ISSN 1991- 8690

الترقيم الدولي ٨٦٩٠ - ١٩٩١

website :http:// jsci.utq.edu.iq Email: utjsci@utq.edu.iq

Fast Video Compression Method based on Wavelet Transform and Simple Efficient Search Algorithm

Hikmat Z. Neima[*] Hzn2007@yahoo.com

Rana M. Ghadban* Rana_ghadban@yahoo.com

Ali A. Yassin** Aliadel2005alamre@yahoo.com

[*] Department of Computer Science- College of Science-University of Basrah- Basrah, Iraq.
[**]Department of Computer Science- College of Education- University of Basrah- Basrah, Iraq.

Abstract

In this paper, a fast video compression method has been proposed. In proposed method, Firstly, frames were transformed using Discrete Wavelet Transform (DWT) in order to reduce computation time. The disparity between each two adjacent frames was estimated by Simple and Efficient Search (SES) algorithm. The result of the Motion Vector (MV) was encoded into a bit stream by Huffman encoding while the remaining part is compressed like the compression that is used in still image. The proposed method produced good results in terms of Peak Signal-to-Noise Ratio (PSNR), CR, and computation time.

Keywords: Video compression, discrete wavelet transform, motion estimation, motion compensation, simple and efficient search.

طريقه سريعه لضغط الفيديو بالأعتماد على التحويل المويجي و خوارزمية البحث البسيطه و الكفوءه

حكمت زنيد نعمه* رنا مجيد غضبان* علي عادل ياسين**

* قسم علوم الحاسبات ، كلية العلوم ، جامعة البصره ، البصره ، العراق.

** قسم علوم الحاسبات ، كلية التربيه ، جامعة البصره ، البصره ، العراق.

الخلاصه

في هذا البحث، تم اقتراح طريقة سريعة لضغط الفيديو. الخطوة الأولى في الطريقة المقترحة، استخدام التحويل المويجي لغرض تحويل اطارات (Frames) الفيديو بهدف تقليل وقت المعالجه المطلوب. تم استخدام خوارزمية البحث البسيطه و الكفوءه (Simple and Efficient Search) لغرض ايجاد متجه الحركة (Motion Vector) والذي يمثل التفاوت بين اطارات الفيديو بهدف تقدير الحركة و من ثم تعويضها. بعد ذلك تم استخدام ترميز هوفمان (Huffman encoding) لترميز متجه الحركة، اما الجزء المتبقي فيتم ضغطه كصورة ثابتة (ضغط الصور). الطريقه المقترحه اعطت نتائج جيده من حيث قمة الأشاره الى الضوضاء (PSNR) و نسبة الضغط و كذلك من حيث وقت المعالجه المستغرق في عملية ضغط الفيديو.

الكلمات الدالة : ضغط الفيديو ، التحويل المويجي المنفصل ، تقدير الحركة ، تعويض الحركة ، خوارزمية البحث البسيطه والكفوءه.

1.Introduction

Video data is recently used in a wide range of applications such as high-definition television (HDTV) and standard definition television (SDTV) broadcasting equipment, video conferencing transmitters, video cellular phones, digital video camcorders, and multimedia services for networking applications. Therefore, digital video compression has become as an essential process due to the large size that required by raw video files either in storage or transmission [1].The high redundancy existing among the successive frames of a video sequence makes it possible to achieve high compression ratio (CR) in video compression. It is possible to exploit this temporal correlation using motion estimation techniques. Motion estimation is the process of evaluating the movements between adjacent frames. There are various algorithms for motion estimation. Motion compensation is an effective method for reducing temporal redundancy found in video sequence compression [2].In video compression process, the first frame often compressed as a single image since it has no prior frame while every one of the rest frames will have its previous frame. Motion estimation will take the previous frame of the current frame in order to estimate the motion that occurred between these two frames [3].Due to its simplicity, block matching algorithm (BMA) is frequently used to estimate the motion of a given block to be encoded. In BMA, all pixels in a block are assumed to undergo a translational motion. An estimate of the current block can therefore be obtained by searching similar blocks in the previous encoded (or original image) frame around the current block [4, 5]. Fig. 1 illustrates flow of the process of video compression.

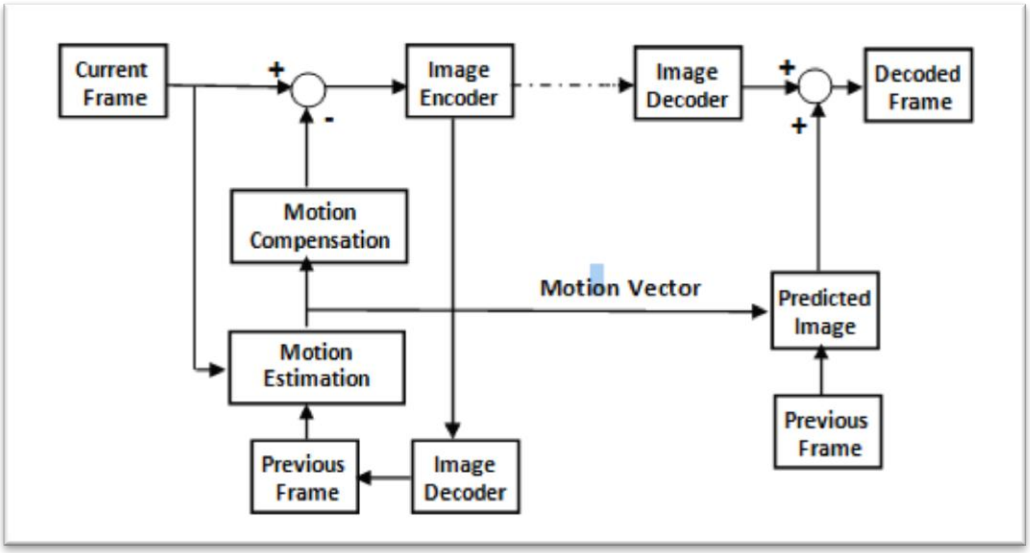


Fig. 1: Video compression flow [5]

Many researchers have proposed several methods to overcome the problem of time consuming in the process of video compression. Most of these researches focus on motion estimation stage as it is the stage that consumes the most time of video compression process. A partial block matching method to speed up the process of motion estimation has been proposed in [6]. Based on histogram, the proposed method seeks the block that will be used in block matching step. So that the number of blocks that is selected will be less than the number of blocks in other block matching methods. Consequently, the time that is consumed will be less.Multiple Block Size Search (MBSS) algorithm is presented in [7]. The presented multiple block-size search (MBSS) algorithm unifies the motion searches for different block sizes into a single searching process instead of independently performing the search for each block size. In this unified search, the suboptimal motion vectors for different block sizes are used to determine the next search steps.A partial distortion scheme has been proposed in [8]. Proposed scheme sacrifices little visual quality for hand-held devices in a given computation constraint condition. The scheme was done by combining computation aware and Directional asymmetric search with prediction (DASP) algorithm to save checking points. They opted to use the pixel decimation to increase the available number of checking points in order to further lower the required computation. They also analyze various computation-aware based block matching algorithms and present a good method to compute the motion vectors. The rest of this paper is organized as follows. Sections 2, 3, 4, and 5 introduce essential background information. In section 6, we present proposed method. Section 7 describes evaluation criteria. Section 8 gives the experimental results. Finally, the paper has been concluded in section 9.

2. Discrete Wavelet Transform (DWT)

Wavelet transform is one of important and useful computation tools for a variety of signal and image processing applications. In image processing field, the main process in wavelet transform is to filter signal of image by two filters, namely, low pass filter (L) and high pass filter. Then, it will down sampled by factor of two leading to compose transform of one level. Repeating of one level transform on the part of low pass output only, results multiple level transform. Two dimensional (2-D) wavelet transform can be obtained by applying 1-D wavelet transform, wavelet filter separately. This computation is done by carrying out 1-D transform on the rows signals one time and on the columns signal another time. As a result of that, it separates image signals into four sub-band images: LL (low frequency in horizon and vertical), LH (low frequency in horizon and high frequency in vertical), HL(high frequency in horizon and low frequency in vertical), HH(high frequency in horizon and vertical). Therefore, it is possible to use different methods for the sake of enhancement of the details in different frequency domain [9]. LL sub-band image often contains the most important information of the original image and it is usually called approximations the three other sub-band images are named as details. HH sub-band normally includes the small coefficients which are more likely due to undesirable noise [10]. Fig. 2 shows Foreman image and its three levels DWT.

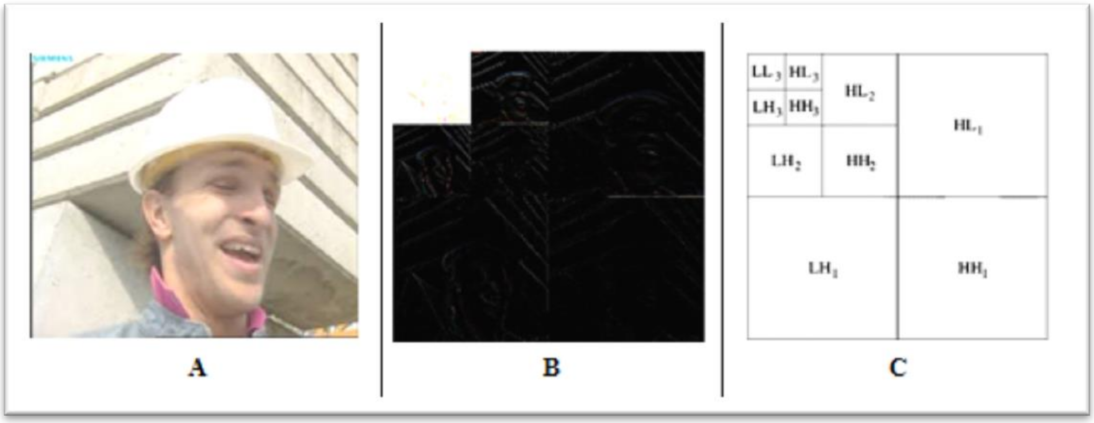
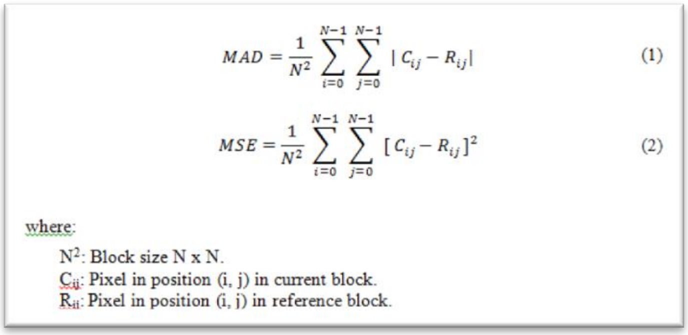


Fig. 2 : A) Foreman image B) Three levels Discrete Wavelet Transform of Lena image C) Low and High sub-bands resulted from three levels DWT [10]

3. Motion Estimation and Motion Compensation

Motion Estimation (ME) and Motion Compensation (MC) are the fundamental parts of video coding systems and form the core of many video processing applications. Motion estimation eliminates temporal redundancy from video by exploiting the temporal correlation between successive frames, so that it reduces the amount of data to be transmitted or stored while maintaining sufficient data quality. However, ME extract temporal motion information from video sequences, while motion compensation uses this motion information for efficient inter frame coding [11]. Motion estimation predicts motion between two successive frames to generate a motion vectors (MVs) which represent the change between them. Consequently, these motion vectors and the prediction error are transmitted instead of the frame itself. With this process, the encoder will have sufficient information to faithfully reproduce the frame sequence. Block- based motion estimation is most used method because of its simplicity and performance. That made made it the standard approach in the video coding systems. The procedure of BMA is to divide the frames into a block of N×N pixels, to match every block of the current frame (CF) with his most similar block inside a research window in the reference frame (RF) and to generate the motion vector [12]. Consequently, for this method, the most important parameters here are the size of the block N and the size of the search window P. However, the block matching is based on minimizing a criterion like the Mean Absolute Error (MAD) or the Mean Square Error (MSE) which is the most common block distortion measure for matching two blocks and it provide more accurate block matching, see Eq.1, and

Eq.2. The MV will be applicable to every pixels of the same block which reduce the computational requirement. To identify the best corresponding block, the simplest way is to evaluate every block in the reference frame (exhaustive search, ES). But, although this method find generally the appropriate block, it consumes a high computations time [13].



$$MAD = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} | C_{ij} - R_{ij}| \qquad (1)$$

$$MSE = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} [C_{ij} - R_{ij}]^2 \qquad (2)$$

where:

N^2: Block size N x N.
C_{ij}: Pixel in position (i, j) in current block.
R_{ij}: Pixel in position (i, j) in reference block.

4. Three Steps Search (TSS)

TSS is one of the earliest attempts at fast block matching algorithms and dates back to mid 1980s. The TSS is the algorithm that limits the number of checking points in a search area. The general idea is represented in Fig. 3, it starts with the search location at the center and sets the 'step size' S = 4, for a usual search parameter value of 7. It then searches at eight locations +/- S pixels around location (0,0). From these nine locations searched so far it picks the one giving least cost and makes it the new search origin. It then sets the new step size S = S/2, and repeats similar search for two more iterations until S = 1.At that point, it finds the location with the least cost function and the macro block at that location is the best match. The calculated motion vector is then saved for transmission. It gives a flat reduction in computation by a factor of 9 [14, 15].

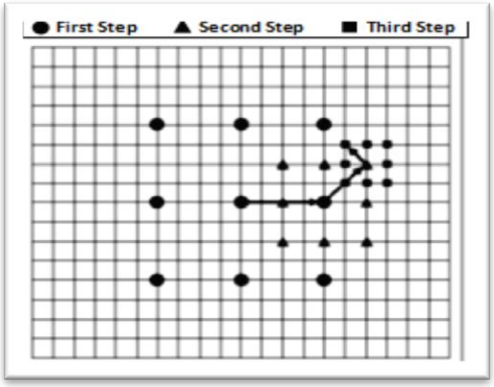


Fig. 3: Three Step Search procedure. The motion vector is (5, -3) [16]

5. Simple and Efficient Search (SES)

SES is an extension to TSS and exploits the assumption of unimodal error surface. The main idea behind the algorithm is that for a unimodal surface there cannot be two minimums in opposite directions and hence the 8 points fixed pattern search of TSS can be changed to incorporate this and save on computations [16].More detailed, in SES algorithm, first the direction from center location to a surrounding location as a search direction will be defined. Then, in the TSS, there are a total of eight search directions, as shown in Fig. 6, involved in each step. It is observed that, due to uniformly distributed search pattern, each direction has an opposite direction which is also searched [17].This is not effective under the unimodal error surface assumption on which the TSS is based. Recall that the unimodal error surface has the property that the block matching error increases monotonically as the search is along the direction away from the position of the global minimum error, which in turn implies that minimum cannot occur in two directions opposite to each other simultaneously. On the other hand, to determine which directions are to be chosen, additional computation is needed. A closer observation indicates that we can first check two arbitrary but orthogonal directions (say directions 1 and 7 in Fig. 6) and then restrict the search area of each step in certain quadrant which contains only three search directions. In that sense, we propose the SES algorithm [18].The SES algorithm still has three steps like TSS, but the innovation is that each step has further two phases. The search area is divided into four quadrants and the algorithm checks three locations A, B and C as shown in Fig. 4. A is at the origin and B and C are S = 4 locations away from A in orthogonal directions. Depending on certain weight distribution amongst the three the second phase selects few additional points Fig4. The rules for determining a search quadrant for seconds phase are as follows [17]:

If MAD(A) > MAD(B) and MAD(A) > MAD(C), select (b);

If MAD(A) > MAD(B) and MAD(A) < MAD(C), select (c);

If MAD(A) < MAD(B) and MAD(A) < MAD(C), select (d);

If MAD(A) < MAD(B) and MAD(A) >MAD(C), select (e);

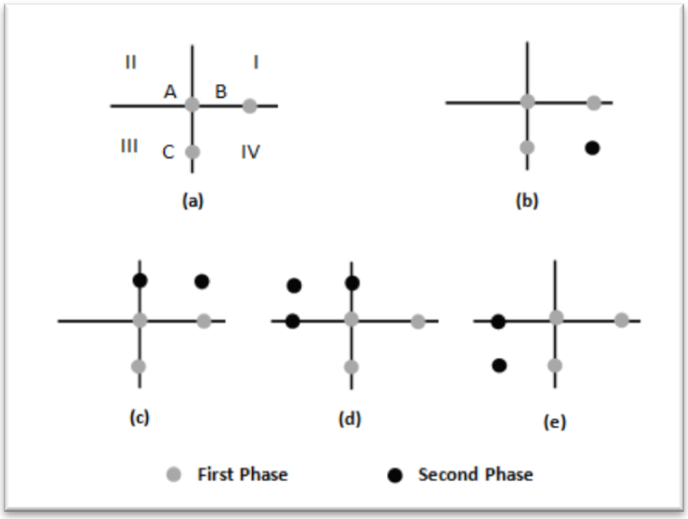


Fig 4: Search patterns corresponding to each selected quadrant: (a) Shows all quadrants (b) quadrant I is selected (c) quadrant II is selected (d) quadrant III is selected (e) quadrant IV is selected [16]

Once we have selected the points to check for in second phase, we find the location with the lowest weight and set it as the origin. We then change the step size similar to TSS and repeat the above SES procedure again until we reach S = 1. The location with the lowest weight is then noted down in terms of motion vectors and transmitted. An example process is illustrated in Fig. 5.

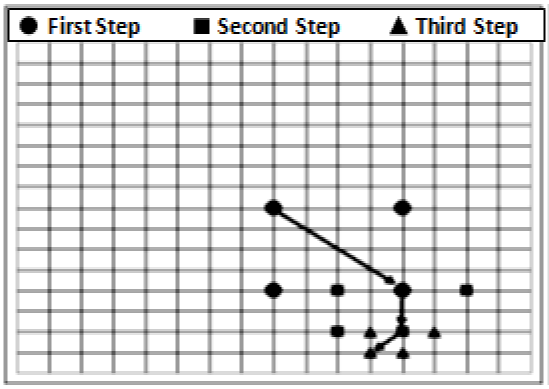


Fig. 5: The SES Algorithm [17]

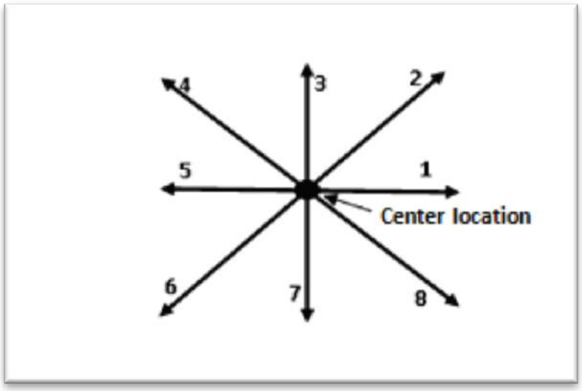


Fig. 6: Search directions in a search pattern of the TSS [18]

6. Proposed Method

In proposed method, there are four main steps. The first step is to fractionate video clip into frames and apply discrete wavelet transform on these frames separately. In the second step, first frame that extracted from the video clip will be compressed as a still image. That is because it has no previous frame to be used for motion vector estimation.The third step is to use the simple and efficient matching algorithm to estimate the motion vector for the remaining frames (from second frame to the last one). After the motion vector is gotten, it will be encoded using Huffman coding.Finally, combine the first compressed frame with the compressed successive frames to compose the completed compressed video clip. Fig. 7 illustrates the flow chart of proposed method.

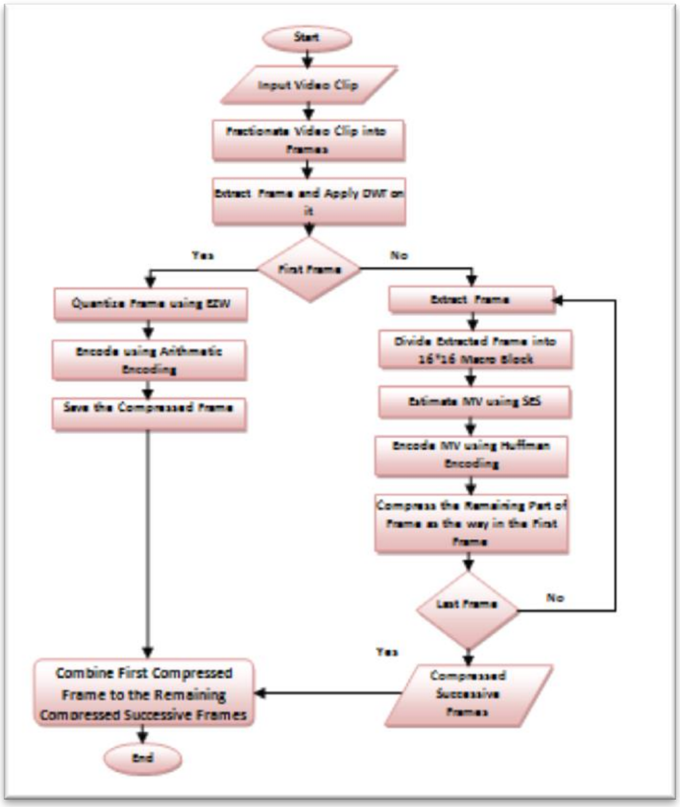

Start
Input Video Clip
Fractionate Video Clip into Frames
Extract Frame and Apply DWT on it
Yes
First Frame
No
Quantize Frame using EZW
Encode using Arithmetic Encoding
Save the Compressed Frame
Extract Frame
Divide Extracted Frame into 16*16 Macro Block
Estimate MV using SES
Encode MV using Huffman Encoding
Compress the Remaining Part of Frame as the way in the First Frame
Last Frame
No
Yes
Compressed Successive Frames
Combine First Compressed Frame to the Remaining Compressed Successive Frames
End

Fig. 7: Flow Chart of Proposed Method

7. Evaluation Criteria

Evaluation criteria that usually used in digital image and video compression are in two directions. First direction is to evaluate quality of the reconstructed image. Second direction is compression ratio (CR). In terms of quality evaluation, two mathematical metrics are used. First one is mean square error (MSE), which measures the cumulative square error between the original and the reconstructed image. Second metric is peak signal-to-noise ratio (PSNR). The formulas for MSE, and PSNR are giving as [19]:

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [\, f(i,j) - g(i,j)]^2 \qquad (3)$$

where:

f(i, j): Original image,
g(i, j): Reconstructed image.
N, M: Dimensions of image.

$$PSNR = 10\log_{10}\frac{255^2}{MSE}$$

The Compression Ratio is the ratio of the number of bits (file size) of the encoded file to the original file size [20]:

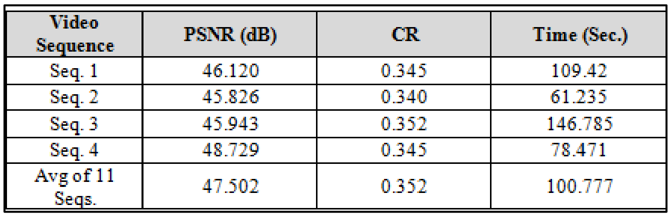
$$CR = \frac{Compressed\ file\ size}{Uncompressed\ file\ size} \qquad (5)$$

In addition to measuring the quality of image, we also measure the compression ratio. Compression ratio is the ratio of the compressed file size to the original file size. In general, the higher the compression ratio, the smaller is the size of the compressed file. Compression speed, on the other hand, is the amount of time required to compress and decompress the image. This value depends on a number of factors, such as the complexity of the algorithm, the efficient of the implementation, and the speed of the processor [20].

8. Experimental Results

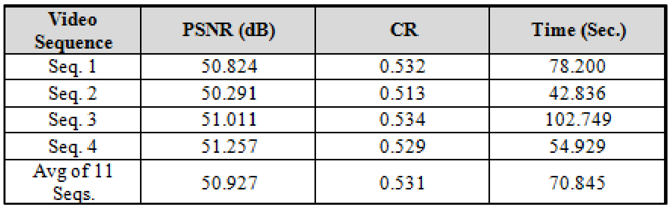
In experiments, the proposed method was implemented using Matlab 7.14.0 program and tested on 11 video sequences in AVI format. Ten of these sequences are standard (i.e. video sequences that are widely used in the video compression research community), while the last one is not standard video sequence. Standard sequences are downloaded from a freewebsite http://see.xidian.edu.cn/vipsl/database_Video.html#video in different properties. For instance, the first sequence (*Foreman*) is composed from 300 frames and its resolution is 192*144. The second sequence (*Football*) contains 120 frames with resolution of 192*144. The third standard sequence is (*Flowergarden*) composed from 360 frames. The resolution of each frame in the third sequence is 352*288. The non standard sequence *(Divers*) is obtained from the website http://www.reefvid.org/topic_clips.php?id=20.It onsists of 200 frames and in resolution of 384*288. Table 1 shows the results of video compression using SES algorithm. Table 1 contains the video sequence that were used, Seq.1, Seq2, Seq3, Seq4, and average of 10 video sequences. PSNR, CR, and computation time that been gotten using SES algorithm have been illustrated in Table 1. Table 2 demonstrates the results of proposed method on the same video sequences that were utilized in SES.

Table 1: PSNR, CR, and computation time of video compression using SES



Video Sequence	PSNR (dB)	CR	Time (Sec.)
Seq. 1	46.120	0.345	109.42
Seq. 2	45.826	0.340	61.235
Seq. 3	45.943	0.352	146.785
Seq. 4	48.729	0.345	78.471
Avg of 11 Seqs.	47.502	0.352	100.777

Table 2: PSNR, CR, and computation time of video compression using Proposed Method



Video Sequence	PSNR (dB)	CR	Time (Sec.)
Seq. 1	50.824	0.532	78.200
Seq. 2	50.291	0.513	42.836
Seq. 3	51.011	0.534	102.749
Seq. 4	51.257	0.529	54.929
Avg of 11 Seqs.	50.927	0.531	70.845

9. Conclusions

In this paper, a method for video compression has been proposed. Proposed method aims to decrease the computation time without much influence on PSNR and compression ratio. Referring to the results that are shown in Table 1, and Table 2, it is obviously that the values of PSNR, CR, and computation time are affected by the length and the resolution of each sequence. Additionally, we can notice clearly that the use of DWT minimized the processing time approximately 30%.

10. References

[1] Alp E., and Sarp E., "Two-Bit Transform for Binary Block Motion Estimation", IEEE Transactions on Circuits and Systems for Video Technology, VOL. 15, NO. 7, pp. 938-946, 2005.

[2] Mohamed G., *et al.*, "Adaptive Motion Estimation Block Matching Algorithms for Video Coding", International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS2006), pp. 427-430, 2005.

[3] Dongming Z., Gang C., and Xiaoguang G., "Improved Motion Estimation Based on Motion Region Identification", International Conference on Systems and Informatics (ICSAI 2012), pp. 2034-2037, 2012.

[4] Cheng K., and Chan S., "Fast Block Matching Algorithms for Motion Estimation", IEEE International Conference On Acoustics, Speech and Signal Processing, Vol. 4, pp. 2311-2314, 1996.

[5] M. H. Bisjerdi and A. Behrad, "Video Compression USING a New Active Mesh Based Motion Compensation Algorithm in Wavelet Sub-Bands," *Journal of Signal & Information Processing*, vol. 3, 2012.

[6] S. Park, *et al.*, "Adaptive Partial Block Matching Algorithm for Fast Motion Estimation", Digest of Technical Papers International Conference on

Consumer Electronics (ICCE), Las Vegas, NV, 9-13 Jan., pp. 371-372, 2010.

[7] K. Ng., *et al.*, "Multiple Block-Size Search Algorithm for Fast Block Motion Estimation", 7th International Conference on Information, Communications and Signal Processing (ICICS2009), Macau, 8-10 Dec., pp. 1-4, 2009.

[8] C. Chang., *et al.*, " Partial Distortion Based Computation-Constraint Motion Estimation", 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), Yichang, 21-23 April, pp. 1956 – 1959, 2012

[9] Q. Yan, and R. Li, " Novel Image Enhancement Algorithm based on Wavelet Multiscale", 3rd International Conference on Intelligent Networks and Intelligent Systems, 01 - 03 November, pp. 39-42, 2010.

[10] L. Passrija, A. Virk, and M. Kuar, "Performance Evaluation of Image Enhancement Techniques in Spatial and Wavelet Domains", International Journal of Computers and Technology, Vol. 3, No. 1, pp. 162-166, 2012.

[11] N. Verma, *et al.*, "Efficient Motion Estimation by Fast Three Step Search Algorithms," ed: Int. Journal of Advanced Research un Electrical, Electronics and Instrumentation Engineering, 2012.

[12] M. Tammen, *et al.*, "Fast Implementation of VC-1 with Modified Motion Estimation and Adaptive Block Transform," *Circuits and Systems,* vol. 1, p. 12, 2010.

[13] R. K. Purwar, *et al.*, "A block matching criterion for interframe coding of video," in *International Conference on Audio, Language and Image Processing, ICALIP 2008.* , pp. 133-137, 2008.

[14] J. Ratnottar, *et al.*, "Review towards the Fast Block Matching Algorithms for Video Motion Estimation," in *International Conference on Communication Systems and Network Technologies (CSNT),* pp. 153-156, 2012.

[15] C. Cheong Seong, *et al.*, "Review of energy efficient block-matching motion estimation algorithms for wireless video sensor networks," in *IEEE Symposium on Computers & Informatics (ISCI),* pp. 241-246, 2012.

[16]
Aroh Barjatya,“ Block Matching Algorithms For Motion Estimation,“ Student Member, IEEE, DIP 6620 Spring 2004 Final Project Paper.

[17] Jianhua L., and Ming L., “A Simple and Efficient Search Algorithm for Block Matching Motion Estimation”, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 7, No. 2, pp. 429-433, April 1997.

[18] P. Muralidhar and C. B. R. Rao, "Analysis of block matching motion estimation algorithms," in *Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT),* pp. 1-4, 2013.

[19] SONKA, M., HLAVAC, V. & BOYLE, R. "Image Processing, Analysis, and Machine Vision", (2008).

[20] Z. He, "Video compression and data flow for video surveillance," *Texas Instruments, SPRY104,* 2007.