

Adaptive Selection of Motion Estimation Block Size for Rate-Distortion Optimization

Dr.V.V.Gohokar

Professor

Electronics Dept. S.S.G.M.C.E.

Shegaon

Dr.V.N.Gohokar

Professor

Electrical Engineering Dept.

S.S.G.M.C.E.Shegaon

ABSTRACT

The employment of the Variable Block size motion estimation technique introduces a new optimization issue for the motion compensated transform coding. An increase in bit rate allocation is necessary to accommodate VBS motion vectors. An algorithm for adaptive selection of motion estimation block size is proposed for R-D optimal motion estimation. It avoids reduction in the block size in visually irrelevant areas. Improvement in R-D performance is obtained at reduced complexity.

General Terms

Video Compression

Keywords

Rate, distortion, video, compression, motion estimation, block

1. INTRODUCTION

Motion Estimation and compensation is the heart of video coding process and always requires great attention for attaining low bit rates with low computational load. Motion estimation is the operation that estimates the motion parameters of moving objects in the image sequence. Motion parameters are estimated based on two or three nearby frames. In this method images are partitioned into regular, nonoverlapped blocks typically 16x16 pixels. This choice of size is a good tradeoff between accuracy and computational cost. However motion estimation techniques may choose different block sizes and may vary the size of the blocks within a given time. Each block is viewed as an independent object. A single displacement vector is estimated for the entire image block under the assumption that all pels in the block share the same displacement vector. The basic operation of block matching is picking up a candidate block and calculating the matching function between the candidate and the current block. This operation is repeated until all the candidate blocks have gone through and then the best-matched candidate is identified. The location of the best-matched candidate becomes the estimated displacement vector. The search is conducted over a predetermined search area. A vector denoting the displacement of the macroblock in the reference frame with respect to macroblock in the current frame is determined. This vector is known as motion vector (MV). When a previous frame is used as reference the prediction is referred to as forward prediction. If the reference frame is a future frame then the prediction is referred to as backward prediction. Backward prediction is typically used with forward prediction and this is referred to as bi-directional prediction. The most frequently used criteria to measure match is the sum of absolute differences (SAD) defined as

$$SAD(P, v) = \sum_{i=1}^N \sum_{j=1}^N |f_t^P(i, j) - f_{t_1}^{P+v}(i, j)| \quad (1)$$

where p - position of the macro block being coded, v - candidate motion vector, N -block width/height. $f_t^P(i, j)$ is the luminance intensity of the pixel (i, j) in the block with position P in the frame at time t . $f_{t_1}^{P+v}(i, j)$ is the luminance intensity of the pixel (i, j) in the block with position P in the frame at time t_1

Block matching method search for the smallest SAD (P, v) by varying v within search window. The full search method computes SAD for every candidate vector in the search window. During reconstruction reference frame is used to predict the current frame using motion vectors. The technique is known as motion compensation. During motion compensation the macroblock in the reference frame that is referenced by the motion vector is copied into the reconstructed frame. In motion estimation the search process can be modified to suit the needs of a particular algorithm. The search area is restricted to lower the computational cost associated with block matching. In many cases the objects in the scene do not have large translational movements between a frame and the next.

Variation in block size has significant effect on coding performance. Small block size gives ambiguity noise problem while large block size gives rise to inaccuracy. Variable block size motion estimation enhances the performance of the motion compensated transform coding system. This technique is observed to be very effective in improving the motion vector precision for areas containing complex motions. It allows for larger blocks to be used when smaller blocks yield little gain, saving bit rates especially for areas containing more complex motion.

However, the employment of the VBS motion estimation technique introduces a new optimization issue for the motion compensated transform coding, because an increase in bit rate allocation is necessary to accommodate VBS motion vectors. An algorithm for adaptive selection of motion estimation block size is developed for R-D optimal motion estimation.

The paper is organized as follows. Section 2 presents an overview block matching method for motion estimation. Section 3 presents effect of variation in block size. Variable block size motion estimation is presented in 4. Section 5 covers the proposed algorithm. Results and conclusion is presented in last section.

2. BLOCK-MATCHING ALGORITHMS

The most accurate block matching method is the full search also known as exhaustive search that compares every possible candidate block in the search window with the best block. This algorithm calculates the cost function at each possible location in the search window. As a result of which it finds

the best possible match and gives the highest PSNR amongst any block-matching algorithm. Equation 1 is evaluated for all possible pixel locations. This is the optimal solution for a given block size and search window. However, the computational complexity of exhaustive block matching algorithm is very high. It spends about 70% of the total encoding time in a video encoder. This heavy computational load limits the performance of the encoder in terms of encoding speed and power consumption. Since the conventional full search algorithm has extremely high computational complexity for real time systems, several faster approaches are suggested.

There are three main classes of Block matching algorithms

1. Methods based on a set of fixed search patterns
2. Methods exploiting interblock correlation
3. Methods applying multiresolution framework

Among these three classes, methods using a set of fixed search patterns are algorithmically the simplest. Three step search(TSS), new three step search, four step search(4SS), Adaptive rood pattern search (ARPS) and diamond search(DS)are few examples. Fast block matching algorithms try to achieve the same PSNR doing as little computation as possible. Three Step Search is one of the earliest attempts at fast block matching algorithms and dates back to mid 1980s. 4SS employs center biased searching and has a halfway stop provision. It sets a fixed pattern size of 2 for the first step, no matter what the search parameter p value is. Search pattern is neither too small nor too big and the fact that there is no limit to the number of steps, this algorithm can find global minimum very accurately. DS algorithm is exactly the same as 4SS, but the search point pattern is changed from a square to a diamond, and there is no limit on the number of steps that the algorithm can take. The end result should see a PSNR close to that of ES while computational expense should be significantly less. In adaptive rood pattern search spatial interblock correlation is utilized to adjust the size of rood shaped search pattern for matching different motion magnitudes. ARPS algorithm makes use of the fact that the general motion in a frame is usually coherent, i.e. if the macroblocks around the current macroblock moved in a particular direction then there is a high probability that the current macro block will also have a similar motion vector. This algorithm uses the motion vector of the macroblock to its immediate left to predict its own motion vector. This improves search speed and search accuracy compared to diamond search. For large and complex motion ARPS yields noticeable degradation. The fast block matching algorithms exploit different search patterns and search strategies for finding the optimum motion vector with drastically reduced number of search points as compared to the FS algorithm [1-4].

3. EFFECT OF VARIATION IN BLOCK SIZE

Different block sizes are used for block matching motion estimation in different standards. The size of the block affects the performance of motion estimation. Small block sizes

afford good approximation to the natural object boundaries, they also provide good approximations to real motion, which is now approximated by piecewise translational movement. However small block sizes produce a large amount of raw motion information. This increases the number of transmission bits or the required data compression complexity to condense this motion information. From performance point of view, small blocks also suffer from object ambiguity problems, which means similar objects may appear at multiple locations inside a picture and may lead to incorrect displacement vectors and random noise problem. Large block size may produce less accurate motion vectors since a large block may likely contain pel moving at different speeds and directions. Fig 1 and 2 shows effect of varying block size on motion compensated frame difference.

Effect of variation in block size is analysed for various sequences. Figure 3 shows the effect of variation in block size on motion compensated frame difference. For the larger block size motion compensated frame difference (mcfd) is more while it reduces for the reduction in block size. Variation in mcfd with variation in block size is considerable for sequences like BBC, which are the sequences with high frame difference between consecutive frames. While for the sequences with little motion like Susie and flower garden the reduction in block size do not have significant effect on mcfd.

4. VARIABLE BLOCK SIZE MOTION ESTIMATION

In block matching motion estimation small block size gives ambiguity noise problem while large block size gives rise to inaccuracy. In addition the boundaries of moving objects do not normally coincide with the boundaries of the blocks used for motion estimation. The objects undergoing different types of motion may exist within the same block. In this case, the motion compensation cannot adequately perform its task by employing a single motion vector. Hence, variable block size motion estimation technique is used to enhance the performance of the motion compensated transform coding system. In the VBS motion estimation, the block size for estimating the motion is varied according to the types of motion in the block. This technique is known to be very effective in improving the motion vector precision for areas containing complex motions. The variable block size (VBS) motion estimation technique allows for larger blocks to be used when smaller blocks yield little gain, saving bit rates especially for areas containing more complex motion. However, the employment of the VBS motion estimation technique introduces a new optimization issue for the motion compensated transform coding, because an increase in bit rate allocation is necessary to accommodate VBS motion vectors. The process of variable block size motion estimation involves partitioning of Image frames into non-overlapped large image blocks. A large size is used at the beginning to obtain a rough estimate of the motion vectors. If a large size is used ambiguity problem can be eliminated. However motion vectors estimated from large blocks are not accurate. If motion compensated estimation error is higher than a threshold, this block is not well compensated, so it is further partitioned into four smaller blocks to improve accuracy [5-8]

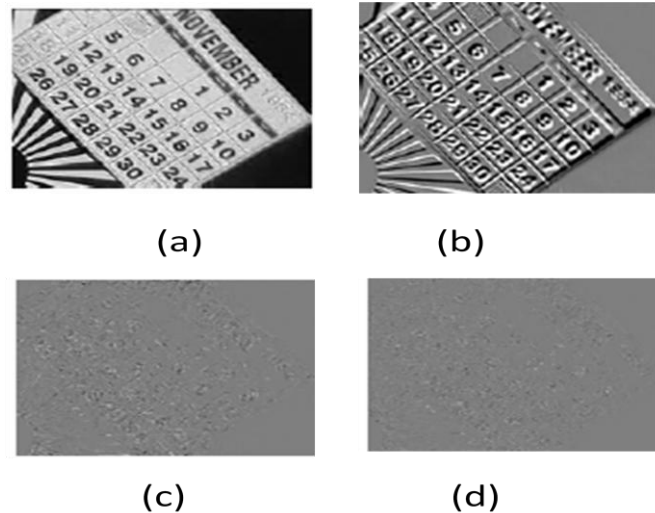
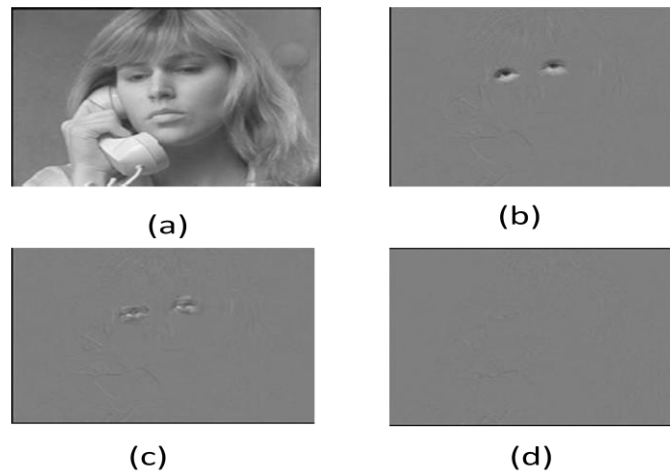


Fig..(1) Effect of variation in block size(BBC)
a. Original frame c. mcfD (16x16)
b. frame difference d. mcfD (4x4)



Fig(2) Effect of variation in block size(Susie)
a. Original frame c. mcfD (16x16)
b. frame difference d. mcfD (4x4)

5. ADAPTIVE SELECTION OF MOTION ESTIMATION BLOCK SIZE

Following observations on human visual perception are used for the algorithm.

1. Visual perception point of view, a hard to predict area does not necessarily catch as much attention as an easily predictable
2. Human visual system is more sensitive to distortion in smooth areas and can tolerate more distortion in highly textured areas
3. Distortion is more visible in brighter area as compared to dark areas
4. Textures featuring a high amount of visible details require high bit rates when coding them using MSE as distortion criteria. Typical representatives of this texture class, called detail-irrelevant may be grass, trees, flowers, cornfield and

water. Exact reconstruction of such texture is not necessary and it can be shown with restricted special accuracy.

5. The contrast sensitivity function of a human visual sensitivity is a band pass filter. When the frequency of input video is too high, the HVS will reach a saturation point that further increase in frequency will no longer increase the visual sensitivity. MSE based encoder may allocate more bits to these visually saturated areas since there are still computable PSNR gains.[9-12]

Analysis of various sequences for variation in block size is done using full search method. Table 1 compares the value of PSNR and the time required to complete the motion estimation. From the table it is clear that the increase in the computational complexity is more as compared to the gain in PSNR for sequences with complex motion and more spatial details. In such situations it is not advantages to reduce the block size.

Table 1

Comparison of PSNR and time required for motion estimation

| Sequence | Blocksize | PSNR | Time |
|---------------|-----------|---------|-----------|
| Susie | 4x4 | 36.7361 | 19.563000 |
| | 8x8 | 35.3327 | 8.359000 |
| | 16x16 | 33.9809 | 5.078000 |
| Flower Garden | 4x4 | 31.2828 | 18.984000 |
| | 8x8 | 30.9195 | 6.375000 |
| | 16x16 | 30.4260 | 3.235000 |
| Tens | 4x4 | 31.2855 | 17.921000 |
| | 8x8 | 30.0015 | 6.344000 |
| | 16x16 | 29.0349 | 3.297000 |
| BBC | 4x4 | 22.8936 | 17.765000 |
| | 8x8 | 20.5568 | 6.375000 |
| | 16x16 | 18.9807 | 3.296000 |

Use of larger blocks when smaller blocks provide little gain can achieve saving the bit rates, especially for areas where human visual system is less sensitive.

The algorithm for adaptive selection of motion estimation block size stops reduction in block size for visually irrelevant areas. In highly textured regions or along edges the energy of the signal is concentrated in the high frequency components while in areas where the image is uniform the energy of the signal is concentrated in the low frequency components. To determine a measure for the texture information within each block based on the energy in the AC coefficients following formula is used.

$$P_T = \log\left(\sum_{i=1}^{63} v_i^2 - v_0^2\right) \quad (2)$$

Where v_i $i=0, 1 \dots 63$ are the 64 DCT coefficients of the 8x8 block that is being considered and v_0 is the value of the DC component of the DCT coefficients. It is not considered when calculating the texture value. M_T is the normalized texture information given by eq.3

$$M_T = \frac{64P_T}{\max(P_T)} \quad (3)$$

STEP 1:

For the first P frame, full search method is used for motion estimation. The value of normalized texture information (M_T) is stored for all blocks.

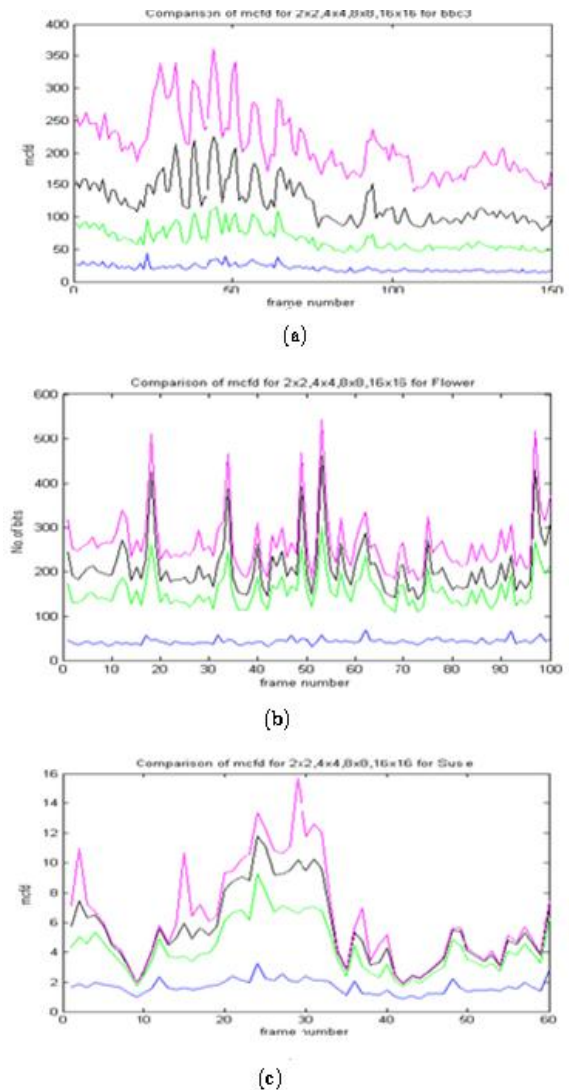
STEP 2:

For all the other frames, initially block size is chosen as 8x8. The search region is categorized using normalized texture information as visually relevant or irrelevant. if the value of M_T is between 0-0.3 it is visually irrelevant for 0.3 to 0.7 it is visually relevant and if it is greater than 0.7 ,the region is smooth, where effect of coding error is more pronounced.

STEP 3:

As Human visual system is more sensitive to distortion in smooth areas and can tolerate more distortion in highly textured areas. The value of block size is modified accordance with the relevance. For highly textured regions higher block size is used as low block size is not going to give any gain in visual relevance point of view, while block size is reduce in smooth areas.

Fig 3 Comparison of mcf for variation in block size for a – BBC , b– Flower garden, c – Susie,



6. TEST RESULTS AND CONCLUSION

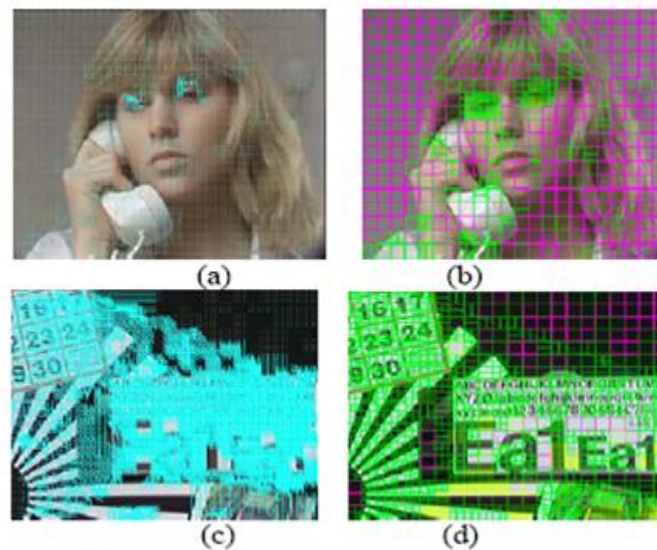
To check the effectiveness of the proposed algorithm .It is tested on various video sequences with variation in visual contents. Figure 4 compares the variation in block size for two video sequences.Sequence1 is Susie which has very less motion and sequence 2 is BBC which has complex rotational motion. For Susie sequence the difference between two

frames is very small or the motion is negligible. For this sequence only for the area where motion occurs block size is reduced, for the remaining region a large block size is sufficient to get the required accuracy. While for the BBC sequence where there is large frame difference between frames, more numbers of blocks are partitioned into small blocks. Table 2 compares performance of the algorithm with respect to standard variable block size algorithm. There is reduction in the time required for motion estimation as well as bit rate at the cost of marginal reduction in PSNR.

Table 2

Comparison of PSNR and time required for motion estimation
 For proposed algorithm

| Video | Time saving % | Bit saving% | Difference PSNR |
|---------------|---------------|-------------|-----------------|
| susie | 20.32 | 5.34 | 0.29 |
| bbc | 15.28 | 2.34 | 0.65 |
| Flower garden | 20.02 | 4.89 | 0.38 |



Figure(4) Motion vectors and block division

7. REFERENCES

- [1] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation" IEEE Trans. Circuits Syst. Video Technol., vol. 4, Aug. 1994, pp. 438–442)
- [2] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation" IEEE Trans. Circuits Syst. Video Technol., vol. 6, June 1996, pp. 313–317
- [3] Yuting Jia, Weisi Lin, and Ashraf A. Kassim, "Estimating Just- Noticeable Distortion for Video" IEEE Transactions On Circuits And Systems For Video Technology, vol. 16, No. 7, July 2006, pp 820-826
- [4] Zhou Wang and Alan C. Bovik, "Mean Squared Error: Love It or Leave It? A New Look at Signal Fidelity Measures", IEEE Signal Processing Magazine, vol. 26, no. 1, pp. 98-117, Jan. 2009
- [5] K. C. Lai, S. C. Wong, and D. Lun, "A rate control algorithm using human visual system for video conferencing systems" Proc. Int. Conf. On Signal Processing, 1, Aug. 2002, pp. 656–660
- [6] Yun-Teng Roan; Pei-Yin Chen "A fuzzy search algorithm for estimation of motion vectors" IEEE Transactions on Broadcasting, vol. 46, Issue 2, Jun 2000, pp 121 –127
- [7] Ming Yunp and Wensheng Wung, "fast macroblock mode selection based on motion content classification in H.264/AVC" 2004 International Conference on Image Processing (ICIP)
- [8] M. A. Ben ayed, a. Samet, s. Marzouk, n. Souissi, m. Ben , n. Masmoudi, "New variable block-size motion estimation algorithm for H.264/avc," ISCCSP 2006
- [9] Zong Chen, "Efficient Block Matching Algorithm for Motion Estimation International Journal of Signal Processing 5;2 2009, 133-137
- [9] Shan Zhu and Kai-Kuang Ma, "A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation" IEEE transactions on image processing, vol. 9, no. 2, February 2000, pp 287-293
- [10] K. C. Lai, S. C. Wong, and D. Lun, "A rate control algorithm using human visual system for video conferencing systems" Proc. Int. Conf. On Signal Processing, 1, Aug. 2002, pp. 656–659
- [11] Weisi Lin, , Li Dong and Ping Xue, "Visual Distortion Gauge Based on Discrimination of Noticeable Contrast Changes" IEEE transactions on circuits and systems for video technology, vol. 15, no. 7, July 2005, pp 900-906
- [12] M. P. Eckert and A. P. Bradley, "Perceptual quality metrics applied to still image compression" Signal Process., vol. 70, 1998, pp 177–200