2012 IEEE International Conference on Information Science and Technology
Wuhan, Hubei, China; March 23-25, 2012

Stereo Matching Based on Support Points Propagation

Hua Wu, Zhan Song, Jian Yao, Liang Li, and Yu Gu

***Abstract*—In this paper, we present a novel propagation-based stereo matching algorithm. Firstly, we select highly reliable depth points (i.e., support points) from both the high-textured areas and the low-textured ones in the rectified images. Then we propagate these support points along adjacent neighboring structure to produce the depths of other points in the image based on the triangulation derived from a set of highly reliable support points. This allows for efficient exploitation of the disparity search space, yielding accurate dense disparity without the need for global optimization. Experimental results successfully demonstrate both the effectiveness and matching accuracy of our proposed method.**

I. INTRODUCTION

WITH the development of the 3D films as well as other 3D applications, stereo matching from a stereo image been one of the hottest research topics in the field of computer vision. In the past decades, a lot of stereo matching algorithms have been proposed. Most of these stereo matching algorithms can be roughly classified into two categories: local optimization method and global optimization one [1].

It has been known that local optimization method can run much faster than the global optimization method under the same experimental conditions. In general, the local optimization method is the best choice for the purpose of real-time applications [1]. However, it typically ignores the disparity optimization focusing on the cost aggregation phase performed on the support window around each point in the image [2]. The quality of the estimated image disparity map mainly depends on the chosen support window size: low matching ratios for small window sizes and border bleeding artefacts for large window ones. However, recently some stereo matching techniques such as the adaptive support weight approach [3] and the segmentation-based adaptive support approach [4] can improve the disparity results with considerable running time. Extensive reviews and evaluation of recently proposed local optimization algorithms for stereo matching can be found in [5], [6].

However, most of global optimization methods can produce accurate results as they incorporate prior constraints into the correlation-based stereo model. In general they formulate the stereo matching problem as an energy minimization function with data constraints, smoothness constraints and other constraints [7], [8] and achieve the optimal results by minimizing the energy function with dynamic programming, graph cuts, belief propagation or fusion move and so on. But they are usually computationally expensive due to the slow converging optimization process. So recently some researchers proposed to reduce the computation time by utilizing the parallel techniques such as CUDA or OpenCL. But this will naturally increase the costs of the hardware systems.

Our proposed method in this paper is mainly motivated by a class of recently proposed methods [9], [10], [11], which leverage the support points to get the accurate disparity map of the stereo image pair. Firstly, these methods will select the high reliable disparity points (i.e., support points) and then propagate these support points along different structures such as scanline and triangulation. If no global optimization is used, the propagation-based methods usually illustrate low computational cost. But there exist two inherent problems in these methods. The first one is that early wrong support points will result in large disparity errors after propagation. The second one is that if these points in the area constructed by the support points don't obey the used propagation structure, it would also lead to wrong disparities. Wei and Quan [12] proposed to use segmented regions as the propagation units, which alleviate these problems to some extent. But the running time of the algorithm is dominated by the expensive segmentation process [9].

In this paper, we propose a novel stereo matching algorithm based on edge segmentation and triangulation propagation in the adjacent neighborhood. Different from the previous methods [9], [10], [11], our proposed method obtains the support points from the high-textured areas and the low-textured ones in the image. We use the edge segmentation as a cue for selecting disparity support points in the low-textured areas. Here we used the fast canny edge extraction algorithm which is not time-consuming. In this way, the obtained support points will become more efficiency. In addition, we combine the pixel color information, the fast changing gradient descriptor named DAISY [13], and the corner feature vector provided by the FAST corner detection [14] for evaluating the pixel matching quality. Finally we utilize adaptive weights for pixel matching, which is familiar with the FBS (fast bilateral stereo) method [2], by not only considering the color similarity but also considering the Euclidean distance. Using the high reliable support points for propagation, we can get high-quality disparity maps with less time.

The remainder of the paper is organized as follows. A full description of our method is provided in Section II. Experiment results are provided in Section III. Finally the conclusions are drawn in Section IV.

Hua Wu is with School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi, P.R. China and Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, P.R. China.

Zhan Song is with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, P.R. China and The Chinese University of Hong Kong, Shatin, NT, Hong Kong, P.R. China (email: zhan.song@siat.ac.cn).

Jian Yao is with School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi, P.R. China and Institute for Transuranium Elements, European Commission-Joint Research Centre (JR- C), Ispra (VA), Italy (email: jianyao@gxu.edu.cn).

Liang Li is with Shenzhen Polytechnic, Shenzhen, Guangdong, P.R. China (email: liliang@szpt.edu.cn).

Yu Gu is with School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi, P.R. China.

II. STEREO MATCHING

In this section we describe our proposed stereo matching algorithm in details. The algorithm flowchart is shown in Figure 1. Given a pair of rectified stereo images, first we will extract the support points from the high-textured areas and the low-texture ones, which are detailedly described in Section II-A and Section II-B respectively. Then we will propagate the support points along the triangulation and adjacent neighborhood structure to get the disparities of other points and filter out the outliers by the left-to-right consistence checking, as described in Section II-C. The last step given in Section II-D is to compute the disparities of the remaining points in the image, which are not successfully propagated.

A. Support Points Extraction from High-Textured Areas

As stated above, we extract the support points in two different ways from high-textured areas and low-textured ones, respectively. The support points in the high-textured areas often stand for the discontinuous disparities. We know that there often exist corner points in the high-textured areas. In our algorithm, we just compute the disparities for the corner points and select the support points from them. We chose the method presented in [14] to get corner points in a pair of rectified stereo images due to that it runs fast and doesn't sacrifice the quality of the corner's responds. Its OpenCV implementation was used in our algorithm implementation. Given a corner point $\mathbf{p} = (x, y)$ with the disparity *d,* the matching cost function is defined as follows:

$$C_{\mathbf{p}}(d) = C_{\text{desc}}(\mathbf{p}, d) + C_{\text{col}}(\mathbf{p}, d) + C_{\text{resp}}(\mathbf{p}, d)\,, \qquad (1)$$

where $C_{\text{desc}}(\mathbf{p}, d)$, $C_{\text{col}}(\mathbf{p}, d)$, $C_{\text{resp}}(\mathbf{p}, d)$ denote three different matching cost functions in term of gradient, color, corner feature similarities, respectively.

The term $C_{\text{desc}}(\mathbf{p}, d)$ denotes the gradient cost function, which is formed by the DAISY descriptor [13], of computing the gradient difference between the left image point $\mathbf{p}$ and the right image point $\mathbf{p}' = (x - d, y)$, i.e., the sum of absolute differences between two gradient vectors provided by the DAISY descriptor. The DAISY descriptor is a good and efficient gradient descriptor for our use. It can quickly change the description size of the circle window around the point $\mathbf{p}$ because it exploits the truth that the large Gaussian kernel convolution can be formed by two consecutive small kernel convolutions. So the DAISY descriptor can run fast for adaptive window sizes.

The term $C_{\text{col}}(\mathbf{p}, d)$ represents the RGB color difference between the left image point $\mathbf{p}$ and the right image point $\mathbf{p}'$, which is defined as follows:

$$C_{\text{col}}(\mathbf{p}, d) = \sum_{c=\mathbf{R,G,B}} |\mathbf{I}_c(\mathbf{p}) - \mathbf{I}'_c(\mathbf{p}')|, \qquad (2)$$

where $\mathbf{I}_c(\mathbf{p})$ denotes the c-th color channel value of the point $\mathbf{p}$ in the left image $\mathbf{I}$, and $\mathbf{I}'$ denotes the right image. The cost function $C_{\text{resp}}(\mathbf{p}, d)$ measures the response difference between the corner features of the left image point $\mathbf{p}$ and the right image point $\mathbf{p}'$. In this paper, the response value of corner feature is provided by the FAST corner detector [14]. We observe that the corner response values of corresponding points in the left and right images provided by the FAST corner detector are similar. To some extent, this term can provide some reliable measurement for the final cost function in Eq. (1).

In order to efficiently and quickly filter out the outliers, we set three suitable thresholds for three above described cost functions. The point $\mathbf{p}$ in the left image with the disparity d (corresponding to the point $\mathbf{p}'$ in the right image) is automatically removed if one of their cost values are larger than their corresponding pre-defined thresholds.

We experimentally find that the proposed combined Gradient-Color-Corner measurement shows the improved matching accuracy than individual Color, Gradient and Corner measurements. By using such the proposed combination measurement, we can find more robust matching support points. In order to efficiently reduce the matching ambiguities, we employ an aggregation step and use adaptive window sizes for aggregation. Similar to the aggregation strategy used in [2], we use the Dual-Cross-Bilateral aggregation. The aggregated cost function $C_{\mathbf{p}}(d)$ with the disparity d is computed over a circle-shape neighborhood window centered at the point $\mathbf{p}$ formed by the DAISY feature detector [13] as follows:

$$C^*_{\mathbf{p}}(d) = \frac{\sum_{\mathbf{q}\in N_c(\mathbf{P})} w(\mathbf{p},\mathbf{q}) \times C_{\mathbf{q}}(d)}{\sum_{\mathbf{q}\in N_c(\mathbf{P})} w(\mathbf{p},\mathbf{q})}, \qquad (3)$$

where $\mathbf{q}$ is the point in the circle-shaped neighborhood window N_c centered at the point $\mathbf{p}$ and $w(\mathbf{p}, \mathbf{q})$ denotes the support-weight value of the point pair **(p, q)** defined as follows:

$$w(\mathbf{p}, \mathbf{q}) = \exp\left(-\left(\frac{\Delta c(\mathbf{p},\mathbf{q})}{\lambda_c} + \frac{\Delta g(\mathbf{p},\mathbf{q})}{\lambda_g}\right)\right), \qquad (4)$$

where $\Delta c(\mathbf{p}, \mathbf{q}) = \|\mathbf{I}(\mathbf{p}) - \mathbf{I}(\mathbf{q})\|$ and $\Delta g(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|$ represent the color similarity and Euclidean distance between the points $\mathbf{p}$ and $\mathbf{q}$, and λ_c and λ_g are two balance controlling parameters.

In this paper we employ a winner-take-all searching strategy to get the optimal disparity $\mathbf{D}(\mathbf{p})$ of the point $\mathbf{p}$ in the left image $\mathbf{I}$ as:

$$\mathbf{D}(\mathbf{p}) = \arg\min_d C^*_{\mathbf{p}}(d), \qquad (5)$$

Based on such a strategy, we will search for reliably matched support points in all extracted corner points. The corner point is regarded as a support point only if all the following conditions are satisfied:

$$\begin{cases} \mathbf{D}(\mathbf{p}) = \mathbf{D}'(\mathbf{p} - (\mathbf{D}(\mathbf{p}), 0)), \\ \frac{C^*_{\mathbf{p}}(d)}{C^*_{\mathbf{p}}(\mathbf{D}(\mathbf{p}))} > \lambda_t, \ \ \forall d \neq \mathbf{D}(\mathbf{p}), \\ |\mathbf{D}(\mathbf{p}) - \mu| < 3\sigma, \end{cases} \qquad (6)$$

where $\mathbf{D}$ and $\mathbf{D}'$ denote the disparity maps for the left and right images $\mathbf{I}$ and $\mathbf{I}'$, respectively.

The first condition in Eq. (6) shows the support point passes the left-right consistency check where $\mathbf{D}'(\mathbf{p})$ stands for the disparity of the point $\mathbf{p}$ in the right image $\mathbf{I}'$. The second condition shows the aggregated cost value $C^*_{\mathbf{p}}(d)$ of the support point $\mathbf{p}$ with the disparity d must be significantly smaller than the competitors with other different disparities where λ_t is a threshold value used to control the quality of reliability. The third condition removes the possibly un-reliable support point based on the disparities in its small neighborhood where μ stands for the mean of disparities of support points in its small neighborhood. It can efficiently filter out possibly incorrect support points. In order to achieve this goal, in our method we will fix a small window size at first, which is possibly increased if there is not sufficient number of support points in this used area until it contains sufficient number of support points or the increased window size is larger than the pre-defined largest window size. The support points are unconditionally removed if their finally checking window sizes without sufficient numbers of support points around them are larger than the pre-defined largest window size.

B. Support Points Extraction from Low-Textured Areas

Here we mainly discuss how to extract some support points from low-textured areas. Firstly, we extract the edge segmentation images for both the left image and right one using the Canny edge detector [15]. Secondly we search the segmented regions where their pixel intensities are zero in the binary edge image, and collect some points in these regions. The numbers of the selected points from these regions depend on the area sizes of these regions. In the large regions we will select more points while in the small regions we will select little few points. At the same time, we set a threshold for the maximal search region λ_{area} to prevent the point selection from a too large region because of the incorrect edge extraction in the Canny algorithm.

To extract robust support points from the low-textured areas, we employ the adaptive window strategy in which the window size at first formed by the DAISY descriptor is adaptively increased. We set the window size depending on the distance between the point to be selected and its nearest edge point in the binary edge image. All the selected support points in low-textured areas must follow the winner-take-all searching strategy in Eq. (5) and satisfy three conditions in Eq. (6) for support points extraction.

C. Delaunay Triangulation and Neighborhood Propagation

As described in [10], we can approximate a small piece of region for disparities into a planar structure (i.e., triangulation structure). So we can integrate this prior into the Bayesian framework for inferring the disparities of other points. Now considering a point $\mathbf{v} = (u_n, v_n)$ inside the n-th triangulation of the left image, we define the prior for the disparity d_n of the point **v** as follows:

$$\mathrm{p}_{\mathbf{v}}(d_n|\mathbf{O}_n, \mathbf{S}) \propto \begin{cases} \gamma + \exp\left(-\dfrac{\left(d_n - \mu(\mathbf{O}_n, \mathbf{S})\right)^2}{2\sigma^2}\right) & \text{if } |d_n - \mu| < 3\sigma \vee d_n \in N_s, \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (7)$$

where $\mathrm{p}_{\mathbf{v}}(d_n|\mathbf{O}_\mathbf{n}, \mathbf{S})$ denotes the probability of the point **v** with the disparity d_{n}. $\mathbf{O}_\mathbf{n}$ stands for the observation of the point **v** which is comprised by the image coordinates and the feature vector. **S** represents the set of the support points. Each support point is constituted by the image coordinates and its disparity, i.e.,(u_i, v_i, d_i) where (u_i, v_i) is the image coordinates and d_i is the disparity. N_s denotes the disparities of the support points around the point v in the triangulation structure. $\mu(\mathbf{O}_\mathbf{n}, \mathbf{S})$ is the mean function linking the support points and the observations (i.e. feature vector). In our method, we use the piecewise linear function used in [10] defined as follows:

$$\mu(\mathbf{O}_n, \mathbf{S}) = u_n \times a_i + v_n \times b_i + c_i, \quad (8)$$

where a_i, b_i, c_i are the planar parameters formed by Delaunay triangulation and (u_n, v_n) is the point coordinates.

The likelihood is defined as the constrained Laplace distribution as follows:

$$p_{\mathbf{v}}(\mathbf{O}_n'|\mathbf{O}_n, \mathrm{d}_\mathrm{n}) \propto \begin{cases} \exp(-\beta\|\mathbf{f_n} - \mathbf{f_n}'\|_1) & \text{if } u_n = u_n' + d_n, v_n = v_n', \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (9)$$

where $\mathrm{p}_{\mathbf{v}}(\mathbf{O_n}'|\mathbf{O_n}, \mathbf{d_n})$ denotes the probability for measuring the similarity between the observation $\mathbf{O_n}$ of the point $\mathbf{v}$=(u_n, v_n) in the left image and the observation $\mathbf{O_n}'$ of the point $\mathbf{v}'$=(u_n', v_n') in the right image. $\mathbf{f_n}$ and $\mathbf{f_n}'$ denotes the feature vectors of the points **v** and $\mathbf{v}'$ in the left and right images respectively which consisting of three parts described in Eq. (1).

The remaining task is to infer the true value of the disparity. In the used Bayesian framework, we use the maximum a-posteriori (MAP) estimation to compute the value d^* as follow:

$$d^* = \arg\min_{d_n}(p_{\mathbf{v}}(d_n|\mathbf{O_n}, \mathbf{S}) \times p_{\mathbf{v}}(\mathbf{O}_n'|\mathbf{O}_n, d_n)). \quad (10)$$

In order to avoid the case that triangulation assumption might be violated, we also propagate the robust support points along its neighborhood. We first collect the eight adjacent support points and then find the minimum and maximum disparities among them. We search the point disparity d_* between the minimum and maximum. We evaluate the quality of the searching disparity using the criterion defined in Eq. (9).

Finally, we get the point disparity *d* of the point **v** by comparing the disparity measurement cost values as follows:

$$d = \begin{cases} d^* & \text{if } \mathsf{C}_\mathbf{v}^1 \geq \mathsf{C}_\mathbf{v}^2 \\ d_* & \text{otherwise} \end{cases} \quad (11)$$

where $\mathsf{C}_\mathbf{v}^1$ denotes the measurement cost value of the point **v** formed in the triangulation strategy and $\mathsf{C}_\mathbf{v}^2$ denotes the measurement cost value of the point **v** formed in the neighborhood propagation.

D. Disparity Post-Processing

This final step focuses on the post-processing of the disparity map by refinement and interpolation. As we know if two points is close enough and their colors are similar in the image, their disparities are the same in most cases. So in our method we use neighborhood propagation as depicted in Section II-C. We first do one iteration for neighborhood propagation into the points in the image whose disparities are not successfully recovered and then check them based the left-right consistence described in

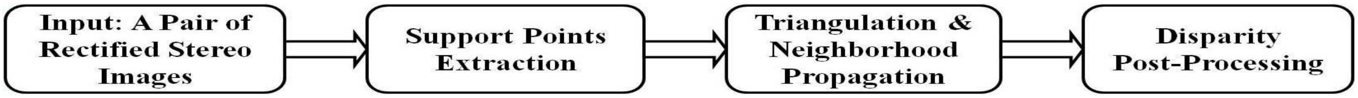


Fig. 1. The flowchart of our proposed propagation-based stereo matching algorithm.

Section II-A. Finally, we perform the morphology operation (erosion and dilation) for filtering out possible outliers and filling in some blobs.

TABLE I
THE MAIN PARAMETERS SETTTING IN OUR PROPOSED STEREO MATCHING ALGORITHM

λ_t	Q (Radius)	T (Angular)	H (Histogram)
0.8	2	8	4

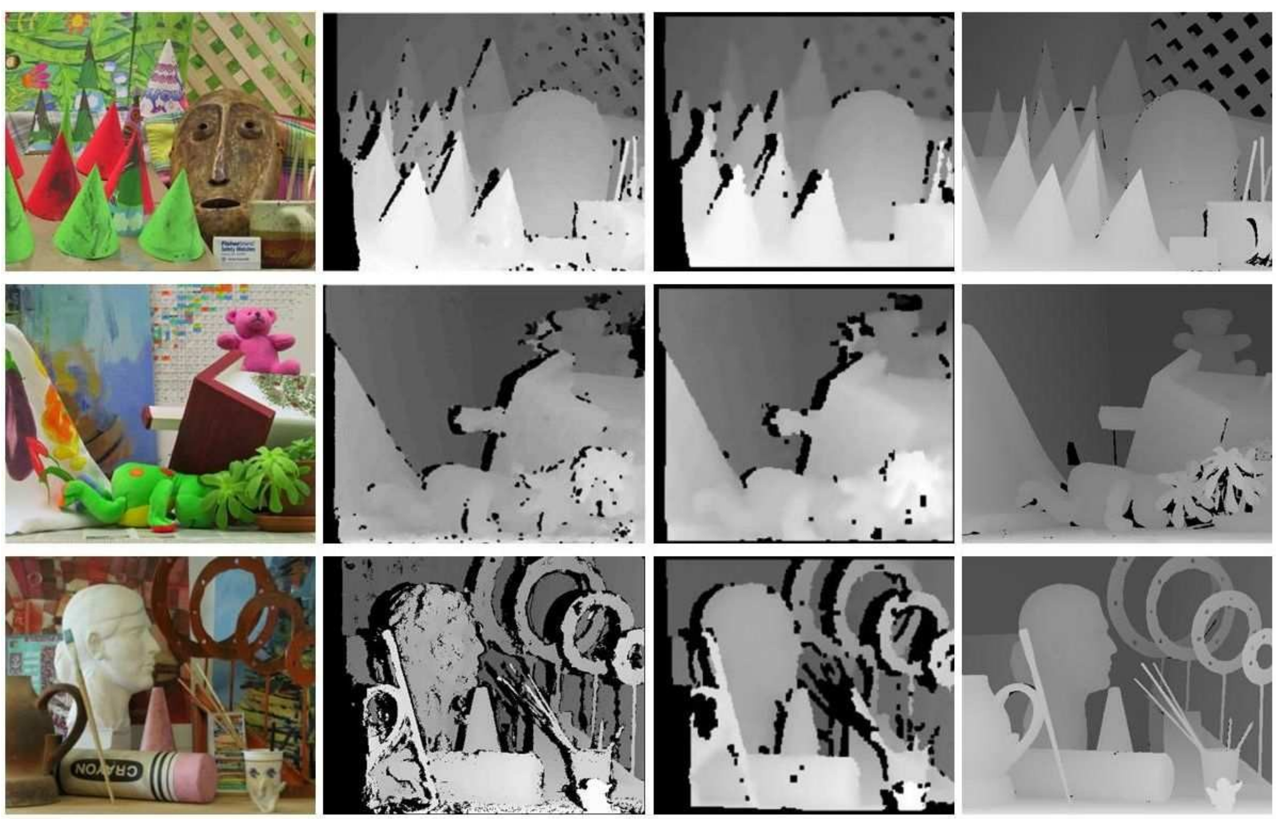
III. EXPERIMENTAL RESULTS

We tested our method on the Middlebury benchmark [1], which has the known ground truth disparity data. Our algorithm was implemented in C++ and tested on a PC with AMD Athlon 64 X2 Dual Core Processor 4600+ 2.4GHz CPU and 2GB memory. The main parameters were set as shown in Table I where λ_t is the threshold for robust support points extraction used in Eq. (6), and Q, T and H are the radius, angular and histogram optimization numbers of the DAISY descriptor [13], whose detailed description and discussion are given in [13]. We compare our method with the ELAS algorithm [10]. The comparison results are shown in Figure 2. From the results shown in Figure 2, we observe that our proposed stereo matching algorithm can successfully recover the disparities in most of small depth discontinuous regions and also get more clear disparities around some depth discontinuous boundaries. In the computation time, our pro- posed algorithm is little worse than the ELAS algorithm due to that some pre-processing steps (e.g., edge segmentation and feature detection) are needed in our algorithm.

At the same time, we observe some errors in our recovered disparity maps. Here we briefly discuss limitations and possible improvements for our methods. Firstly, our method doesn't consider the radiometric variation such as illumination direction, illuminant color are imaging device changes, which are often occurred in the outdoor scene. The result is expected to be improved if we can integrate the radiometric factor used in [16] into our method. Secondly, our method doesn't remove some disparity errors caused by the pre-processing, such as filtering process. Thirdly, the result will be become more well if we consider the sub-pixel interpolation techniques [1], [17] for pixel matching.

IV. CONCLUSIONS

In this paper we develop a simple and effective method for stereo matching, which is based on the support points propagation. Our method first collects a set of reliably matched support points and then quickly searches the disparities of other points by propagation. Experimental results on the Middlebury benchmark [1] successfully demonstrate the effectiveness of our method. In the future we will apply our method to real-time applications by implementing it in the high-parallel GPU hardware.



(a) (b) (c) (d)

Fig. 2. Comparison results of our proposed stereo matching algorithm vs the ELAS algorithm [10]: (a) the left images of tested stereo image pairs; (b) the disparity maps estimated by our algorithm; (c) the disparity maps estimated by the ELAS algorithm [10]; (d) the ground truth disparity maps.

REFERENCES

[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[2] S. Mattoccia, M. Viti, and F. Ries, "Near real-time fast bilateral stereo on the gpu," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2011, pp. 136–143,.

[3] K. Yoon and I. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.

[4] F. Tombari, S. Mattoccia, and L. D. Stefano, "Segmentation-based adaptive support for accurate stereo correspondence," *Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, 2007, pp. 427–438,.

[5] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda, "Classification and evaluation of cost aggregation methods for stereo correspond dence," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8.

[6] L. Wang, M. Gong, M. Gong, and R. Yang, "How far can we go with local optimization in real-time stereo matching," *The Third International Symposium on 3D Data Processing,Visualization, and Transmission*, 2006, pp. 129–136.

[7] L. Wang and R. Yang, "Global stereo matching leveraged by sparse ground control points," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3033–3040,.

[8] M. Bleyer, C. Rother, P. Kohli, D. Scharstein, and S. Sinha, "Object stereo -joint stereo matching and object segmentation," *Computer Vision and Pattern Recognition*, pp. 3081–3088, 2011.

[9] X. Sun, X. Mei, S. Jiao, M. Zhou, and H. Wang, "Stereo matching with reliable disparity propagation," *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, pp.132 –139, 2011.

[10] A. Geiger, M. Roser, and R. Urtasun, "Efficient large scale stereo matching," *Asian Conference on Computer Vision (ACCV)*, 2010, pp. 25–38,.

[11] Q. Yang, L. Wang, and N. Ahuja, "A constant-space belief propagation algorithm for stereo matching," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1458–1465, 2010.

[12] Y. Wei and L. Quan, "Region-based progressive stereo matching," *IEEE Conference on Computer Vision and Pattern Recognition*, 2004, vol. 1, pp. 106–113.

[13] E. Tola, V. Lepetit, and P. Fua, "A fast local descriptor for dense matching," *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[14] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.

[15] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[16] Y. Heo, K. Lee, and S. Lee, "Robust stereo matching using adaptive normalized cross-correlation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 807–822, 2011.

[17] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister, "Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 492–504, 2009.