Stereo Matching Using Random Walks

Rui Shen, Irene Cheng, Xiaobo Li and Anup Basu
Department of Computing Science, University of Alberta, Canada
{rshen,lin,li,anup}@cs.ualberta.ca

Abstract

This paper presents a novel two-phase stereo matching algorithm using the random walks framework. At first, a set of reliable matching pixels is extracted with prior matrices defined on the penalties of different disparity configurations and Laplacian matrices defined on the neighbourhood information of pixels. Following this, using the reliable set as seeds, the disparities of unreliable regions are determined by solving a Dirichlet problem. The variance of illumination across different images is taken into account when building the prior matrices and the Laplacian matrices, which improves the accuracy of the resulting disparity maps. Even though random walks have been used in other applications, our work is the first application of random walks in stereo matching. The proposed algorithm demonstrates good performance using the Middlebury stereo datasets.

1. Introduction

Stereo vision is a very active research area in computer vision. One crucial and traditional task in stereo vision is stereo matching, the goal of which is to determine the disparities of corresponding pixels in a pair of stereo images. Many algorithms have been proposed to solve this problem. According to the comprehensive survey by Scharstein and Szeliski [8], the existing algorithms can be roughly classified into two categories: local (window-based) algorithms [1] that compute disparities within a finite window, and global algorithms [1] that find the best disparity configuration by minimizing a global energy function. Compared with local methods, global methods produce more accurate results using optimization frameworks, such as graph cuts (GC) [1], belief propagation (BP) [6] and dynamic programming (DP) [7].

A recent application of random walks (RW) in multilabel image segmentation [2][4] shows great potential for its application in stereo matching. As shown in [3], RW demonstrates superior performance over GC in segmentation. The segmentation is achieved by minimizing the Dirichlet integral, which is actually a kind of energy function that can be defined in the form of the typical energy function used by global stereo algorithms:

$$E(f) = E_{data}(f) + \lambda E_{smooth}(f). \tag{1}$$

The major advantage of random walks is that an exact and unique minimum solution of an energy function of the above form can be produced, while other approaches like GC can only produce an approximation. Therefore, it is very likely that RW can achieve better performance than GC and other optimization frameworks in stereo matching. Hence, we develop a new stereo matching algorithm using random walks. Our contribution lies in that our work is the first application of random walks in the area of stereo matching. Two major phases are considered: first, a set of reliable matching pixels is computed using RW with prior models [2] (Section 4); second, with the reliable set serving as seeds, the disparities of unreliable pixels are determined using the original RW [4] (Section 5). As demonstrated in Section 6, our algorithm produces more accurate disparity maps than many of GC-based and DP-based methods, and some of BP-based methods.

2. Random walks

Random walks works on a graph representation $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ of an image, where $\boldsymbol{V}$ is the set of all the pixels and $\boldsymbol{E}$ is the set of weighted edges. Given a set $\boldsymbol{S}$ of k labels, the probability of a random walker starting from an unlabeled node u and reaching a seed (labeled node) is calculated. u is assigned a label s, if it has the highest probability of reaching a seed with label s. The calculation of the probabilities is equivalent to the Dirichlet problem, and the solution can be found by solving the equation:

$$\boldsymbol{L_U x^s} = -\boldsymbol{B^T f^s}, \tag{2}$$

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

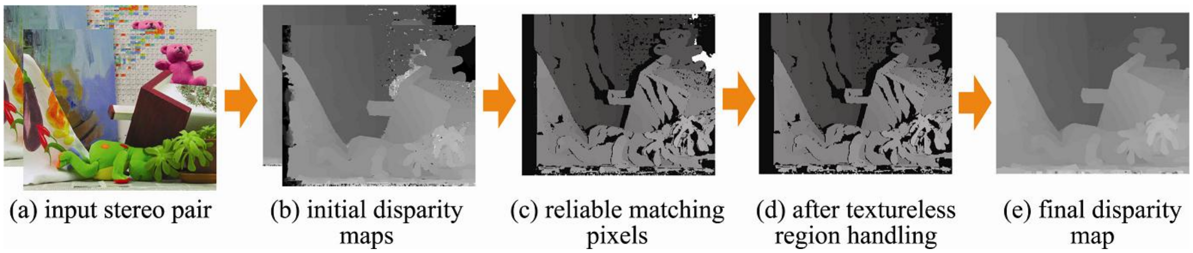

Figure 1. Processing procedure of the proposed algorithm.

where x^s is the vector to be calculated that represents the probability of each unlabeled node being assigned label s. L_U is a submatrix of the Laplacian matrix L, containing weights of the edges connecting unlabeled nodes. B^T is another submatrix of L, containing weights of the edges connecting a seed and an unlabeled node. f^s is a binary function defined on V and S.

Prior terms can also be incorporated using the improved RW algorithm [2]. In addition to L, a set of priors $\{\lambda_i^s\}$ is added to represent the probability that a node v_i belongs to label s. Then, the segmentation (labeling) problem can be formulated as minimizing an energy function of the form:

$$E^s_{total} = E^s_{spatial} + \gamma E^s_{aspatial}. \tag{3}$$

Note the resemblance of Equation 3 to Equation 1. The solution is obtained by solving the equation:

$$(L + \gamma \sum_{r=1}^{k} \Lambda^r) x^s = \lambda^s, \tag{4}$$

where $\Lambda = diag(\lambda^s)$. We term the matrix containing all the priors as the prior matrix $P = [\lambda_i^s]$.

3. Algorithm overview

Like graph cuts [1], we define the set of labels as the set of all possible disparities between $[0, d_{max}]$. Thus, the stereo matching problem is formulated in the RW framework as finding the minimum of Equation 3. The whole procedure is illustrated in Figure 1. An initial disparity estimation for both images (I_l and I_r) is obtained by solving Equation 4. Left-right checking (Section 4.1) is applied to determine a set of reliable matching pixels. Unreliable pixels are marked in black in Figure 1(c)(d). Special consideration is given to textureless regions (marked in white in Figure 1(c)). Using the intermediate result, an optimal disparity map is computed by solving Equation 2. Between each step, the intermediate disparity maps are smoothed (*e.g.,* using a median filter) to reduce noise. Our algorithm works in the YC_bC_r space, and the parameters are adjusted automatically according to the illumination condition of the input image pair.

4. Reliable matching pixel computation

In order to formulate the stereo matching problem in the RW framework, the Laplacian matrix L and the prior matrix P need to be built first. The weight of an edge connecting pixel p_i and pixel p_j is defined as:

$$w_{ij} = e^{-\beta \|p_i - p_j\|} + \epsilon, \tag{5}$$

where $\|p_i - p_j\|$ is the Euclidean distance between p_i and p_j in the YC_bC_r space, and ϵ (we set $\epsilon = 10^{-5}$) is a small positive constant. In practice, $\|p_i - p_j\|$ is normalized between $[0, 1]$. β is a weighting factor to reduce the effect of variance of illumination across different images. Intuitively, the lower the brightness of an image, the smaller the difference between similar pixels in the YC_bC_r space; thus, a larger β is needed to emphasize the difference. We use the average luminance value Y_{avg} as the measurement of the brightness of an image. Hence, β is adjusted according to the equation:

$$\beta = \max(\beta_{ini} - \delta_\beta Y_{avg}, \theta_\beta), \tag{6}$$

where we set $\beta_{ini} = 295, \delta_\beta = 380, \theta_\beta = 0.1\delta_\beta$. After calculating all the edge weights, the Laplacian matrix L is built using the equation from [4]:

$$L_{ij} = \begin{cases} \sum_k w_{ik}, & i = j; \\ -w_{ij}, & p_i \text{ and } p_j \text{ are adjacent;} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

To build the prior matrix P, a prior λ_i^d is defined as:

$$\lambda_i^d = \begin{cases} \Theta, & min_{d \in [0, d_{max}]}\{C(p_i, d)\} \geq \theta_1; \\ e^{-\sigma(0.1\theta_2)}, & C(p_i, d) \leq \theta_2; \\ e^{-\sigma C(p_i, d)}, & otherwise. \end{cases} \tag{8}$$

where we take $\theta_1 = 0.03, \theta_2 = \max\{0.1Y_{avg} - 0.04, 10^{-4}\}, \Theta = 150$. $C(p_i, d)$ is defined in a similar way as in [1] using the pixel dissimilarity measure:

$$C(p_i, d) = \min\{C_{fwd}(p_i, d), C_{rev}(p_i, d)\}. \quad (9)$$

In practice, $C(p_i, d)$ is normalized between $[0, 1]$. We found that $C_{rev}(p_i, d)$ only calculated for the reference image is sufficient for a final disparity map with comparable quality. Like β, σ is determined by:

$$\sigma = \max(\sigma_{ini} - \delta_\sigma Y_{avg}, \theta_\sigma), \quad (10)$$

where we set $\sigma_{ini} = 499, \delta_\sigma = 720, \theta_\sigma = 0.1\delta_\sigma$.

Given $\boldsymbol{L}$ and $\boldsymbol{P}$, the probability x_i^d of a pixel being assigned a disparity $d \in [0, d_{max}]$ can be calculated by solving Equation 4, where we set $\gamma = 10^{-5}$. Pixel p_i is assigned the disparity d with the highest probability, *i.e.,* $D(p_i) = \arg\max_d(x_i^d), \forall d$.

4.1. Left-right checking

A left-right checking technique similar to the one used in [7] is applied to extract a set $\boldsymbol{M}$ of reliable matching pixels. The remaining pixels are categorized into the unreliable set $\boldsymbol{U}$. Since the left image $\boldsymbol{I_l}$ is used as the reference image, this reliable set is only determined on $\boldsymbol{I_l}$. For two matching pixels $(p_l, d_l) \in \boldsymbol{I_l}$ and $(p_r, d_r) \in \boldsymbol{I_r}$ in the two images (*i.e.,* $p_r.x = p_l.x + d_l$ and $p_r.y = p_l.y$), the reliability of (p_l, d_l) is determined by:

$$(p_l, d_l) \in \begin{cases} \boldsymbol{M}, & d_l \in [-d_r - \theta_3, -d_r]; \\ \boldsymbol{U}, & \text{otherwise.} \end{cases} \quad (11)$$

where we take $\theta_3 = \lceil \frac{d'_{max}}{7} \rceil$ with d'_{max} as the maximum disparity in the initial disparity maps. Pixels with $p_l.x \leq \lfloor \frac{d'_{max}}{2} \rfloor$ are directly labeled as unreliable without left-right checking, because they are occluded in $\boldsymbol{I_r}$.

4.2. Textureless region handling

Textureless regions tend to cause wrong disparity estimation. Due to the high similarity of a pixel to its neighbouring pixels, a small disparity is very likely to be assigned while the true disparity may be much larger. Therefore, after left-right checking, every pixel (p_b, d_b) on the left boundary of a region $\boldsymbol{R} \subset \boldsymbol{M}$ with disparity $d \leq \theta_3, \forall(p, d) \in \boldsymbol{R}$ is selected to go through a disparity validation test. The validation procedure is simply a linear search from $\max\{p_b.x - 1, \lceil \frac{d'_{max}}{2} \rceil\}$ to $\max\{p_b.x - d'_{max}, \lceil \frac{d'_{max}}{2} \rceil\}$ to find the matching pixel p_m in $\boldsymbol{I_r}$ with the largest disparity d_m that satisfies:

$$\|p_b - p_j\| \leq \theta_2, \forall p_j(p_j.x \in [p_b.x - d_m, p_b.x - 1]). \quad (12)$$

If p_m is found, $d_b = d_m$; otherwise, d_b is unchanged. The remaining pixels of $\boldsymbol{R}$ are categorized into $\boldsymbol{U}$.

5. Unreliable pixel labeling

The disparity assignment of unreliable pixels $\boldsymbol{U}$ based on the disparities of reliable pixels $\boldsymbol{M}$ is exactly the same as the Dirichlet problem defined in the original RW framework [4]. $\boldsymbol{f_i^d}$ is defined as:

$$f_i^d = \begin{cases} 1, & ((p_i, d_i) \in \boldsymbol{V}) \bigwedge (d_i = d); \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Matrices $\boldsymbol{L_U}$ and $\boldsymbol{B^T}$ are obtained by reformatting $\boldsymbol{L}$ as:

$$\boldsymbol{L} = \begin{pmatrix} L_M & B \\ B^T & L_U \end{pmatrix}. \quad (14)$$

After this step, the disparities of unreliable pixels are computed by solving Equation 2.

6. Experimental results & discussion

Our algorithm was evaluated using the Middlebury test bed (http://vision.middlebury.edu/stereo/) designed by the authors of [8]. A set of constant parameters is used for all images. The current ranking of our algorithm is 17th among 41, and it is compared with top three algorithms in Table 1. Each decimal entry from the third column to the last column represents the percentage of bad pixels for different regions with absolute disparity error larger than one pixel. The disparity maps (top row) are shown against the ground truths (bottom row) in Figure 2. The computational complexity of RW is $O(n)$ as given in [2]. As our algorithm mainly works by solving two consecutive Dirichlet problems using the RW framework, its computational complexity is also $O(n)$. Our algorithm is currently implemented in Matlab. The computational time ranges from 17 (Tsukuba) to 41 seconds (Teddy) on a 2.41GHz Athlon 64 dual-core computer with 2GB memory. The most time-consuming part is building $\boldsymbol{P}$, which takes between 4 (Tsukuba) and 21 seconds (Teddy). A C++ or GPU (graphics processing unit) implementation would be much faster. Besides this, solving the Dirichlet problems takes between 11 (Tsukuba) and 17 seconds (Teddy). A GPU implementation of random walks was proposed in [5], which demonstrated an approximately 10-fold speedup. Therefore, we expect at least a 10-time speedup once our algorithm is implemented in GPU, which will be significantly faster than many other global methods using a different optimization framework (*e.g.,* [1][6][7]).

Table 1. The Middlebury stereo evaluation results of the proposed algorithm.

Algorithm	Avg. Rank	Tsukuba			Venus			Teddy			Cones		
		nonocc	*all*	*disc*	*nonocc*	*all*	*disc*	*nonocc*	*all*	*disc*	*nonocc*	*all*	*disc*
DoubleBP2	2.8	0.88⋆ 1∗	1.29 1	4.76 1	0.13 3	0.45 5	1.87 5	3.53 2	8.30 3	9.63 1	2.90 3	8.78 7	7.79 2
AdaptingBP [6]	3.0	1.11 7	1.37 3	5.79 8	0.10 1	0.21 2	1.44 1	4.22 4	7.06 2	11.8 4	2.48 1	7.92 2	7.32 1
DoubleBP	4.8	0.88 2	1.29 2	4.76 2	0.14 5	0.60 12	2.00 7	3.55 3	8.71 5	9.70 2	2.90 4	9.24 10	7.80 3
...	...	...	...	...	...	...	...	...	...	...	...	...	...
Proposed Algorithm	**17.2**	**0.96** 4	**1.47** 5	**4.91** 3	**1.41** 27	**1.71** 23	**3.46** 16	**7.83** 19	**13.2** 17	**17.8** 18	**5.76** 28	**11.6** 25	**11.8** 22

∗: The subscripted integers are the ranks of each algorithm in each column; ⋆: percentage (%) of bad pixels.

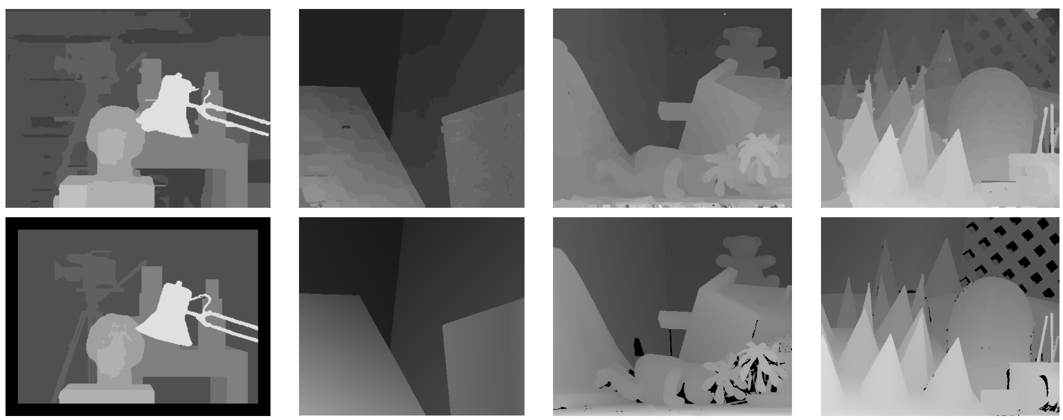

Figure 2. Dense disparity maps for the Middlebury stereo datasets. Our results are in the top row and the ground truths are in the bottom row.

7. Conclusion

In this paper, we not only proposed a new algorithm for stereo matching, but also demonstrated the feasibility of a new framework, random walks, for solving the stereo matching problem. The major parameters (β and σ) used in our algorithm are adapted automatically according to the illumination condition of the input image pair. The evaluation results of our algorithm using the Middlebury datasets demonstrated promising performance. One limitation of our algorithm is that it works in the YC_bC_r space, which limits it to color images. In the future, we plan to incorporate segment-based methods (*e.g.,* [6]) to label a more reliable set of initial matching pixels, which will be used to initialize the random walks optimization procedure. This combination, together with GPU implementation, will achieve a much more accurate and faster disparity estimation.

References

[1] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE PAMI*, 23(11):1222–1239, 2001.

[2] L. Grady. Multilabel random walker image segmentation using prior models. In *Proceedings of CVPR'05*, pages 763–770, 2005.

[3] L. Grady. Random walks for image segmentation. *IEEE PAMI*, 28(11):1768–1783, 2006.

[4] L. Grady and G. Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In *Proceedings of ECCV'04 Workshops on CVAMIA and MMBIA*, pages 230–245, 2004.

[5] L. Grady, T. Schiwietz, S. Aharon, and R. Westermann. Random walks for interactive organ segmentation in two and three dimensions: implementation and validation. In *Proceedings of MICCAI'05*, pages 773–780, 2005.

[6] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Proceedings of ICPR'06*, pages 15–18, 2006.

[7] C. Lei, J. Selzer, and Y.-H. Yang. Region-tree based stereo using dynamic programming optimization. In *Proceedings of CVPR'06*, pages 2378–238, 2006.

[8] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1-3):7–42, 2002.