-constrained linear system:

$$\begin{bmatrix} A_d \\ A_r \\ A_l \end{bmatrix} \begin{bmatrix} D_L \\ D_R \end{bmatrix} = \begin{bmatrix} b_d \\ b_r \\ b_l \end{bmatrix}. \quad (19)$$

In the above, A_d, A_r are diagonal matrices corresponding to E_d and E_r with weights ω_i and $\lambda_s \lambda_r$, respectively, and A_l is a symmetric matrix corresponding to E_l, where $A_l(i, j) = \lambda_s \omega_c \omega_p$ as defined in Eq. (11). D_L and D_R are the disparities for the left and right views. b_d and b_r

are known values from Eq. (18) and Eq. (13) respectively, and b_l is a zero vector. When the number of pixels in an image is n, we have $D_L, D_R \in \mathbb{R}^n$, $A_d, A_r \in \mathbb{R}^{2n\times 2n}$, $A_l \in \mathbb{R}^{8n\times 2n}$, $b_d, b_r \in \mathbb{R}^{2n}$, and $b_l \in \mathbb{R}^{8n}$. We use the sparse QR factorization [9] in SuiteSparse[1] to solve the linear system.

5. Experiments

We evaluate our method using the Middlebury dataset [1]. For our algorithm, the weighting factors $(\lambda_s, \lambda_r, \alpha)$ in Eqs. (1, 10, 3) are fixed to $(2.5, 0.04, 0.9)$, and the truncation values (τ_c, τ_g) in Eq. (3) are fixed to $(7, 2)$ when the intensity range is $[0, 255]$. The sensitivity parameters $(\sigma_d, \sigma_c, \sigma_p)$ in Eqs. (6, 12) are set to $(0.4, 1.73, 1.22)$. We take the winner-takes-all result of the local approach in [21] to initialize the disparity maps. Since our method estimates continuous disparities, we evaluate performance for 0.5-pixel accuracy.

Performance comparison on Middlebury dataset We first compare our method to four related techniques that explicitly label outliers, namely SymBP+occ [26], OutlierConf [28], VarMOSH [4], and LLR [31], which is similar to ours in its iterative optimization scheme. Figure 4 shows the estimated disparity and error maps for non-occlusion (denoted as nonocc) regions. In the figure, the blue and orange boxes highlight occlusion and texture-less regions, respectively. Since our method estimates continuous disparities, it generally yields higher accuracy than the discrete disparity methods SymBP+occ [26] and OutlierConf [28], especially in large flat regions. Moreover, our use of adaptively updated outlier confidence leads to less error in occlusion regions such as the blue box in Teddy. Our results in texture-less regions tend to be better than that of [4], especially in the Tsukuba dataset, because of our consideration of ordering constraints. Our results are closer in quality to LLR [31], but our method can better recover from pixels with poor initializations such as in the orange box of Teddy, since ordering constraints help to detect such outliers.

Table 4 lists the rankings and percentages of inaccurately estimated pixels for this comparison. Our method generally outperforms the other outlier handling methods and demonstrates competitive accuracy to LLR with more than 10 times greater computational efficiency.

Adaptive confidence weight ω We examine the significance of our soft, adaptive confidence weight model by comparing it to versions of our method that instead use a uniform data weight and the fixed confidence weight defined in [31]. Comparison results are shown in Figure 5.

[1]SuiteSparse: a Suite of Sparse matrix packages, at http://www.cise.ufl.edu/research/sparse/SuiteSparse/

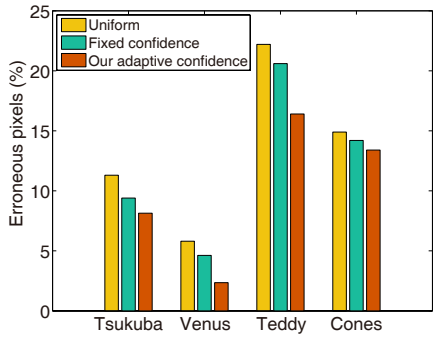


Figure 3. Percentage of erroneous pixels with different weighting factors. Yellow bars: uniform data weight. Green bars: fixed confidence weights. Orange bars: our adaptive confidence weights.

Nonocc err.	Tsukuba	Venus	Teddy	Cones
w/o LR	14.3	5.15	11.2	10.3
w/o ordering	10.6	3.53	13.7	7.92
Our method	7.09	1.85	9.88	6.55

Table 1. Numerical evaluation of left-right consistency and ordering constraints.

For the uniform data weight, we carefully tuned the parameter ω_i in Eq. (2) and use this value for all the pixels. For the fixed confidence weight, we set the value ω_i for each pixel according to the scheme in [31].

In Figure 5 (b) with the uniform data weight, occlusion regions (blue and red) and repetitive texture regions (red and green) are treated the same as other pixels, which results in substantial errors. For the fixed confidence weight [31], the repetitive region (green) is better handled, but the occlusion region (red) still suffers from incorrect weight settings as shown in Figure 5 (c). With our outlier confidence weight (Figure 5 (d)), accurate disparities are obtained in both occlusion and repetitive texture regions. Figure 3 shows the percentage of pixels with incorrectly estimated disparity in four different scenes.

Left-right consistency and ordering constraints Figure 6 shows the importance of incorporating both left-right consistency constraints and ordering constraints in our soft, adaptive confidence weight. Their effect is examined by removing the left-right consistency constraint in Figure 6 (b) and omitting the ordering constraint in Figure 6 (c). Removing either of the two constraints increases the chance that mismatches are not identified as outliers, as shown for the repetitive texture in the red boxes. Likewise, it can also lead to missed occlusions as illustrated in the blue boxes. Application of both constraints together results in appreciable improvements, as shown in Figure 6 (d) and numerical improvement in Tab. 1, since outliers have a much lower likelihood of passing both constraints.

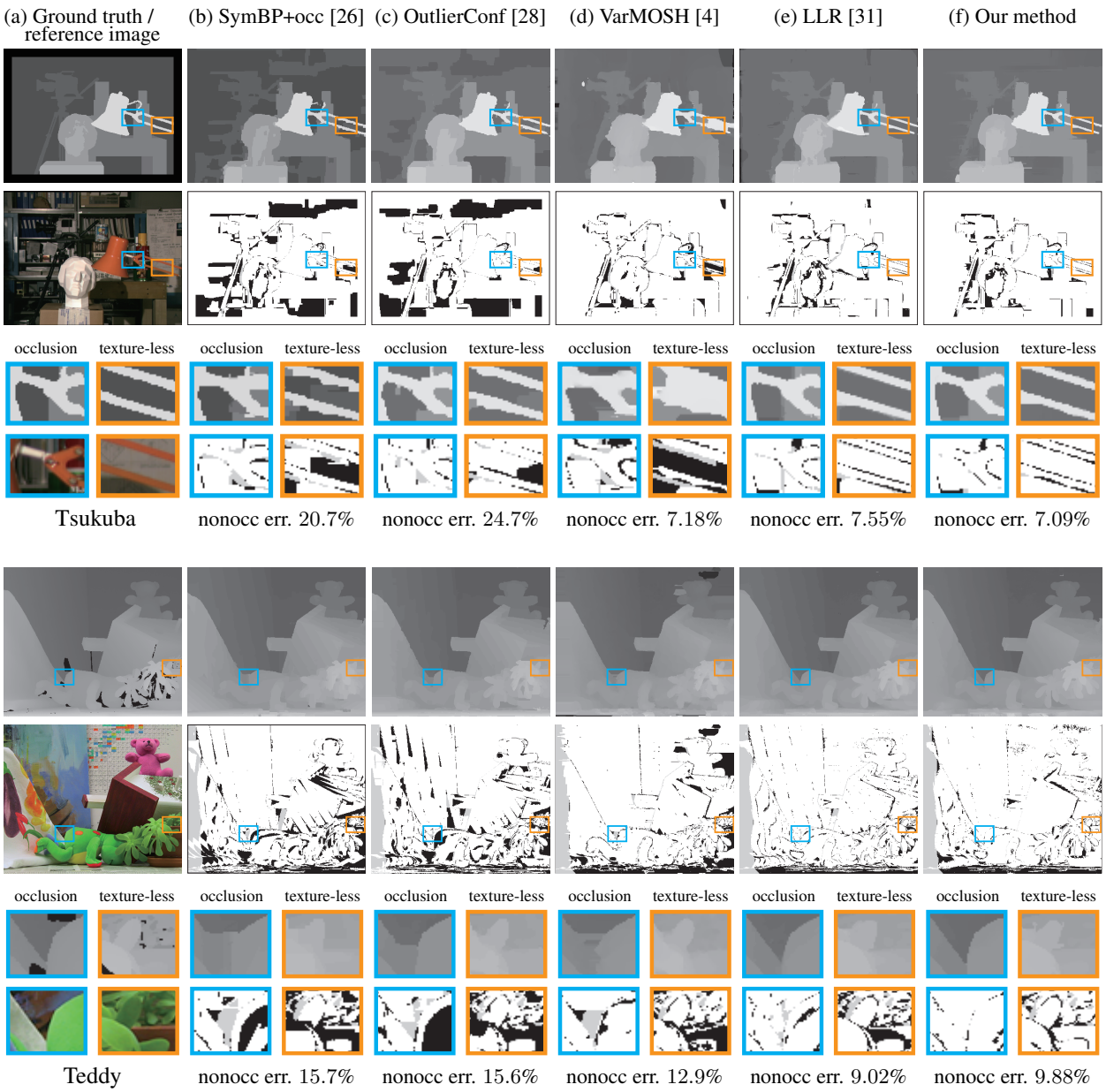


Figure 4. Comparisons of our method to SymBP+occ [26], OutlierConf [28], VarMOSH [4] and LLR [31] on the Tsukuba and Teddy images. Our method shows higher accuracy in large flat regions, occlusion regions (blue boxes), and texture-less regions (orange boxes) in comparison to the related methods. More results are available in the supplementary material.

Results on outdoor images Besides the Middlebury dataset, outdoor images are also used to test our approach. The three images shown in Figure 7(a) are selected from an image sequence captured along a street and rectified wiht a pre-process. The sky region is automatically segmented. We use [12] to obtain an initialization. The resolution of each image is 2200×1900, and it takes about 30 minutes on our PC to process each image pair. Figure 7(c) shows the reconstructed point cloud obtained by merging the three results. The trees, streetlights and structure of the building are nicely reconstructed by our method. From the use of the proposed adaptive confidence weights, the boundaries between the foregrounds and background are correctly determined, and occlusion regions are also handled accurately.

Complexity and convergence analysis Our computation is performed on an Intel Core i7 CPU with 4 cores and 8G RAM. In all of our experiments with the Middlebury datasets, convergence of our iterative method is reached within 10 iterations and a total of 1.5 minutes per $166K$-

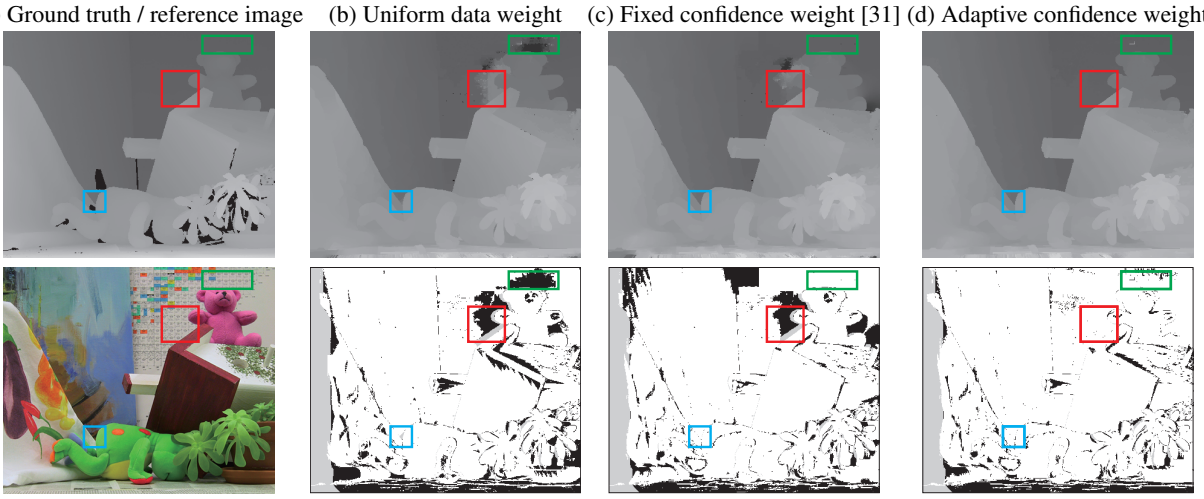


Figure 5. Comparison of weighting strategies for the data term. (a) ground truth disparity map and a reference image. (b) disparity and error map with a uniform data weight. (c) results with the fixed confidence weight of [31]. (d) results with our adaptive confidence weight in Eq. (5).

Figure 7. Results on outdoor images. (a) Reference images. (b) Estimated disparity maps. The disparity range is mapped to wavelengths of visible light (380nm to 710nm) for better visualization. (c) Reconstructed point cloud by merging the three results.

Resolution	166K	360K	1440K
Time per image pair (mins.)	1.5	3.5	14.5

Table 2. Computation time for different image resolutions with the Middlebury datasets [22, 16].

Graph-cut	BP [26, 28]	LLR [31]	PMF [18]	**Ours**
$O(M^2L)$	$O(NML)$	$O(WNM)$	$O(NM\log(L))$	$O(NM)$

Table 3. Time complexity comparisons.

pixel image pair. The processing times for other image resolutions are given in Table 2. We note that in this computation our method computes disparity for two views jointly, rather than just one. The most time-consuming component of our algorithm is the sparse linear solver, which can potentially be accelerated by $5\times$ via GPU[2].

Our technique is relatively efficient compared to previous adaptive occlusion handling methods based on global optimization, as shown in Table 3. With the image resolution denoted by M, the discrete disparity search range as L, and the iteration number as N, the computational time complexity of graph-cut optimization is $O(M^2L)$ [5] and the computational time complexity of belief propagation is $O(NML)$ [10]. In practice, the computational complexity of sparse QR factorization is linear on average to the number of nonzero entries of A in Eq. (19) [9]. While computational complexity is also dependent on matrix structure, we note that the structure of A is relatively simple, as it is built from three submatrices, two of which are diagonal, with the other being symmetric with two entries per row

[2]cuSPARSE:`https://developer.nvidia.com/cuSPARSE`.

Algorithm	Rank	Tsukuba			Venus			Teddy			Cones			Avg.
		nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	
Our method	12	7.09 10	7.61 10	15.3 24	1.85 21	2.35 20	6.57 11	9.88 23	16.3 26	22.1 19	6.55 36	13.2 40	13.5 32	10.2
LLR [31]	19	7.55 18	8.63 22	14.1 14	6.25 70	7.13 79	13.0 77	9.02 19	15.8 25	20.8 15	6.46 33	13.0 36	13.0 26	11.2
VarMSOH [4]	27	7.18 16	8.56 21	20.1 87	1.46 17	2.12 18	7.87 21	12.9 67	19.4 68	27.5 71	6.22 29	12.6 31	15.8 56	11.8
SymBP+occ [26]	87	20.7 114	21.6 114	19.5 79	5.96 58	6.27 55	10.2 32	15.7 97	20.9 87	31.7 110	11.4 103	17.5 101	18.9 92	16.7
OutlierConf [28]	104	24.7 145	25.0 137	17.4 56	8.01 107	8.27 99	13.7 89	15.6 96	20.5 83	28.7 80	10.9 98	17.3 97	17.4 81	17.3

Table 4. Evaluation of our method on the Middlebury benchmark for 0.5-pixel accuracy. The table entries show the percentage of inaccurately estimated pixels and the rank (blue) in the benchmark. The rightmost column lists the average percentage of incorrect pixels.

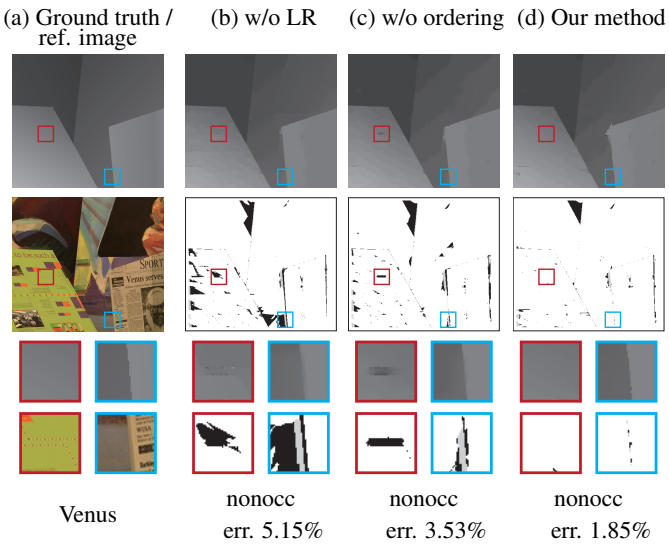
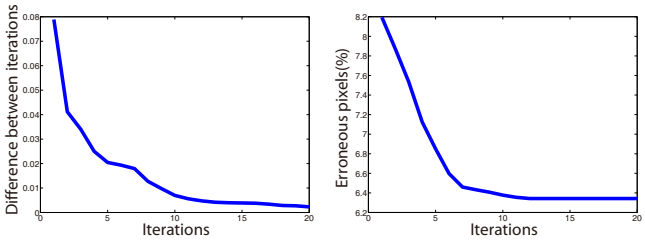


Figure 6. Evaluation of left-right consistency and ordering constraints. (a) ground truth and reference image. (b) only ordering constraints. (c) only left-right consistency constraints. (d) with both constraints. The left-right consistency and ordering constraints significantly improve the results in regions with occlusions (blue box), without texture, and with repeated patterns (red box).

(see Eq. (19)). This linear relationship is also experimentally reflected in the timings reported in Table 2. The time complexity of our optimization, $O(MN)$, where N is less than 10, represents a significant speed-up over graph cut and belief propagation. The complexity of LLR [31], which has an optimization framework similar to ours, is a function of the window size W(*e.g.*, 5×5, or 7×7) for their smoothness term. By contrast, the soft and adaptive confidence weights of our method allow for a similar quality of results with smoothness computed among only 4-connected neighbors, which leads to a more than $10\times$ speed up over LLR. Besides the previous occlusion handling methods, our method is also more efficient than approaches which are slightly more accurate than ours. For example, PM-Huber [15] takes 2 mins with GPU acceleration, which is slower than our CPU implementation. The time complexity of PMF [18] is



(a) (b)

Figure 8. Convergence analysis. (a) Average difference in disparity estimates between each iteration. (b) Average percentage of error pixels.

$O(NM\log(L))$ which will be inefficient when L is large.

Although our optimization framework does not guarantee convergence, all image pairs converge within 10 iterations as mentioned before, and we have not observed oscillatory behavior in our experiments. Figure 8 provides a basic convergence analysis on the Middlebury datasets. We plot the average difference in disparity estimates between two consecutive iterations in Figure 8 (a), and the average percentage of pixel errors[3] in Figure 8 (b). The two curves are both monotonically decreasing, which indicates convergent behavior in practice.

6. Conclusion

In this paper, we presented a continuous stereo matching algorithm with soft and adaptive handling of outliers. For efficient accounting of left-right consistency and ordering constraints, we proposed continuous outlier confidence weights that are iteratively updated as the matching results are gradually refined. An empirical analysis of convergence and complexity are presented, and our experiments support the proposed approach in comparison to related methods. Because of its combination of accuracy and speed, we believe that our method would be well-suited to applications involving large-scale data, such as 3D reconstruction of urban scenes.

[3] Averaged over the Tsukuba, Venus, Teddy and Cones datasets.

Acknowledgements

This work was partially supported by NSFC (No. 61272305) and the National Program for Special Support of Eminent Professionals. Lap-Fai Yu is supported by the University of Massachusetts Boston StartUp Grant P20150000029280 and by the Joseph P. Healey Research Grant Program provided by the Office of the Vice Provost for Research and Strategic Initiatives & Dean of Graduate Studies of the University of Massachusetts Boston.

References

[1] Middlebury stereo vision page. `http://vision.middlebury.edu/stereo/`.

[2] P. Belhumeur. A bayesian-approach to binocular stereopsis. 19(3):237–260, August 1996.

[3] P. Belhumeur and D. Mumford. A bayesian treatment of the stereo correspondence problem using half-occluded regions. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 506–512, Jun 1992.

[4] R. Ben-Ari and N. Sochen. Stereo matching with mumford-shah regularization and occlusion handling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(11):2071–2084, 2010.

[5] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1998.

[6] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, 2001.

[7] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(3):500–513, 2011.

[8] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, 2002.

[9] T. A. Davis. Algorithm 915, suitesparseqr: Multifrontal multithreaded rank-revealing sparse qr factorization. *ACM Trans. Math. Softw.*, 38(1):8:1–8:22, 2011.

[10] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *Int'l Journal of Computer Vision*, 70(1):41–54, 2006.

[11] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *Int'l Journal of Computer Vision*, 40(1):25–47, 2000.

[12] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *Proc. of Asian Conf. on Computer Vision (ACCV)*, 2011.

[13] D. Geiger, B. Ladendorf, and A. L. Yuille. Occlusions and binocular stereo. *Int'l Journal of Computer Vision*, 14(3):211–226, 1995.

[14] K. He, J. Sun, and X. Tang. Guided image filtering. In *Proc. of European Conf. on Computer Vision (ECCV)*, 2010.

[15] P. Heise, S. Klose, B. Jensen, and A. Knoll. Pm-huber: Patchmatch with huber regularization for stereo matching. In *Proc. of Int'l Conf. on Computer Vision (ICCV)*, pages 2360–2367, 2013.

[16] H. Hirschmuller. Evaluation of cost functions for stereo matching. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[17] S. B. Kang and R. Szeliski. Extracting view-dependent depth maps from a collection of images. *Int'l Journal of Computer Vision*, 58:139–163, 2004.

[18] J. Lu, H. Yang, D. Min, and M. N. Do. Patch match filter: Efficient edge-aware filtering meets randomized search for fast correspondence field estimation. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, CVPR '13, pages 1854–1861, Washington, DC, USA, 2013. IEEE Computer Society.

[19] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 42:577–685, 1989.

[20] J. Park, H. Kim, Y.-W. Tai, M. Brown, and I. Kweon. High quality depth map upsampling for 3d-tof cameras. In *Proc. of Int'l Conf. on Computer Vision (ICCV)*, 2011.

[21] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[22] D. Scharstein. Learning conditional random fields for stereo. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

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

[24] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[25] C. Strecha, R. Fransens, and L. Van Gool. Combined depth and outlier estimation in multi-view stereo. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[26] J. Sun, Y. Li, S. B. Kang, and H.-Y. Shum. Symmetric stereo matching for occlusion handling. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 399–406, 2005.

[27] L. Wang and R. Yang. Global stereo matching leveraged by sparse ground control points. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[28] L. Xu and J. Jia. Stereo matching: An outlier confidence approach. In *Proc. of European Conf. on Computer Vision (ECCV)*, pages 775–787, 2008.

[29] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(3):492–504, 2009.

[30] K.-J. Yoon and I. S. Kweon. Adaptive support-weight approach for correspondence search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):650–656, Apr. 2006.

[31] S. Zhu, L. Zhang, and H. Jin. A locally linear regression model for boundary preserving regularization in stereo matching. In *Proc. of European Conf. on Computer Vision (ECCV)*, pages 101–115, 2012.