Horizontal Adaptive Disparity Estimation Scheme for Stereoscopic Images

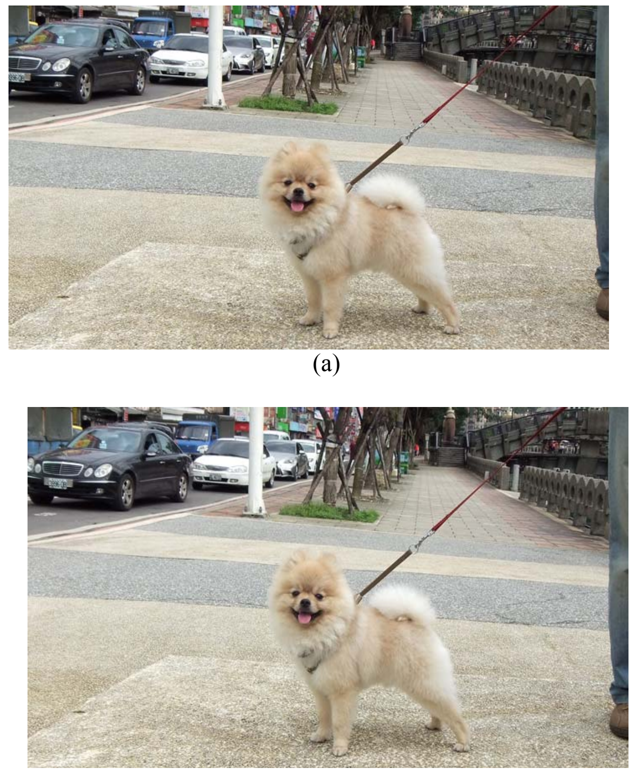
Szu-Wei Fu, Jian-Jiun Ding *, Ching-Wen Hsiao, and Po-Jen Chen
Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan
Email: r019420319@ntu.edu.tw, jjding@ntu.edu.tw, r02942031@ntu.edu.tw, and r03942106@ntu.edu.tw
TEL: +886-2-33669652

***Abstract*— Stereo image compression is more and more important because of new display technologies and the needs of 3D movies. As a video sequence, a pair of stereo images is very similar to each other. Therefore, there are usually a lot of redundancies between them. To improve the compression efficiency, an effective method to estimate the target image from the reference image is needed. In this paper, based on some peculiar properties of stereoscopic images, a new 3-state fast block searching algorithm for disparity estimation is proposed. It applies the horizontal search with a fat rectangular search window and uses the thresholding and prediction scheme. Moreover, the technique of variable block size mode is also adopted. This novel algorithm can achieve a performance close to (even a little better than) that of full search, while only 2.7% computation cost is required. In addition, simulations show that the entropy for coding disparity vectors can also be greatly reduced compared to other state-of-the-art searching methods.**

I. Introduction

Stereoscopic images can be easily got from a system with two cameras separated by an appropriate distance analogous to the left and right eyes of a person. Recently, because of some great 3-D effect movies, people have more interest on this special experience and the stereoscopic image processing techniques become more and more important in industry.

A pair of stereo images is very similar to each other, since they are taken at the same time with only small displacement, as shown in Figure 1. Since the involved scene is almost the same for both images, there should be a lot of redundant information. Therefore, an efficient way to compress them is to encode one image firstly and predict the other one from it. For stereoscopic images, if the left and right images are encoded independently, the file size is twice of that a single JPEG image, which is inefficient. To solve this issue, the encoding block diagram for stereo images can be designed as in Figure 2. This paper mainly focuses on the part of disparity estimation. After getting those vectors, they can be coded by differential coding or other more sophisticated methods [1, 2]. As suggested in some researches [3], a display system can always tolerate a slight amount of asymmetric image quality for stereo viewing. Therefore, if the target image can be estimated pretty well, maybe one does not need to implement residue coding [4-8].



(a) (b)

Fig. 1 A pair of stereo images. (a) Left (b) Right.

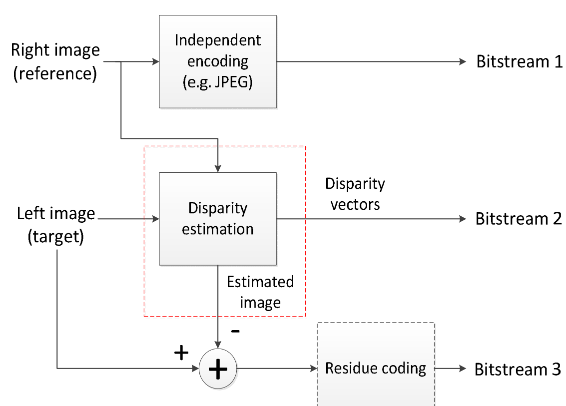


Fig. 2 Coding structures of stereo images.

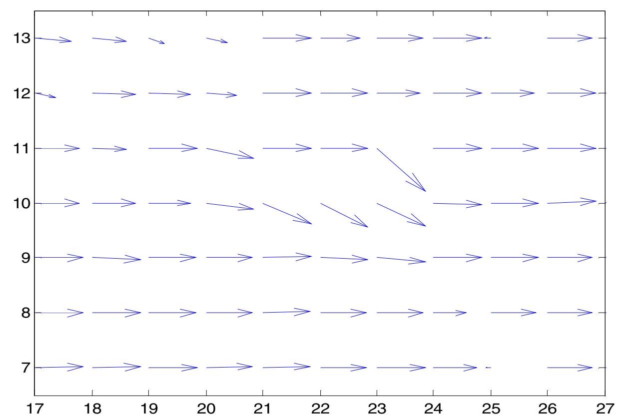


Fig. 3 Part of disparity vector fields in Figure 1.

As a video sequence, an effective and popular method to reduce the redundancy of two images is based on block-matching, which has been widely adopted in various video coding standards. However, the process of searching the best matching block usually consumes most computation cost. Therefore, a fast and accurate search technique is highly desirable to much reduce the processing time while maintaining good estimated image quality. Therefore, many fast block-matching algorithms have been developed for video coding, e.g., hierarchical search [9, 10], Adaptive Rood Pattern Search (ARPS) [11], overlapped block searching [12], and Diamond search (DS) [13], etc. These fast algorithms exploit different search strategies for finding the optimum motion vector with drastically reducing the number of search points. Although, they may be efficient for video sequences, some important geometric characteristics of stereo images are not adopted. In this paper, we proposed an efficient search method for stereo images by observing two important properties related to disparity vectors:

(Property I). In usual, there is a large horizontal shift but only a small vertical shift between left and right images.

(Property II). The shifts of the blocks in same depth of the scene are very close. Therefore, the disparity vectors of adjacent blocks may also be very similar.

The detail of how to apply these properties for disparity estimation will be discussed in the next section.

The rest of this paper is organized as follows. The detail and flow chart of the proposed searching algorithm will be introduced in Section 2. In Section 3, several simulations are given to compare the reconstruction qualities and the computation complexities for several searching methods. Finally, a conclusion is made in Section 4.

II. Proposed Algorithm

A. Fixed Block Size Mode

As mentioned in the previous section, the two peculiar properties of stereo images are the basis of our efficient search method. One can easily verify these two characteristics by observing the disparity vector field shown in Figure 3. Note that the horizontal component of the disparity vector is usually larger than the vertical component. Moreover, the disparity vectors of adjacent blocks are usually very similar.

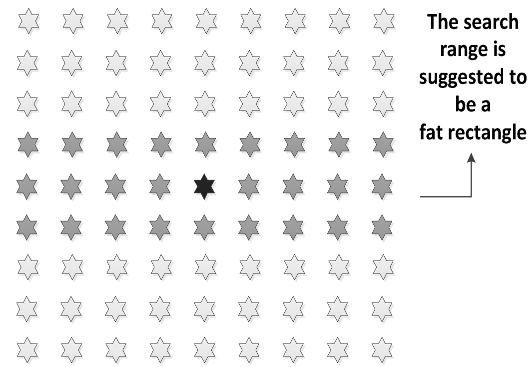


Fig. 4 Search range of stereo images and a video sequence.

Therefore, from first observation, the search range of stereo images should be a “fat rectangle” instead of a square used in video coding, as illustrated in Figure 4.

Since every candidate position in the search window should be compared to find the best matching block, the computation cost of full search is very high. Even if the optimal solution has already been found, one still needs to waste much time to check the rest of candidates. To solve this inefficiency, a simple thresholding method is introduced. For example, if the error between the current block in the target image and the candidate block in the reference image is below a predefined threshold, then it is accepted as a good solution and the search process can be stopped. This greatly reduces the computational time and keeps a good estimated image. Note that this simple method can perform much well if the disparity vectors of adjacent blocks are the same, which is usually the case.

In addition to horizontal search (using a fat rectangle search window), the proposed algorithm also applies thresholding and prediction techniques. Its detail is described as follows.

First, select two thresholds *th1* and *th2* where *th2* >*th1*. We define *err* as the error between current block in the target image and the candidate block in the reference image when the disparity vector of the previous block is treated as the predicted disparity vector of the current block. Then, according to the relation between *err* and *th1* or *th2*, the search process can be divided into three states:

(i) State 1 (previous vector): *err* < *th1*

If *err* is smaller than *th1*, then the disparity vector of the previous block can be seen as a good prediction of the current disparity vector. Then the search process can be stopped.

(ii) State 2 (modified previous vector): *th1* < *err* < *th2*

If *err* is between *th1* and *th2*, then the disparity vectors of the previous and the current blocks may have a little difference and a better matching block may be close to the predicted matching block. Hence, a small search window (e.g., a window with size 5x5) centered at the predicted matching block is applied to find the best matching block.

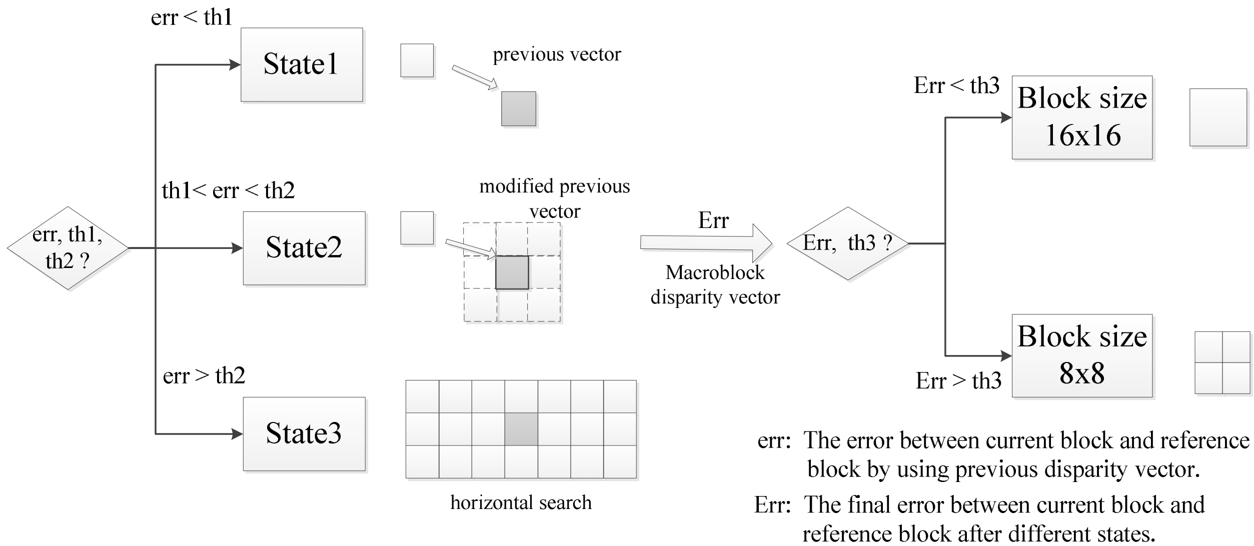


Fig. 5 Flow chart of the proposed search algorithm.

(iii) State 3 (horizontal search): *err > th2*
If *err* is larger than *th2*, then the disparity vector of the previous block is not a reliable prediction. In this case, one should use the horizontal search with a flat rectangular search window as in Fig. 3 to find the best matching block.

Note that, unlike most of the other search methods mentioned before, the search range of this algorithm is not a subset of that in full search. For example, if the search range used in full search is 64x64, then the absolute value of the disparity vector will never exceed 32. However, in the proposed search method, if the disparity vector of previous block is 31 and the current state is 2, then there may be a small positive refinement (e.g. +4) which makes the current vector larger than 32. This is a very good property, since the search range will no longer be restricted by a predefined value. In other words, it provides some flexibility to adaptively modify the search range. As will be shown in the next section, because of this property, the quality of the estimated image by the proposed algorithm is better than that by other methods. In summary, state 1 can dramatically reduce the computation while states 2 and 3 can both reduce the computation and improve the quality at the same time.

In addition, there is another advantage by adopting the proposed algorithm. Since in states 1 and 2, the disparity vector is predicted by the previous block, the correlation between adjacent disparity vectors is usually higher than other search methods**.** This will greatly reduce the entropy when encoding disparity vectors by differential coding. On the other hand, due to the high correlation, the blocking effect, which is an intrinsic problem in block-based estimation, can also be solved.

B. Variable Block Size Mode

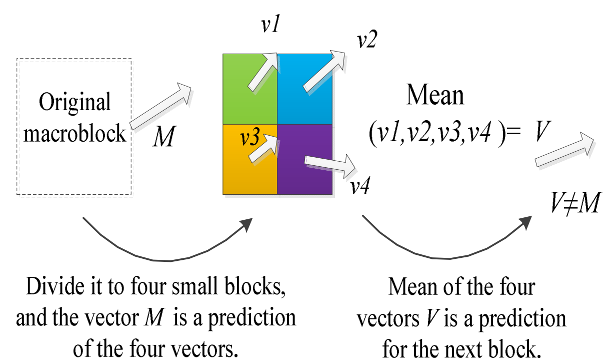
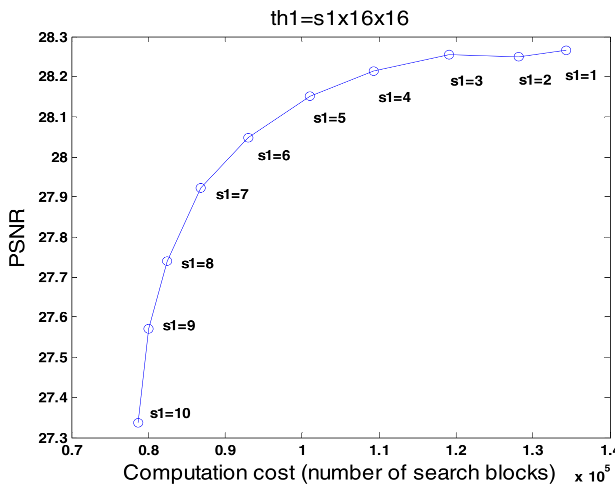


Fig. 6 Detail of the variable block size mode.

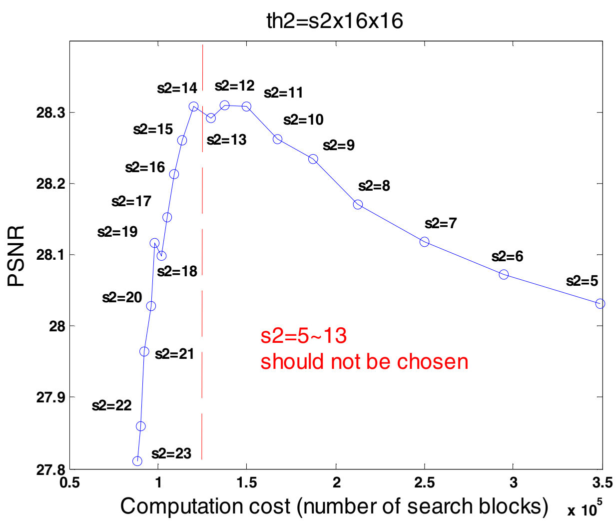
As in many new video coding standards, in order to gain higher compression efficiency, a variable block size mode is also proposed. If the block size is smaller, a better estimated image will be got but the computational time greatly increases. Therefore, it is more efficient to adaptively modify the block size. When the macroblock has simple texture or belongs to the same object, a larger block size is more suitable for disparity estimation. On the other hand, when more than one object is in the macroblock, a smaller block size is more suitable.

Our method to decide the size of block is quite easy to implement. If the error of the current block *Err* exceeds a threshold *th3*, then the block may contain more than one object and it will be divided to 4 smaller blocks.

The overall flow chart of the proposed method is presented in Figure 5. To save computation, the disparity vector *M* of the original macroblock will be used as an initial guess of those 4 blocks and small refinement will be made afterward. Then, the mean of the four vectors *V* is used as a prediction for the next block, as depicted in Figure 6. In the next section, simulations will show that this new vector *V* is a better prediction than the old vector *M*.



(a)



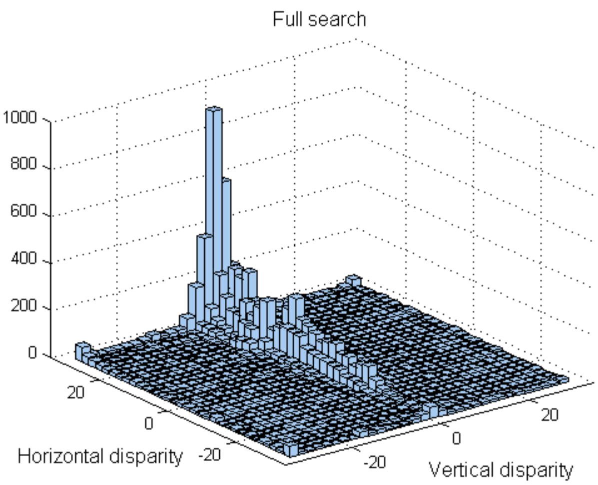
(b)

Fig. 7 The relation between different thresholds and search efficiency, (a) fixing th2 and adjusting th1 (b) fixing th1 and adjusting th2.

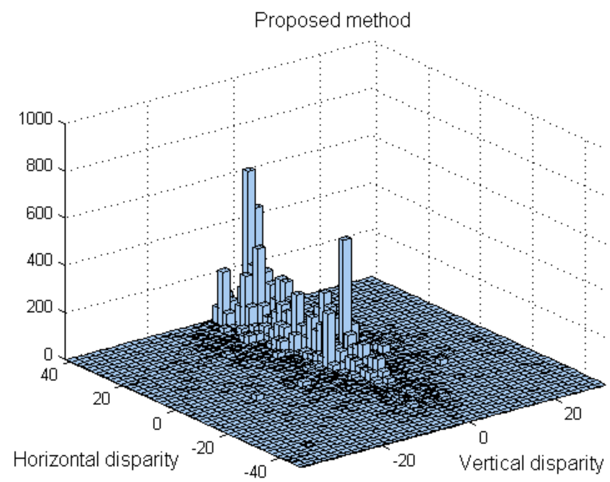
III. Simulations

Twenty image pairs with different types of scenes were collected for evaluating the performance of the proposed algorithm. In addition to Fig. 7, the parameters *th1* and *th2* are set to be 1024 and 4096, respectively, in our simulations.

Under this setting, the ratio of average numbers of state 1: state 2: state 3 is roughly to be 0.3472: 0.4942: 0.1586. The relation between different thresholds and searching efficiency is shown in Figure 7. As one may expect, the case for *th1* is relative simple. A larger threshold will save computation but with worse estimated quality. However, this is not the case for *th2*, as illustrated in Figure 7(b). Since the relation is like a concave function, all the points in the right of the red dash line can find another left-up points, which is more efficient (because of less computation and higher PSNR). Therefore, the red line acts like a road sign.



(a) Full search



(b) Proposed method

Fig. 8 The distribution of disparity vectors.

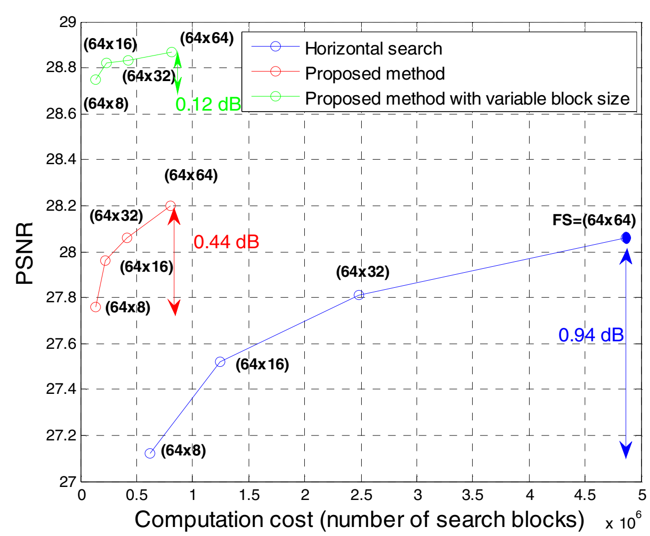


Fig. 9 Comparison of the horizontal searching method and the proposed method under different search range.

The thresholds in the right side of it should not be chosen and those in the left side provide a common trade-off between quality and the computation cost, as *th1*. The reason why we can improve the quality and reduce the computation at the same time in the right side part is mainly because some better matching blocks are located outside the predefined search range. Hence, as mentioned before, state 2 can make the search range unrestricted and increase the PSNR.

In our simulations, the block size is set to be 16x16 and *err* is defined as the sum of the absolute difference (SAD) between the pixels in the reference block and the current block. Therefore, if the average error is smaller than 4 (1024/16/16=4) per pixel, then the prediction is considered to be very reliable and the scheme goes to the first state. The other twelve image pairs are evaluated under those parameters and compared to other search methods as shown in table 1. In this simulation, the search range of full search is 64x64 and that of horizontal search and state 3 in the proposed method is 64x8. The computation cost here is based on how many "equivalent" search blocks are needed to complete the search process and compare the relative ratio with respect to full search. The term equivalent means if the block size is 8x8 or 4x4 (in variable block size mode or hierarchical-based disparity estimation) the computation cost per search is only 1/4 and 1/16 of that by a 16x16 block. The entropy of disparity vectors coded by differential coding are also presented.

It is easy to see that horizontal search can save large computation while only sacrifice a moderate degree of image quality. Since the search range of horizontal search is a subset of that in full search, its PSNR will never exceed it. Note that compared to horizontal search, the proposed method can reduce the implement time and improve estimated image quality at the same time. Moreover, since highly correlated, the entropy of differential coded disparity vectors can be dramatically reduced by our method. If we observe the distribution of disparity vectors as shown in Figure 8, we can see that most of vertical components are concentrated near zero, and hence lead to less entropy compared to horizontal component. This also verifies the first property of stereo images. Note that, due to the second state, the search range of the proposed method does not be confined in ±32.

Although diamond search and ARPS are very efficient for video coding, due to their diamond pattern, they may find the local minimum which is away from horizontal direction. Hence, their estimated image quality is unacceptable even if the search range is set to be unrestricted. On the other hand, hierarchical-based disparity estimation can perform quite well, but there are some drawbacks that cannot be avoided. First, during the encoding process, it needs extra storage and preprocessing step (not included in the computation cost here, e.g., low-pass filtering, down sampling, etc.) to keep and get pictures at different resolutions. Second, since each block is independently searched in a coarse-to-fine method, there is only little relation between adjacent estimated vectors and cause very large entropy and severe blocking effect. Some estimated results by different search methods are shown in Figs. 10 and 11.

In the proposed method with variable block size, the estimated image quality can be greatly improved, since many complex regions are represented with smaller blocks. To save computation, the information of disparity from the original macroblock is kept, and serves as a predictor for its sub-blocks. Thus, only 0.14% extra computation is needed compared to the fixed block size mode. In addition, prediction by the mean of the four sub-vectors is more accurate than that by the original macroblock. If we just adopt this better prediction and leave the size of macroblock unchanged, then the PSNR can increase from 27.76 to 27.89 dB. In Fig. 10(h), those blocks that need to be divided to four smaller sub-blocks are indicated by the four red squares. However, when compared to Fig. 10(g), one can easily observe that other regions may also be improved. For example, although the object inside the black circle does not be represented by smaller blocks, it is obviously modified due to the better prediction. In summary, this variable block size mode can enhance the estimated image quality by both smaller block size and better prediction.

The situation for different search ranges is also compared in Figure 9. Note that the proposed method can have better image quality and less computation at same time when compared to full search and horizontal search. Another good property of our algorithm is that it is not so sensitive to the choice of search range. Since the second state can adaptively modify the range, there is no severe degradation when the setting is not enough. This can be easily observed by their quality difference under different search range.

IV. Conclusions and Future Work

In this paper, we proposed a fast algorithm to search the disparity vectors in a stereo image pairs. The efficiency of this method is mainly based on two important observations. First, the horizontal shift is usually much larger than vertical shift and this suggests that the search range should be a fat rectangle instead of a square. Second, the objects in the same depth should have similar disparity, and hence disparity vectors between adjacent blocks may also be very similar. This is the basis of the prediction from previous block. Combing these two properties with thresholding, three different states can be defined. The second state plays an important role that the search range will no longer be restricted by a predefined value. In other words, the search range of this algorithm is not a subset of that in full search, so this provides the chance to get better quality. Simulation results show that our method can achieve close performance as full search, while only 2.7% computation cost on average is required. In order to gain higher compression efficiency, a variable block size mode is also proposed. To save computation, the information of disparity from the original macroblock is used as a predictor for its sub-blocks.

TABLE I. *Comparison of different searching methods*

	Average PSNR (dB)	Entropy of differential coded Vertical and Horizontal disparity vector	Number of searched blocks	Relative computation cost (FS=100%)
Full search (64x64)	28.06	V=3.9757 H=4.8717	4866052	100%
Hierarchical search (3 layers)	27.49	V=4.8018 H=5.7578	176405	3.63%
Diamond search (unrestricted search range)	22.93	V= 4.3931 H= 5.3646	51412	1.06%
ARPS (unrestricted search range)	25.32	V= 3.6330 H= 4.5004	25651	0.53%
Horizontal search (64x8)	27.12	V=2.4852 H=4.4945	622402	12.8%
Proposed method (64x8)	**27.76**	**V=1.9852 H=2.7854**	**130696**	**2.69%**
Proposed method with variable block size (64x8)	**28.75**	**V=2.1502 H=2.8949**	**137844**	**2.83%**

Simulation results show that this mode can enhance the estimated image quality by using smaller block size and better prediction.

Although this paper is mainly focused on still stereo image pairs, the proposed method can also be applied on stereo videos. In the case of videos, the search efficiency is a very important issue. How to unify our method with temporal redundancy to efficiently get motion and disparity vector is our future topic.

REFERENCES

[1] M. Cagnazzo, V. Valentin, M. Antonini, and M. Barlaud, “Motion vector estimation and encoding for motion compensated DWT,” *Visual Content Processing and Representation*, pp. 233-242, 2003.

[2] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the scalable video coding extension of the H. 264/AVC Standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103-1120, 2007.

[3] P. Seuntiens, L. Meesters, and W. Ijsselsteijn, “Perceived quality of compressed stereoscopic images: effects of symmetric and asymmetric JPEG coding and camera separation,” *ACM Transactions on Applied Perception (TAP)*, vol. 3, no. 2, pp. 95-109, 2006.

[4] G. J. Sullivan, J. Ohm, W. J. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649-1668, 2012.

[5] M. S. Moellenhoff, and M. W. Maier, “Transform coding of stereo image residuals,” *IEEE Trans. Image Processing*, vol. 7, no. 6, pp. 804-812, June 1998.

[6] D. Tzovaras, N. Grammalidis, and M. G. Strintzis, “Object-based coding of stereo image sequences using joint 3-D motion/disparity compensation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 312-327, 1997.

[7] A. Ortis, F. Rundo, G. Di Giore, and S. Battiato, “Adaptive compression of stereoscopic images,” *Image Analysis and Processing (ICIAP)*, pp. 391-399, 2013.

[8] B. Schenkel, C. Luo, P. Frossard, and F. Wu, “Joint decoding of stereo JPEG image pairs,” *IEEE International Conference on Image Processing*, pp. 2633-2636, 2010.

[9] K. M. Nam, J. S. Kim, R. H. Park, and Y. S. Shim, “A fast hierarchical motion vector estimation algorithm using mean pyramid,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 4, pp. 344-351, August 1995.

[10] S. Sethuraman, A. G. Jordan, and M. W. Siegel, “Multiresolution based hierarchical disparity estimation for stereo image pair compression,” *Proc. of the Symposium on Application of Subbands and Wavelets*, March 1994.

[11] Y. Nie, and K. K. Ma, “Adaptive rood pattern search for fast block-matching motion estimation,” *IEEE Trans. on Image Processing*, vol. 11, no. 12, pp. 1442-1449, December 2002.

[12] W. Woo, and A. Ortega, “Overlapped block disparity compensation with adaptive windows for stereo image coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 2, pp. 194-200, March 2000.

[13] S. Zhu, and K. K. Ma, “A new diamond search algorithm for fast block-matching motion estimation,” *IEEE Trans. Image Processing*, vol. 9, no. 2, pp. 287-290,2000.

(a) Left (target)

(b) Right (reference)

(c) Full search

(d) Hierarchical search

(e) Diamond search

(f) ARPS

(g) Proposed method

(h) Proposed method with variable block size
(red squares mean the sub-blocks using the method in Fig. 6)

Fig. 10 Estimated images by different searching methods.

(a) left (target)

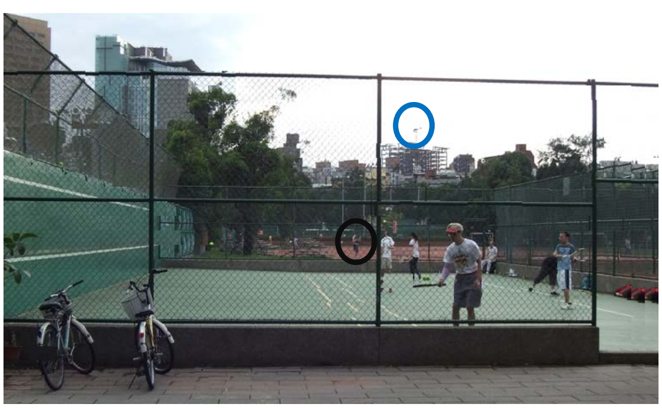
(b) right (reference)

(c) full search

(d) Hierarchical

(e) Diamond search

(f) ARPS



(g) proposed method

(h) proposed method with variable block size

Fig. 11 Estimated images by different searching methods.