

Feature Match for Underwater Image via Superpixel Tracking

Shu Zhang¹, Junyu Dong², Hui Yu^{1,*}

¹School of Creative Technologies, University of Portsmouth, Portsmouth, UK

²Department of Computer Science and Technology, Ocean University of China, Qingdao, China

*hui.yu@port.ac.uk

Abstract— Feature matching is fundamental to many vision tasks. Due to the low visibility of images in underwater environments, traditional pixels-based matching methods suffer from miss-matching or error-matching. Recently, Superpixel based features have been applied to image feature analysis. However, most of existing methods dedicate to rectified stereo matching with images captured in the air. This paper presents a novel feature matching scheme aiming at underwater images. It targets the unrectified image pair from the video sequence. The Superpixel matching process is fulfilled with multiclass labelling based on Markov Random Field (MRF). Experiments show that the proposed method produces competitive performance.

Keywords-component; formatting; style; styling; insert (key words)

I. INTRODUCTION

Feature matching is the basis of many computer vision tasks, such as object tracking, stitching, and 3D reconstruction. It has been well studied for images with fine visibility. However, vision tasks may happen in a condition with low visibility, which presents a real challenge for feature matching. This paper presents a novel algorithm to match features in underwater images with low visibility. We employ superpixel as a higher level representation of image features to deal with low visibility problems underwater.

Superpixel can be generated by image segmentation methods. It has been proven to be effective representations of small regions in an image [16, 11, 22]. Pixels in one superpixel are assumed to come from the same projection of a 3D planar patch [13]. Therefore, superpixel can represent a collection of the pixels with continuous depths. Inspiringly, this paper treats those pixel collections as features to match in the images captured in underwater environment. The rationale is that compared to the pixels, the superpixels can be easily observed in low visibility. Our matching process is based on the MRF using graph based energy minimization. MRF was first introduced by Boykov et al. [5], which has the advantage of efficient convergence [19].

II. RELATED WORK

Feature matching has been a long-studied issue since computer vision became popular. It forms the basis for other algorithms such as tracking, stitching, and 3D reconstruction. The most basic feature matching method is

Harris Corners [7] introduced by C. Harris. Recently, SIFT based feature matching [12] has become the most commonly used algorithm. Those matching methods are all pixel based and confront significant challenges when dealing with the images captured in underwater environment [2]. Meanwhile, higher level representation methods, such as those based on the superpixel, may fulfil the tasks in a more perceptive way [9, 10].

Superpixel-based feature matching has drawn a lot attentions recently. Barnes et al. [20] presented a superpixel matching scheme for low resolution depth generation. They aggregated matching by summing the dissimilarity [4] over a superpixel region. Their algorithm only works on images with fine visibility and on rectified stereo image pair, where the searching path is only on the epipolar line. They cannot deal with the matching problems without extrinsic parameters of the cameras. Hartley et al. [23] also introduced a superpixel matching algorithm using a graph-based superpixel algorithm [21] for superpixel generation. However, their algorithm only works on rectified stereo image pair obtained under a fine visibility condition as well.

III. SUPERPIXEL GENERATING FOR UNDERWATER IMAGE PAIR

Superpixel was first introduced in [16]. A superpixel in an image is a group of pixels that have continuous depths [13]. Therefore, the matched pixel features in the image pairs should also lie in the matched superpixels. We have studied a range of superpixel generating algorithms, including SLIC [15], Lattices [14] and Turbopixels [11]. We choose SLIC proposed by Radhakrishna et al. due to its low computational cost and strong edge response. Based on SLIC, we propose a new superpixel generating process for image pairs by taking into account the motion trend between those two images.

Superpixels are produced by seeds placed over the image pixels. It calculates the similarities between each pixel and its neighbouring seeds according to the color distances and spatial distances. These two types of distances determine which Superpixel region the pixel should belong to, as shown in (1).

$$\begin{cases} D_{total} = \sqrt{D_{lab}^2} + \sqrt{Weight * D_{xy}^2} \\ Weight = ((Step_x / Compactness) * (Step_y / Compactness))^{-1} \end{cases} \quad (1)$$

where D_{lab} is the distance of colors; D_{xy} is the distance of positions; Compactness is a predefined parameter, which

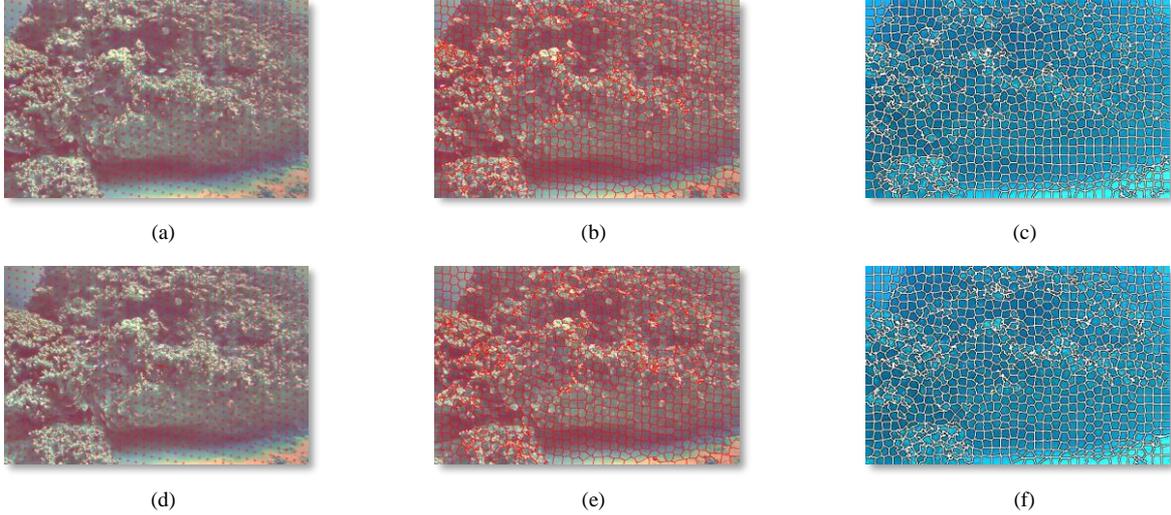


Figure 1: (a) and (d) are two sets of superpixel seeds on two frames generated using the proposed method. These two frames are pre-processed using the gray-world algorithm; (b) and (e) are the superpixels generated from the seeds in (a) and (d); (c) and (f) are the superpixels by the original SLIC from the same image pair. Apparently, the superpixels generated using our method are easier to be matched compared with the traditional ones.

adjusts the proportion of these two distances. $Step_x$ and $Step_y$ are the seeds' intervals along x axis and y axis respectively.

The superpixels produced from an image can be very different if different seeds being given over the pixels. This adds difficulties to the matching process between the image pairs. In the original superpixel generating process, the consistent arrangement of seeds between image pairs makes it hard for superpixels generated across these two frames to remain similar. Therefore, we propose a new algorithm of dynamic self-adaptive seeds arrangement for superpixel generating. It makes the relationship between superpixel seeds and image's pixels remain as still as possible across a pair of images, which adds more similarities between two sets of superpixels generated from these two images, as shown in Fig. 1 (a) and (d).

$$\begin{cases} Step_x = Step * \frac{|x_{Optical_Flow}|}{\sqrt{x_{Optical_Flow}^2 + y_{Optical_Flow}^2}} \\ Step_y = Step * \frac{|y_{Optical_Flow}|}{\sqrt{x_{Optical_Flow}^2 + y_{Optical_Flow}^2}} \\ Step = \sqrt{height_{image} * width_{image} / Count_{superpixel} + 0.5} \end{cases} \quad (2)$$

To achieve the superpixel tracking, the underwater images are firstly pre-processed using the gray world algorithm, and then the optical flow field is calculated from the image pair. The average optical flow vector is finally extracted according to the optical flow field to represent the movement trend between these two image frames. The arrangement of the superpixel seeds is done according to this average optical flow. The proposed method is shown in (2), where $Step_x$ and $Step_y$ are defined as in the (1); $x_{Optical_Flow}$ and $y_{Optical_Flow}$ are the x and y components of the average optical flow respectively. $Count_{superpixel}$ is the total superpixel number in an image, which is 900 in our method. According to (2), we can obtain a superpixel seed field illustrated in Fig. 1 (a) for the first frame. As for the second frame, the seeds field needs a shift along the average optical flow, as shown in Fig. 1 (d). The superpixels generated from these two sets

of seeds are shown in Fig. 1 (b) and (e), which are obviously more match-able than those generated from the traditional superpixel algorithm, as shown in Fig. 1 (c) and (f).

IV. MRF BASED SUPERPIXEL MATCHING

The matching process is based on the intuitive concept of finding similar superpixels. As observed, matched superpixels share similarities with colors, sizes, and neighborhoods. The known motion trend between images will also contribute to the matching. For instance, in traditional stereo matching, the motion is along the epipolar lines, which makes the match restricted along horizontal lines. We transform the matching process into a labelling problem. During the matching, the second frame's superpixel index is treated as the Label (L) of first frame's superpixel index (I). We employ MRF for the proposed algorithm to conduct the labelling process. MRF is suitable for image processing since the content in an image is neighbour-related.

A. Energy function

The energy function consists of the Data cost (D) and Smoothness cost (S). D refers to the energy for superpixel index I with label L assigned. The better L and I is matched, the less D cost is. S refers to the energy that neighbour superpixels' indices are assigned with a pair of labels at the same time.

$$\begin{cases} D_{Total_cost} = \sum_{i=1}^{Count_{superpixel}} D(I_i, L_i) \\ S_{Total_cost} = \sum_{Neighbor(i,j)=true} S(I_i, I_j, L_i, L_j) \\ Energy = D_{Total_cost} + \lambda S_{Total_cost} \end{cases} \quad (3)$$

Intuitively, finding similar superpixels should appeal to the cues that human vision system uses. So we employ superpixels' color, size, position, and neighbor structure into the cost of each label assignment to calculate the total D_{Total_cost} :

- Values from CIELAB color space are used to represent the color cost. CIELAB is a three

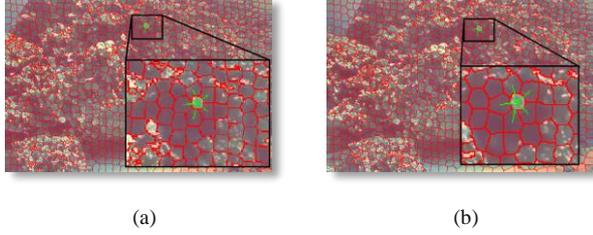


Figure 2: Illustration of matched superpixels' neighbour structure likeness between frames. There are five neighbour superpixels for each.

dimensional lookup table for color definition, where L is about lightness, and A , B are the coordinates of two axes of a predefined color table, as shown in (4).

$$Color_{cost} = \sqrt{(L_i - L_L)^2 + (A_i - A_L)^2 + (B_i - B_L)^2} \quad (4)$$

- The size cost is simply the pixel number in a superpixel.

$$Size_{cost} = |pcount_i - pcount_L| \quad (5)$$

- With large possibilities, the matched superpixel pair should lie along the motion trend of frame pair, which can be described by obtained average optical flow. Therefore, as (6) shows, position cost is calculated using a cos angle, where dx and dy refer to the distance between superpixel I and its label L in x and y axis respectively.

$$Position_{cost} = 1 - \frac{dx * x_{Optical_Flow} + dy * y_{Optical_Flow}}{\sqrt{dx^2 + dy^2} * \sqrt{x_{Optical_Flow}^2 + y_{Optical_Flow}^2}} \quad (6)$$

- The information of the neighbour structure including the neighbouring superpixel's count and their relative positions, will also be taken into the data cost computation, as shown in Fig. 2.

$$D = w_1 * Color_{cost} + w_2 * Size_{cost} + w_3 * Position_{cost} + w_4 * Neighbors_{cost} \quad (7)$$

As shown in (7), those distances are weightedly summed together to form up the D function.

The Smoothness cost brings penalties to a pair of labelling for neighboring superpixels. As (8) shows, the proposed smoothness cost is based on the assumption that when two superpixels L_i and L_j are matched with the neighboring superpixels of I_i and I_j respectively, the distance between L_i and L_j should be short since the distance between I_i and I_j is short.

$$S = Distance[Distance(I_i, I_j), Distance(L_i, L_j)] \quad (8)$$

B. Energy minimization

As in (3), with Data and Smoothness costs, the energy function is then constructed based on the summation. The next step is to minimize the energy function to achieve matching. Several algorithms for energy minimization are available, however, since the network structure of the superpixels is not a grid graph, we choose Graph Cut based minimization algorithm using α -expansion for energy optimization.

TABLE I. PERFORMANCE COMPARISONS

Feature Matching Method		Accuracy Rate with [Count of Matched Pairs] in square brackets			
		Experiment 1	Experiment 2	Experiment 3	Experiment 4
Proposed Method in This Paper		0.97 [283]	0.98 [594]	0.96 [407]	0.97 [403]
Other Testing Methods					
Detector	Descriptor				
Harris	Sift	0.69 [1000]	0.95 [120]	0.94 [96]	0.73 [687]
	Surf [3]	0.04 [1000]	0.78 [120]	0.30 [96]	0.03 [687]
	Brief [6]	0.58 [854]	0.00 [3]	0.92 [91]	0.68 [632]
	Orb [18]	0.54 [833]	0.00 [3]	0.91 [91]	0.64 [621]
	Freak [1]	0.02 [876]	0.00 [4]	0.10 [92]	0.03 [641]
Sift	Sift	0.48 [1436]	0.73 [73]	0.78 [196]	0.57 [665]
	Surf	0.05 [1436]	0.52 [73]	0.63 [196]	0.38 [665]
	Brief	0.62 [1173]	0.00 [9]	0.81 [179]	0.65 [614]
	Orb	0.00 [0]	0.00 [0]	0.00 [0]	0.00 [0]
	Freak	0.03 [1219]	0.00 [12]	0.08 [175]	0.04 [619]
Surf	Sift	0.58 [1445]	0.74 [158]	0.73 [737]	0.62 [1206]
	Surf	0.52 [1445]	0.77 [158]	0.72 [737]	0.62 [1206]
	Brief	0.61 [1321]	0.71 [105]	0.74 [709]	0.67 [1141]
	Orb	0.32 [1285]	0.66 [96]	0.57 [704]	0.44 [1120]
	Freak	0.03 [773]	0.31 [29]	0.05 [464]	0.03 [753]
Orb	Sift	0.25 [500]	0.90 [21]	0.77 [496]	0.59 [500]
	Surf	0.36 [500]	0.95 [21]	0.89 [496]	0.71 [500]
	Brief	0.44 [500]	0.85 [21]	0.92 [496]	0.81 [500]
	Orb	0.20 [500]	0.95 [21]	0.76 [496]	0.55 [500]
	Freak	0.11 [126]	0.00 [5]	0.08 [212]	0.11 [190]
Fast [17]	Sift	0.83 [3565]	0.82 [369]	0.85 [332]	0.72 [1062]
	Surf	0.13 [3565]	0.57 [369]	0.55 [332]	0.23 [1062]
	Brief	0.65 [2863]	0.61 [102]	0.81 [300]	0.61 [984]
	Orb	0.57 [2784]	0.66 [89]	0.82 [296]	0.52 [967]
	Freak	0.02 [2957]	0.14 [126]	0.04 [307]	0.03 [997]

V. EXPERIMENTS

To demonstrate the matching performance, the proposed method is tested with the frame pairs extracted from the underwater videos. We compare the proposed method with the state-of-the-art feature matching methods accomplished based on the different combinations of feature detectors and descriptors. Experiment results are demonstrated in Table I.

According to the experiments, the accuracy rates and

matching feature pair numbers of other existing methods can hardly result in good performance simultaneously. Moreover, most of the time they produce poor results. For example, in Table 1, the best accuracy rate using other feature matching method in experiments is the one with Harris detector and Sift descriptor for experiment 2, which results in 95%. However, the totally number of matched pixel pairs is only 120, as shown in Fig. 3 (2b). The best matched pixel number of other existing feature matching method is the one using Fast detector and Sift descriptor

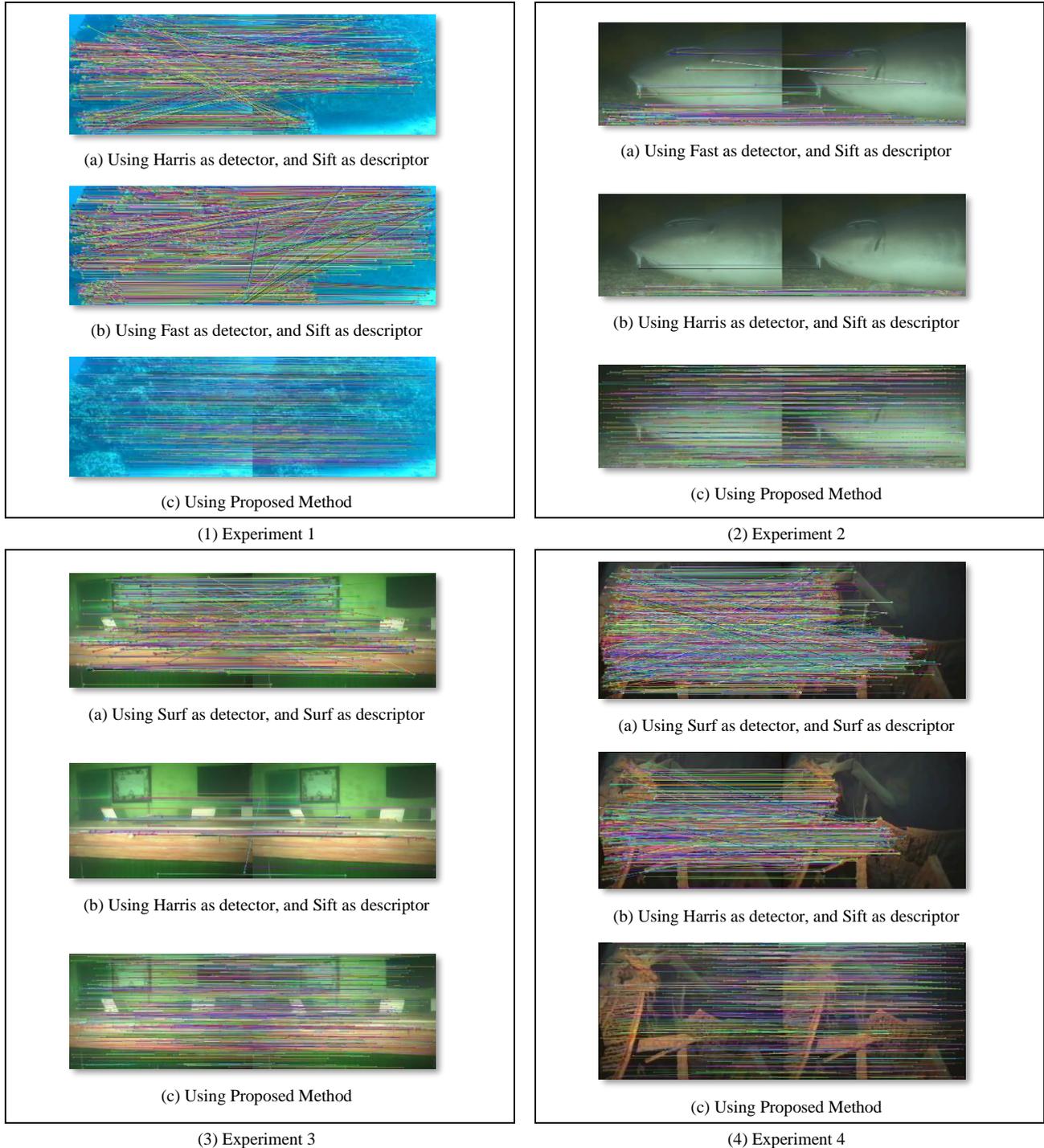


Figure 3: The tested frame pairs are extracted from underwater videos: (1) From tourist video of coral from Saipan1; (2) From the underwater video of "Myanmar Magic", website of Bubble Vision1; (3) From submerged Costa Concordia wreckage footage, news in "The Telegraph"1; (4) From the Titanic wreckage expedition footage, website of NOAA1; (a) and (b) are the results using different combinations of the feature detectors and descriptors; (c) are the results by proposed algorithm.

for experiment 1, which results in 3565. However, the accuracy rate is only 83%, as in shown Fig. 3 (1b), while our algorithm reaches 97%. Furthermore, the number of the matched pairs of this testing method drops sharply when computing for other frame pairs. The stability of its performance is poor.

As for the proposed algorithm based on superpixel, the accuracy rates are steadily at a higher performance, which can reach up to 98%. Meanwhile, the numbers of the matched pairs are in the unit of superpixel, which contains approximate 250 pixels in each in our implementation. Every pixel in the superpixel can find its own matched one in the corresponding superpixel in the other image. This will make the feature matching denser than it looks like. The matched feature numbers of other existing methods are in the unit of pixel. There is a very high possibility that many matched pixels by other methods are located in a same superpixel region. Therefore, the numbers of matched superpixel pairs by the proposed algorithm is far more efficient than the numbers of matched pixel pairs by other testing algorithms. With the settings of our implementation, the total number of the superpixels in one image is around 900. Meantime, the total pixel number in one image is more than 230400.

VI. CONCLUSION

This paper presents a novel algorithm to match features for frame pairs extracted from the underwater videos. Superpixels are used as features in the images. Generation of superpixel enhances perception for the image pairs in poor visibility. The designed energy functions are used for MRF to find the matched superpixels between image pairs. In the experiments, the results by the proposed algorithm demonstrate the competitive performance compared with those matching results using other existing feature matching algorithms. Further matching between the pixels in the matched Superpixel pairs can obtain a denser matching result.

ACKNOWLEDGMENT

This work is supported by the Engineering and Physical Sciences Research Council Project, UK (EPSRC) (No. EP/N025849/1).

REFERENCES

- [1] Alahi, A., Ortiz, R., & Vandergheynst, P. (2012, June). Freak: Fast retina keypoint. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (pp. 510-517). IEEE.
- [2] Aulinas, J., Carreras, M., Llado, X., Salvi, J., Garcia, R., Prados, R., & Petillot, Y. R. (2011, June). Feature extraction for underwater visual SLAM. In *OCEANS, 2011 IEEE-Spain* (pp. 1-7). IEEE.
- [3] Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In *Computer vision—ECCV 2006* (pp. 404-417). Springer Berlin Heidelberg.
- [4] Birchfield, S., & Tomasi, C. (1998). A pixel dissimilarity measure that is insensitive to image sampling. *PAMI, IEEE Transactions on*, 20(4), 401-406.
- [5] Boykov, Y., Veksler, O., & Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *PAMI, IEEE Transactions on*, 23(11), 1222-1239.
- [6] Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). Brief: Binary robust independent elementary features. In *Computer Vision—ECCV 2010* (pp. 778-792). Springer Berlin Heidelberg.
- [7] Derpanis, K. G. (2004). *The harris corner detector*. York University.
- [8] Egnal, G., & Wildes, R. P. (2002). Detecting binocular half-occlusions: empirical comparisons of five approaches. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(8), 1127-1133.
- [9] Hong, L., & Chen, G. (2004, July). Segment-based stereo matching using graph cuts. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on* (Vol. 1, pp. I-74). IEEE.
- [10] Klaus, A., Sormann, M., & Karner, K. (2006, August). Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on* (Vol. 3, pp. 15-18). IEEE.
- [11] Levinshtein, A., Stere, A., Kutulakos, K. N., Fleet, D. J., Dickinson, S. J., & Siddiqi, K. (2009). Turbopixels: Fast superpixels using geometric flows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12), 2290-2297.
- [12] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2), 91-110.
- [13] Mičušík, B., & Košecká, J. (2010). Multi-view superpixel stereo in urban environments. *International Journal of Computer Vision*, 89(1), 106-119.
- [14] Moore, A. P., Prince, S., Warrell, J., Mohammed, U., & Jones, G. (2008, June). Superpixel lattices. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (pp. 1-8). IEEE.
- [15] Radhakrishna, A., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Susstrunk, S. (2010). Slic superpixels. *Dept. School Comput. Commun. Sci., EPFL, Lausanne, Switzerland, Tech. Rep.*, 149300.
- [16] Ren, X., & Malik, J. (2003, October). Learning a classification model for segmentation. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on* (pp. 10-17). IEEE.
- [17] Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. In *Computer Vision—ECCV 2006* (pp. 430-443). Springer Berlin Heidelberg.
- [18] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, November). ORB: an efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 2564-2571). IEEE.
- [19] Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., & Rother, C. (2008). A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(6), 1068-1080.
- [20] Tong, H., Liu, S., Liu, N., & Barnes, N. (2010, August). A novel object-oriented stereo matching on multi-scale superpixels for low-resolution depth mapping. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE* (pp. 5046-5049). IEEE.
- [21] Veksler, O., Boykov, Y., & Mehrani, P. (2010). Superpixels and supervoxels in an energy optimization framework. In *Computer Vision—ECCV 2010* (pp. 211-224). Springer Berlin Heidelberg.
- [22] Wang, S., Lu, H., Yang, F., & Yang, M. H. (2011, November). Superpixel tracking. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 1323-1330). IEEE.
- [23] Zhang, Y., Hartley, R., Mashford, J., & Burn, S. (2011, December). Superpixels, occlusion and stereo. In *Digital Image Computing Techniques and Applications (DICTA), 2011 International Conference on* (pp. 84-91)

