Published in final edited form as:

Int J Adv Comput Technol. 2012 December ; 4(22): 36–47. doi:10.4156/ijact.vol4.issue22.5.

Bayesian Stereo Matching Method Based on Edge Constraints

Jie Li¹, Wenxuan Shi², Dexiang Deng³, Wenyan Jia⁴, and Mingui Sun⁵

Jie Li: jielonline@gmail.com; Wenxuan Shi: shiwxdsp@gmail.com; Dexiang Deng: whuddx@gmail.com; Wenyan Jia: jiawenyan@gmail.com; Mingui Sun: drsun@pitt.edu

¹School of Electronic Information, Wuhan University, China

²School of Electronic Information, Wuhan University, China

³School of Electronic Information, Wuhan University, China

⁴Department of Neurosurgery, University of Pittsburgh, USA

⁵Departments of Neurosurgery, Electrical engineering and Bioengineering, University of Pittsburgh, USA

Abstract

A new global stereo matching method is presented that focuses on the handling of disparity, discontinuity and occlusion. The Bayesian approach is utilized for dense stereo matching problem formulated as a maximum a posteriori Markov Random Field (MAP-MRF) problem. In order to improve stereo matching performance, edges are incorporated into the Bayesian model as a soft constraint. Accelerated belief propagation is applied to obtain the maximum a posteriori estimates in the Markov random field. The proposed algorithm is evaluated using the Middlebury stereo benchmark. Our experimental results comparing with some state-of-the-art stereo matching methods demonstrate that the proposed method provides superior disparity maps with a subpixel precision.

Keywords

MAP-MRF; local affine model; edge classify; belief propagation

1. Introduction

Binocular stereo vision infers 3D scene geometry from a pair of images with different viewpoints. Dense stereo matching is a key computational step for 3D reconstruction and has been one of the most active research topics in the stereo vision field. Given a pair of images, the aim of stereo matching is to determine the disparity values of the pixels belonging to the image selected as reference view. However, dense stereo matching is difficult for the following reasons: (1) light variations, optical blurring, and sensor noise corrupts the input images, (2) lacking texture in input images may lead to ineffectiveness in using intensity consistency constraints, (3) disparity discontinuities, caused by sudden disparity changes, often appear at object boundaries and are difficult to detect; the

smoothness constraint may break-down in these boundaries, and (4) some portions of objects are visible by one camera but not by the other camera (occlusion), causing difficulties in computing reasonable disparity values in affected regions.

In general, depending on whether the matching method relies on local window-based computation or the minimization of a global energy function, dense stereo matching can be broadly classified into two categories: local methods and global methods. In local methods, the disparity computed at a given point depends only on intensity values within a finite window. Many effective local algorithms for stereo matching have been reported [1–3]. However, when local structures in an image are similar, it may be very difficult to find their correspondences in the other image without global reasoning. Although a high accuracy is not guaranteed in local methods, they have advantages in computational speed and parallelism over global methods.

A stereo algorithm belongs to global methods if there is a global cost function to be optimized, which considers the correlation of disparities in the neighborhood. Hence, the key of global algorithms is not only to define a good cost function but also to provide an effective computational method for global optimization. Among various global methods, the Bayesian approach [4] treats the stereo matching problem as finding the “best guessed” solution. Depending on the computation model, this approach can be classified into two categories: dynamic programming-based and MRFs-based.

The dynamic programming method [5] assumes that the occlusion cost is identical in each scanline. Ignoring the dependency between scanlines leads to the well-known “streaking” artifacts. In contrast, the MRFs-model based methods rely on energy minimization algorithms, such as Graph Cut (GC) [6], and Belief Propagation (BP) [4]. Both of these algorithms can be implemented efficiently to compute the minimum for an MAP-MRF whose energy function is Potts or Generalized Potts [4]. Experiments have shown [7] that, the solution produced by GC tends to be smoother than BP, but BP is significantly faster than GC, even up to real-time [8], while achieving a similar performance with GC.

Traditional global algorithms [4][6][9] usually give good results in most regions of an image. However, in the regions containing occlusions and disparity discontinuities, disparity errors often increase substantially. Hence, explicit smoothness assumptions have to be incorporated in these particular regions. Segment-based priors have been widely explored [4][10–13] due to their first-rank performance. The matching results greatly depend on whether the segmentation results are coincident in the locations where the occlusions and disparity discontinuities take place. But the consistency can hardly be achieved when tackling the highly-textured areas or the nonplanar surfaces with a uniform color. This problem can be solved by planar fitting after segmentation with impressive results [10][11]. However, the computational complexity increases.

Different from the methods mentioned above, our algorithm emphasizes on the edge information of the images. Assuming that discontinuities and occlusions occur on some edges, we explore the characteristics of the normal edges and the disparity discontinuities in reference image of the input image pairs. The edge information is incorporated into our

MAP-MRF stereo matching model to obtain better stereo matching results. Another contribution of the proposed algorithm is that a local affine model is used to aggregate the matching costs. The aggregated matching costs are updated based on the current understanding of which pixels in the reference image are occluded.

This paper is organized as follows: Section 2 gives a mathematic model of the proposed algorithm. In Section 3, the framework of the proposed algorithm is described. Section 4 presents experimental results in which our algorithm and other algorithms are compared based on images in the Middlebury dataset [14]. Finally, conclusion and the future work are discussed in Section 5.

2. Related Works

In this section, we review the related models, the Maximum a posteriori Markov Random Field (MAP-MRF) model and the local affine model, which are used in the proposed algorithm.

2.1 MAP-MRF Model

The Bayesian approach provides a promising way for solving the problems of stereo matching, that is to find the best disparity map D given the observations I_L and I_R , according to maximizing the posterior probability $P(D|I_L, I_R)$. By Bayes' rule $P(D|I_L, I_R) \propto P(I_L, I_R|D)P(D)$, where $P(D)$ is called the prior. Accordingly, the disparity problem can be modeled as pair-wise Markov Random Field (MRF), encoding the smoothness. If $N(p)$ is the neighborhood of pixel p , pixel q belongs to $N(p)$, and disparities of p and q are d_p and d_q , respectively. According to the Markovian property and Gibbs distribution, the basic Bayesian model can be expanded as:

$$P(D|I_L, I_R) \propto \prod_p \exp(-\Phi(d_p)) \prod_p \prod_{q \in N(p)} \exp(-\psi(d_p, d_q)) \quad (1)$$

where $\psi(d_p, d_q)$ penalizes the different assignments of neighbor sites when no discontinuity exists between them. $\Phi(d_p)$ is the matching cost function of pixel p with disparity d_p . If (1) is written as a Gibbs distribution with energy function: $E(D|I_L, I_R) = -\log(P(D|I_L, I_R))$, then, (1) has the familiar energy form:

$$E(D|I_L, I_R) = E_{data} + E_{smooth} \quad (2)$$

where E_{smooth} is defined to measure the disparity smoothness between neighboring segment pairs, and E_{data} measures the disagreement of segments and their matching regions based on the assumed disparity planes. Depending on the different solving methods, we can maximize the posterior probability $P(D|I_L, I_R)$ or minimize the energy function $E(D|I_L, I_R)$.

Sun *et al* [4] incorporated segmentation results into (1) as soft constraints under a probabilistic framework. They point out that low-level visual cues (e.g. edge, corner) can also be incorporated into the basic Bayesian model. Motivated by Sun's method, we incorporate the edge classification result into (1) as a soft constraint. In this way, the probabilistic framework becomes:

$$P(D|I_L, I_R) \propto \prod_p \exp(-\Phi(d_p)) \prod_p \prod_{q \in N(p)} \exp(-\psi(d_p, d_q)) \exp(-\rho_{edge}(d_p, d_q)) \quad (3)$$

where $\rho_{edge}(d_p, d_q)$ is the edge constraint between neighbor sites, which can be defined as:

$$\rho_{edge}(d_p, d_q) = \begin{cases} 0, & edge(p) = edge(q) \\ \lambda_{edge}, & edge(p) \neq edge(q) \end{cases} \quad (4)$$

where $edge(p)$ is a label, which is used to judge whether the pixel p is in a discontinuity and λ_{edge} is a constant. In general, the larger the λ_{edge} is, the more difficult to pass the message between neighbor sites.

2.2 Local Affine Model

An aggregation step is needed to aggregate each pixel's matching cost over a weight region to reduce the matching ambiguities and noises in the initial cost volume. In order to deal with disparity discontinuities and ambiguous regions (textureless areas, repetitive patterns, etc), an ideal cost aggregation strategy should modify its support at each position according to image content so that only those points with the same (unknown) disparity are included. Different from the traditional weight-based aggregation [10], which tries to change the color weight and distance weight to complete weight calculation, the proposed method uses a local affine model to aggregate the cost, which has been proven useful in image denoising [15] and image matting [16][17]. Assume there are two small local matching regions (Ω_p and Ω_q) centered on pixel p and pixel q in the two rectified images L and R (obtained from input pair I_L and I_R), respectively, an affine relation between the intensities of the local matching region is described as:

$$R_{k'} = M_p L_k + B_p, \quad k \in \Omega_p, k' \in \Omega_q \quad (5)$$

where M_p and B_p are linear coefficients which are constants in Ω_p . Assuming that the left image is the reference image, we can get the disparity information from $D_k = R_{k'} - L_k$. Combining with (5), D_k is given by

$$D_k = R_{k'} - L_k = M_p L_k + B_p - L_k = A_p L_k + B_p \quad (6)$$

with $A_p = M_p - 1$. Equation (6) gives the relationship between the disparity image D and input image L in local region Ω_p . Additionally, the local affine model satisfies $D_k = A_p \nabla L_k$, which ensures that the image D has an edge if image L also has an edge. In this way, disparity discontinuities are also parts of edges in the reference image.

In order to obtain A_p and B_p , we minimize the difference between input and output disparities p_k and disparity D_k respectively, using a least-square optimal estimator in a local window:

$$E(A_p, B_p) = \min \left(\sum_{k \in \Omega_p} ((A_p L_k + B_p - p_k)^2 + \varepsilon A_p^2) \right) \quad (7)$$

where p_k is a pixel in the window Ω_p in an initial input disparity map, denoted by $\hat{D}$. ε is a factor which prevents A_p from being too large. A guided filter (GF) has been utilized to solve (7) using a least-squares estimator [15]. The optimal A_p and B_p are given by:

$$A_p = \frac{\frac{1}{|\Omega_p|} \sum_{k \in \Omega_p} L_k p_k - \mu_p \bar{p}_p}{\sigma_k^2 + \varepsilon}, \quad B_p = \bar{p}_p - A_p \mu_p \quad (8)$$

where μ_p and σ_k^2 are the mean and variance, respectively, of input image L in window Ω_p . $|\Omega_p|$ is the number of pixels in Ω_p , and $\bar{p}_p$ is the mean of p in Ω_p . Since p is included in many windows centered by its neighborhood, correspondingly, there are many A_p and B_p in different windows including p . Hence, the GF simply averages all possible values of D_k by

$$\bar{D}_k = \bar{A}_k L_k + \bar{B}_k, \quad \bar{A}_k = \frac{1}{|\Omega_k|} \sum_{p \in \Omega_k} A_p, \quad \bar{B}_k = \frac{1}{|\Omega_k|} \sum_{p \in \Omega_k} B_p. \quad (9)$$

Additionally, from the (9), we found that all the summations can be calculated by box filtering. Therefore, these terms can be computed within the time proportional to N , i.e., $O(N)$ time.

In the proposed algorithm, we expand (9) to disparity space image. In addition, we use a color image as the reference image in order to obtain a more reliable data term. Then, (9) is rewritten as:

$$D_k(x, y) = \sum_{i \in \Omega_k} (\mathbf{A}_k^T \mathbf{L}_i + B_k) \quad (10)$$

where the bold faced $\mathbf{L}_i$ and $\mathbf{A}_k$ are both 3×1 vectors, and $\mathbf{A}_k^T \mathbf{L}_i = A_k^r L_i^r + A_k^g L_i^g + A_k^b L_i^b$, with r, g and b denoting color components.

3. Proposed Algorithm

In order to ease description, we partition our algorithm into three modules: initialization, global optimization and enhancement.

In the initialization module, the initial matching cost is first computed. Then, the reference image and local affine model are used to obtain aggregation matching costs. This process will be applied twice, because both the left and right input images are needed to be reference images. The initial data term $\mathcal{A}(d_p)$ is computed from the aggregation matching cost. Hence, this step generates the following outputs: the initial left disparity map D_L , initial right disparity map D_R , initial disparity image space C_D^0 , aggregated disparity image space C_D^1 , and data term $\mathcal{A}(d_p)$. Another important output of the initialization module is the labels of the edge classification from which ρ_{edge} is incorporated as the constraint in the basic Bayesian model. In the global optimization module, the method of accelerated belief propagation is employed for iterative optimization that trades off between the data term and the prior term. The prior is decreased at discontinuities using the edge constraint, since the

discontinuities are likely to coincide with edges in the reference image. Finally, the goal of the enhancement module is to improve the disparity result in a subpixel level.

3.1 Initialization

Step1: Matching cost computation—Typically, a matching cost is computed at each pixel for all disparities under consideration, which constructs the disparity space image. Recently, Klaus proposed a simple self-adapting dissimilarity measurement which linearly combines SAD and a gradient for computing the matching cost [11]. Considering both the good performance and simplicity of Klaus' method, in this paper, this measurement is used for constructing the disparity space image.

Assume that the left image is a reference image, C_{CSAD} is the absolute color difference and C_{CGRAD} is the color gradient difference. The matching cost $C(x, y, d)$ can be defined by

$$C(x, y, d) = (1 - \alpha)C_{CSAD}(x, y, d) + \alpha C_{CGRAD}(x, y, d) \quad (11)$$

where C_{CSAD} and C_{CGRAD} are given by:

$$\begin{aligned} C_{CSAD}(x, y, d) &= \sum_{(i,j) \in \Omega(x,y)} \min((\sum_{c \in \{r,g,b\}} |I_l^c(i, j) - I_r^c(i+d, j)|) / 3, \tau_c), \\ C_{CGRAD}(x, y, d) &= \sum_{(i,j) \in \Omega(x,y)} \min(|\nabla_x I_l(i, j) - \nabla_x I_r(i+d, j)| + |\nabla_y I_l(i, j) - \nabla_y I_r(i+d, j)|, \tau_g) \end{aligned} \quad (12)$$

where $C(x, y, d)$ is the matching cost of pixel (x, y) in the disparity d , α balances the color and gradient terms, ∇_x and ∇_y represent the horizontal and vertical gradients, respectively, and τ_c and τ_g are truncation values. In order to suppress noise, a small supporting window $\Omega(x, y)$ centered at (x, y) is used to calculate the matching cost. After this step, the initial disparity space image C_D^0 is constructed by $C(x, y, d)$ for all the pixels in the reference image.

Step2: Aggregation—To obtain more accurate results in both smooth and discontinuous regions, the local affine model is used to initialize a reliable matching cost. The local affine model can be expanded to the RGB space as well by rewriting (10) as:

$$C_{new}(x, y, d) = \bar{\mathbf{A}}^T(x, y) \mathbf{I}(x, y) + \bar{\mathbf{B}}(x, y) \quad (13)$$

(x, y) and $\bar{\mathbf{B}}(x, y)$ are given by:

$$\begin{aligned} \bar{\mathbf{A}}(x, y) &= \frac{1}{|\Omega|} \sum_{(i,j) \in \Omega(x,y)} \mathbf{A}(i, j), \quad \mathbf{A}(x, y) = \frac{\frac{1}{|\Omega|} \sum_{(i,j) \in \Omega(x,y)} \mathbf{I}(i, j) C(i, j, d) - \mu(x, y) \bar{\mathbf{C}}(x, y, d)}{\Lambda + \varepsilon \mathbf{U}}, \\ \bar{\mathbf{B}}(x, y) &= \frac{1}{|\Omega|} \sum_{(i,j) \in \Omega(x,y)} \mathbf{B}(i, j), \quad \mathbf{B}(x, y) = \bar{\mathbf{C}}(x, y, d) - \bar{\mathbf{A}}^T(x, y) \mu(x, y). \end{aligned} \quad (14)$$

In (13) and (14), $\mathbf{I}$ is a 3×1 vector, μ is a mean and Λ is a 3×3 covariance matrix, respectively, of $\mathbf{I}$ in local window Ω . $\mathbf{A}$ and $\bar{\mathbf{A}}$ are 3×1 coefficient vectors, $\mathbf{U}$ is a 3×3 identity matrix. $C_{new}(x, y, d)$ constructs the new disparity space image C_D^1 , which will replace the initial disparity space image C_D^0 .

In a similar manner, we compute the disparity space image for the right reference image. When two disparity space images are obtained, the Winner-Take-All (WTA) [1] strategy is applied to form two initial disparity maps D_L and D_R .

In a reliable disparity map, any stable pixel is expected to satisfy the mutual consistency condition, which requires the pixels on the pixel grids in the left and right disparity maps to be perfectly consistent with each other (i.e., having the same disparity value). This is performed by a subsequent mutual consistency check (often called left-right checking [1]) that divides all the pixels into stable or unstable pixels:

$$D_L(x_l) - D_R(x_l - D_L(x_l)) \leq T_{lf} \quad (15)$$

where T_{lf} is a threshold. We mark pixels in the left disparity map as occluded pixels if (15) does not hold true.

After the left-right checking step, the data term can be defined by the occluded pixels and stable pixels as:

$$\Phi(x, y, d) = \begin{cases} C_{new}(x, y, d), & (x, y) \in \text{stable} \\ 0, & (x, y) \in \text{occluded} \end{cases} \quad (16)$$

The constant 0 in (16) reflects the fact that the occluded pixels need the most regularization.

Step3: Edge classification—This step is based on the assumption that an intensity edge in an image indicates a depth discontinuity in the scene. In our case, the edges in the reference image are detected by the Sobel edge detector [18]. If D_k is the mean disparity value of the window Ω_k centered at pixel k , and i is a pixel in the neighborhood of pixel k with disparity D_i , then, the standard deviation (Std) is computed to estimate the edge variation in the disparity map D_L :

$$Std(k) = \sqrt{\sum_{i \in \Omega_k} (D_i - \bar{D}_k)^2}, i \in \text{Sobel edge}. \quad (17)$$

Hence, if $Std(k)$ is smaller than the threshold λ , the edge belongs to a normal edge. Otherwise, the edge is a discontinuity. The edge classification is thus given by

$$edge(k) = \begin{cases} 0, & Std(k) < \lambda \\ 1, & Std(k) > \lambda \end{cases} \quad (18)$$

Fig. 1 provides an example of edge classification using (17) and (18). It can be observed that most disparity discontinuities are successfully identified.

3.2 Global Optimization

The data term $\Phi(d_p)$ is determined by updating the matching cost in eq.(16), and the prior term $\psi(d_p, d_q)$ in (3) is defined by the Potts model. Assuming that two neighboring pixels p

and q are likely to have the same disparity if their intensities satisfies $I(p) \approx I(q)$. This contextual information is incorporated into $\psi(d_p, d_q)$, which is computed in the same fashion as that in [7]:

$$\psi(d_p, d_q) = \begin{cases} 0, & d_p = d_q \\ \beta, & \text{otherwise} \end{cases}, \quad \beta(\Delta I) = \begin{cases} s, & \text{otherwise} \\ P \times s, & \text{if } \Delta I < T \end{cases} \quad (19)$$

where T is a gradient threshold, and ΔI is the difference between I_p and I_q , s is a penalty term for violating the smoothness constraint and P is a penalty term when the gradient has a small magnitude. Note that T , P and s are constants over the whole image. Assuming that there is a matching cost, denoted by C , in order to use belief propagation, C should be converted to a compatibility form by calculating e^{-C} . For numerical reasons (e.g. e^{-C} is extremely small), we include a positive constant M and define the compatibility as $e^{-C/M}$.

Belief Propagation (BP) is an iterative inference algorithm that propagates messages in a network. Among several high performance algorithms, we adopted the max-product BP algorithm [4][7] which works by passing messages within a graph defined by an image with four-connected pixel grid. Each message is a vector of the dimension given by the number of possible labels. During a iteration, each node uses the messages received in the previous iteration from neighboring nodes. A new message is then calculated and sent to its neighbors.

A message update scheme determines when a message is sent to a node where it will be used to compute subsequent messages for neighbors of the node. One of the update schemes is to propagate messages in one direction and update each node immediately. In this work, we use the “accelerated” BP updating scheme (seen in Fig. 2) proposed by Tappen *et al* [7]. The “up-down-left-right” message passing scheme enables the BP algorithm to converge rapidly.

Let us consider the case where node p is located to the right side of node q (Fig. 2(a)). Let $m_{right}^t(d_q)$ be the message that node p sends to q in the current iteration t , which contains node p 's belief about each possible state of node q . This message is computed from the messages that p has received from its neighbors (up, down, right) at iteration $t-1$. Then, the new message (for the right side case) is updated as:

$$m_{right}^t(d_q) \leftarrow \max_{d_p} \psi(d_p, d_q) \times \Phi(d_p) \times m_{right}^{t-1}(d_p) m_{up}^{t-1}(d_p) m_{down}^{t-1}(d_p) \quad (20)$$

where $m_{right}^{t-1}(d_p)$, $m_{up}^{t-1}(d_p)$ and $m_{down}^{t-1}(d_p)$ are the messages received by p from the nodes' right, above and below at iteration $t-1$. The new updated left, above and below messages (Fig. 2(b)–(d)) are computed similarly to Eq.(20). After iteration T , the beliefs at node q can be computed by

$$b(d_q) = \Phi(q, d) m_{left}^T(d_p) m_{right}^T(d_p) m_{up}^T(d_p) m_{down}^T(d_p). \quad (21)$$

Then, the best disparity d_q^{MAP} in pixel q is obtained from the maximum belief:

$$d_q^{MAP} = \operatorname{argmax}_{d_q} (b(d_q)). \quad (22)$$

3.3 Enhancement

To reduce the discontinuities error caused by the quantization effect in the disparity, a method producing subpixel precision is proposed. Assuming that d_c is the disparity after accelerated BP. The final disparities d_f can be achieved by:

$$d_f = d_c + 0.5 * \frac{C(x, y, d_c - 1) - C(x, y, d_c + 1)}{C(x, y, d_c - 1) + C(x, y, d_c + 1) - 2C(x, y, d_c)}. \quad (23)$$

Since a disparity map has been produced from the belief propagation module and the matching cost of the three candidates $C(x, y, d_c - 1)$, $C(x, y, d_c + 1)$ and $C(x, y, d_c)$ are obtained from the aggregated disparity space image in the aggregation step, the enhanced disparity map can be easily computed.

Our algorithm was summarized in Fig. 3.

4. Experimental Results

In this section, we evaluate the performance of our stereo matching algorithm by the Middlebury stereo benchmark dataset [14]. This dataset consists of four pairs of images: *Tsukuba*, *Venus*, *Teddy* and *Cones*. Our experimental study includes the following components: parameters setting, evaluation of results, and comparison with several state-of-the-art stereo matching methods [10][11][13][19][20].

4.1 Parameter Setting

We first describe the settings of parameter used in our algorithm. Note that the parameters are constant across all datasets. In order to improve stability, we normalize the pixel intensity to [0,1].

Cost computation—The computation of matching cost is parallel at each pixel and each disparity level. To obtain appropriate thresholds, we analyze the parameters statistically by experiments shown in Fig. 4. Then, the truncation values τ_c and τ_g are set, respectively, to 0.0028 and 0.016 in (12). The balance term α is set to 0.92. Ω is chosen to be a 3×3 local window.

Cost aggregation—We use the Eq. (13) for the aggregation. The parameters are chosen empirically as follows: Ω_k is a 15×15 window and ϵ is 0.0001.

Occlusion detection and edge constraint—Since there is a certain error in computation, we choose T_{lf} to be 1 in Eq. (15). After experiments, λ in Eq.(18) is set to 0.18 in order to distinguish the disparity discontinuity from normal edges.

Belief Propagation—Parameters s , T and P in Eq. (19) of the BP module are set to 0.000153, 0.06 and 8, respectively. The constant M is set to 0.1961 in the global optimization module of section II. The iteration time is set to 5 by experiments.

Fig. 4 shows the disparity maps obtained by our algorithm using the above parameters. The results after different intermediate stages provided visual explanations that how the different stages in the algorithm improve the results. The ground truths are also given for comparison.

4.2 Evaluation

The performance of stereo matching algorithms can be evaluated quantitatively. Two methods are often utilized based on the RMS (root-mean-squared) error and the percentage of bad matching pixels with respect to the ground truth data included in the Middlebury datasets [1]. In this work, the second method is utilized. First, we compute B which reflects the percentage of bad matching pixels:

$$B = \frac{1}{N} \sum_{(x,y)} (|d_c(x,y) - d_T(x,y)| > \delta_d) \quad (24)$$

where δ_d is the disparity error tolerance (Table 1 list results for $\delta_d=0.5$, 0.75 and 1). The result of each image pair in the Middlebury set is computed by measuring B in pixels of three regions: nonoccluded pixels (denoted by **noc**), discontinuity pixels (denoted by **disc**) and either nonoccluded or half-occluded (denoted by **all**). The value of B in the whole disparity image is considered (the second column marked by **Ave. Error** in Tables 1 and 2). It can be seen that the result of image pair *Venus* is most accurate when $\delta_d=1$ (Table 1 in the rows with bold faces). This is because our algorithm performs well when the scene is mainly composed of planar objects. Note that, in both tables, the subscript of each number denotes the rank in the Middlebury stereo benchmark which contains approximately 140 algorithms.

4.3 Comparison

Table 2 summarizes the quantitative performance of our method and those of other global BP stereo methods [10][11][13][19][20]. The results are ranked roughly in descending order of overall performance. The evaluation of our final results after disparity enhancement is presented in the first row. Additionally, the evaluation results with and without edge constrain incorporated into stereo matching are shown in the second and third rows (in bold face), respectively. After incorporating the edge constraints, the matching result is improved, which is indicated by smaller average error in Table 2. Note that our algorithm produces the best performance in subpixel level (e.g. $\delta_d=0.5$), and the rank is up to 5 in approximately 140 algorithms.

The *Tsukuba* image pairs contain some dark and noisy regions near the lamp and the desk, which usually lead to incorrect support regions for aggregation. Our method performs very well in *Tsukuba*, including high texture parts (such as the lamp stem in Figure 4.(f)) and textureless parts (such as the box, desk). That is because our method preserves edges while smoothing the relatively flat regions.

Although a reliable local affine model prevents edge smoothing obtaining an accurate affine model is very difficult. Therefore, in the proposed algorithm, we just average possible local affine model parameters, which lead to the inequality $\nabla D_k < \nabla L_k$ in high texture regions. In this case, certain details are lost, which can be seen in *Tsukuba* of Figure 4(c) where the lamp holder appears to be broken.

Additionally, our algorithm performs well when the scene is mainly composed of planar objects. However, if the scene is mainly composed of smooth, curved surfaces, the performance of the proposed algorithm may decrease. This is because the smoothness prior used in the proposed method is the first-order prior [21][22], which indeed favors low-curvature fronto-parallel surfaces. Even in man-made scenes (seen in the “stair-case” in *Venes* of Fig. 4(g)), the prior is maximized by fronto-parallel planes, leading to inaccurate depth estimates.

In the left-right checking stage, all unstable disparities are replaced by 0. Since the noise causes inconsistency in the left and right disparity maps, some discarded unstable pixels may have a correct disparity. Therefore, it can be explained why the error increases after left-right checking (seen in Fig. 4(c)–(d)).

5. Conclusion

In this paper, stereo matching is formulated as a Bayesian inference problem. The edge information is integrated into the basic MAP-MRF stereo model as a soft constraint. Then, the accelerated belief propagation algorithm is adopted to this MAP-MRF problem effectively. In order to obtain more reliable cost volume, a local affine model is used to aggregate the stereo matching process. Our experimental results have demonstrated high performance of the proposed algorithm. Additionally, our algorithm performs well when the scene is mainly composed of planar objects. However, the performance of the proposed algorithm may decrease if the scene is mainly composed of smooth and curved surfaces. Hence, combining more visibility reasoning and second-order smoothness in an optimization framework will be considered in the future work.

Acknowledgments

This research was supported by the Chinese National Natural Science Foundation under grant No.61072135 and National Institutes of Health grants U01HL91736 and R01CA165255 of the United States.

References

1. Scharstein, Daniel; Szeliski, Richard. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, Springer. 2002; 47(1–3):7–42.
2. Yoon, Kuk-Jin; Kweon, In-So. Adaptive Support-Weight Approach for Correspondence Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE. 2006; 28(4):650–656.
3. Rhemann, Christoph; Hosni, Asmaa; Bleyer, Michael; Rother, Carsten; Gelautz, Margrit. Fast cost-volume filtering for visual correspondence and beyond. *Proceedings of International Conference on Computer Vision*; 2011. p. 3017–3024.
4. Sun, Jian; Shum, Heung-Yeung; Zheng, Nan-Ning. Stereo Matching using Belief Propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE. 2003; 25(7):787–800.

5. Veksler, Olga. Stereo Correspondence by Dynamic Programming on a Tree. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*; IEEE. 2005. p. 384-390.
6. Boykov, Yuri; Veksler, Olga; Zabih, Ramin. Fast Approximate Energy Minimization via Graph Cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE. 2001; 23(11):1222–1239.
7. Tappen, Marshall F.; Freeman, William T. Comparison of Graph Cuts with Belief Propagation for Stereo, using Identical MRF Parameters. *Proceedings of IEEE International Conference on Computer Vision*; IEEE. 2003. p. 900-907.
8. Yang, Qingxiong; Wang, Liang; Yang, Ruigang; Wang, Shengnan; Liao, Miao; Nistér, David. Real-time Global Stereo Matching Using Hierarchical Belief Propagation. *Proceedings of British Machine Vision Conference*; Springer. 2006. p. 989-998.
9. Felzenszwalb, Pedro F.; Huttenlocher, Daniel P. Efficient Belief Propagation for Early Vision. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*; IEEE. 2004. p. 261-268.
10. Yang, Qingxiong; Wang, Liang; Yang, Ruigang; Stewenius, Henrik; Nister, David. Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*; IEEE. 2009. p. 492-504.
11. Klaus, Andreas; Sormann, Mario; Karner, Konrad F. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. *Proceedings of International Conference on Pattern Recognition*; IEEE. 2006. p. 15-18.
12. Zhang, Shuying; Li, Dongmei. Stereo Match Algorithm Based on Image Color Segments. *AISS: Advances in Information Sciences and Service Sciences, AICIT*. 2012; 4(17):519–526.
13. Taguchi, Yuichi; Wilburn, Bennett; Zitnick, Lawrence. Stereo reconstruction with mixed pixels using adaptive over-segmentation. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*; IEEE. 2008. p. 1-8.
14. <http://vision.middlebury.edu/stereo>.
15. He, Kaiming; Sun, Jian; Tang, Xiaoou. Guided Image Filtering. *Proceedings of European Conference on Computer Vision*; Springer. 2010. p. 1-14.
16. Levin, Aant; Rav-Acha, Alex; Lischinski, Dani. Spectral Matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE. 2008; 30:1699–1712.
17. Levin, Anat; Lischinski, Dani; Weis, Yair. A Closed Form Solution to Natural Image Matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE. 2008; 30:228–242.
18. González, Rafael C.; Woods, Richard E. *Digital Image Processing*. Addison Wesley; USA: 1992. p. 414-428.
19. Yang, Qingxiong; Yang, Ruigang; Davis, James; Nistér, David. Spatial-depth super resolution for range images. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*; IEEE. 2007. p. 1-8.
20. Zitnick, Lawrence; Kang, Sing Bing. Stereo for image-based rendering using image over-segmentation. *International Journal of Computer Vision*, Springer. 2007; 75:49–65.
21. Woodford, Oliver J.; Torr, Philip HS.; Reid, Ian D.; Fitzgibbon, Andrew W. Global Stereo Reconstruction under Second Order Smoothness Priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE. 2009; 31(12):2115–2128.
22. Fu, Jibin; Wu, Huixin; Zhai, Zhengang. A Probabilistic Stereo Matching Model for Slanted Surface. *IJACT: International Journal of Advancements in Computing Technology, AICIT*. 2012; 4(15):226–234.

Figure 1.

(a) Left image in a pair of stereo images; (b) Edges detected by Sobel detector; (c) Edges corresponding to disparity discontinuities detected by Eq.(17) and Eq.(18).

Figure 2.

Accelerated BP message updating scheme: (a) from right to left, (b) from left to right, (c) from down to up, (d) from up to down.

Figure 3.
Framework of the proposed method

Figure 4.

Intermediate results of four different standard test images by our algorithm compared to the ground truth. (a) Left images of the stereo image pairs; (b) edge detection result; (c) WTA disparity maps after local affine model; (d) disparity maps after applying left-right checking ; (e) disparity maps after BP propagation; (f) disparity maps after refinement; (g) ground truth.

Table 1

Result of the proposed method with different error thresholds

Error Threshold	Avg. Error	Tsukuba			Venus			Teddy			Cones		
		Noce	All	Disc	Noce	All	Disc	Noce	All	Disc	Noce	All	Disc
$\delta_d = 1$	5.88	1.29 ₃₇	2.08 ₅₂	5.71 ₁₈	0.09 ₃	0.38 ₁₉	1.31 ₄	7.14 ₆₁	13.2 ₆₄	15.7 ₃₅	3.49 ₄₆	9.62 ₅₉	10.3 ₆₄
$\delta_d = 0.75$	8.26	7.15 ₂₃	8.27 ₂₂	12.6 ₇	0.43 ₁₇	0.83 ₂₂	3.62 ₁₆	7.91 ₃₈	14.7 ₅₃	17.6 ₂₅	4.05 ₃₄	10.3 ₄₂	11.6 ₅₀
$\delta_d = 0.5$	10.5	7.15 ₉	8.27 ₁₁	12.6 ₅	3.76 ₂₈	4.34 ₂₇	9.83 ₁₉	10.3 ₁₃	17.5 ₂₄	22.7 ₁₅	4.64 ₆	11.8 ₁₅	13.1 ₁₄

Table 2

Results of Comparison ($\delta_d=0.5$)

Algorithm	Avg. Error	Tsukuba			Venus			Teddy			Cones		
		Noce	All	Disc	Noce	All	Disc	Noce	All	Disc	Noce	All	Disc
Our method(final)	10.5	7.15 ₉	8.27 ₁₁	12.6 ₅	3.76 ₂₈	4.34 ₂₇	9.83 ₁₉	10.3 ₁₃	17.5 ₂₄	22.7 ₁₅	4.64 ₆	11.8 ₁₅	13.1 ₁₄
Our method(cons)	10.7	7.36 ₁₀	8.48 ₁₁	13.8 ₇	3.69 ₂₇	4.27 ₂₇	9.51 ₁₉	10.4 ₁₃	17.4 ₂₂	23.0 ₁₆	4.79 ₈	12.0 ₁₅	13.3 ₁₆
Our method(no)	10.8	7.53 ₁₀	8.73 ₁₃	14.1 ₈	3.72 ₂₈	4.30 ₂₇	9.88 ₁₉	10.3 ₁₃	17.7 ₂₅	23.0 ₁₆	4.93 ₁₀	12.4 ₁₉	13.7 ₁₈
SubPixDoubleBP _[19]	10.7	8.78 ₁₈	9.45 ₁₅	14.9 ₁₀	0.72 ₃	1.12 ₅	5.24 ₃	10.1 ₁₂	16.4 ₁₅	21.3 ₇	8.49 ₄₅	14.7 ₄₃	16.5 ₄₁
AdaptOvrSegBP _[13]	11.9	5.98 ₃	6.56 ₃	9.09 ₁	3.66 ₂₅	3.96 ₂₂	13.2 ₅₂	13.0 ₄₃	18.9 ₃₇	26.4 ₃₅	9.48 ₅₃	14.9 ₄₅	17.2 ₄₈
OvrSegBP _[20]	12.4	7.75 ₁₄	8.17 ₉	13.8 ₇	4.33 ₃₃	4.73 ₃₀	16.8 ₈₄	13.2 ₄₄	19.3 ₄₁	27.5 ₄₀	6.53 ₂₄	12.6 ₂₂	14.0 ₂₁
AdaptingBP _[11]	13.6	19.1 ₆₉	19.3 ₆₅	17.4 ₂₇	4.84 ₃₉	5.08 ₃₅	7.84 ₁₁	12.8 ₃₉	16.7 ₁₇	26.3 ₃₄	7.02 ₃₀	13.2 ₂₈	14.0 ₂₀
DoubleBP _[10]	15.7	18.7 ₆₄	19.1 ₆₂	15.8 ₁₈	7.82 ₇₇	8.22 ₇₃	11.3 ₃₂	14.4 ₅₄	19.9 ₄₇	24.4 ₂₄	11.8 ₇₇	17.6 ₆₈	19.7 ₆₈