A Multiple Graph Cut Based Approach for Stereo Analysis

Ulas Vural and Yusuf Sinan Akgul

GIT Vision Lab, Department Of Computer Engineering
Gebze Institute Of Technology, Cayirova, Gebze, Kocaeli 41400, Turkey
{uvural, akgul}@bilmuh.gyte.edu.tr

Abstract. This paper presents an optimization framework for the 3D reconstruction of the surfaces from stereo image pairs. The method is based on employing popular graph cut methods under the dual mesh optimization technique. The constructed system produces noticeably better results by running two separate optimization processes that communicate with each other. The communication mechanism makes our system more robust against local minima and it produces extra side information about the scene such as the unreliable image sections. We validated our system by running experiments on real data with ground truth and we compared our results with the other optimization methods, which showed the accuracy and effectiveness of our method.

1 Introduction

The classical breakdown of 3D surface recovery from stereo suggests that first the correspondences between the image pairs should be established and then the 3D surface is reconstructed using these correspondences[9]. The newer techniques take the approach of a global solution by incorporating the correspondence and the 3D reconstruction steps into the same process. This process is larger and more complex but the results are far better than the classical methods if the problem complexity is addressed properly. One common method to manage this larger problem is to pose it as an energy optimization task. An energy functional that penalizes locally unsmooth and discontinuous 3D structure is formulated. Optimization of this functional on the stereo image pairs would produce the desired 3D surface. Despite the elegance and unified nature of such systems, optimizing these functionals are not trivial. The problem is fundamentally NP-Hard and the approximation methods are sensitive to initializations, local minima, and image noise.

Recently, graph cut methods gained popularity in optimizing energy functionals of Computer Vision problems. Graph cuts can guarantee optimal functional values for some restricted cases[10][8]. For the other cases, they guarantee an upper bound in error from the optimal result[6]. Furthermore, since they are based on the deep theory of graph and flow algorithms, there are numerically stable and efficient algorithms for performing cuts[4]. Although the types of energy functionals that can be optimized by graph cuts are limited[13][7], the limitations are

K. Franke et al. (Eds.): DAGM 2006, LNCS 4174, pp. 677–687, 2006.

not very restrictive. As a result, graph cuts were applied to many stereo problems including multi-camera scene reconstruction[12], occlusion detection[11], and stereo with plane fitting and layering[3].

Energy optimization with the dual mesh approach was proposed for depth estimation from stereo pairs[2] and tracking of ultrasound tongue sequences[1]. The dual mesh method is a framework that employs two instances of a known energy optimization method. It works on the principle of two simultaneous and interacting optimization processes. The energy optimizations start from the two ends of the search space and the optimizations continue until they find the same position in the search space. The interaction between the optimization processes is used to force the mesh with the high energy towards the other. The dual mesh method was shown to be relatively insensitive to local minima due to its two-way sweep of the search space. It does not have any initialization problems. However, the system can only be used for continuous depth recovery and it might be sensitive to local minima depending on the optimization methods used.

In this paper, we describe a system that uses graph cut energy optimization methods under the framework of dual mesh optimization. The system introduces a number of novel enhancements to both graph cuts and dual mesh framework to achieve a noticeably better energy optimization which results in more accurate 3D surface recovery. The system eliminates the dependency on the initial configuration because the initializations are done exactly the same way for all system input. The proposed system produces other important side information about the 3D scene such as the unreliable image parts for 3D reconstruction without any additional computational load.

Section 2 formally introduces the dual mesh energy. The details of graph cut optimization under the dual mesh framework is explained in Section 3. Experiments and validation work is discussed in Section 4. Section 5 concludes the paper.

2 The Dual Mesh Energy

A deformable mesh is a set of horizontally and vertically connected points in 3D space. Each point m_{ij} is a mesh element and the mesh elements form a 3D surface or set of 3D surfaces. The mesh elements have fixed x and y positions. The z positions of the mesh elements can change. The z position of a mesh element m_{ij} is also called the depth value of the element and it is given by the function $depth(m_{ij})$. A deformable mesh is positioned in a 3D volume and it interacts with the contents of the volume to localize any desirable 3D surface while maintaining the surface properties such as local continuity and smoothness. The movements of the deformable mesh is governed by an energy functional, which forces the deformable mesh to move towards the 3D surface positions that overlap with the existing real world surfaces. For our system, the energy functional of the deformable mesh M is dependent on the mesh N and it is written as

$$E_{Mesh}(M,N) = \sum_{i=1}\sum_{j=1} E_{Data}(m_{ij}) + E_{Smoothness}(m_{ij}) + E_{Tension}(m_{ij}, n_{ij}) \tag{1}$$

The term $E_{Smoothness}(m_{ij})$ is for satisfying the smoothness constraint of the mesh. Regularization based approaches or using convex functions as smoothness term extend smoothness everywhere. Although these kinds smoothness terms makes the resulting systems more efficient and robust against noise, they do not work well at the object boundaries. For example, [2] can only recover continuous surfaces because it uses such a smoothness term. Non-convex functions can preserve discontinuities but they are more sensitive to local minima and optimizing such functions require more computational power.

Potts style smoothness terms are very simple but effective. They preserve discontinuities and their computational complexity is fair. We use a Potts style smoothness term to keep the discontinuities with a four-neighborhood system.

$$E_{Smoothness}(m_{ij}) = V(m_{ij}, m_{i+1j}) + V(m_{ij}, m_{ij+1}) \tag{2}$$

where

$$V(m_{ij}, m_{kl}) = \begin{cases} 0 & depth(m_{ij}) = depth(m_{kl}), \\ \lambda_1 & |depth(m_{ij}) - depth(m_{kl})| \leq thresh_1, \\ \lambda_2 & otherwise. \end{cases}$$

The tension energy is not always active. It is a mechanism that the dual mesh optimization framework employs to communicate information between the two separately deforming meshes. Under this framework, the functionals of two meshes are optimized separately at different initial 3D positions and it is expected that local minima, occlusions, and discontinuities will prevent them finding the same position at the end of the optimizations. When this happens, the tension term is activated to push the mesh with the worse position towards the other mesh. The details of this term is explained in section 3. We again use a Potts based model for the tension term.

$$E_{Tension}(m_{ij}, n_{ij}) \\ = \begin{cases} 0 & depth(m_{ij}) = depth(n_{ij}) \\ \infty & depth(m_{ij}) > depth(n_{ij}) \\ \lambda_3 * (depth(n_{ij}) - depth(m_{ij})) & (depth(n_{ij}) - depth(m_{ij})) \leq thresh_2, \\ \lambda_3 * thresh_2 & otherwise. \end{cases}$$

Note that due to the working mechanism of the dual mesh framework, the elements n_{ij} of mesh N normally cannot have depth values less than the depth values of m_{ij} of mesh M.

Data energy is the term responsible for the deformation of the deformable mesh with the scene data. If the stereo pair is viewing a volume V (Fig. 1), and if the point P in this volume is visible from both cameras, then classical stereo analysis states that the image points p_l and p_r on the left and right images should belong to similar image regions. Therefore, the Sum of Squared Differences (SSD) between the local neighborhoods around the corresponding image points is a good data energy term. For a given mesh element m at 3D point P in volume V (Fig. 1), the data energy term is written as

$$E_{Data}(m) = \sum_i \sum_j (IL_{ij} - IR_{ij})^2, \tag{3}$$

where IL_{ij} and IR_{ij} are the left and right image neighborhoods around the image points p_l and p_r of point P. Note that m does not have to be on a real surface in volume V. For any 3D position inside V the above data term can be calculated. If m is on a real world 3D surface, it is expected that the data energy term stays smaller.

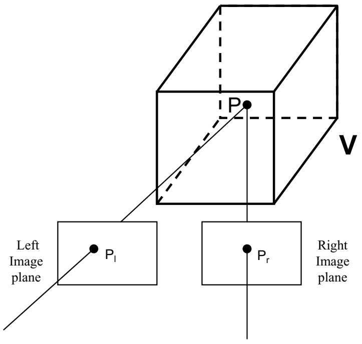

Fig. 1. A volume V is viewed by a stereo camera system

3 Graph Cut Based Dual Mesh Optimization for 3D Surface Recovery

Classically, stereo can be viewed as the problem of assigning a depth value label for each pixel p in the images. The depth value labels are chosen to be integers representing the distance between the image planes and the 3D point whose projection on the image plane is the pixel p (Fig. 1). There are only a finite number of labels. Therefore, we can formally define this labeling in terms of depth value sets. Let L and R be two sets of pixels in the left and right images respectively on the same epipolar line pair, and let D be the set of possible depth values. For any pixel p_l in set L there is a corresponding depth value label in set D that ties p_l to the pixel p_r in set R, which contains only the epipolar conjugate pixels of L. Note that choosing a label from set D for a pixel p_l is equivalent to choosing a corresponding pixel p_r for p_l and vice versa. A labeling f represents the complete matching of all pixels in the set L to their labels in the set D.

It is possible to minimize only the data energy term (Equation 3) to find the D labels by choosing the p_l and p_r pixels that maximize the data energy term. This mapping would be in polynomial time complexity. However, this mapping would also produce a rough depth map even at good image regions because of the very local decision base. We therefore need to figure out a more global formulation for the labeling problem.

3.1 Graph Cut Optimization

A more robust labeling can be achieved with the help of the deformable mesh structure defined in Section 2. The elements of the mesh would represent the pixels and their depths in a stereo image. Therefore, the x and y positions of the mesh elements cannot change. The z position of the mesh would represent the estimated depth value of the pixel. The labels that minimize the associated mesh energy would produce a 3D surface or surface set that would satisfy smoothness and good SSD values between the pixel correspondences. The brute force implementation of the above deformable mesh optimization is NP-hard, hence an approximation method is required. Recently, graph cuts became popular in approximate function optimization after the introduction of α-expansion algorithm[6], which proves the existence of an upper bound on error from the optimal result.

Graph cut based optimization methods need a special type of graph constructed first. We add a new node to the graph for each mesh element and since each mesh element represents a pixel, the special graph has a node for each pixel in the image. Two terminal nodes named the source (s) and the sink (t) are also added to the graph. Each node in the graph is connected to the terminal nodes with links called $t-links$. The weight of a $t-link$ is chosen as the E_{Data} term of the mesh element corresponding to the node. Semantically, any graph node connected to s node has the depth label of s. Similarly, nodes connected to t node has the depth label of t. All pixel nodes are connected to their neighbors by n-links with the cost $E_{Smoothness}$ of the pixel node(Fig. 2-a). The initial labeling of the graph is called f_0.

The α-expansion algorithm uses z position values of the volume as the depth labels. For each depth label, a new graph is constructed with the sink node representing the current configuration and the source node representing the new depth label. Note that every time the graph is reconstructed, weights of the links are recalculated. The α-expansion algorithm performs an $s-t$ cut on this graph by using one of the max flow/min cut algorithms from the literature. After the partitioning, the nodes will have only one t-link, which means that each node will have only one label(Fig. 2-b). In other words, after one α-expansion some mesh elements can change their labels to α, and the others will keep their labels. Once all depth labels are tried as the source node, an iteration is completed. The α-expansion continues with other iterations until there is no improvement in the total energy.

The above method was shown to be a very effective approximation method in assigning labels to pixels when there is a special type of energy functional involved. However, since it is still an approximation method, a better approximation is always helpful for a number of applications including stereo analysis. The next section explains how we use the dual mesh framework to achieve a better approximation.

3.2 Graph Cuts Under Dual Mesh Framework

Dual mesh optimization framework[2,1] was originally developed for finding the approximate optimal values of contour positions in ultrasound or depth labels

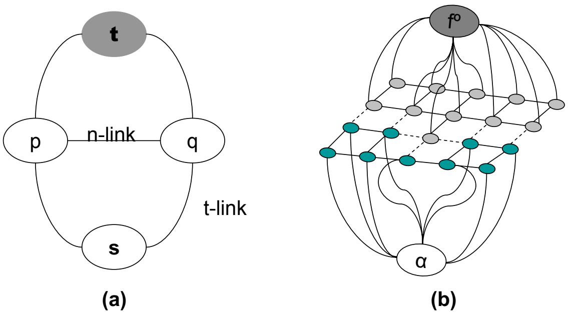

Fig. 2. (a) The α-expansion special graph with two nodes p and q. (b) The new labeling of the nodes after one α-expansion step.

in stereo. The basic idea of dual mesh optimization is to pose the problem as a label assignment problem for each pixel or contour element. The continuity and smoothness of the labels are satisfied by optimizing an energy functional of a deformable mesh that assigns a depth position for each mesh element. The main argument of dual mesh approach is to employ two separate deformable mesh structures and initialize them at the opposite ends of the search space of the labels. By employing known optimization methods from the literature, the mesh energies are minimized separately and they start deforming independently(Figure 3). The deformable meshes usually stop deforming at different 3D positions due to local minima, which is a common problem in optimizing complex energy functionals. The dual mesh framework addresses the local minima problem by pushing the deformable mesh with the larger energy towards the other mesh. The biggest advantage of the dual mesh structure is that it can employ any optimization method to optimize each deformable mesh and the resulting 3D mesh positions will be better than what that specific optimization method can achieve.

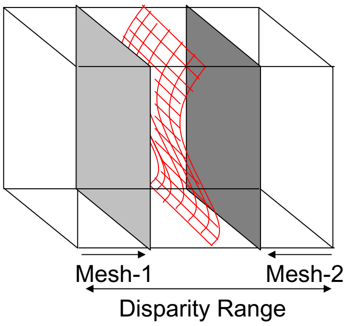

Fig. 3. Two deformable mesh structure localizing the same 3D surface

We borrow the idea of the dual mesh framework to use it with graph cut based optimization methods. The dual mesh approach provides a facility for the graph cut method to improve the results by showing a better direction of deformations which cannot be achieved by the graph cut method itself. The proposed optimization method stages are as follows:

- Start two graph cut optimization processes to minimize the dual mesh energies defined by Equation 1. The first optimization will use the maximum possible depth values as the initial label (mesh N). The other optimization will use the minimum possible depth values as the initial label (mesh M). The $E_{Tension}$ term of the energy will not be used at this phase.
- After the optimization, take the final labeling of the corresponding mesh elements m_{ij} and n_{ij} in the two meshes and activate the $E_{Tension}$ term for each mesh element that has different labeling in the other mesh. The $E_{Tension}$ element is activated by adding its cost to the $t - link$ of the mesh element.
- The above step will bias the mesh elements of the two meshes to find the same depth positions. However, it is not guaranteed that they will find the same positions due to other energy terms.
- If all of the two mesh elements find the same depth positions, then the optimizations ends. Otherwise, we deactivate the $E_{Tension}$ term and start doing the same steps until convergence occurs or we see no improvement in the overall mesh energy.
- At the end of this process, the final labels are assigned to each stereo image pixels. If the corresponding mesh elements m_{ij} and n_{ij} have the same depth label, then pixel p_{ij} is assigned the same label. Otherwise, pixel p_{ij} gets the label of the smaller energy element.

The above procedure has several advantages. First, when compared to the graph cut methods, it produces noticeably closer results to the optimal value. This advantage is expected because we compare two almost identical graph cut optimization processes to bias the estimations towards the better one. The second advantage of this method is that, the corresponding mesh elements m_{ij} and n_{ij} that do not find the same depth positions would give us valuable information about the scene. These kinds of positions are actually problematic for stereo analysis because they correspond to occlusions, depth discontinuities, or textureless image areas. This information is very important in knowing what depth estimation values are more reliable than the others. Finally, unlike the original dual mesh method, our new method allows recovery of discontinuous 3D surface patches due to the employment of the Potts style smoothness function.

4 Experiments and System Validation

We implemented our system by using the graph cut library provided by [13]. We tested our system exhaustively to observe its performance in real world against the other methods and to validate the claims we made. For these experiments, we employed BVZ algorithm [5] as the underlying graph cut method of the dual mesh framework for its simplicity, though we could have used any graph cut method. For all the experiments, we used the stereo data and the ground truth provided by the Middlebury image base[14].

There are three main experiments we performed. First, we compared our results with the popular graph cut systems, BVZ[5], KZ1[12], and KZ2[11]. Table 1 shows that our method is always better than the BVZ, which is the underlying graph cut algorithm for our method. In some cases, the errors get very close to KZ1 and KZ2 algorithms which are much more sophisticated than the BVZ algorithm, which is very encouraging. Note that it is not fair to compare our dual mesh method to KZ1 and KZ2 methods directly because dual mesh method is dependent on the BVZ optimization method. We provided the numbers for the other methods just to show the scale of the difference between methods. We are

Table 1. Comparison of the dual mesh algorithm with other graph cut based algorithms. The numbers are percentage errors compared to ground truth on non-occluded, discontinuous, and all image regions.

	Tsukuba			Venus			Teddy			Cones		
Algorithm	Non-Occ	All	Disc	Non-Occ	All	Disc	Non-Occ	All	Disc	Non-Occ	All	Disc
BVZ	1.96	4.20	9.71	2.03	3.69	12.1	17.3	25.8	28.8	19.2	28.3	25.7
KZ1	1.83	2.48	6.42	1.06	1.52	5.53	12.0	17.9	22.4	5.78	12.9	13.2
KZ2	1.33	2.15	6.94	1.22	1.78	5.99	12.5	18.8	22.1	6.08	13.2	13.3
Dual Mesh	1.91	4.13	9.50	1.64	3.29	10.5	13.0	21.9	25.2	9.37	19.6	17.5

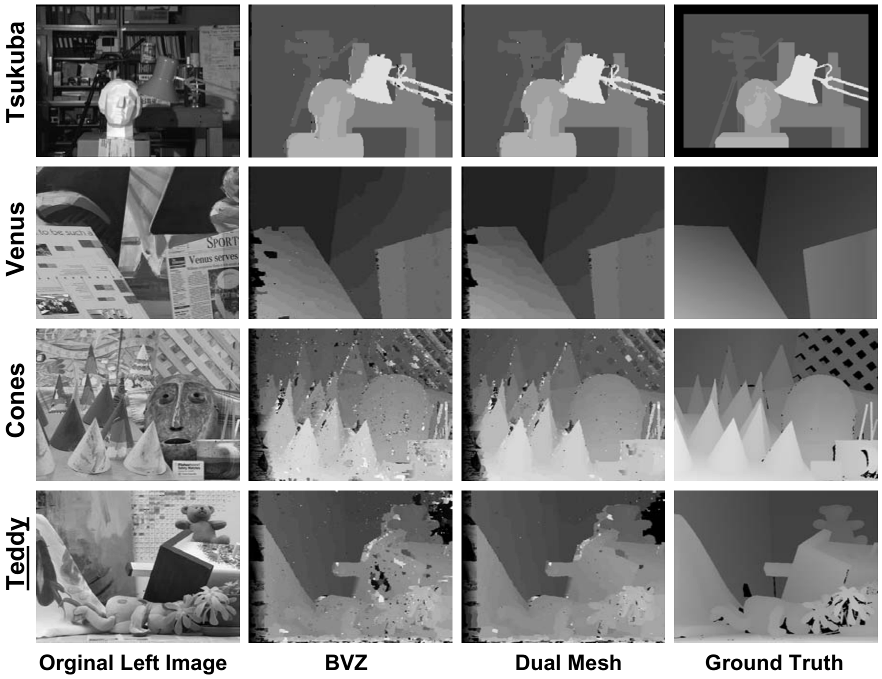

Fig. 4. The dual mesh disparity values compared with the ground truth and BVZ

working on implementing the KZ1 and KZ2 based dual mesh algorithms and we expect that such systems would produce better results than the KZ1 and KZ2 systems. Figure 4 shows the obtained disparity images from our system compared to BVZ method and the ground truth for the images of Table 3. Visual inspection of this data shows the accuracy and effectiveness of our system especially in images with discontinuities, occlusion and textureless regions. Note that we used exactly the same parameter set as the BVZ method when we compare our dual mesh method to BVZ in all the experiments.

For the second experiment, we tested the dual mesh capability of capturing the unreliable image areas. Textureless, occluded and discontinuous regions are problematic parts of the images for stereo analysis. Knowing such regions would make the subsequent processing more convenient. Dual mesh structure can be used for partial detection of these areas by checking the intermediate positions of the mesh elements during the optimization. If there are mesh elements m_{ij} and n_{ij} that do not find the same positions, then we mark these areas as problematic areas. Table 2 shows the percentage of the problematic pixels detected and the overlap of these pixels with the occluded image regions from the ground truth of the Tsukuba image. Notice that 38% of the problematic pixels are occluded. We visually verified that the rest of the problematic pixels are from textureless regions and depth discontinuities. We are working on producing the ground truth data to quantitatively verify this claim.

For the third experiment, we like to show that the results obtained by the dual mesh method cannot be obtained by a single optimization process. We run the BVZ method with the random labeling as suggested by [5]. Due to the randomness factor, we repeated the run 10 times and recorded the best, the worst and the average errors. We also modified the BVZ method so that it takes

Table 2. Detection of unreliable pixels. Occluded pixels are obtained from ground truth.

	Number	Percentage
Pixels	109921	100.0
Problematic Meshels of Dual Mesh	1312	1.19
Problematic And Occluded	504	0.45

Table 3. The effects of using different labeling methods

Algorithm	Non-occluded Areas	All Areas	Discontinuity Areas
BVZ Random (Best)	18.9	28.0	25.5
BVZ Random (Avg)	19.3	28.4	25.7
BVZ Random (Worst)	19.6	28.6	26.1
BVZ Regular-1 (One Mesh)	19.0	28.1	25.4
BVZ Regular-2 (One Mesh)	18.9	28.0	25.5
Dual Mesh	9.37	19.6	17.5

the same labeling order as our dual mesh labelings. Since we have two separate optimizations, there are two different labeling orders(regular-1 and regular-2). Table 3 shows the percentage errors from each run compared to our dual mesh on the Cones image. We obtained similar results on other images. As the results clearly show, the dual mesh approach produces better results.

5 Conclusions

Graph cut methods gained popularity in optimizing energy functionals of Computer Vision problems due to their effectiveness and closeness to the optimality. We presented an optimization framework that noticeably improves the graph cut based stereo methods. The system is based on deformable dual mesh optimization that employs two graph cut optimization processes. The processes communicate with each other to come up with a better 3D surface that cannot be achieved by a single optimization. Furthermore, the communication mechanism makes our system more robust against local minima. The method can also extract other useful information about the scene such as the unreliable image sections. Although we employed the BVZ algorithm as the underlying graph cut method, our framework can employ any graph cut methods to improve the results.

We validated the system by running several experiments and compared our results with the ground truth and other stereo algorithms. We quantitatively observed that our method noticeably improves the graph cut optimization results. We also visually observed the improvements. Overall, the results are very encouraging.

The system is open to further enhancements. We are working on implementing more sophisticated graph cut algorithms to be used as optimization methods. We are also working on making the system computationally more efficient by using a tighter communication mechanism between the meshes. It is also planned to use this system as a general label assignment method for image/volume segmentation, contour tracking, and motion analysis.

Acknowledgements

This work is supported by TUBITAK Career Project 105E097.

References

1. Y. Akgul, C. Kambhamettu, and M. Stone. A task-specific contour tracker for ultrasound. In *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, 2000.
2. Yusuf Sinan Akgul and Chandra Kambhamettu. Recovery and tracking of continuous 3d surfaces from stereo data using a deformable dual-mesh. In *International Conference on Computer Vision*, pages 765–772, 1999.
3. M. Bleyer and M. Gelautz. A layered stereo matching algorithm using image segmentation and global visibility constraints. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(3):128–150, May 2005.

4. Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In *EMMCVPR02*, page 359 ff., 2002.
5. Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *IEEE Computer Vision and Pattern Recognition or CVPR*, pages 648–655, 1998.
6. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222 – 1239, Nov 2001.
7. D. Freedman and P. Drineas. Energy minimization via graph cuts: settling what is possible. In *IEEE Computer Vision and Pattern Recognition*, pages II:939–946, 2004.
8. D. Greig, B. Porteous, and A. Seheult. Exact maximum a posterori estimation for binary images. *Journal Royal Statistical Society*, B: 51(2):271–279, 1989.
9. B.K.P. Horn. *Robot Vision*. The MIT Press, 1986.
10. H. Ishikawa. Exact optimization for markov random fields with convex priors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(10):1333–1336, October 2003.
11. V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions via graph cuts. In *International Conference on Computer Vision*, pages II: 508–515, 2001.
12. V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *European Conference on Computer Vision*, page III: 82 ff., 2002.
13. V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.
14. D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, April 2002.