# A Window-Based Adaptive Correspondence Search Algorithm Using Mean Shift and Disparity Estimation

Shujun Zhang, Jianbo Zhang and Yun Liu

College of Information Science & Technology, Qingdao University of Science and Technology

***Abstract*—Current methods to solve the problem of binocular stereo matching can be divided into two categories: sparse points based methods and dense points based methods. However, both of them have different shortcomings and limitations. There is no perfect method to solve the disparity problem. Dense points based techniques relatively obtain more accurate results but with higher computation. A large number of window-based adaptive corres-pondence techniques have emerged in recent years. In order to solve the problem of high time complexity and large amount of calculation in matching process, we propose a new window-based correspondence search algorithm using mean shift and disparity estimation. Mean shift can aggregate the same or similar colors so it can be applied to pre-process the source images to reduce their dynamic color range. Disparity estimation is conducted on the pre-processed two images to compute disparities of uniform texture regions. Adaptive window matching through similarity computation and window-based support aggregation is finally executed and exact depth map is obtained. Experimental results show that our algorithm is more efficient and keeps smooth dis-parity better than the prior window method.**

***Index Terms* —Window-based adaptive correspondence, mean shift, disparity estimation, , binocular stereo matching.**

## I. INTRODUCTION

The research of binocular stereo vision is the foundation and hotspot of computer vision. It is widely used in various fields including pattern recognition, stereo television, satellite remote sensing, intelligent robotics, industrial inspection and identification, etc. Stereo matching is the core technique for stereo vision. That is finding out the relationship between pixels which are in the same position but in different view point of two images or multiple images. In recent years, with the development of virtual reality and 3D reconstruction research, higher requirements to stereo correspondence are put forward.

Binocular vision is the process of recovering depth from the left image and the right image with the same height and width in a certain distance, similar to human vision principle. During the process, stereo matching is the key point, which means to find the correspondence pixels of the same physical spatial point on both images. The relationship between two pixels can be expressed by disparity. When a point in 3D space is projected to two different-position camera planes, there will be a position distance between two points on those two images. That distance is the so-called disparity. If the two images are calibrated, the distance will relate to the offset *x* only. This offset *x* is disparity.

Binocular stereo matching algorithms fall into two categories. One is based on sparse points, and the other is based on dense points. The former one emphasizes more on local region, and matching is constrained by local points. This category includes region matching [1], feature-points matching [2, 3], phase matching algorithm [4, 5] and so on. They only use interest points and neighboring points to compute disparity. Matching procedure in these methods needs less information, so the calculation amount is small and the time complexity is low. The advantages are high efficiency and low memory occupancy. But their disadvantages are also obvious for they are sensitive to noise and cannot reach a satisfactory result when matching region is of single color with no texture area or occlusion area.

In contrast, the latter category can overcome those shortcomings. The algorithm based on dense points finds all constraints from the whole image not local parts. This will keep the structure information of an image, and save the real information of the image. Algorithms based on dense points are better than those based on sparse points on accuracy but bring high time complexity. There are many binocular dense-point matching methods in which the representatives are BirchField algorithm [6] and Yoon algorithm [7, 8].

BirchField algorithm is used to match two gray-scale images. It applies dynamic programming into matching epipolar lines. It is efficient but inaccurate. Subsequent improvement method [9] of BirchField is based on each pixel's eight neighborhoods and improves the matching accuracy to some extent. In the study of color image matching, Yoon's method of adaptive window algorithm [7] is a milestone and it also becomes the main focus of research in recent years. When aggregating support to measure the similarity between image pixels, support from a neighboring pixel is valid only when the neighboring pixel is from the same depth – it has the same disparity – as the pixel under consideration. Therefore, the support-weight should be in proportion to the probability that the pixels have the same disparity. Yoon' algorithm is a global traversal algorithm, when matching two images it travels point by point until it obtains the final disparity image. Its time complexity is: O (image width * image height * the default maximum image

Manuscript Received on June 2, 2011
E-mail: lindazsj@163.com

disparity * square of matching window size * matching computation). The strength of this algorithm is accurate, but the time complexity is comparably high. It cannot be used for real-time systems. And it shows not so good result in uniform texture regions. The improvement methods currently include GPU acceleration [10], BP algorithm, graph cut method [11, 12] and so on [15, 16, 17, 18, 19].

The existing algorithms are lack of accuracy, such as Birchfield algorithm, or the time complexity is too high, such as Yoon's algorithm. And there are also many other problems. In this paper, we propose a rapid adaptive window matching algorithm based on mean shift and disparity estimation. We will call our algorithm rapid adaptive window correspondence for short below. The major innovative point is to combine color aggregation with local disparity estimation and adaptive window correspondence. It is able to accomplish a better matching accuracy while effectively reducing the time complexity so as to improve the performance of the algorithm. Experimental results show that our algorithm is a satisfactory compromise method in binocular stereo matching between time consuming and calculation accuracy. Firstly, rapid adaptive window correspondence aggregates all color pixels in reference image and target image using mean shift filtering. Then we obtain two images with low dynamic color range. Secondly, rapid adaptive window correspondence deals with these two images using Brichfield algorithm for binocular stereo matching to acquire an initial depth map. Through this procedure disparity is rapidly estimated and a general depth map is supplied for the next step. Finally, our algorithm matchs reference image to target image using the rapid adaptive window correspondence method and completes the final exact depth map.

The steps of our algorithm are shown as below:

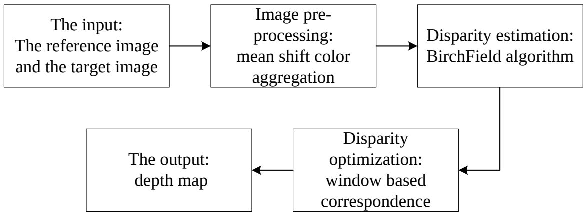


Fig. 1. The steps of the proposed algorithm

## II. MEAN SHIFT COLOR AGGREGATION

Mean shift algorithm was first proposed in estimation of probability density function [8] and its original meaning was "shift the mean vector". Afterwards, it was successfully used to process images or track videos [9]. It becomes an important algorithm in computer vision and image processing field. The advantage of mean shift is reducing the high dynamic color range to low color range which makes the image simpler.

In the first step of rapid adaptive window correspondence, we use mean shift for image pre-processing. This step is very important in our algorithm in the aspect of reducing computation amount. One of the reason of high time complexity binocular stereo correspondence of dense point is that the dynamic color range of the image is large, therefore the color difference between each pixel pair is various even they are in the same depth., The key idea to solve this problem is using a suitable method to reduce the dynamic range of the images especially at the same disparity area in order to cut down the redundant matching computation and enhance the efficiency.

In view of the strength of mean shift in filtering noise and reaching median value, we choose mean shift technique to perform color aggregation function. Mean shift is a kind of median filtering method, but different from the traditional median method, mean shift will not obliterate the necessary information. Mean shift is a more useful and rational method. As to a color image, mean shift will play a kind of clustering role. Given a set of multi-dimensional data points, it uses a defined window to scan the image space to find the “cluster block” with the highest data density and merges the block into one category in order to reduce the computation of data processing. In binocular vision, in consideration of the characteristics of the object itself, the objects with the same or similar color on the left and right eye images usually have the same depth and disparity, therefore, we can use mean shift algorithm to preprocess the images and reduce the dynamic color range. The cluster with the same or similar color becomes one category in the color image. The total amount of color of the image drops down so as to facilitate further disparity estimation and matching. Through mean shift color aggregation, large uniform texture regions occur and this greatly reduces the disparity computation complexity. Correspondence searching can be batch processed and the efficiency of the matching algorithm is enhanced. Color aggregation using mean shift is shown in Fig.2.

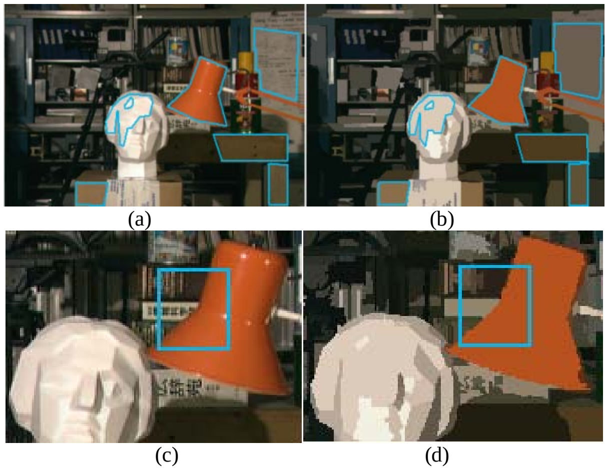


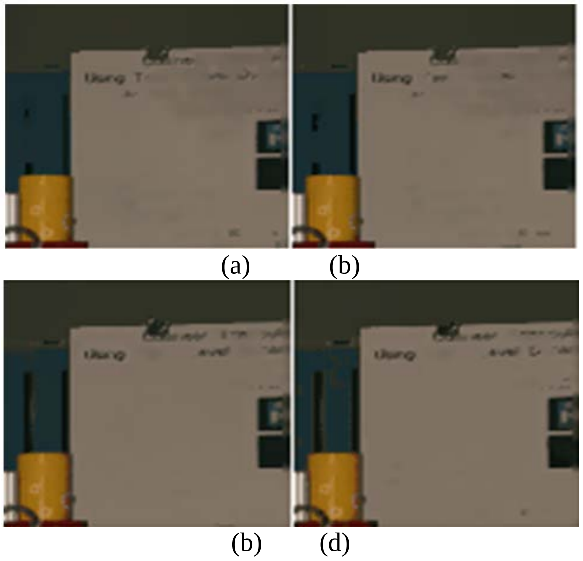
Fig. 2. Color aggregation using mean shift
(a) original image of Tsukuba (b) image processed by mean shift
(c) part of the original image (d) filtered part after color aggregation

In Fig.2 (a) (b), we can see that using mean shift method, the objects in the image such as the lamp, desks, white board and other features separately cluster into one same or approximate same color, which form a large number of non-textured areas. In Fig.2 (c)(d), the effect of aggregation only affects the same or similar color regions. That means, unlike median algorithm, mean shift will not play a role in quite different color areas.

There are two most important parameters in implementing the mean shift algorithm, Spatial Radius (*Rs*) and Color Radius (*Rc*). Now we give detailed discussion about these two parameters.

### 2.1 Spatial Radius

Spatial radius is used to determine the window size which is used to search the color image through its total space. It will directly affect the statistics when scanning the color image, thereby affecting the amount of the shift of the mean. Some experiments have been made to show the significance of Spatial Radius, as is shown in Fig.3. When Color Radius is determined, the increase of Spatial Radius will lead to more areas affected by mean shift.



(a) (b)

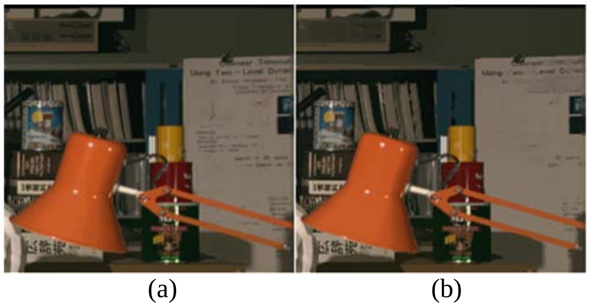
(b) (d)

Fig. 3. The processing result of Tsukuba using mean shift with different *Rs*.
(a) *Rs* = 8 (b) *Rs* = 16 (c) *Rs* = 20 (d) *Rs* = 40

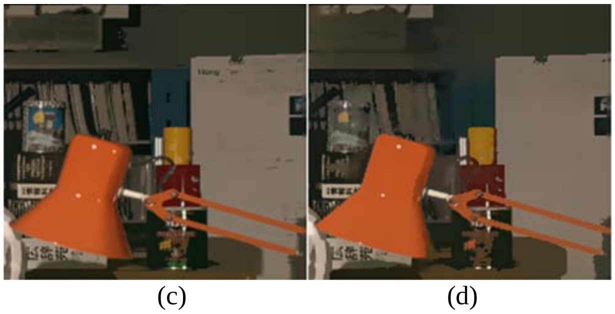
We can see that in Fig.3 from (a) to (d) when Spatial Radius increases, the color aggregation region expands as well. These colors are the same or similar under the threshold of *Rs*. In (a), there are patches of gray fuzzy-areas on the top right of the white board, but we can see almost nothing but single white color with no texture in (d).

### 2.2 Color Radius

Color radius is the color threshold for color aggregation. If one pixel's color is in the color radius neighborhood of the reference pixel's color, it will be filtered to this color. Increasing the Color Radius will aggregate more similar colors together. The influence of the Color Radius is illustrated in Fig. 4.



(a) (b)



(c) (d)

Fig. 4. The processing result of Tsukuba using mean shift with different Rc.
(a) Rc = 16 (b) Rc = 20 (c) Rc = 32 (d) Rc = 40

Observing the surface of the white board in Fig.4, we can recognize some black letters in (a). But along with Color Radius increasing, we see all letters aggregate to the same white color in (d).

Thus, by mean shift algorithm, we can aggregate large areas of similar colors into single color or texture-less region. In most binocular vision situation, the same or similar color areas are probably in the same depth range. Mean shift processing can remove patches of complex texture and form a new pattern of large single color regions without texture. Disparity of the same region can be represented by one pixel's disparity. The correspondence searching times is effectively reduced and efficiency of the algorithm is improved.

## III. ADAPTIVE CORRESPONDENCE SEARCHING

After color aggregation, we make disparity estimation and optimization with window-based adaptive correspondence searching algorithm to recover the depth map from the simplified left image and the right image.

### 3.1 Disparity Estimation

First, we make use of Birchfield image matching algorithm to carry out the disparity estimation in order to narrow the searching range of window correspondence. Dynamic programming is applied to do epipolar line matching to each pixel. The advantage is that it can rapidly obtain relatively reliable points of depth discontinuity and unmatched points while computing the depth map.

Define a global cost function $\gamma(M)$ to represent the credibility of a match sequence *M*:

$$\gamma(M) = N_{occ}k_{occ} - N_m k_r + \sum_{i=1}^{N_m} d(x_i, y_i) \tag{1}$$

Where, $k_{occ}$ represents an occlusion punishment value, $k_r$ is a match bonus, $d(x_i, y_i)$ represents the dissimilarity between point $x_i$ and $y_i$. $N_{occ}$ And $N_m$ respectively mean the number of occlusion region and that of matching region. Using this cost function a depth map can be acquired which is segmented but continuous within each segment.

Disparity estimation based on Birchfield uses the similarity criteria from insensitive sampling pixels. It avoids the problem in image sampling and solves the problem of no-match of large areas without texture. Moreover, non-similar points are cut branches and the computing time of the algorithm is greatly reduced.

We can rapidly reach a depth map using Birchfield in the processing of binocular stereo matching. But as we know, the matching result is not satisfactory. Experiments show that the accuracy of Birchfield is sometime too low to be accepted, especially on the boundary of the image. From a global perspective, Birchfield algorithm can reach a fast speed and high efficiency as a result of computing less information. It only searchs the pixels on the same row as the reference pixel and takes the one with the largest matching probability as the dispatiry source. Therefore wrong matching is inevitable. Only taking single row pixels into consideration to match will lead that each row has nothing to do with other rows. What's more, it will lead to such situations that the disparity of pixels on the neighboring two rows, up and down, may become ragged or with a great gap which is nearly impossible in fact.. Thus, when matching images, rapid adaptive window correspondence only use Birchfield algorithm to estimate disparity, and use rapid adaptive window method to match further in order to reach the depth map of high accuracy.

### 3.2 Window-Based Correspondence

Take the disparity map estimated from the above step as the initial value, now we use window-based correspondence method to optimize the result. When matching by this method, weight and the function $w(p,q)$ should be used to determine the relationship between the two pixels in each image. Weight is calculated as the follow formula:

$$w(p,q) = f\left(\Delta c_{pq}, \Delta g_{pq}\right) \tag{2}$$

Where $\Delta c_{pq}$ and $\Delta g_{pq}$ are color similarity and geometric proximity.

The so-called color similarity degree is the distance between the color values of the two pixels. In RGB color space, if pixel *p* and pixel *q* stand for two different colors, the color similarity degree can be defined as $\Delta C_{pq}$:

$$\Delta C_{pq} = \sum_{c\in\{r,g,b\}} \left| I_c(p) - I_c(q) \right| \tag{3}$$

Geometric proximity is the distance from the position of the reference pixel to the target pixel when they are on the same plane. Generally, we use Euler distance to compute geometric proximity distance.

When matching the pixels, one pixel in reference image may have a good matching with many pixels in target image. But the fact tells us that one pixel can only match one single pixel at any time. So another value, dissimilarity should be used to measure the difference when matching two pixels in order to avoid the situation above. Obviously, the smaller the dissimilarity is, the two pixels match better. The more dissimilarity reflects that those two pixels do not match.

Due to the existence of disparity, the correspondence needs to use windows for weight adjustment. The disparity value is the distance between the reference pixel and the pixel which has the smallest dissimilarity in target image.
The formula to compute dissimilarity between two pixels:

For each pixel $p$ and its neighborhood $N_p$ in reference image, the corresponding pixel $p_d$ and its neighborhood $N_{p_d}$ in target image, define the dissimilarity $E(p, p_d)$ between the two windows:

$$E(p, p_d) = \frac{\sum_{q\in N_p, q_d\in N_{p_d}} w(p,q) w(p_d, q_d) e(q, q_d)}{\sum_{q\in N_p, q_d\in N_{p_d}} w(p,q) w(p_d, q_d)} \tag{4}$$

Where $e(q,q_d)$ is absolute difference and $w(p,q)$ is the adaptive weight:

$$e(q,q_d) = \min\{ \sum_{c\in\{c,g,b\}} | I_c(q) - I_c(q_c) |, T\} \tag{5}$$

### 3.3 Optimized Disparity Computation

In last paragraph we have introduced the disparity calculating method. The disparity value is the distance between the reference pixel and the pixel with the smallest dissimilarity in target image. In prior research, window matching methods such as Yoon's, to each pixel *p* in reference image, it needs to search all the pixels within the pre-defined parameter MaxDisparity to obtain the final result. This sounds reasonable and may result in high accuracy and reliability. But the computation is vast.

Through analysis of this issue we find that the estimated disparity can be used to improve the result. Compared with ground truth, the estimated depth map shows us that there are only ±1~2 pixels with different disparities in single texture area. But there are about ±3~10 pixels different on border areas. That is to say, disparity in single texture area can be used to accelerate our algorithm.

Consequently, after disparity estimation in the second step, we effectively drop down the cost and reduce the searching times from the max disparity to possible smaller range. The final optimized depth our algorithm achieves at point *p* is:

$$Disp_p = \arg\min_{d\in S_d} E(p, p_d) \tag{6}$$

where $S_d$ is the possible range of disparity:

$$S_d = \overline{Disp_p} \pm K_n \tag{7}$$

And $\overline{Disp_p}$ is the estimated disparity at point $p$; $K_n$ is the disparity estimation threshold. By setting suitable parameter $K_n$, we can reduce the dynamic matching range, and use adaptive space instead of the original fixed space. Thus the searching range is shortened and the cost of the algorithm is cut down .

On the other hand, through priori knowledge we know that simple texture region in most cases is in the same depth.

Therefore, using this rule in matching points between the two images will save time when matching this type of areas. We only need to check whether this point is in the simple texture region, instead of matching this point with target image. If yes, it will be assigned with the same depth as the point processed before it.

We can inspect whether $I_{p(x,y)}$ , $I_{p(x,y-1)}$ and $I_{p(x-1,y)}$ are in the same color to determine the reference point $p(x,y)$ in uniform texture region. Only when $I_{p(x,y)}$ , $I_{p(x,y-1)}$ and $I_{p(x-1,y)}$ are in the same color and $Disp_{p(x-1,y)}$ and $Disp_{p(x,y-1)}$ are in the same depth can we determine point $p(x,y)$ is in the uniform texture region. Then we assign $Disp_{p(x,y)} = Disp_{p(x-1,y)}$.

### 3.4 *Pesudo Code*

```
Input : LeftImage, RightImage // left and right eye images
Output :  Dep_img / / Depth Map
Begin
Step 1:
        L_img = Mean Shift (LeftImage, Rs, Rc)
// using mean shift algorithm processing LeftImage to L_img
        R_img = Mean Shift (RightImage, Rs, Rc)
// using mean shift algorithm processing RightImage to R_img
Step 2:
        Dep_img = Birchfield (L_img, R_img, MaxDisp, kocc, kr)
// using BirchField algorithm processing L_img and R_img to obtain Dep_img
Step 3:
      // this loop traverse all points in reference image to calculate disparity
    for i = 1: Dep_img-> height
      for j = 1: Dep_img-> width
          {
              if $p \in N_{simple}$
        // if point p belongs to a single texture region
              {
                $Disp_{p(i,j)} = Disp_{p(i-1,j)}$;
        // directly assigning the depth
              }
              else
               {
                     for k = $\overline{Disp_p} - K_n : \overline{Disp_p} + K_n$
        // possible disparity range
                     {
                         $S_k = E(p, p_k)$
        // Window matching
                     }
                     $Disp_p = \arg\min d(d \in S_k)$
        // assigning the depth after calculating
                 }
           }
End
```

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

We made experiments to verify the proposed algorithm using the standard image data (four images: Cones, Teddy, Tsukuba and Venus) on the website of MiddleBury.

The experimental environment is shown in Table 1.

TBALE 1. EXPERIMENTAL ENVIRONMENT

| Category | Description |
|---|---|
| Host configuration | Lenovo Think Station, Windows Vista Business, Intel Xeon CPU E5540 2.53GHz （2CPU）, Memory 8GB, 64-bit operating system, NVIDIA Quadro FX 1800 |
| Software | VS2008，OpenCV 1.0 |

Now we demonstrate the results of the three steps of the algorithm.

### 4.1 *Mean Shift Processing of Color Image*

The parameters of mean shift aggregation are: Space Radius *Rs* = 40, Color Radius *Rc* = 32. The results of color aggregation are shown in Fig. 5.

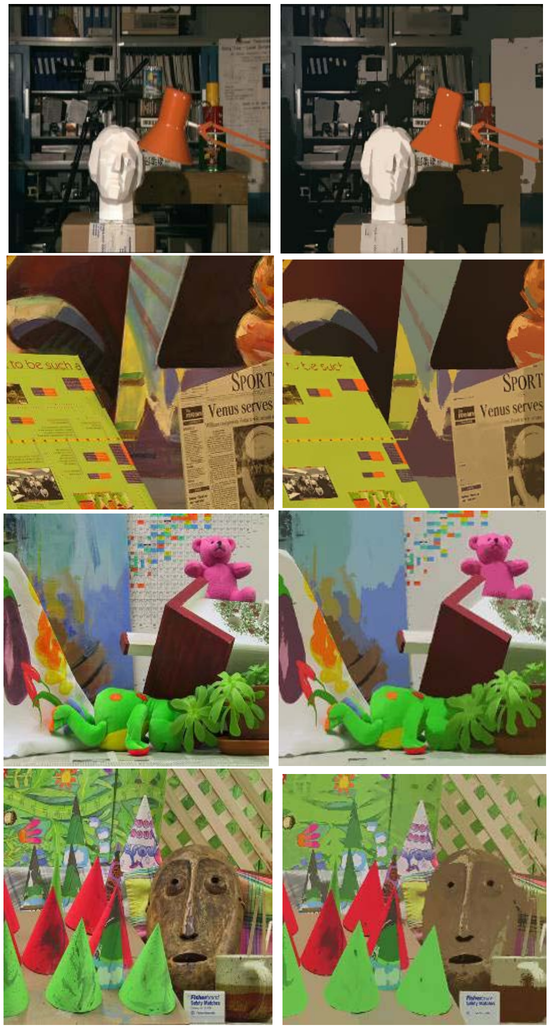


Fig.5. Images with color aggregation.
The images in the left column are the original images and the images in the right are the corresponding images after color aggregation.

From Fig.5 we can see, the complicated similar texture regions of the same depth in each image are reasonably aggregated into large uniform texture areas.

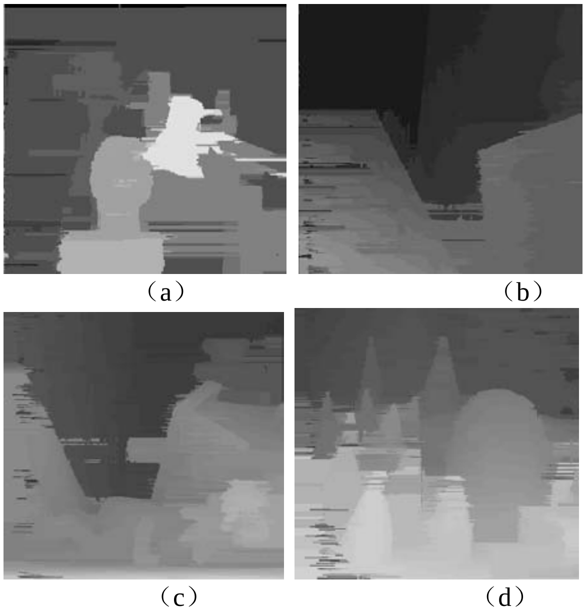


Fig.6. Disparity estimation result of four images (a) Tsukuba (b) Venus (c)Teddy (d)Cones

### *4.2 Disparity Estimation*

In this step, we defined different parameters to these four images. To Tsukuba and Venus, the initial maximum disparity is 20 and to Teddy and Cone, MaxDisp = 60. The results after disparity estimation are shown in Fig.6. It can be seen that the disparity of the uniform texture regions of each image has been estimated properly.

### *4.3 Window-based Correspondence Search*

In window-based matching procedure, the correspondence search range is reduced to the estimated disparity plus or minus 5. The final depth maps we obtained are shown in Fig.7.

From Fig.7, we can see that the result of this algorithm is more accurate than the algorithm of Yoon in the region of single object and uniform texture. Our algorithm is near to the ground truth unless on the boundary area.

We also tested the time complexity of the proposed algorithm and the result is shown in Table 2.

TABLE 2. ALGORITHM RUNNING TIME

| Time (s) | Tsukuba | Venus | Teddy | Cones |
|---|---|---|---|---|
| Yoon’s | 1800 | 2700 | 3600 | 3600 |
| Our algorithm | 496 | 527 | 880 | 670 |

As Table 2 shows, our algorithm is more efficient than

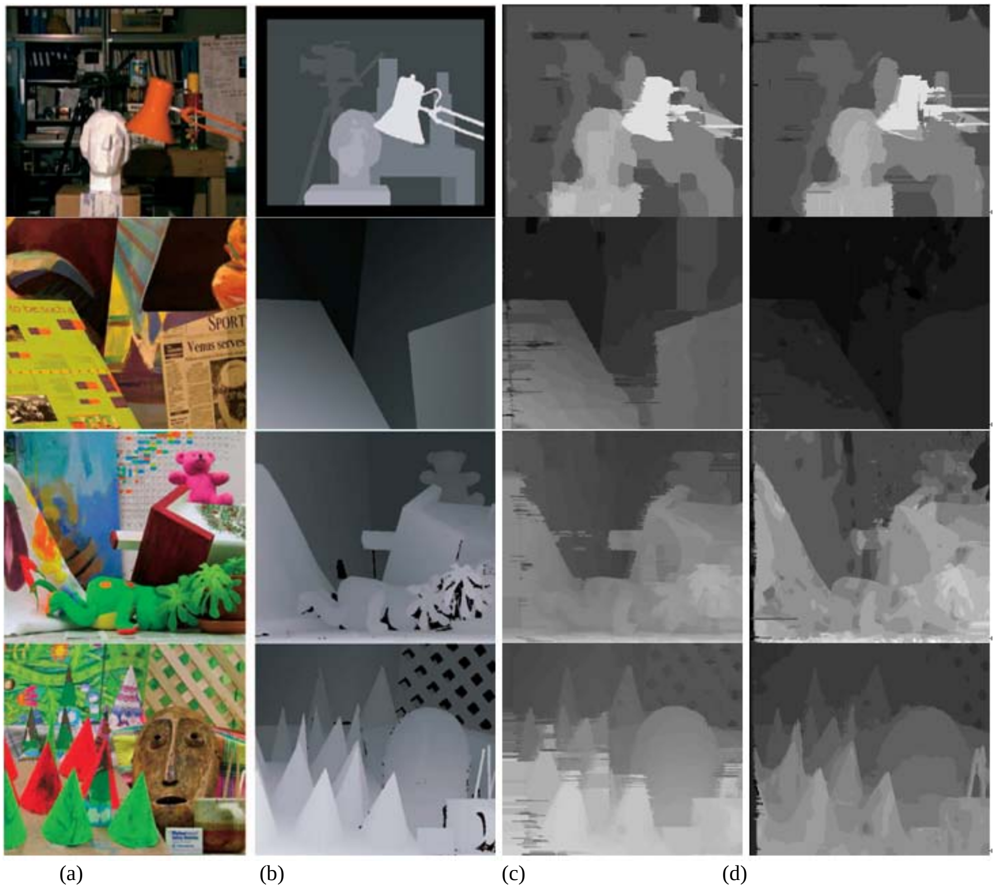


Fig.7. Depth map result
(a) Original left images (b) Ground Truth (c) Our algorithm (d) Yoon's algor

Yoon's. We will further make GPU acceleration in future work to support real-time application.

## V. CONCLUSIONS

This paper puts forward a rapid window-based adaptive correspondence search algorithm using mean shift and disparity estimation. The core idea of this algorithm is to combine color aggregation and local disparity estimation into matching cost aggregation, so that the color dynamic range of the original image can be reduced and pixels of complex texture regions become simple uniform texture areas. Under the combined effect, the time complexity is reduced and the performance of the algorithm is improved.

At present, rapid adaptive windows correspondence is one of the algorithms which improve window-based correspondence, but not the best one. As an advanced algorithm in binocular stereo matching, how to balance the accuracy and time complexity is still potential for study. In this paper, we made efforts in both aspects. The next work will be further acceleration and global optimization of disparity computation and multi-view reconstruction.

## ACKNOWLEDGEMENT

This paper is supported by the National Natural Science Foundation (No. 60903064 and No. 61040047) and National Grand 973 Program of China (No. 2009CB320805).

## REFERENCES

[1] B.J. Super, W.N. Kiariaquist, (1997). Patch-based stereo in general binocular viewing. *Theory and Experiment, PAMI,* 3, 19, (1997), pp. 247-252.

[2] Raymond van Ee, Clifton, M. Schor (2000). Unconstrained stereoscopic matching of lines. *Vision Research, 40,* (2000), pp. 151-162.

[3] A. Kui, jper I.M.J. Florack, (2001). On the creations of critical points in scale space with applications to medical image Analysis. UU-CS, 25, (2001), pp. 316-319.

[4] J. Zhou, Y. Xu, W.R. Yu, (2002). Phase matching with multi-resolution wavelet transform. *In Proc SPIE,* (2002), pp. 82-91.

[5] M.R.M. Jenkin, A.D. Jepson, (1994). Recovering local surface structure through local phase difference methods. *CVGIP: Image Understanding*, 59, (1994), pp. 72-93.

[6] S. Birchfield, C. Tomasi, (2001). Depth discontinuities by pixel-to-pixel stereo. *International Journal of Computer Vision (IJCV)*, 35, 3, (1999), pp. 269-293.

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

[8] K. J. Yoon, I. S. Kweon, (2005). Locally adaptive support-weight approach for visual correspondence search. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2, (2005), pp. 924-931.

[9] S. Birchfield, C. Tomasi, (1998). A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI),* 20, 4, (1998), pp.401-406.

[10] I. Ernst, H. Heiko, (2008). Mutual information based semi-global stereo matching on the GPU. *International Symposium on Visual Computing, ISVC*, (2008), pp.228-239.

[11] M. Bleyer, M. Gelautz, (2007). Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions. *Image Communication*, 22, (2007), pp. 127-143.

[12] L. Hong, G. Chen, (2004). Segment-based stereo matching using graph cuts. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004, Volume 1, pp.74-81.

[13] K. Fukunaga, L. Hostetler (1975), The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 1975, 21(1), pp. 32-40.

[14] D. Comaniciu, P. Meer, (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI),* 2002, 24, 5, pp.603-619.

[15] M.R.M. Jenkin, A.D. Jepson, (1994). Recovering local surface structure through local phase difference methods. *CVGIP: Image Understanding,* 59, (1994), pp. 72-93.

[16] S.J.D. Prince, R.D. Eagle, (2000). Weighted directed energy model for human stereo correspondence. *Vision Research*, 40, (2000), pp. 151-162.

[17] M. EI, Ansari, L. Masmoudi, L. Radouane, (2000). A new region matching method for stereoscopic images. Pattern Recognition Letters, 21, 4, (2000), pp. 283-294.

[18] L. Vinet, A1. Gagalowicz, (1991). Region matching using neighborhood consistency. *In Proceedings of the 8th Congress of Form Recognition and Artificial Intelligence. Lyon2Villeurbanne,* (1991), pp. 821-826.

[19] E. Zagrouba, C. Krey, (1992). Region matching by adjacency propagation in stereo vision. *In Proceedings of the 2nd International Conference on Automation,* Robotics and Computer Vision. Hyatt Regency, Singapore,(1992). pp. 851-855.

**Shujun Zhang** was born on Oct.6th, 1980. She received her Ph.D degree on computer science from Ocean University of China in 2007. She completed her post-doctoral research in State Key Laboratory of Virtual Reality Technology and Systems, Beihang University from 2007 to 2009 and now is a master tutor in Qingdao University of Science & Technology. Her major field of research is virtual reality, including: 1) image-based modeling; 2) computer graphics and computer vision; 3)V-R interaction and immersion.

**Jianbo Zhang** was born on Aug.3rd, 1988. He received his Bachelor Degree of Electronic and Information Engineering from Qingdao University of Science & Technology (QUST) in 2010. Now, he is a postgraduate in QUST, majoring in Computer Software and Theory. His direction of studyincludes information and signal processing, computer vision and virtual reality.

**Yun Liu** is a professor and master tutor in College of Information Science & Technology from Qingdao University of Science and Technology. He received his Ph.D degree in 2001 from China University of Mining and Technology. Now he is the Dean of College of Information Science & Technology of QUST, high-level member of China Computer Federation, high-level member of Chinese Institute of Electronics. His major research direction is image processing and pattern recognition.