Scientific Research

Video Compression USING a New Active Mesh Based Motion Compensation Algorithm in Wavelet Sub-Bands

Mohammad Hossein Bisjerdi, Alireza Behrad

Electrical Engineering Department, Faculty of Engineering, Shahed University, Tehran, Iran.
Email: bisjerdi@shahed.ac.ir, behrad@shahed.ac.ir

ABSTRACT

In this paper, a new mesh based algorithm is applied for motion estimation and compensation in the wavelet domain. The first major contribution of this work is the introduction of a new active mesh based method for motion estimation and compensation. The proposed algorithm is based on the mesh energy minimization with novel sets of energy functions. The proposed energy functions have appropriate features, which improve the accuracy of motion estimation and compensation algorithm. We employ the proposed motion estimation algorithm in two different manners for video compression. In the first approach, the proposed algorithm is employed for motion estimation of consecutive frames. In the second approach, the algorithm is applied for motion estimation and compensation in the wavelet sub-bands. The experimental results reveal that the incorporation of active mesh based motion-compensated temporal filtering into wavelet sub-bands significantly improves the distortion performance rate of the video compression. We also use a new wavelet coder for the coding of the 3D volume of coefficients based on the retained energy criteria. This coder gives the maximum retained energy in all sub-bands. The proposed algorithm was tested with some video sequences and the results showed that the use of the proposed active mesh method for motion compensation and its implementation in sub-bands yields significant improvement in PSNR performance.

1. Introduction

Recently, digital video communication and digital video storage have found many applications like digital TV, computer multimedia and video conferencing, to name a few. To reduce transmission bandwidth and storage capacity, it is necessary to apply an appropriate compression algorithm to videos. The Discrete Wavelet Transform (DWT) is the transform of choice in recent video compression algorithms. Adopted by the JPEG2000 image compression standard [1], it significantly outperforms the algorithms based on other transforms, such as the discrete cosine transform, in terms of objective metrics as well as perceptual image quality [2]. The success of the DWT stems from its ease of computation and its inherent decomposition of an image into non-overlapping subbands that enables the design of efficient quantization algorithms and allows for the incorporation of the human visual system.

One of the main stages in video compression is motion estimation and compensation. In most of the frames in a video, the only difference between subsequent frames is the result of either camera's or objects' motion in the frames. Motion estimation and compensation algorithms take advantage of this clue to provide a way for creating frames of a movie from a reference frame. Different algorithms have been proposed for motion estimation and compensation such as block-displacement methods [3,4], optical flow [5], feature matching [6], mesh based algorithms [7] and model based methods [8].

In block-displacement methods, the matching process uses a block based method. That is, video frames are divided into blocks and motion vectors of the blocks in the current frame point to the closest matching blocks in the reference frame. If there is no motion or only pure translational motion, the motion vectors provide a one-to-one mapping between pixels in the reference frame and pixels in the current frame. However, in more realistic video sequences, motion is usually much more complex and yield one-to-many mappings for some pixels in the reference frame and no mapping for others. The latter pixels are thus unconnected.

Although block-displacement techniques are popular, they have a number of drawbacks. First, the rigid block motion model fails to capture all aspects of the motion field, leaving a significant number of pixels unconnected between the frames. These unconnected pixels are coded

separately, which decreases coding efficiency. Second, it is difficult to achieve sub-pixel accuracy in the block-displacement techniques while maintaining invertibility of the temporal transform. Finally, implementation of block-displacement techniques is hindered by numerous unconnected pixels.

Motion estimation using image intensity is another method for motion compensation in video compression. Two usual methods of this category are optical flow and feature matching. The optical flow and the feature matching methods are similar. In optical flow methods, motion vectors are calculated for all pixels of the image; however, in the feature matching algorithms, motion vectors are only calculated for strong features in the image. Considering the similarity between optical flow and feature matching algorithms, they are also called dense and sparse optical flow algorithms, respectively.

In both optical flow algorithms, it is assumed that intensity changes only because of the motion of pixels. Optical flow equation has two unknown variables; thus, the motion vector cannot be estimated without other constraints, which is called aperture problem. Therefore, different constraints are added to calculate motion vectors. Constant optical flow method (Lucas-Kanade algorithm) [9] assumes that motion fields are well approximated by a constant vector within any small region of the image plane. To increase the speed and accuracy of the method, multi-resolution implementation of the algorithm using image pyramids may be also employed [10].

Optical flow algorithms mainly suffer from the aperture problem. Because motion vectors are calculated independently, they are somewhat chaotic as well.

Mesh based algorithms [11] use some nodal points to generate a triangular or rectangular mesh on the image. Then, by matching mesh elements in different frames, motion vectors are calculated. The model based methods employ a 2-D or 3-D geometric model for motion estimation.

Model based algorithms are proper for global motion estimation like camera motion; however, they fail to calculate local motion in the video frames.

In this paper, a new method is proposed for video compression in the wavelet domain. The proposed algorithm is based on the new motion estimation and compensation method in wavelet sub-bands. To calculate motion for different pixels in the image a mesh model based on new mesh energy is proposed.

The organization of this paper is as follows: in the next section, different 2-D DWT algorithms are reviewed. In Section 3, the proposed active mesh based algorithm is described. The proposed video compression procedure is described in Section 4. Experimental results appear in Section 5 and conclusion is provided in Section 6.

2. 2-D Discrete Wavelet Transform

A 2-D separable discrete wavelet transform is equivalent to two consecutive 1-D transforms. For an image, a 2-D DWT is implemented as a 1-D row transform followed by a 1-D column transform. Transform coefficients are obtained by projecting the 2-D input image $X(u,v)$ onto a set of 2-D basis functions, which are expressed as the product of two 1-D basis, as shown in Equation (1).

$$\begin{aligned}
\phi(u,v) &= \phi(u)\phi(v) \\
\psi_1(u,v) &= \psi(u)\phi(v) \\
\psi_2(u,v) &= \phi(u)\psi(v) \\
\psi_3(u,v) &= \psi(u)\psi(v)
\end{aligned} \tag{1}$$

The 2-D DWT analysis can be expressed as the set of equations as shown in Equation (2). The scaling function $\phi(u,v)$, wavelets $\psi_1(u,v)$, $\psi_2(u,v)$, $\psi_3(u,v)$ and the corresponding transform coefficients $X(N,j,m)$, $X^{(2)}(N,j,m)$ and $X^{(3)}(N,j,m)$ correspond to different sub-bands in the decomposition. $X(N,j,m)$ are the coarse coefficients that constitute the LL sub-band. The $X^{(1)}(N,j,m)$ coefficients contain the vertical details and correspond to the LH sub-band. The $X^{(2)}(N,j,m)$ coefficients contain the horizontal details and correspond to the HL sub-band. The $X^{(3)}(N,j,m)$ coefficients represent the diagonal details in the image and constitute the HH sub-band. Thus, single-level decomposition at scale $(N+1)$ has four sub-bands as shown in **Figure 1**.

$$\begin{aligned}
&X(N,j,m) \\
&= \iint x(u,v)2^N \phi(2^N u - j)\phi(2^N v - m)\,\mathrm{d}u\mathrm{d}v \Rightarrow LL \\
&X^{(1)}(N,j,m) \\
&= \iint x(u,v)2^N \psi(2^N u - j)\phi(2^N v - m)\,\mathrm{d}u\mathrm{d}v \Rightarrow LH \\
&X^{(2)}(N,j,m) \\
&= \iint x(u,v)2^N \phi(2^N u - j)\psi(2^N v - m)\,\mathrm{d}u\mathrm{d}v \Rightarrow HL \\
&X^{(3)}(N,j,m) \\
&= \iint x(u,v)2^N \psi(2^N u - j)\psi(2^N v - m)\,\mathrm{d}u\mathrm{d}v \Rightarrow HH
\end{aligned} \tag{2}$$

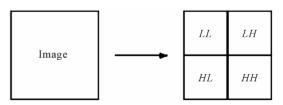The synthesis bank performs the 2-D IDWT to reconstruct $X(u,v)$. 2-D IDWT is given in Equation (3).

Figure 1. Single-level 2-D wavelet decomposition (analysis).

$$X(u,v) = \sum_j \sum_m X(N, j, m) 2^N \phi(2^N u - j) \phi(2^N v - m)$$

$$+ \sum_{i=N}^{M-1} \sum_j \sum_m X^{(1)}(i, j, m) 2^i \psi(2^i u - j) \phi(2^i v - m)$$

$$+ X^{(2)}(i, j, m) 2^i \phi(2^i u - j) \psi(2^i v - m)$$

$$+ X^{(3)}(i, j, m) 2^i \psi(2^i u - j) \psi(2^i v - m)$$

$$(3)$$

A single stage of a 2-D filter bank is shown in **Figure 2**. First, the rows of the input image are filtered by the high-pass and low-pass filters. The outputs of these filters are down-sampled by two and then the columns of the outputs are filtered and down sampled to decompose the image into four sub-bands. The synthesis stage performs up-sampling and filtering to reconstruct the original image.

3. Motion Estimation Using the Proposed Active Mesh Method

To find a remedy for the problem of motion estimation methods in video compression, a new algorithm is used based on an active mesh model. The method uses a combination of 2-D formable meshes and feature matching algorithm to calculate motion vectors more precisely. In this algorithm, image features (interest points) are extracted using the Kanade-Lucas-Tomasi (KLT) feature extractor technique [12]. Interest points are used as mesh vertices in order to generate an unstructured mesh over the video frame by Delaunay triangulation algorithm [13]. To compute motion vectors, the matching points of the features are found using the improved Lucas-Kanade feature matching algorithm [11], which utilizes image pyramids for the calculation of match points. The method calculates match point of each feature individually with sub-pixel accuracy. However, because motion vectors are calculated independently, motion vectors are somewhat chaotic. Thus, the match points of the previous step are not considered as true match points. In order to find true match points, mesh energies are defined based on the

location of feature points, their matches and other attributes of the generated mesh and video frames. The tracking of mesh is performed by minimizing the mesh energy which considers the motion information of nearby features to remove erroneous matches and enhance the accuracy.

3.1. Mesh Energy

The mesh energy is the combination of internal and external energies. To achieve high accuracy, different energy functions are defined for mesh model including matching, interest points, correlation, sum of squared differences (SSD) and corner energies. These energies are calculated for match points as well as predefined neighborhoods around them. The mesh energy is the weighted sum of internal and external energies. By minimizing the mesh energy in each frame, the location of image features and the related mesh are calculated in the subsequent frames.

3.2. Internal Energy

The normalized external energy is defined for every mesh vertex as follows:

$$E_{\text{int}} = \lambda L_{cur}(x) + \lambda L_{cur(y)} \qquad (4)$$

where $L_{cur}(x)$ and $L_{cur(y)}$ are the x, y components for length of mesh lines. The value of λ is given by:

$$\lambda = \left(\frac{L_{pre} - L_{cur}}{\alpha_{Line} L_{cur}} \right) \qquad (5)$$

where L_{pre} and L_{cur} are lengths of the mesh lines in the previous and current frames, respectively. The α_{line} parameter is a coefficient that regularizes effect of the applied energy. Internal energy for origin and destination nodes is given by:

$$E_{org}(i) = -E_{int} \qquad (7)$$
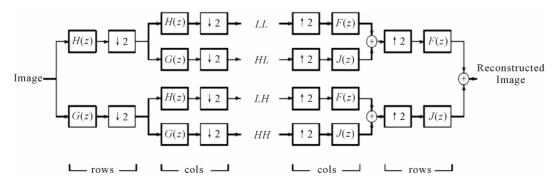
$$E_{dst}(i) = +E_{int} \qquad (8)$$

Figure 2. One level filterbank for computing 2-D DWT and IDWT.

Definition of the internal energy using the mentioned equations gives a rigid-elastic structure to the mesh, which is useful for motion estimation.

3.3. External Energy

To test different motion estimation algorithms in the spatial domain, they were applied in video compression algorithm, as demonstrated in **Figure 3**. This figure illustrates the utilizing process of motion estimation algorithm in compression. As can be seen in **Figure 3**, the difference between the current frame and the previous one, which was compensated with one of these previous, mentioned methods, will be send to the encoder. As a result of using the difference between two sequences, we managed to send the minimum amount of information to the decoder. Then the residual image will be encoded and decoded by the SPIHT method, which is based on wavelet transformation. While the motion estimation process is being done, the motion vectors obtained by these three methods are send to the decoder. In the decoder section, the previous frame which was completely send to the decoder produced the restored image by means of motion vectors. This image will be added to the encoders residual image and in this way the main image will be produced. This process is performed for all of the images in a image sequence and all images are restored by this method.

3.4. Video Compression Steps in Wavelet Sub-Bands

To test different motion estimation algorithms in the wavelet sub bands, they were applied in video compression algorithm, as demonstrated in **Figure 4**. For video compression, optimized active mesh motion estimation in wavelet sub bands was used. Then, motion compensation algorithm was used with sub-pixel accuracy. Followed by motion compensation algorithm, an embedded wavelet-based coding was used which was applied to residual frames. The wavelet based encoding was based on SPIHT algorithm [14-16] with the specified wavelet and number of levels. The SPIHT algorithm involved a 2-D DWT followed by a progressive bit plane coding of wavelet coefficients using a zero tree-like quantization structure. For decompression, the wavelet-based decoding algorithm was applied and using the motion vectors obtained by motion estimation algorithm, the original frame was reconstructed.

The external energy utilizes image information such as intensity and texture information to locate the true matching points of mesh nodes. The external energy was defined based on the following intuitions:

- Considering the short interval between frames, nodes are mostly matched with similar points in the next frames.
- Since nodes are the interest points (corners) of the first frame, the nodes are probably matched with corners or interest points in the subsequent frames.

Based on the above intuitions, the normalized external energy was defined for every mesh vertex as follows:

$$E_{ext} = d(x) S_{FM}\left(\frac{r-d}{r}\right) + d(y) S_{FM}\left(\frac{r-d}{r}\right) \quad (8)$$

where $d(x)$ and $d(y)$ are the x and y components related to the distance between feature points in the previous frame and matching points in the current frame, r is the radius of the search area and S_{FM} is the correlation criterion which is given by:
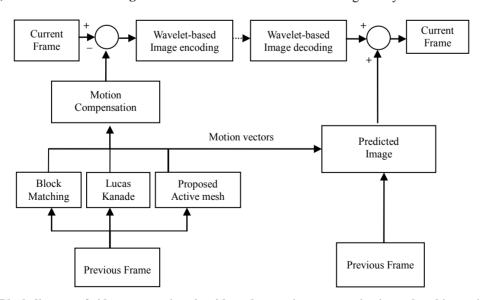

Figure 3. Block diagram of video compression algorithm when motion comensation is employed in spatial domain.
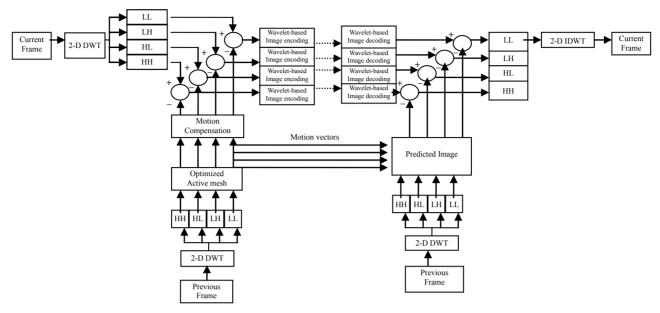
Figure 4. Block diagram of video compression algorithm when motion comensation is employed in wavelet sub-bands.

$$S_{FM} = \frac{F_{corr} + 1}{2}$$

where F_{corr} is the correlation coefficient between matching points. The sum of the internal and external energy is considered as the node energy.

3.5. Energy Minimization

The Greedy algorithm [17] is used here to minimize the calculated mesh energy in deformable meshes. The Greedy algorithm for minimization of the mesh energy and tracking mesh nodes has the following steps:

1) Select one of the unselected nodes of mesh.

2) Fix the location of other vertices and move the selected vertex in the window centered on the initial location of the selected vertex.

3) Calculate the total mesh energy for all possible locations of Step 2.

4) Move the vertex to the point with the minimum energy.

5) Repeat Steps 1 to 4 to reach the minimum point and/or maximum iteration.

6) Update mesh elements and parameters.

3.6. The Proposed Video Compression Algorithm

We use two different methods for video compression. In the first method, which is shown in **Figure 3**, motion estimation and compensation is applied in spatial domain.

As the figure shows, we utilize different motion estimation algorithms for comparison. In this method, the motion compensation is carried out in spatial domain to obtain the predicted image. Then difference of the current frame and the predicted image is fed for wavelet decomposition. In the case of proper motion estimation, the difference or residual image has less amount of information for decoding. Then the residual image are encoded and decoded by the SPIHT method [14-16], which is based on wavelet transformation. The SPIHT algorithm involves a 2-D DWT followed by a progressive bit plane coding of wavelet coefficients using a zero tree-like quantization structure.

In the decoder section, by using the motion vectors and the decompressed previous frame the predicted image is also reconstructed. By adding the predicted image to encoded residual image the decompressed image is obtained.

Our second approach benefit from motion estimation in wavelet sub-bands as shown in **Figure 4**. In this approach the proposed active mesh method for motion estimation is employed in wavelet sub-bands. The motion estimation algorithm calculates motion vectors with subpixel accuracy. In this algorithm, the predicted image is calculated for all images in the wavelet sub-bands. Then SPIHT algorithm is utilized for video coding in subbands. Motion vectors for all sub-bands are sent for video decoding as well. For decompression, the wavelet-based decoding algorithm is applied and using the motion vectors and the predicted sub-band images, the original frame is reconstructed.

4. Experimental Results

The proposed algorithm was implemented using a Microsoft Visual C++ program and tested with several vid-

eo sequences. To show the efficiency of the proposed algorithm, we also implemented Lucas-Kanade feature matching algorithm and motion estimation algorithm using block matching. Algorithms were tested with different videos including the Coast Guard video (100 frames), Motor video (100 frames), Snake video (100 frames), NASCAR video (100 frames), Foreman video (100 frames), Mother video (100 frames), Akiyo video (100 frames), Hall video (100 frames), Silent video (100 frames) and Boat video (100 frames). The first video sequence has the spatial resolution of 320×240 pixels and the temporal sampling rate of 30 frames/sec. The second sequence has the spatial resolution of 160×120 pixels and the temporal sampling rate of 25 frames/sec. The third sequence has the spatial resolution of 352×240 pixels and the temporal sampling rate of 29 frames/ sec. The last five sequences have a spatial resolution of 160×120 pixels and the temporal sampling rate of 10 frames/sec.

For block matching method, the block size for motion estimation is 16×16. Other parameters for block match-

ing and feature matching algorithm were selected in order to give the best results.

Figure 5 shows motion vectors of two consecutive frames for two sample video sequences. As the figure shows, the proposed algorithm has less outliers or incorrect matches. **Figures 6** and **7** demonstrate the result of the proposed active mesh model for motion estimation in spatial domain and wavelet sub-bands respectively. Figures show the results of the proposed algorithm for mother video sequence.

Figure 8 shows the frame-by-frame PSNR performance for the proposed video compression algorithm. To compare the results of the proposed method with those other methods the results of the following algorithm are shown in the figure:

- Proposed video compression algorithm with active mesh based motion compensation in wavelet sub-bands (Wavelet active mesh).
- Proposed video compression algorithm with active mesh based motion compensation in spatial domain (Spatial active mesh).
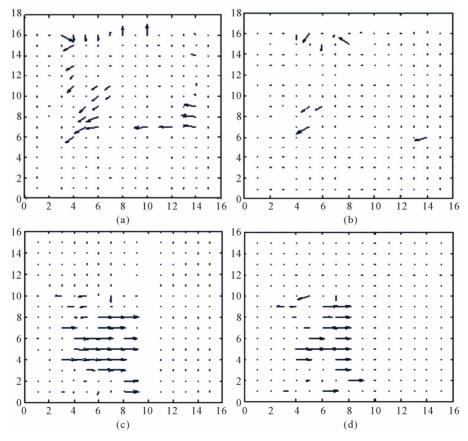

Figure 5. Motion vectors of two sample consecutive frames for Coastguard and Motor video sequences. (a) Motion vectors for sample consecutive frame of Coastguard video using the proposed active mesh model; (b) Motion vectors for sample consecutive frame of Coastguard video using Luca-Kanade method; (c) Motion vectors for sample consecutive frame of Motor video using the proposed active mesh model; (d) Motion vectors for sample consecutive frame of Motor video using Luca-Kanade method.
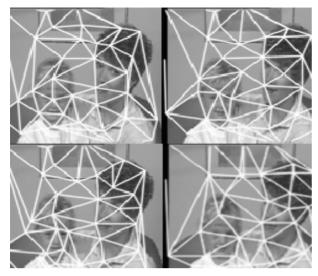
Figure 6. The results of the proposed active mesh method for motion estimation in spatial domain and its evolution over video frames.
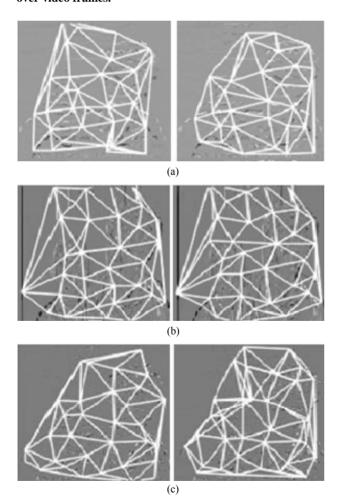

(a)

(b)

(c)

Figure 7. The results of the proposed active mesh method for motion estimation in wavelet sub-bands, (a) LH subband; (b) HL subband; (c) HH subband.

- Video compression using block matching algorithm. The structure of **Figure 3** is used for compression.
- Video compression using Lucas-Kanade feature matching algorithm. The structure of **Figure 3** is used for compression.
- Video compression using Redundant Discrete Wavelet Transform (RDWT) algorithm [18].

We tested the algorithms with the Coastguard, Motor, Snake, Nascar, Foreman, Mother, Akiyo, Hall, Silent and boat videos. **Figure 8** shows the frame-by-frame PSNR for Coastguard, Motor, Snake, Nascar, Foreman, Mother, Silent and boat videos, respectively. Bit rates for all of these sequences are 0.5 bpp (1.3 Mbps). **Table 1** shows the average PSNR for different test videos. According to the results of **Figure 8** and **Table 1**, it can be observed that the proposed active mesh model in wavelet sub-bands yields significant improvement in PSNR performance. **Table 2** shows the execution time of different algorithms for various test videos. According to the results of **Table 2**, it can be observed that the video compression algorithm with Lucas-Kanade motion estimation algorithm is the fastest method. It is also obvious that the proposed algorithm has proper execution speed comparing other methods.

5. Conclusion

In this paper, a new method for motion estimation and compensation and its application for video compression were proposed. The proposed motion estimation algorithm is based on mesh model and new sets of energy functions. The proposed motion estimation algorithm was applied in spatial and wavelet sub-bands for video

Table 1. Average PSNR performance (dB) for different videos at 1.3 Mbps (0.5 bpp).

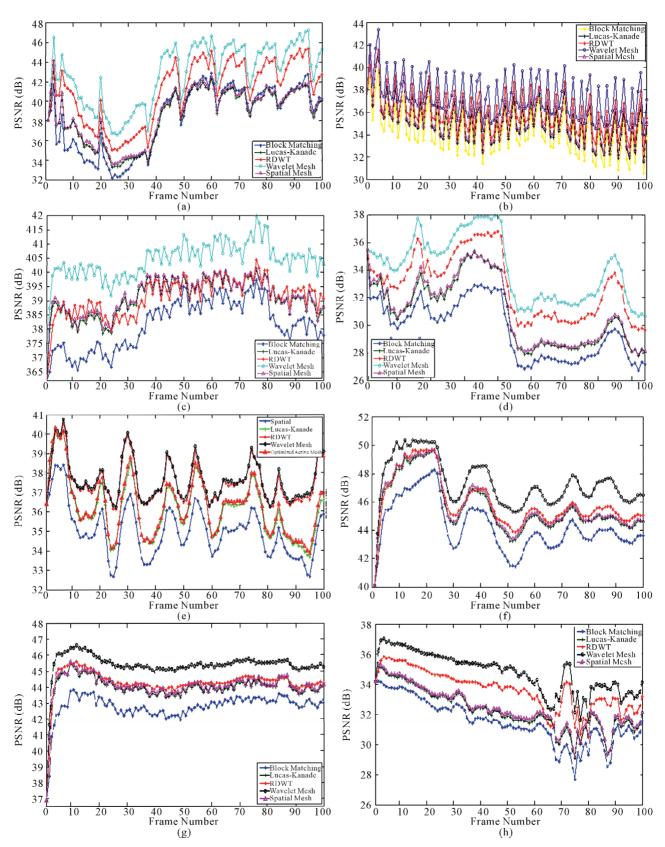| Video Sequences	Compression Methods				
	Lucas-Kanade	Spatial Active Mesh	Block Matching	RDWT	Wavelet Active Mesh
Nascar	32.48	32.73	31.18	32.34	33.58
Coastguard	36.21	36.45	34.67	41	42.79
Motor	36.44	36.68	35.22	36.02	37.54
Snake	38.35	38.47	37.05	39.11	40.42
Foreman	34.98	36.14	36.28	37.52	37.64
Mother	44.26	45.63	45.79	45.97	47.34
Akiyo	49.17	50.42	50.61	50.74	53.05
Hall	42.74	43.96	44.09	44.25	45.38
Silent	40.02	41.99	42.26	44.88	47.15
Boat	31.49	32.22	32.38	32.63	34.76

Figure 8. Frame-by-frame PSNR for sample videos at 0.5 bpp (1.3 Mbps), (a) Coastguard; (b) Motor; (c) Snake; (d) Nascar; (e) Foreman; (f) Mother; (g) Boat; (h) Silent.

Table 2. Execution time in second for the proposed and various implemented algorithms.

Video Sequences	Compression Methods				
	Lucas-Kanade	Spatial Active Mesh	Block Matching	RDWT	Wavelet Active Mesh
Nascar	64	65	62	64	66
Coastguard	72	73	70	82	84
Motor	72	74	72	73	75
Snake	76	75	78	78	80
Foreman	68	71	74	77	76
Mother	86	90	90	90	94
Akiyo	95	100	100	103	106
Hall	84	86	88	88	90
Silent	80	82	84	90	94
Boat	75.9	64	65	67	68

compression. We tested the proposed algorithm with several video sequences and compared its results with those of other methods. Experimental results showed that the active mesh model in wavelet sub-bands enhances the PSNR considerably.

REFERENCES

[1] ITU-T Recommendation T.800. JPEG2000 Image Coding System, Part 1, ITU Std., July 2002.

[2] A. Skodras, C. Christopoulis and T. Ebrahimi, "The JPEG2000 Still Image Compression Standard," *IEEE Signal Processing Magazine*, Vol. 18, No. 5, 2001, pp. 36-58. doi:10.1109/79.952804

[3] S.-J. Choi and J. W. Woods, "Motion-Compensated 3-D Subband Coding of Video," *IEEE Transactions on Image Processing*, Vol. 8, No. 2, 1999, pp. 155-167. doi:10.1109/83.743851

[4] P. Chen and J. W. Woods, "Bidirectional MC-EZBC with Lifting Implementation," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 14, No. 10, 2004, pp. 1183-1194. doi:10.1109/TCSVT.2004.833165

[5] Y. M. Chi and T. D. Tran and R. Etinne-cummings, "Optical Flow Approximation of Sub-Pixel Accurate Block Matching for Video Coding," *Proceedings of IEEE International Conference on Acoustic*, *Speech and Signal Processing*, Honolulu, 15-20 April 2007, pp. 1017-1020.

[6] D. C. Liu and S. Cheng, "A Brief Introduction of Feature Matching," *Proceedings of* 2008 *IEEE Region* 5 *Confer-*
ence*, Kansas, 17-20 April 2008, pp. 1-4.

[7] M. Eckert, D. Ruiz, J. I. Ronda and N. Garcia, "Evaluation of DWT and DCT for Irregular Mesh-Based Motion Compensation in Predictive Video Coding," In: K. N. Ngan, T. Sikora and M.-T. Sun, Eds., *Visual Communications and Image Processing*, *Proceedings of SPIE* 4067, 2000, pp. 447-456.

[8] B. Song, A. Roy-Chowdhury and E. Tuncel, "A Multi-Terminal Model-Based Video Compression Algorithm," *Proceedings of IEEE International Conference on Image Processing*, Atlanta, 8-11 October 2006, pp. 265-268.

[9] B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereovision," *Proceeding of DARPA Image Understanding Workshop*, 1981, pp. 121-130.

[10] J.-Y. Bouguet, "Pyramidal Implementation of Lucas Kanade Feature Tracker Description of the Algorithm," Intel Corporation, Microprocessor Research Labs, OpenCV Documentation, May 2001.

[11] N. Božinović and J. Konrad, "Mesh-Based Motion Models for Wavelet Video Coding," *Proceedings of IEEE International Conference on Acoustics Speech Signal Processing*, Vol. 3, 17-21 May 2004, pp. 141-144.

[12] J. Shi and C. Tomasi, "Good Features to Track," *IEEE International Conference on Computer Vision and Pattern Recognition*, Seattle, 21-23 June1994, pp. 593-600.

[13] J. R. Shewchuk, "Delaunay Refinement Algorithms for Triangular Mesh Generation," *Computational Geometry*, Vol. 22, No. 1-3, 2002, pp. 21-74.

[14] A. Said and W. A. Pearlman, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 6, No. 3, 1996, pp. 243-250. doi:10.1109/76.499834

[15] S. Li and W. Li, "Shape-Adaptive Discrete Wavelet Transforms for Arbitrarily Shaped Visual Object Coding," *IEEE Transactions on Circuits and Systems for Video Coding*, Vol. 10, No. 5, 2000, pp. 725-743.

[16] G. Minami, Z. Xiong, A. Wang and S. Mehrota, "3-D Wavelet Coding of Video with Arbitrary Regions of Support," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 11, No. 9, 2001, pp. 1063-1068. doi:10.1109/76.946523

[17] K. M. Lam and H. Yan, "Locating Head Boundary by Snakes," *International Symposium on Speech*, *Image Processing and Neural Networks*, Vol. 1, 13-16 April 1994, pp. 17-20.

[18] Y. Wang, S. Cui, and J. E. Fowler, "3D Video Coding Using Redundant-Wavelet Multihypothesis and Motion-Compensated Temporal Filtering," *Proceedings of the IEEE International Conference on Image Processing*, Barcelona, 14-17 September 2003, pp. 755-758.