SEGMENT-TREE BASED COST AGGREGATION FOR STEREO MATCHING WITH ENHANCED SEGMENTATION ADVANTAGE

Peng Yao[12], *Hua Zhang*[12], *Yanbing Xue*[12], *Mian Zhou*[12], *Guangping Xu*[12], *Zan Gao*[12], *Shengyong Chen*[12]
1 Key Laboratory of Computer Vision and System (Ministry of Education), Tianjin University of Technology, Tianjin, China
2 Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin University of Technology, Tianjin, China

ABSTRACT

Segment-tree (ST) based cost aggregation algorithm for stereo matching successfully integrates the information of segmentation with non-local cost aggregation framework. The tree structure which is generated by the segmentation strategy directly determines the final results for this kind of algorithms. However, the original strategy performs unreasonable due to its coarse performance and ignores to meet the disparity consistency assumption. To improve these weaknesses we propose a novel segmentation algorithm for constructing a more faithful ST with enhanced segmentation advantage according to a robust initial over-segmentation. Then we implement non-local cost aggregation framework on this new ST structure and obtain improved disparity maps. Performance evaluations on all *31* Middlebury stereo pairs show that the proposed algorithm outperforms than other *five* state-of-the-art aggregated based algorithms and also keeps time efficiency.

Index Terms— Segment-tree, stereo matching, non-local aggregation, segmentation advantage

1. INTRODUCTION

According to the related works which have been reported in [1-4], stereo matching algorithms can be categorized into two groups: global and local algorithms. Both categories are often implemented with one or all of the following *four* steps: matching cost computation; cost aggregation; disparity computation; and disparity refinement. Global algorithms, such as Belief Propagation algorithm [5] makes explicit smoothness assumptions and minimize a predefined energy function for obtaining optimal result. Despite the reliable results generated, they still perform time-consumed. Local algorithms, such as Adaptive Support Weight (ASW) algorithm [6] performs matching cost computation at first and then implement cost aggregation within a local window. After that, a Winner-Takes-All (WTA) strategy is employed for disparity computation which selects the corresponding disparity value of the minimum aggregated cost for each pixel. At last, some disparity refinement strategies such as Left-Right Cross-check (LRC) and Occlusion Filling (OF) are widely used for obtaining final results. Unfortunately, they perform faster but still inferior in accuracy.

Yang proposed a non-local cost aggregation algorithm [7, 8] which performed much better than traditional local ones. However, his work is limited by solving the matching problem on a Minimum Spanning Tree (MST) structure. As a variant, Mei *et al.* proposed a Segment-tree (ST) based algorithm [9] which introduced segmentation information into non-local cost aggregation framework and was reported to perform better than MST based algorithm. More recently, a Cross-Scale framework has also been proposed for improving some existing local and non-local aggregated based algorithms [10, 11].

For ST based algorithm, the tree structure is directly determined by the segmentation strategy. However, the original strategy performs too relaxed at the beginning of grouping pixels into segments and too constrained at last, which inevitably leads coarse performances on initial segmentation results. More importantly, these imperfect segments weakly match the disparity consistency assumption where the pixels are more likely to share similar disparities and get higher supported weights for each other in one natural segment. That means the original ST based algorithm limitedly utilize the segmentation advantage while implementing non-local cost aggregation framework.

The main contributions to our work are we propose a novel segmentation strategy and effectively employ it into ST based algorithmic framework. The most praiseworthy thing is the improved ST based algorithm not only achieves better results but also keeps time efficiency.

This paper is organized as the following. We firstly have a brief review of original ST based algorithm. Then a novel segmentation strategy is proposed for constructing a more faithful ST structure. Based on this new tree structure, we implement the non-local cost aggregation and yield superior results. Performance evaluations on all *31* Middlebury stereo pairs with other *five* state-of-the-art aggregated algorithms are evaluated in experimental section. We draw a conclusion at the last section.

2. SEGMENT-TREE BASED ALGORITHM

Based on the main idea of [12], ST based algorithm performs two connected neighbored pixels with one edge and computes the color dissimilarity as the tree's weight on the whole graph. Each edge weight connects a pair of neighbored pixels s and r can be computed as:

$$\omega_e = \omega(s,r) = \left| I(s) - I(r) \right| \tag{1}$$

For a color image I, the edge weight $\omega(s,r)$ is the maximum value estimated from *three* RGB channels.

In ST based algorithm, the final aggregated cost $C_d^A(p)$ is:

$$C_d^A(p) = \sum_{q \in I} S(p,q) \cdot C_d(q) \tag{2}$$

Where $C_d(q)$ denotes the matching cost volume of disparity value d with the color-gradient based method [13, 14] and

$$S(p,q) = \exp\left(\frac{-D(p,q)}{\sigma}\right) \tag{3}$$

denotes the supported weight from p to any pixel $q \in I$ on a unique path $P(p,q)$ of the tree; $D(p,q)$ denotes the weights summation of (1) on $P(p,q)$, σ is a parameter of user-specified. It is clear that the best supported weight $S(p,q)$ is related to the minimum of $D(p,q)$, which determines the aggregated results of (2). At the same time, from (3) we can infer that the pixels lie far from or dissimilar to p would not receive relative larger supported weights.

The procedures of constructing the ST structure can be illustrated as in **Fig.1**. For the sake of simplicity, we only use *four* nodes to exhibit. In *initialization* step (**Fig.1 (a)**), all the pixels are treated as subtrees on the graph; and for all pairs of subtrees' connected weights which satisfied as:

$$\omega_{e_j} \le \min\left(Int(T_p) + \frac{k}{|T_p|}, Int(T_q) + \frac{k}{|T_q|}\right) \tag{4}$$

Two subtrees would be connected as a new larger subtree in *grouping* step (**Fig.1 (b)**). Where $Int(T_p)$ and $|T_p|$ denote the maximum edge weight and the region size of subtree T_p respectively; k is a user specified parameter. It is easily to infer that the ST structure is directly determined by (4). After *grouping* step, a *linking* step is enforced to integrate each subtree into a whole ST structure (**Fig.1 (c)**). Then the tree-based non-local cost aggregation can be implemented on it. The same as in [7-11, 15], the intermediate *two-pass* cost aggregation on a tree structure can be represented as:

$$\begin{aligned} C_d^A(p) &= S(\Pr(p), p) \cdot C_d^A(\Pr(p)) \\ &\quad + (1 - S^2(p, \Pr(p))) \cdot C_d^{A\uparrow}(p) \end{aligned} \tag{5}$$

Where $Pr(p)$ denotes the parent of pixel p. After that, WTA strategy is enforced as disparity computation for obtaining the initial results. This is called ST-1 algorithm.

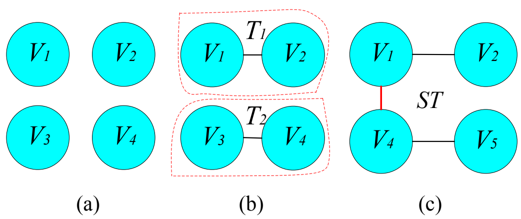

Fig.1 The procedures of constructing a ST structure. (a) *Initialization*. (b) *Grouping*. (c) *Linking*. Here we assume that in *grouping* V_1 and V_2, V_3 and V_4 are connected to construct two subtrees; in *linking* V_1 and V_4 are connected (the connected line segment is marked with red color in (c)).

Further an enhanced strategy of color-depth weight is employed after ST-1 algorithm. The main idea behind this is to introduce depth information into weight function and rebuild the ST structure. The jointed color-depth weight can be represented as:

$$\omega_e = \lambda \frac{|I(s) - I(r)|}{\Delta_I} + (1-\lambda)\frac{|D(s) - D(r)|}{\Delta_D} \tag{6}$$

Where Δ_I and Δ_D are the normalized parameters for color image I and depth image D, λ used for balancing the relative contributions of color and depth. D is computed by the ST-1 algorithm. With the color-depth weight, an enhanced ST is constructed and the non-local cost aggregation is also implemented. According to the second enforcement of WTA strategy, ST-2 algorithm is performed. At last, with the disparity refinement, final results are yielded.

3. PROPOSED ALGORITHM

As mentioned above we can infer that the ST structure directly determines the final result of non-local cost aggregation. That means the segmentation algorithm plays an important role for constructing the ST structure and provides segmentation advantage while implementing non-local cost aggregation. However, in formula (4) the original segmentation algorithm generally performs underwhelming due to the following drawbacks: Firstly, k does not control the desired region size very well. A large k leads to a much relaxed threshold at the beginning of *grouping* and performs under-segmentation regions with inconsistent boundaries. On the contrary, a small k performs better at the beginning of *grouping* but inevitably leads to consistent over-segmentation and too many boundaries. In this circumstance, small segments with too few pixels would not reach relative effective supported weights from neighbors. That means too many small segments could lead the non-local aggregation trap into "localization". Secondly, as the region size $|T|$ grows, the term $k/|T|$ decreases dramatically which becomes increasingly unreliable for discriminating between regions of the same type. At the same time, when $|T|$ increases, the connecting decision function would perform much too constrained. It will break the homogeneous regions into different segments and contrary to the disparity consistency assumption. More importantly, while implementing non-local cost aggregation on this imperfect ST structure would lead the substandard utilization of segmentation advantage.

Based on the reasons we have mentioned above, an algorithm is proposed for constructing a new ST structure. In our implementation, the connected decision function can be formulated as follows:

$$\omega_{e_j} \le \min\left(Int(T_p) + k \cdot \sqrt{|T_p|}, Int(T_q) + k \cdot \sqrt{|T_q|}\right) \tag{7}$$

Here we choose k as *0.02*. We perform the algorithm mainly on the following *two* considerations: For one thing, the proposed algorithm begins with a more constrained threshold while *grouping*. Compared with original larger k (*k=1200* in original segmentation algorithm), this choice of k keeps most of the boundaries are coincident with true region borders. For another thing, with the increasing of the region size $|T|$, the proposed segmentation algorithm performs a more relaxed threshold. It could enforce more similar regions merged and perform more compatible with disparity consistency assumption. Due to these reasons, the proposed algorithm could provide enhanced segmentation advantage. Moreover, in order to prevent the fast increasing of region size $|T|$, we also perform a square root of it. In general, the proposed algorithm provides a constrained threshold at the beginning and a relaxed threshold while merging regions. What's more, it also better meets the disparity consistency assumption.

With formula (7), an Improved-ST (IST) based non-local cost aggregation algorithm for stereo matching can be implemented, which is descripted as follows:

Alogrithm1 Improved-ST Algorithm

for each $p \in I$
  for $d \in [0, maxdis]$
    Compute matching cost
  end for
end for
Compute D by ***ST*** based algorithm with formula (7)
Construct updated ***ST*** with Color-Depth weight:

$$\omega(s,r) = \lambda \frac{|I(s) - I(r)|}{\Delta_I} + (1-\lambda)\frac{|D(s) - D(r)|}{\Delta_D}$$

Aggregate costs and get D_I as updated result

4. EXPERIMENTAL RESULTS

In this section, we have two parts to show the experimental results. To ensure fairness, there is no disparity refinement step introduced in **4.1**; and we also set the same error threshold for each disparity map (*1.0* pixel). For all the algorithms we have tested the error rates in *non-occluded* regions. But for final results with disparity refinement in **4.2**, we also have evaluated the error rates of *all* pixels. For testing the time efficiency in fair, all the algorithms are implemented on a same PC platform with a *3.60GHz Intel Core i7* CPU, *16GB* RAM and *64-bits* OS; furthermore, all the implementations are estimated by using the *C++* code.

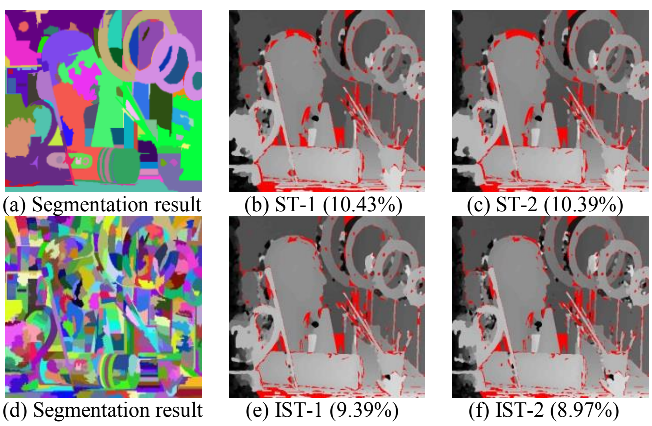

Fig.2. Comparisons of different ST based stereo matching algorithms. (a) Result of original segmentation algorithm on color image. (b), (c) Results of ST-1 and ST-2. (d) Result of proposed segmentation algorithm on color image. (e), (f) Results of IST-1 and IST-2. All the bad pixels are marked with red dots and the error rates in *non-occluded* regions are indicated below. Error threshold is *1.0* pixel.

At first, **Fig.2** visualizes the segmentation results of color images and disparity maps of *Art* with different algorithms. We name IST-1, IST-2 as the improved ST-1 and ST-2 algorithms. To ensure fairness, here for all the impleme-ntations we employ the same textureless handling matching cost computation method which has been proposed in [15]. In addition, there is no disparity refinement step introduced. Compared with the original color segmentation result (**Fig.2 (a)**), ours (**Fig.2 (d)**) provides more segments and preserves more important boundaries. Due to this segmentation advantage, the proposed algorithm constructs a more faithful ST structure and perform lower error rates in *non-occluded* regions (as shown in **Fig.2 (e)** and **Fig.2 (f)**).

4.1. Evaluations without Disparity Refinement

In this part, our proposed IST-2 and other *five* state-of-the-arts aggregated based algorithms are estimated: MST [7], ST-2 [9], CS-GF (Guided Filter), CS-MST and CS-ST; where "CS" denotes the Cross-Scale algorithm [10, 11].

For IST-2 we set $\{\sigma, k, \lambda\}$ = {*0.08, 0.02, 0.5*} on *Tsukuba*, *Venus*, *Teddy* and *Cones*; and other *27* Middlebury stereo pairs $\{\sigma, k, \lambda\}$ = {*0.04, 0.02, 0.5*}. Here we use the matching cost computation method which was proposed by us in [15]:

$$C'(p,p') = \log(1 + \exp(C(p,p'))) \quad (8)$$

Where

$$C(p,p') = \alpha \cdot \min(\|I_p - I_{p'}\|, \tau_{col}) + (1 - \alpha) \cdot \min(\|\nabla I_p - \nabla I_{p'}\|, \tau_{grd}) \quad (9)$$

And we set the parameters as: $\{\alpha, \tau_{col}, \tau_{grad}\}$ = {*0.11, 7, 2*} in IST-2. For other *five* estimated algorithms, we remain holding the original parameters in their implementations.

Table 1 Performance evaluations without disparity refinement on all *31* Middlebury stereo pairs by *six* algorithms. Error threshold is *1.0* pixel and unit of time is second. Only *non-occluded* regions are evaluated.

Stereo Pairs	**CS-GF**	**MST**	**ST-2**	**CS-MST**	**CS-ST**	**IST-2**
Tsukuba	2.35_{6}	2.12_{5}	1.65_{2}	1.72_{3}	1.90_{4}	**1.45_{1}**
Venus	1.26_{4}	0.91_{3}	0.52_{2}	2.13_{5}	2.63_{6}	**0.37_{1}**
Teddy	6.87_{4}	7.61_{6}	7.48_{5}	**5.25_{1}**	5.53_{2}	6.83_{3}
Cones	3.19_{2}	4.10_{4}	3.50_{3}	4.29_{5}	4.55_{6}	**3.05_{1}**
Aloe	5.61_{6}	4.16_{2}	4.51_{3}	4.79_{4}	4.90_{5}	**3.55_{1}**
Art	9.20_{2}	9.79_{3}	10.99_{6}	10.84_{5}	10.46_{4}	**8.97_{1}**
Baby1	3.98_{2}	7.46_{5}	4.79_{4}	8.17_{6}	4.43_{3}	**3.30_{1}**
Baby2	**3.46_{1}**	12.03_{3}	16.32_{6}	13.53_{4}	15.15_{5}	6.91_{2}
Baby3	5.68_{3}	5.69_{4}	5.27_{2}	7.15_{6}	6.27_{5}	**2.69_{1}**
Books	8.32_{2}	9.57_{3}	9.57_{4}	10.02_{6}	9.84_{5}	**7.26_{1}**
Bowling1	13.03_{2}	17.13_{4}	16.48_{3}	20.84_{5}	22.11_{6}	**8.94_{1}**
Bowling2	6.62_{2}	9.32_{3}	10.29_{4}	10.41_{5}	10.91_{6}	**5.33_{1}**
Cloth1	1.31_{6}	0.51_{3}	0.41_{2}	0.72_{4}	0.77_{5}	**0.17_{1}**
Cloth2	3.52_{3}	2.85_{2}	3.55_{4}	4.19_{6}	4.06_{5}	**1.84_{1}**
Cloth3	2.26_{4}	1.79_{3}	1.69_{2}	2.63_{6}	2.60_{5}	**0.89_{1}**
Cloth4	1.75_{5}	1.29_{3}	1.13_{2}	1.87_{6}	1.69_{4}	**0.71_{1}**
Dolls	4.95_{2}	4.98_{3}	5.92_{5}	5.93_{6}	5.65_{4}	**4.25_{1}**
Flowerpots	12.22_{2}	16.69_{5}	16.02_{4}	18.81_{6}	15.77_{3}	**4.81_{1}**
Lampshade1	**8.88_{1}**	10.43_{3}	10.79_{5}	11.79_{6}	10.63_{4}	9.81_{2}
Lampshade2	16.53_{3}	23.90_{6}	21.17_{5}	16.97_{4}	15.35_{2}	**8.21_{1}**
Laundry	13.12_{4}	13.69_{5}	13.04_{3}	12.85_{2}	14.69_{6}	**12.67_{1}**
Midd1	36.92_{5}	32.39_{4}	32.97_{6}	26.98_{3}	26.47_{2}	**24.12_{1}**
Midd2	30.06_{3}	34.80_{6}	32.44_{5}	31.79_{4}	**23.21_{1}**	25.20_{2}
Moebius	9.56_{6}	**7.67_{1}**	8.36_{3}	8.74_{5}	8.67_{4}	7.69_{2}
Monopoly	25.11_{6}	24.18_{4}	24.57_{5}	23.32_{3}	22.96_{2}	**12.70_{1}**
Plastic	**29.55_{1}**	42.53_{5}	35.79_{3}	46.52_{6}	42.32_{4}	34.30_{2}
Reindeer	7.23_{3}	9.56_{5}	7.01_{2}	9.74_{6}	8.33_{4}	**6.16_{1}**
Rocks1	2.43_{4}	2.35_{3}	2.12_{2}	2.82_{6}	2.61_{5}	**1.70_{1}**
Rocks2	1.59_{4}	1.59_{3}	1.54_{2}	2.15_{6}	2.07_{5}	**1.09_{1}**
Wood1	4.30_{2}	8.70_{5}	5.17_{3}	10.59_{6}	5.69_{4}	**3.19_{1}**
Wood2	2.92_{4}	0.99_{2}	2.75_{3}	4.88_{5}	6.10_{6}	**0.97_{1}**
Avg. Error	9.15_{2}	10.67_{4}	10.25_{3}	11.05_{6}	10.27_{5}	**7.07_{1}**
Avg. Rank	3.35_{2}	3.74_{4}	3.55_{3}	4.87_{6}	4.26_{5}	**1.23_{1}**
Avg. Time(s)	7.67_{6}	**0.56_{1}**	0.87_{2}	1.69_{5}	1.64_{4}	1.33_{3}

Table 1 shows the detailed results on all *31* Middlebury stereo pairs. The normal numbers are the percentages of the error pixels in *non-occluded* regions and the subscript numbers are the relative rank in each row. Among the *31* Middlebury stereo pairs, IST-2 ranks *1* of *25/31* stereo pairs (marked with bold fonts in corresponding row). The average rank and average error rate are computed in the reciprocal second and third rows. We can easily conclude that IST-2 performs best overall accuracy and the best overall rank among *six* algorithms. For time efficiency shows in the last row, IST-2 ranks *three* among *six* algorithms.

Corresponding to **Table 1, Fig.3** visualizes some of our initial results compared with other *five* algorithms. For all the disparity maps are indicated with error rates in *non-occluded* regions below and the error pixels are also marked with red color dots. Error threshold is still set as *1.0* pixel. It shows definitely that IST-2 performs much better than other *five* evaluated algorithms in matching accuracy.

4.2. Evaluations with Disparity Refinement

To obtain final results, a disparity refinement step is also needed in stereo matching algorithm. For all the estimated algorithms, we have implemented with the same disparity refinement strategy. Different from the former contents in **4.1**, here we also have measured the error rates of *all* pixels (error threshold is also set as *1.0* pixel). **Table 2** simply shows the average error rates and running time of all the *six* algorithms. It reflects that with disparity refinement all the results have reached improvement than themselves without disparity refinement in *non-occluded* regions. Even in this circumstance, our proposed algorithm remains keeping the top *1* rank among *six* algorithms both in *non-occluded* regions and *all* pixels. Although the running time increases, our algorithm still performs shorter time than CS-MST, CS-ST and CS-GF algorithms. Especially for CS-GF, it performs the longest implementing time.

Table 2 Performance evaluations with disparity refinement on *31* Middlebury stereo pairs by *six* algorithms. Error threshold is *1.0* pixel and unit of time is second.

Algorithms	CS-GF	MST	ST-2	CS-MST	CS-ST	IST-2
Avg. Error(non-occ)	8.46	9.83	9.42	9.93	8.40	**6.52**
Avg. Error(all)	15.77	16.98	18.32	17.71	16.31	**13.78**
Avg. Time(s)	9.10	**0.75**	1.12	3.38	3.27	1.45

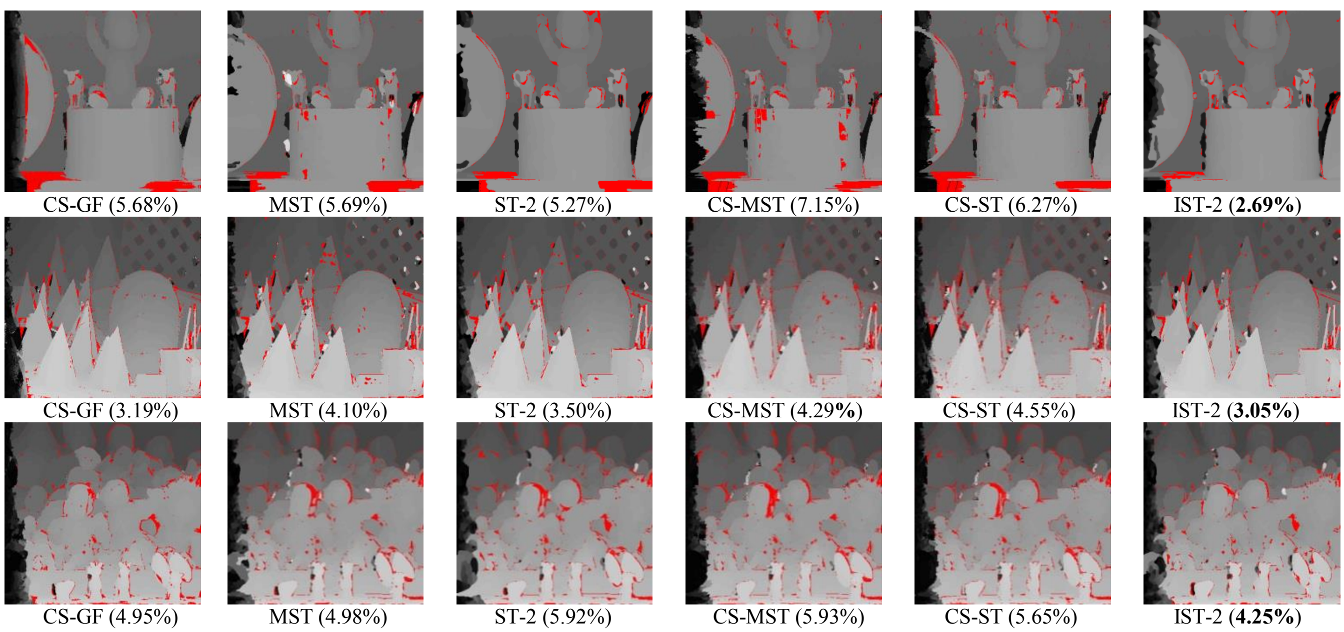

Fig.3. Disparity maps of *Baby3, Cones* and *Dolls* without disparity refinement ordered from top to bottom rows. The error rates in *non-occluded* regions are indicated below and the bold fonts represent the lowest. The error pixels are marked with red color dots and the error threshold is *1.0* pixel.

5. CONCLUSION

In this paper, we improved the ST based stereo algorithm by using a novel segmentation algorithm. Firstly, it provides an enhanced segmentation advantage and constructs a more faithful ST structure. Secondly, it also better meets the disparity consistency assumption. Based on this new tree structure, an Improved Segment-tree (IST) non-local cost aggregation algorithm can be performed. Performance evaluations show that the proposed algorithm outperforms than other *five* aggregated based algorithms on all *31* Middlebury stereo pairs and time consuming does not increase too much.

In the future, we plan to employ various improved or other segmentation algorithms into the ST based stereo algorithmic framework for obtaining superior results.

6. ACKNOWLEDGEMENTS

This research has been supported by National Natural Science Foundation of China (U1509207, 61325019, 61472278, 61403281 and 61572357), Key project of Natural Science Foundation of Tianjin (14JCZDJC31700).

7. REFERENCES

[1] D.Scharstein, R.Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *In SPRINGER International Journal of Computer Vision (IJCV), 2002, pp. 3017-3024.*

[2] F.Tombari, S.Mattoccia and L.Stefano *et al*., "Classification and evaluation of cost aggregation methods for stereo correspondence," *In Computer Vision and Pattern Recognition (CVPR*), 2008, *pp. 1-8*.

[3] M.Gong , R.Yang and L.Wang *et al.*, "A performance study on different cost aggregation approaches used in real-time stereo matching", *In International Journal of Computer Vision (IJCV), 2007, pp. 283-296*.

[4] W.Yu, T.Chen and J.C.Hoe, "Real time stereo vision using exponential step cost aggregation on GPU," *In IEEE International Conference on Image Processing (ICIP)*, 2009, *pp. 4281-4284.*

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

[6] K.-J.Yoon, I.S.Kweon, "Adaptive support-weight approach for correspondence search," *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 28(4), 2006, pp.650-656.*

[7] Q.Yang, "A Non-Local Cost Aggregation Method for Stereo Matching," *In IEEE Computer Vision and Pattern Recognition (CVPR), 2012, pp. 1402-1409.*

[8] Q.Yang, "Stereo Matching Using Tree Filtering," *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 37(4), 2015, pp. 834-846.*

[9] X.Mei, X.Sun and W.Dong *et al*., "Segment-Tree based Cost Aggregation for Stereo Matching," *In IEEE Computer Vision and Pattern Recognition (CVPR), 2013, pp. 313-320.*

[10] K.Zhang, Y.Fang and D.Min *et al*., "Cross-Scale Cost Aggregation for Stereo Matching," *In IEEE Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1590-1597.*

[11] K.Zhang, Y.Fang and D.Min *et al*., "Cross-Scale Cost Aggregation for Stereo Matching," *In IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), 2016, pp. 1-1*.

[12] P.F.Felzenszwalb, D.P.Huttenlocher, "Efficient graph based image segmentation," *In SPRINGER International Journal of Computer Vision (IJCV), 59 (2), 2004, pp.167-181.*

[13] C.Rhemann, A.Hosni and M.Bleyer *et al*., "Fast cost-volume computation for visual correspondence and beyond," *In IEEE Computer Vision and Pattern Recognition (CVPR), 2011, pp. 3017-3024.*

[14] A.Hosni, C.Rhemann and M.Bleyer *et al*., "Fast cost-volume computation for visual correspondence and beyond," *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 32(2), 2013, pp. 504-511.*

[15] P.Yao, H.Zhang and Y.Xue *et al.*, "Iterative Color-Depth MST Cost Aggregation for Stereo Matching," *In IEEE International Conference on Multimedia and Expo (ICME), 2016, pp.1-6.*