A Multi-Constraint Strategy Algorithm for Stereo Matching

Yingjiang Li¹*, Yuzhong Zhong², Libo He¹ and Shuqiu Tan¹
¹Chongqing university of technology, Chongqing, China
²Computer college, Sichuan University, Chengdu, China

*Corresponding author’s e-mail: 45487743@qq.com

Abstract. An improved multi-constraint stereo matching strategy is proposed herein. First, the original reference images are downsampled to reduce the matching scale. Absolute difference and rank transform are used as the matching cost. We establish the associated disparity map and the matching cost matrix to constrain the matching process. We use the associated disparity map to apply a smoothing constraint on disparity values. We then apply a uniqueness constraint using the associated disparity map and the matching cost matrix. We also apply a continuity constraint to improve the efficiency of the algorithm. Second, for the downsampled disparity map, the disparity map of the original image is obtained by interpolation and filling. Lastly, the final disparity map is obtained using a sub-pixel refinement strategy. The experimental results demonstrate that the algorithm is obviously superior to the traditional method. In addition, a multi-constraint matching process may be extracted as a framework for combination with other algorithms. Compared with the original algorithms, the fusion algorithm improved matching accuracy by 3% on average.

1. Introduction
Stereo matching is a key step to obtaining depth information in computer vision [1]. The depth information of an object in an image is obtained by calculating its disparity value, and the disparity value is obtained by searching for the corresponding object relation in a pair of stereo images. Many researchers have proposed different methods for stereo matching, a basic problem in stereo vision processing; however, obtaining accurate disparity is still a challenge for some areas in the image, such as occluded areas, repetitively textured areas, and smooth areas.

According to the methods of cost aggregation and disparity calculation, stereo matching algorithms can be divided into local stereo matching, global stereo matching, and other matching method. Local stereo matching aggregates the matching cost through the support domain and selects the matching point with the minimum matching cost to obtain the disparity value [2-5]. Existing local cost aggregation methods include adaptive support weight [2], guided filter [3, 4], and trilateral filter [5]. The window-based matching cost methods are sensitive to noise and brightness, a cross-correlation [6] and adaptive NCC[7] are used to obtain a robust matching cost against variations in image illumination. A robust matching cost can also be obtained by rank and census transformation [8-10]. Kim et al. [11] constrain the disparity of smooth regions and correct disparity in regions of discontinuity using edge and texture information. Zhan et al. [12] added a guided image in the matching process to enhance the matching accuracy of edges and smooth texture regions.

Global stereo matching methods assume a definite smoothness. The matching process is the optimization of the global energy function composed of data and smoothness terms over the entire image. Effective methods used to optimize the energy function include dynamic programming [13],...
graph cuts [14, 15], and belief propagation [16]. Yamaguchi et al. [17] generated stereo matching using a hybrid Markov random field consisting of a continuous random variable for a slant 3D plane and a discrete random variable for the occluding edges. Jung et al. [18] used the consistency criterion between actual and virtual intermediate views, and minimized the energy function, including the consistency terms.

Other matching algorithms, including semi-global matching (SGM) [19] and minimum spanning tree (MST) [20] are excellent matching algorithms. In recent years, the matching algorithm based on convolutional neural network (CNN) [21] and deep learning [22] has become a popular algorithm and has achieved extremely high matching accuracy.

The above algorithms have made significant contributions to the development of stereo matching, but some are still flawed in terms of practical application. Herein, we propose a local stereo matching algorithm based on downsampling and multiple constraints. The algorithm uses absolute difference (AD) and rank [23] as the matching cost. We use a new multi-constraint strategy to reduce wrong matching and repeat matching.

2. Related work

2.1. Coarse-to-Fine strategy
Early research in stereo vision adopted stereo matching using a coarse-to-fine (CTF) strategy. The disparity value was first obtained at a coarse resolution, and then the high-resolution disparity search space was reduced by using the coarse disparity. This CTF strategy has been widely used stereo matching [24, 25]. In this paper, we use a similar method. To improve matching efficiency, we use Gaussian downsampling to obtain a semi-resolution image. In addition, the following stereo matching process is applied to the semi-resolution image.

2.2. Stereo matching constraints
Thus far, a number of constraint strategies have been proposed for the stereo matching process. A constraint strategy can reduce the matching search space and improve the matching accuracy. The existing matching constraint strategies are including: epipolar constraints, uniqueness constraint, continuity constraint, similarity constraints, sequential consistency constraint, left and right consistency constraint, etc.

Although a number of constraints have been proposed, some are inaccurate. For continuity constraints, it is sometimes difficult to distinguish which regions belong to the same object. For similarity constraints, a different effect may occur on the color or grayscale values under different illumination. For sequential constraints, if there are occluded objects in the foreground, the foreground can sometimes cause significant interference to the background when the image is captured by two cameras. Some constraint strategies cannot be applied to all scenes, and some of the constraints cannot be mandatorily used in a stereo matching algorithm.

In our matching algorithm, we proposed a new multi-constraint strategy, including the improved smoothness constraint, uniqueness constraint, continuity constraint, and left and right consistency constraints. We propose a new smoothness constraint method, and a concept of the Associated Disparity Map (ADM). The ADM and the current disparity map are used to constrain the smooth regions of the image, improve the matching accuracy. At the same time, the matching cost of each point is recorded, the matching cost matrix (MCM) is generated, and the ADM and the MCM are used to carry out the uniqueness constraint so that the multiple matching problems of the points in the right reference image are reduced. Finally, the left and right consistency constraints are used to remove the unstable and occluded points in the disparity map.

3. Proposed algorithm
The disadvantage of local stereo matching is that there is no global consideration, and it is easy for the algorithm to fall into a local minimum. The best matching point is sometimes not the correct matching
point, and thus we add a multiple constraint strategy to our matching process to prevent the matching point from falling into a local optimum. Our algorithm mainly includes the following steps: downsampling and rank transformation; multi-constrained stereo matching; disparity interpolation and filling; fine matching and sub-pixel refinement.

3.1 Downsampling
The purpose of downsampling an image is to improve the efficiency of the stereo matching algorithm. The downsampling method uses Gaussian subsampling with a $5 \times 5$ Gaussian kernel and $\sigma = 1$. After the downsampling, the size of the entire image becomes one-quarter of the original, the maximum disparity range becomes one-half, and the size of the cost aggregation window can be appropriately smaller without affecting the matching accuracy, which can further improve the matching speed. In theory, if the matching accuracy is not strict, the algorithm usually increases the speed by about 10-fold when using the downsampled image for matching. In addition, if the original image resolution is too large, downsampling may be applied multiple times.

3.2 Stereo matching
The original reference images at full resolution are $I_L$ and $I_R$. The images after downsampling are $I_{DL}$ and $I_{DR}$. The algorithm first applies a rank transformation on the downsampled image, and then uses multiple constraints to conduct stereo matching.

3.2.1 Rank Transformation. The matching cost is also a very important measure in stereo matching, and herein we use the image intensity combined with rank transformation as the matching cost. The rank transformation (RT) uses the method described in [10], the reliability of which was proven in [26]. The RT is chosen as the matching cost measure because this type of transform is insensitive to differences in noise and brightness in the image. This RT method can be described as follows: First, the difference between the center pixel $p$ and the average value of the other pixels in the neighborhood window of $p$ are calculated, and then the intensity difference is divided into five levels: smaller, equal, bigger, and biggest—as shown in Formula (1).

$$\text{rank} = \begin{cases} 
-2 & \text{if Diff.} < -s, \\
-1 & \text{if } -s \leq \text{Diff.} \leq -t, \\
0 & \text{if } -t \leq \text{Diff.} \leq t, \\
1 & \text{if } t \leq \text{Diff.} \leq s, \\
2 & \text{if } \text{Diff.} > s, 
\end{cases}$$

Here, $s$ and $t$ are graded parameters. The RT matrices $I_{DL\text{rank}}$ and $I_{DR\text{rank}}$ can be obtained by applying a RT on all points of the two downsampled images $I_{DL}$ and $I_{DR}$.

3.2.2 Stereo matching based on multiple constraints. In our stereo matching algorithm, the calculation of the matching cost is composed of absolute differences (AD) and RT. The combination of AD in intensity and RT can be used to achieve a more accurate matching cost. At the same time, in order to reduce the interference of pixel points in different disparity planes, and protect the edge information of the image during the process of cost aggregation, we use an adaptive weight aggregation method based on bilateral filtering [2]. In the support window, the similarity and proximity between each point and the center point are calculated. AD and RT use different evaluation criteria, and the initial matching cost is inconsistent, so we cannot add the two types of data directly. Therefore, the two measurement functions need to be normalized to $[0, 1]$ using the cost aggregation formula, which is shown below.

$$C(p, d) = \Sigma_{q \in P} w(p, q) (\rho(C_{AD}(p, d), \lambda_{AD}) + \rho(C_{\text{rank}}(p, d), \lambda_{\text{rank}}))$$

$$w(p, q) = \exp\left(-\frac{\Delta_{C_{pq}}}{\gamma_c} + \frac{\Delta_{\gamma_{pq}}}{\gamma_g}\right)$$
\[
\rho(C, \lambda) = 1 - \exp\left(-\frac{C}{\lambda}\right) \tag{4}
\]

\[
\Delta C_{pq} = \| f(i,j) - f(k,l) \|, \quad \Delta G_{pq} = \sqrt{(i-k)^2 + (j-l)^2} \tag{5}
\]

In Formula (2), \( p \in I_{DL} \), \( C(p, d) \) is the total matching cost between \( p(x, y) \) and \( p' (x-d, y) \) in the right reference image \( I_{DR} \); \( w(p, q) \) is the adaptive support weight function; \( \Omega_p \) is the pixel in the support window of point \( p \); \( \rho \) is the normalization function; \( C_{AD} \) and \( C_{rank} \) are the matching costs of AD and rank, respectively; and \( \lambda_{AD} \) and \( \lambda_{rank} \) are normalized coefficients. In Formula (3), \( \Delta C_{pq} \) and \( \Delta G_{pq} \) are the similarity and proximity between pixel \( q \) in the support window and the center pixel \( p \), respectively, the calculation method for which is shown in Formula (5); and \( \gamma_c \) and \( \gamma_g \) are user-defined parameters, proportional to the size of the support window, i.e., \( \gamma_c, \gamma_g \propto \text{(window size)} \). Formula (4) is used to normalize \( C_{AD} \) and \( C_{rank} \). Finally, \( f(i,j) \) in Formula (5) is the center point of the support window, and \( f(k,l) \) is a point in the support window.

In the matching process, we also establish two matrices of the same size as the downsampled image to store two sets of information. For one, when a point \( p(x, y) \) in the left reference image matches a point \( p'(x-d, y) \) in the right reference image and disparity \( d \) is recorded on the point \( (x-d, y) \) of a new matrix, which we call an ADM. For the other, when a point \( p(x, y) \) on the left reference image matches a point \( p'(x-d, y) \) on the right reference image, the matching cost on the current point \( p(x, y) \) is recorded in another matrix, which we call the MCM, the initial value of which is 255. Using these two matrices, we can apply smoothing and uniqueness constraints during the matching process.

In Figure 1, (a) is the disparity obtained from Adirondack of the Middlebury stereo dataset, (b) is the associated disparity map, and (c) is the MCM (the value is amplified 10 times, and the initial value of the matrix is 255). No constraint strategy is applied to any of the images in Figure 1.

**Figure 1.** Disparity map and two matrices without a constraint strategy: (a) left disparity map, (b) associated disparity map (ADM), and (c) matching cost matrix (MCM).

The disparities have incorrectly matched in areas of low texture and occlusion; in particular, the red ellipse region in Figure 2 (a) is more obvious, and in (b) we can see the position and value of the incorrect matching points, i.e., in the ADM. In addition, in (c) the matching costs of the incorrect matching points and occluded area are higher. The dark region of the MCM show where the matching cost is low; the bright regions indicate points of occlusion, points of high matching cost, and regions that are not matched.

ADM is similar to the right disparity map, but its method of generation is different. It corresponds to the disparity map of the left reference image \( I_{DL} \). For example, a pixel point \( p(x, y) \) in the left reference image finds a matching point in the right reference image \( p'(x-d, y) \), and the disparity value of point \( p \) is then \( d \), and the disparity value of point \( p' \) on the ADM is also \( d \). If the matching cost between \( p \) and \( p' \) is \( c_p \), then the value of point \( p' \) on the MCM is \( c_p \).

- **Smoothness constraints.** The smoothing constraint consists of two parts: one is given by the neighboring points around the current point in the left reference image, and the other is given by the neighboring points of the corresponding matching point in the ADM. The constraint formula is as follows:

\[
P(p, d) = \sum_{q \in \Omega_p} w(p, q) |d_p - d_q| + \sum_{q' \in \Omega_{p'}} w(p', q') |d_{p'} - d_{q'}| \tag{6}
\]

Here, \( \Omega_p \) is the set of all points within the support window of point \( p \) in the left reference image,
point \( p' \) is the matching point of \( p \), \( \Omega_{p'} \) is the set of points in the support window of \( p' \) in the right reference image, and \( w(p, q) \) and \( w(p', q') \) are the support weight functions. To reduce the error constraints that are similar to the current point intensity but not in the same disparity plane, we also use a weight calculation method for bilateral filtering. The calculation is the same as in Formula (3), where \( d_i \) denotes the disparity value of pixel \(*\). From Formula (6), we see that the first term is the smoothing constraint on \( p \) using the support window of the current pixel \( p \), and the second is the smoothing constraint on \( p' \) using the support window of point \( p' \) on the ADM, i.e., the smoothing constraints on point \( p \). The intensity of the constraint is related to the proximity, similarity, and disparity difference between \( p \) and \( q \).

- **Uniqueness constraints.** Here, we use the ADM and MCM to determine the matching reliability. MCM records the matching cost \( C(p', d) \) of each matching point in the right reference image. We first judge whether the matching cost of matching point \( p' \) in the MCM is smaller than the threshold value \( T_c \), and then determine whether the disparity value in the support window of \( p' \) on the ADM is smooth. If these two conditions are satisfied, we consider this point to have been matched reliably, and thus it can no longer be matched with other points.

- **Continuity constraint.** Here, we only use a reliable and correct disparity value to spread to the subsequent point. The method compares whether the current point \( p \) and the adjacent point \( p + 1 \) are similar in intensity, if so, the disparity of the point is propagated toward the right, i.e., \( d_{p+1} = d_p \).

Before calculating the disparity value for each point, we first determine whether the current point has a preset disparity value in the disparity matrix, and if so, we apply matching within a small range \([d-3, d+3]\). If the matching cost is less than the threshold value \( T_c \), the resulting disparity value is saved and propagated to subsequent points; otherwise, the disparity search range extends to the maximum disparity value to re-search the matching point. With the addition of smoothness and uniqueness constraints, the measure we use to search for the best matching point is determined based on the new energy function, and Formula (7) is obtained from Formula (2) and (6).

\[
E(p, \hat{d}) = C(p, \hat{d}) + \beta \times P(p, \hat{d}) \tag{7}
\]

Here, \( \hat{d} \) indicates that the target matching point corresponding to point \( p \) is not a point that has been reliably matched, and thus one-to-many error matches can be avoided, and \( \beta \) is the weighting coefficient of the smoothing term. The method for judging whether a point has been reliably matched uses the following formula:

\[
C(p', d) \leq T_c \text{ and } \left| \text{aver}(\sum_{q' \in \Omega_{p'}} q' \mid g_{p'} \sim g_q) - d_{p'} \right| \leq \omega \tag{8}
\]

Here, \( p' \in ADM, T_c \) is the threshold of reliable matching cost, \( \Omega_{p'} \) is the set of all points within the support window of \( p' \), \( g_{p'} \sim g_q \) indicates that \( p' \) has a similar intensity to \( q' \), \( \text{aver}(\cdot) \) is the mean value, \( d_{p'} \) is the disparity value of \( p' \) in ADM, and \( \omega \) is a user-defined threshold parameter.

Thus, the matching point searched for in Formula (7) must be a point that does not satisfy Formula (8), and if it does, its disparity value can be propagated backward as a reliable disparity value. According to the energy function, Formula (7), the final disparity is obtained by adopting an accepted rule of winner-take-all, and the final dense disparity \( D \) is obtained by minimizing the energy function, as shown in Formula (9).

\[
D_p = \arg\min_d E(p, \hat{d}) \tag{9}
\]

- **Left and right consistency constraints.** Using the cost calculation and constraint methods proposed above to calculate the disparity map of the reference image, the stable, unstable, and occluded points in the disparity map are found according to the left and right consistency constraints, the disparity between unstable and occluded points is removed, holes are filled in using the bilateral filter weight method according to the disparity value of the stable points, and a preliminary disparity
result is obtained. Taking Adirondack as an example, the results of the disparity obtained after applying the various constraint algorithms are as shown in Figure 2.

Figure 2. Disparity results obtained using multiple constraint strategies: (a) disparity obtained from a multi-constrained matching algorithm, where we observe that the disparity map contains some holes caused by occlusions or high matching cost regions, and (b) result of holes filled in using a bilateral filter weight method and mean filtering.

3.2.3 Disparity calculation at the original resolution. After obtaining a subsampling disparity map, we use it to generate the disparity map at the original resolution. Here, we use the interpolation method to fill in the disparity map. The resulting disparity map is shown in Figure 3 (a). The holes in the disparity map then need to be filled in, using a method that fills the holes according to the disparity value of the point that is most similar in intensity to the current hole. The result after filling is shown in Figure 3 (b). At this time, the disparity map can be seen as a rough disparity map, on the basis of which we will obtain the final disparity results through a fine matching process. Searching for a smoother matching point within a small range according to the roughly matching disparity values, the range is \([d - \alpha, d + \alpha]\), where \(d\) is the rough disparity value for each pixel, and \(\alpha\) is the user-defined search parameter. Fine matching also uses Formula (7). Next, we use a weighted mean filter to smooth the disparity map according to the intensity of the image. The result after fine matching is shown in Figure 3 (c).

Figure 3. Results of disparity at the original resolution: disparity maps (a) at the original resolution after interpolation, (b) after filling in the empty disparity areas according to similar adjacent points, and (c) after fine matching

3.3. Sub-pixel refinement
The purpose of sub-pixel refinement is to make the discrete disparity value continuous. Because point-to-point stereo matching can only obtain integer disparity, the accuracy of the resulting disparity map is insufficient. We use the method described in [27] to perform sub-pixel refinement. This method uses a quadratic polynomial interpolation to approximate the cost function between three discrete depth candidates: \(d\), \(d_-\), and \(d_+\), where \(d\) is the discrete depth value of the current point, and \(d_- = d - 1\) and \(d_+ = d + 1\).

\[
f(x) = ax^2 + bx + c
\]

\[
x_{\text{min}} = -\frac{b}{2a}
\]

Here, \(f(x_{\text{min}})\) is the minimum value of the function \(f(x)\), and given \(d\), \(f(d)\), \(f(d_-)\), and \(f(d_+)\), the values of parameters \(a\) and \(b\) can be calculated. Thus,

\[
x_{\text{min}} = d - \frac{f(d_+) - f(d_-)}{2(f(d_+) + f(d_-) - 2f(d))}
\]

where \(x_{\text{min}}\) is the depth with the minimum quadratic cost function \(f(x)\). The resulting sub-pixel refinement disparity map is shown in Figure 4.
4. Experiment and Discussion

4.1. Parameter setting
In our experiments, we used the Middlebury 3.0 benchmark [28], and Middlebury 2003 and 2006 datasets. The program runs on a 3.4 GHz Intel i7 6700 CPU using 8 GB of memory and a GTX 960 image card. For the set of program parameters, the values of the classification parameters s and t in the RT are 5 and 2, respectively. In the downsampling stereo matching process, the cost aggregation window size was $9 \times 9$, and the parameters $\gamma_c$ and $\gamma_g$ used to adjust the similarity and proximity weights in Formula (3) are set to 5 and 9. In Formula (6), the size of the support window for smoothing constraints is $r_s = 5$. For weight parameter $\beta$ of the smoothing constraint term in Formula (7), when applied to the downsampling disparity calculation, $\beta = 0.1$, and when applied to the fine matching of the original image, $\beta = 0.5$. With the uniqueness constraint, to judge whether the matching of a point was reliable, the judgment threshold of matching cost is $T_c = 10$. In Formula (8), the threshold $\omega$ of the difference between the current point disparity and the average disparity of similar points in the window is set to 3, and the neighborhood search range is $\alpha = 5$ for each pixel at the original resolution.

4.2. Experiment results
Our experiment mainly includes two parts: in one, the AD plus rank multi-constraint matching algorithm proposed in this paper is used for comparison with other algorithms on the Middlebury benchmark database; in the other, the results before and after the multi-constraint framework proposed in this paper combined with several classical algorithms are compared.

4.2.1 Experiment results for Middlebury benchmark. Experiments were carried out on the standard Middlebury benchmark dataset [28] to test the performance of the proposed method; here, we use a resolution of one-half for benchmarking. This test benchmark includes three image pairs (Teddy, Adirondack, and Motorcycle), and the experiment results are as shown in Figure 5.

Figure 4. Disparity map after sub-pixel refinement: (a) disparity map by our algorithm, and (b) the ground truth.

Figure 5. Our experiment results on the Middlebury dataset. From top to bottom: Teddy, Adirondack, and Motorcycle. From left to right: left stereo image, ground truth, subsampling disparity map, the final disparity map of our algorithm, and the error of our disparity map. The disparity map errors in non-occluded and occluded regions are marked in black and gray, respectively.
Table 1. Performance of comparison between different stereo matching method for Nonocc, All and Disc (bad2.0 error). The best performance is marked in bold and underlined.

| Algorithm | NTDE | IGSM | MC-CNN-fst | Our method |
|-----------|------|------|------------|------------|
| nonocc    |      |      |            |            |
| Tsukuba   | 0.65 | 0.93 | 1.15       | 1.22       |
| all       | 1.41 | 2.47 | 2.80       | 1.36       |
| disc      | 6.93 | 6.25 | 7.86       | 6.30       |
| nonocc    | 2.73 | 4.08 | 3.91       | 4.07       |
| Teddy     |      |      |            |            |
| all       | 10.90| 14.73| 13.62      | 12.22      |
| disc      | 14.65| 15.78| 16.84      | 14.55      |
| nonocc    | 2.91 | 2.14 | 1.81       | 2.90       |
| Cones     |      |      |            |            |
| all       | 7.28 | 6.64 | 9.66       | 5.92       |
| disc      | 9.75 | 7.50 | 10.35      | 7.73       |
| nonocc    | 8.35 | 7.69 | 8.13       | 7.93       |
| Recycle   |      |      |            |            |
| all       | 12.74| 13.27| 14.22      | 12.35      |
| disc      | 15.69| 14.36| 16.37      | 15.14      |
| Average Error(all) | 8.08 | 9.28 | 10.08 | 7.96 |

In this paper, we use other state-of-the-art stereo matching algorithms to compare with our algorithm, i.e., NTDE [11], IGSM [12], and MC-CNN-fst [21]. The results are shown in Table 1, where “Nonocc” denotes non-occluded areas, “all” denotes all areas, and “disc” denotes discontinuous areas. The performance of our algorithm in the non-occluded regions is not the best but has the highest accuracy within the entire image region because the algorithm constrains the disparity matching in the smooth region, and in the edge area we use a bilateral filter weight similar to IGSM [12]; thus, our algorithm achieves a better performance in the smooth and discontinuous regions, and the final matching effect of our algorithm is the best among these stereo matching algorithms.

4.2.2 Performance of multi-constraint framework combined with other stereo matching methods. The core of our algorithm is the addition of a multi-constraint method to improve the accuracy of matching during the cost aggregation process. We use our multi-constraint method as a framework to combine with other existing algorithms. Here, we use the classic NL [29], BF [2], GF [3], and CT [9] for comparison. Taking the Cones and Tsukuba images as an example, we first use the NL, BF, GF, and CT algorithms to compute the corresponding disparity maps, and then combine our proposed constraint framework (described in Section 3.2.2) into each algorithm to recalculate the disparity maps. The experiment results are as shown in Figure 6.

The experiment results are not refinement results. Based on these results, our proposed constraint framework combined with these algorithms improved accuracy, and the average error matching rate decreased by more than 3%. It shows that our constraint strategy is effective, and can be used as a separate framework to combine with other matching algorithms to improve matching accuracy.
5. Conclusion
This paper proposed a stereo matching algorithm based on an improved multi-constraint strategy. The original reference images were downsampled to reduce the search space and complexity of the stereo matching. During the matching process of the downsampled images, using AD and rank as the matching cost, we establish two matrices: ADM and MCM. We use the proposed smoothing constraints, uniqueness constraints, continuity constraints, and left and right consistency constraints to build a stereo matching constraint framework and perform stereo matching. Experimental results show that the proposed multi-constraint method can obtain more accurate disparity values, especially in large smooth regions and edge regions. Furthermore, our constraint strategy as a framework, can be extracted to combine with other algorithms, thereby significantly improving the accuracy of the other algorithms as well.

Acknowledgement
This work was supported by the Scientific Research Foundation of Chongqing University of Technology, China (2019ZD112)

Reference:
[1] D. Scharstein and R. Szeliski, “High-accuracy stereo depth maps using structured light,” in Proc. IEEE Comput. Soc. Conf. CVPR, vol. 1, pp. I-195–I-202, Jun. 2003.
[2] K.-J. Yoon and I. S. Kweon, “Adaptive support-weight approach for correspondence search,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 4, pp. 650–656, 2006.
[3] A. Hosni, C. Rhemann, et al., “Fast cost-volume filtering for visual correspondence and beyond,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 2, pp. 504–511, 2013.
[4] Hong, Gwang Soo, and B. G. Kim. "A local stereo matching algorithm based on weighted guided image filtering for improving the generation of depth range images." Displays 49(2017):80-87.
[5] D. Chen, M. Ardabilian, and L. Chen, “A novel trilateral filter based adaptive support weight method for stereo matching,” in Proc. Brit. Mach. Vis. Conf., pp. 1–11, Oct. 2013.
[6] S. Kim, B. Ham, B. Kim, and K. Sohn, “Mahalanobis distance cross-correlation for illumination-invariant stereo matching,” IEEE Trans. Circuits Syst. Video Technol., vol. 24, no. 11, pp. 1844–1859, Nov. 2014.
[7] Y. S. Heo, K. M. Lee, and S. U. Lee, “Robust stereo matching using adaptive normalized cross-correlation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 4, pp. 807–822, Apr. 2011.
[8] Son, H. S., et al. "A real-time stereo matching hardware architecture based on the AD-Census." International Journal of Multimedia & Ubiquitous Engineering, Vol. 19, pp:195 - 198, 2013.

[9] W. S. Fife and J. K. Archibald, “Improved census transforms for resource-optimized stereo vision,” IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 1, pp. 60–73, Jan. 2013.

[10] Gu, Zheng, et al. "Local stereo matching with adaptive support-weight, rank transform and disparity calibration." Pattern Recognition Letters 29.9:1230-1235, 2008.

[11] Kim, Kyung Rae, and C. S. Kim. "Adaptive smoothness constraints for efficient stereo matching using texture and edge information." IEEE International Conference on Image Processing IEEE, 3429-3433, 2016.

[12] Zhan, Yunlong, et al. "Accurate Image-Guided Stereo Matching With Efficient Matching Cost and Disparity Refinement." IEEE Transactions on Circuits & Systems for Video Technology 26.9:1632-1645, 2016.

[13] S. Birchfield and C. Tomasi, “Depth discontinuities by pixel-to-pixel stereo,” Int. J. Comput. Vis., vol. 35, no. 3, pp. 269–293, 1999.

[14] Cao, Guozhen, and H. Peng. "Graph Cut Stereo Matching Based on Region Consistency." Journal of Northwestern Polytechnical University 35.1(2017):160-163.

[15] Yang, M., et al. "A Denoising Method for Randomly Clustered Noise in ICCD Sensing Images Based on Hypergraph Cut and Down Sampling." Sensors 17.12(2017):2778.

[16] Ahn, Sungsoo, et al. "Maximum Weight Matching using Odd-sized Cycles: Max-Product Belief Propagation and Half-Integrality." IEEE Transactions on Information Theory 64.3(2018):1471-1480.

[17] K. Yamaguchi, T. Hazan, D. MCAllester, and R. Urtasun,"Continuous markov random fields for robust stereo estimation," in Proc. IEEE ECCV, pp. 45–58, Oct. 2012.

[18] Park, Hae-sol, and K. M. Lee. "Look Wider to Match Image Patches with Convolutional Neural Networks." IEEE Signal Processing Letters PP.99:1-1, 2016.

[19] Hirshmuller, Heiko. "Stereo Vision in Structured Environments by Consistent Semi-Global Matching." CVPR, 2006.

[20] Peng, Yao, et al. "Iterative color-depth MST cost aggregation for stereo matching." IEEE International Conference on Multimedia & Expo 2016.

[21] J.Žbontar and Y. LeCun, “Stereo matching by training a convolutional neural network to compare image patches,” arXiv:1510.05970, 2015.

[22] Luo, Wenjie, A. G. Schwing, and R. Urtasun. "Efficient Deep Learning for Stereo Matching." Computer Vision & Pattern Recognition 2016.pp: 5695-5703.

[23] Gu, Zheng, et al. "Local stereo matching with adaptive support-weight, rank transform and disparity calibration." Pattern Recognition Letters 29.9:1230-1235, 2008.

[24] S. Hermann and R. Klette, "Evaluation of a new coarse-to-fine strategy for fast semi-global stereo matching," in Advances in Image and Video Technology, vol. 7087. Springer, pp. 395–406, 2011.

[25] Q. Yang, "A non-local cost aggregation method for stereo matching," in Proc. IEEE Conf. CVPR, pp. 1402-1409, Jun. 2012.