

Selection of Wavelet Video Codec Parameters to Optimize Coding Time

Andrzej Popławski

Abstract—In the paper, results of experimental research on wavelet video codec are presented. The author has put emphasis on the optimal selection of the coder parameters to optimize the coding time. The author has used six test sequences (Basket, City, Crew, Harbour, Ice and Soccer in 352x288, 30 fps) for the experiments. The sequences used in the research are commonly used to assess the effectiveness of video sequence compression. The measurement of the image quality is based on the PSNR. The results of optimal selection of wavelet video codec parameters due to the coding time are presented. The main goal of the paper is to show the coder parameters that provide the best values of the PSNR for a reduced coding time.

Keywords—Optimization, wavelet video coding, scalability

I. INTRODUCTION

NOWADAYS, video data compression is widely used in a great number of devices and deployments. This technique is applied in simple devices as well as advanced video systems, for example in: cheap video cameras, car video recorders, mobile phones, movie and TV cameras, terrestrial and satellite digital television systems (DVB-T, DVB-S) and the like. Devices which are able to record and play video data are in most cases fitted with a processing unit that ensures processing all incoming data in real time. The cheaper the recording unit, the less efficient the processing unit. This makes it necessary to use less efficient video compression techniques and thereby at a lower numeric cost.

In recent years, we have observed significant progress in the area of image and video compression. New and more efficient compression techniques of video sequences have been proposed. Numerous standards of video compression have appeared (e.g. MPEG-4, H.263, H.264) which use hybrid coding techniques. Together with the development and growth of coding effectiveness, requirements for the computing power of processing units increase. That is why some solutions can not be used in devices with insufficient processing power.

Coders based on the standards mentioned above, often called hybrid coders, use the Discrete Cosine Transform (DCT) [1]. In recent years, many researchers show interest in the Discrete Wavelet Transform (DWT) [2]–[4]. Wavelet-based coders are an attractive alternative to the well-known hybrid coders, owing to their many advantages. The most important ones are:

- fully embedded bitstream that enables to progressively decode with once coded data stream (SNR scalability),

- natural scalability in the spatial domain which allows to obtain correct video sequences at reduced spatial dimensions from one data stream,
- natural scalability in the time domain which allows to obtain correct video sequences at reduced framerate from one data stream,
- ease to get an exact bitrate (thanks to fully embedded bitstream).

The above mentioned advantages of wavelet video coders have special significance in cordless systems (where flow capacity can change in time) and also in the transmission of video sequences via heterogeneous networks [5]. The use of a scalable wavelet coder can facilitate broadcast for a wide range of customers using: a network with different transmission rates, devices with different screen resolutions and different computational power. By changing the values of the wavelet coder parameters, it is possible to influence the coding effectiveness and thereby the limitation of requirements for computational power. In many cases such a solution allows to use cheaper components in video devices.

During encoding video sequences, sending data to the receiver and decoding video data, three types of delays can occur:

- delay of the processing unit that codes or decodes data,
- delay resulting from the necessity to send data to the coder via the transmission channel,
- delay resulting from algorithms used in a coder/decoder which sometimes require waiting for a number of future images.

The paper does not analyze the coding delay resulting from algorithms applied in a coder. This aspect has been analyzed in [6]. The time that is necessary to send data to a coder via a transmission channel has not been taken into account either. The focus is mainly on the processing time of the test sequence by the processing unit.

In the paper, results of experimental research on a wavelet video codec are presented. The author has put emphasis on an optimal selection of the coder parameters to optimize the coding time. For the experiments, six test sequences were taken: Basket, City, Crew, Harbour, Ice and Soccer in format CIF 30Hz (352x288). The sequences selected are commonly used to assess the effectiveness of video sequence compression.

II. WAVELET CODER OF VIDEO SEQUENCES

In order to achieve a good performance of a wavelet video coder, the pictures of the sequence are grouped into groups of

A. Popławski is with the Institute of Computer Engineering and Electronics, University of Zielona Góra, Z. Szafrana 2, 65-516 Zielona Góra, Poland (e-mail: A.Poplawski@iie.uz.zgora.pl).

pictures (GOP) and processed jointly. The number of frames in a GOP is equal to k -th power of 2, where k is the number of levels of a temporal analysis. Such a group is filtered with motion compensation producing one low frequency picture and seven high frequency pictures [7]–[11]. The GOP size has influence on both the coding complexity and the coding effectiveness.

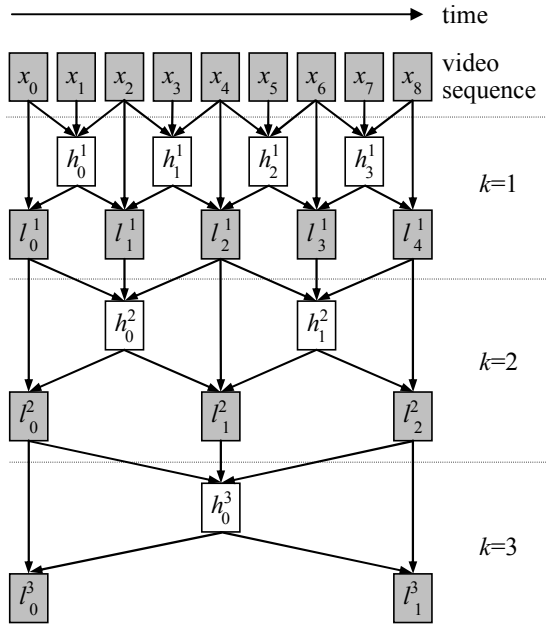


Fig. 1. Three levels of temporal wavelet video analysis based on LeGall 5/3 filters.

Figure 1 presents a temporal filtering scheme for a group of eight pictures. The analysis begins at the first level of temporal decomposition ($k = 1$) and consists in filtering successive threesomes of input pictures $x_{2t}, x_{2t+1}, x_{2t+2}$. As a result, each of the three produces two filtered images: the low frequency component l_t^1 and the high frequency component h_t^1 (Fig. 1). At the next level of the temporal decomposition ($k = 2$), only low frequency components are processed, producing their own low frequency component l_t^2 and the high frequency components h_t^2 . The same scheme is repeated in the following decomposition levels, until only one low frequency and one high frequency component have been obtained. For a group of eight pictures, the final result is: one low frequency component and seven high frequency components.

In practice, a wavelet analysis is performed with the use of the so called lifting structure [12]–[14]. The lifting is based on two steps: prediction – which represents highpass filtering and update – which represents lowpass filtering. The first stage of such analysis is splitting the input signal y into the even and odd samples y_0 and y_1 (Fig. 2). The next step is the actual analysis: the output component y'_1 is the result of the operation of the predictor P (which represents high frequency filtering), and the y'_0 component is produced by the update operator U (representing low frequency filtering). The operation can be presented as in equation (1):

$$\begin{aligned} y'_1 &= y_1 - P(y_0), \\ y'_0 &= y_0 + U(y'_1). \end{aligned} \quad (1)$$

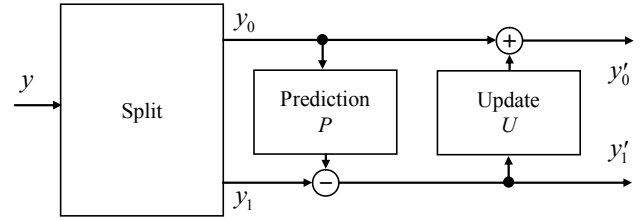


Fig. 2. Lifting filtering scheme in wavelet analysis.

A synthesis can be done by using the same lifting steps but in a reverse order and with opposite signs (change + into – in update step and change – into + in prediction step). The first step is the actual synthesis: the component y_0 is the result of the operation of the update operator U , and the y_1 component is produced by the prediction operator P (Fig. 3). Finally, the last operation merges the even and odd samples y_0 and y_1 . This procedure produces the output signal y . The operation can be presented as in equation (2):

$$\begin{aligned} y_0 &= y'_0 - U(y'_1), \\ y_1 &= y'_1 + P(y_0). \end{aligned} \quad (2)$$

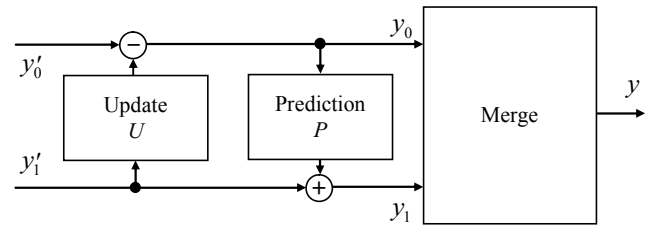


Fig. 3. Lifting filtering scheme in wavelet synthesis.

Lifting implementations of the temporal filters in wavelet video coders include motion-compensated prediction and update operations. Motion compensated temporal filtering performed with the use of a lifting scheme [15] and based on the LeGall 5/3 filters [16] is described by equation (3) – the analysis, and (4) – the synthesis:

$$\begin{aligned} h_t &= x_{2t+1} - 1/2 \left(MC(x_{2t}, mv_{2t+1}^+) + \right. \\ &\quad \left. + MC(x_{2t+2}, mv_{2t+1}^-) \right), \end{aligned} \quad (3)$$

$$\begin{aligned} l_t &= x_{2t} + 1/4 \left(MC^{-1}(h_{t-1}, mv_{2t-1}^-) + \right. \\ &\quad \left. + MC^{-1}(h_t, mv_{2t+1}^+) \right), \end{aligned}$$

$$\begin{aligned} x_{2t} &= l_t - 1/4 \left(MC^{-1}(h_{t-1}, mv_{2t-1}^-) + \right. \\ &\quad \left. + MC^{-1}(h_t, mv_{2t+1}^+) \right), \end{aligned} \quad (4)$$

$$\begin{aligned} x_{2t+1} &= h_t + 1/2 \left(MC(x_{2t}, mv_{2t+1}^+) + \right. \\ &\quad \left. + MC(x_{2t+2}, mv_{2t+1}^-) \right), \end{aligned}$$

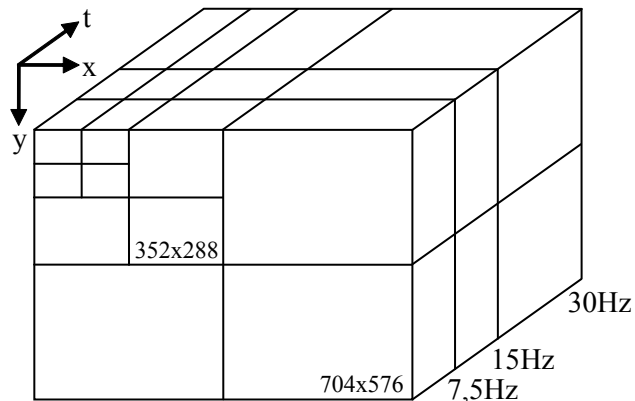


Fig. 4. An output bitstream cube representation of a wavelet video coder.

where:

- x_t – picture of input sequence in time t ,
- h_t – sample of high-band signal in time t ,
- l_t – sample of low-band signal in time t ,
- mv_{2t+1}^+ – set of motion vectors between pictures x_{2t+1} and x_{2t} ,
- mv_{2t+1}^- – set of motion vectors between pictures x_{2t+1} i x_{2t+2} ,
- (m, n) – spatial coordinates of the picture,
- $MC(x_t, mv_t)$ – motion compensation operator, defined as $x_t(m, n - mv_t(m, n))$.

Components produced in the process of motion compensated temporal filtering are in the next step analyzed in a two-dimensional spatial domain [4]. The Daubechies 9/7 wavelet filters are used in this step of analysis [17], [18]. As a result, we obtain a three-dimensional wavelet analysis of the input video sequence (one dimension in time and two dimensions in 2D space). The resulting three-dimensional components, taking the form of an output bitstream, can be presented in a visual form as a cube (Fig. 4). Taking suitable parts of the cube, it is possible to obtain spatial and temporal scalability with no need to perform the coding procedure again.

Motion estimation is the most time-consuming stage of wavelet video coding. It consumes about 80% of the whole time needed for compression. A fundamental part of the prediction with motion compensation is picture division into smaller components. A picture of a video sequence based on a predetermined algorithm is divided into so-called blocks. Then for each of the blocks, a motion vector is calculated. The block, after moving according to the motion vector, should be as similar as possible to a reference picture. The mean absolute error (MAE) is a quantity used to measure how similar the referenced part of a picture and a block are to each other.

The size of a block can vary from 64x64 points to 4x4 points. Spatial filtering is performed in the first step of motion estimation. As a result of this filtering, the picture size is decreased fourfold (Fig. 5). At the beginning, the motion estimation is performed for pictures with reduced spatial dimensions for a block size of 64x64 points. In the next step, the resolution of the pictures is doubled and each block is divided into four equal subblocks. Now, motion vectors are calculated for the subblocks taking into account

the motion vector calculation in the previous step. Then, using equation (5)

$$D(m, n) = \sum_{m, n} MAE(B(m, n) - A(m - d_m, n - d_n)) \quad (5)$$

where:

- (m, n) – picture spatial coordinates,
- (d_m, d_n) – motion vector for coordinates of the picture at point (m, n) ,
- $A(m, n)$ – one point of previous picture at point (m, n) ,
- $B(m, n)$ – one point of current picture at point (m, n) ,

a comparison is made of the motion estimation error of a block with the mean of the error in the subblocks. If the mean of this error in a subblock is smaller than that in a block, then the division at the time block is kept. After determining the motion vectors in the picture with full spatial dimensions, the motion estimation calculation ends. In the end, we have obtained a motion vector tree (Fig. 5) [10].

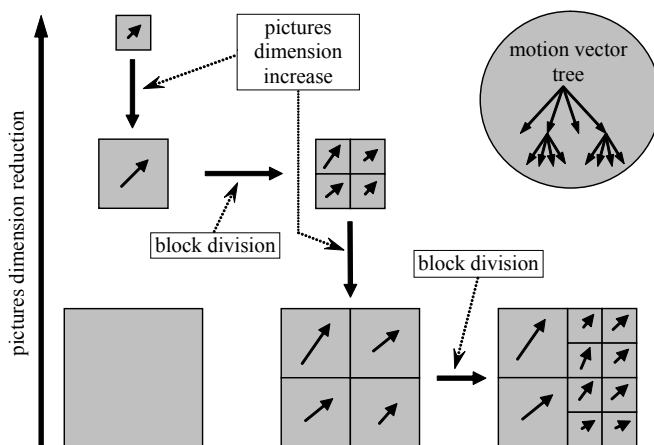


Fig. 5. An illustration of motion vector calculation.

Motion vectors are calculated for each type h component. The more levels of decomposition in the time domain, the more effort required to perform the motion estimation procedure.

In order to carry out the motion estimation with an accuracy higher than one spatial sampling interval, at first this procedure is done with one pixel accuracy. Then, by interpolating, the size of the picture is doubled. Next, a new motion vector is searched in the neighbourhood of the previously found vector. This step can be repeated in order to achieve the required accuracy of the calculation of motion vectors. Usually, the higher the accuracy of the motion vector calculation, the higher the coding effectiveness. More accurate motion vector calculation implies the necessity of calculating a higher number of motion vectors, producing more data to process and increasing the numerical cost of the operation.

A. Parameters of Wavelet Video Coder

Among many parameters that affect the coding process, four of them are crucial from the point of view of the complexity and coding effectiveness. Depending on the values

of the parameters, the computational complexity increases or decreases, and thereby the time required to encode a video sequence does too. The importance of these parameters is characterized and described below.

tPL – defines the level of subband decomposition in the time domain. The value of the parameter is usually from 1 to 5. A higher value of the parameter usually means better coding efficiency, but it also leads to the growth of coding complexity.

MV – determines the accuracy of motion vectors calculation. Possible values of the sampling interval are: 1, 1/2, 1/4, 1/8. The smaller the sampling interval, the greater the coding accuracy of motion vectors. The more accurate the motion vector calculation, the higher the coding complexity.

MB – defines the level of block division into subblocks. The standard size of an image block is 64x64. A block can assume the following values: 64x64, 32x32, 16x16, 8x8, 4x4. For example, if a parameter value is equal to 16x16, the subblock size is not less than 16x16 (but may be larger). For each subblock, a motion vector is calculated. A greater number of smaller subblocks contributes to the growth of the coding complexity. Searching motion vectors is a time consuming operation.

SR – determines the search range (in pixels) of the current block in the reference frame for the first level of wavelet decomposition in time. The value of the parameter is doubled on each succeeding level of decomposition in time. The larger the search range, the higher the coding complexity. But on the other hand, this may contribute to a more accurate motion vector calculation for a given block. The value of this parameter is 8 or 16 points.

III. EXPERIMENTAL RESEARCH

The main aim of the experimental research was to determine such values of the coder parameters which permit to obtain significant reduction of the coding time with an as small as possible drop in the coding effectiveness. Here, the coding time is understood as the time which is needed to process data by a processing unit, e.g. by a processor. Solutions presented in the literature, focus primarily on reducing the coding delay introduced by the filters of analysis and synthesis in the time domain [6], [19], [20]. They rely on such a modification of the filter system that provides a lower coding delay. Such solutions do not substantially affect the coding time associated with the processing of data by a processing unit.

In the paper, the symbol T means the coding time for which the value of PSNR (6) was the biggest. Next, the most advantageous coding parameters for the coding time equal to 50% and 25% of the time T (1/2T and 1/4T) were determined. For the experiments, six test sequences were used (Basket, City, Crew, Harbour, Ice, Soccer) in the format CIF (352x288) and a frame frequency equal to 30Hz. The selected sequences are commonly used to assess the effectiveness of video sequence compression. They are characterized by variable dynamic both in the foreground and in the background.

The test sequences used were represented in the YUV format, comprising the luminance component Y and two chrominance components U and V, with the sampling scheme 4:2:0 [21]. To measure the quality, the PSNR (Peak Signal-to-Noise Ratio) defined as in equation (6) was used:

$$PSNR = 10 \log \frac{255^2 N^2}{\sum_i e_i^2} \quad (6)$$

where:

- 255 – dynamic range of the signal,
- N – number of pixels in the picture,
- e_i – difference between the i -th pixel of the original and processed images.

Because subjective quality measures [22] take much time and effort, they were not carried out. Since the codecs compared use similar compression methods, the type of distortion introduced is approximate and according to [23] the PSNR is a satisfactory measure of estimating the results. The measurements were performed for the luminance component by calculating the mean value of the PSNR for all the pictures in a given sequence. This method is widely applied by many researchers.

TABLE I
ENCODER PARAMETER VALUES USED IN EXPERIMENTS

Parametr	Parameter values used in experiments
tPL	2, 3, 4, 5
MV	1/2, 1/4, 1/8
MB	16x16, 8x8, 4x4
SR	8, 16

The experiments consisted in encoding and then decoding test video sequences with three various assumed bitrates: 512kbps, 768kbps and 1024kbps. A decoded sequence was compared to the original sequence, and the quality of the decoded sequence was measured. In Table I, a list of wavelet coder parameters which were employed in the experiments is presented. All the possible combinations of the parameters mentioned in Tab. I were examined (a total of 72 combinations). During the examination, the encoding time was measured with an accuracy of one second. Since the encoding time is a few dozen minutes, such an accuracy is fully sufficient. Experiments were carried out on a personal computer with an Intel Core i7 2700K processor. For each particular sequence, 72 operations of encoding and 216 operations of decoding were performed (a total of 432 operations of encoding and 1296 operations of decoding were performed). As a result, all the possible coding variants for parameter values depicted in Tab. I were checked.

IV. EXPERIMENTAL RESULTS

In the first step of the analysis of the experimental results, the impact of different coding parameters on the compression time was taken into account. At this moment, the coding effectiveness was not analyzed. For all the examined test



Fig. 6. A frame of the City video sequence a) for the longest coding time, PSNR=38.06dB, b) for the shortest coding time, PSNR=32.06dB; bitrate equal to 512kbps.

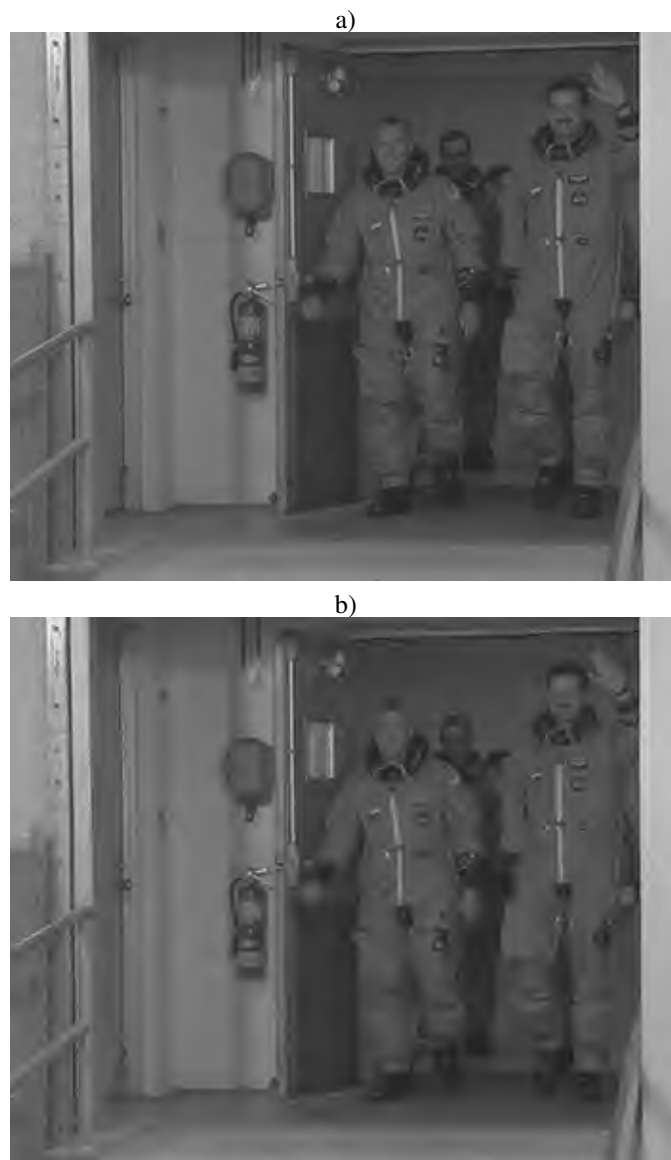


Fig. 7. A frame of the Crew video sequence a) for the longest coding time, PSNR=34.33dB, b) for the shortest coding time, PSNR=33.11dB; bitrate equal to 512kbps.

video sequences, the shortest coding time was obtained for the following parameter values:

- tPL=2,
- MV=1/2,
- MB=16x16,
- SR=8.

For this parameter compilation, the lowest PSNR values were obtained. For all the examined test video sequences, the longest coding time was obtained for the following parameter values:

- tPL=5,
- MV=1/8,
- MB=4x4,
- SR=16.

For this parameter compilation, the highest PSNR values were obtained. The tPL parameter has the biggest impact on

the increase in the encoding time. Changing the value of the tPL parameter from 2 to 5 contributes to the increase in the encoding time by an average of between 213% and 427%, depending on the sequence and values of the remaining parameters. These values for each sequence are as follows:

- Basket – 259%,
- City – 221%,
- Crew – 427%,
- Harbour – 272%,
- Ice – 213%,
- Soccer – 229%.

In Figure 6, a frame of the City sequence for the longest and the shortest coding time is shown. In Figure 7, a frame of the Crew sequence for the longest and the shortest coding time is shown. We can see the deterioration in the image quality with the reduced coding time. There are two sequences for which

TABLE II
AVERAGE CODING TIMES DEPENDING ON TPL PARAMETER VALUE

Sequence name	Average coding time for tPL parameter [h:m:s]			
	2	3	4	5
Basket	0:23:53	0:53:20	1:01:48	1:25:38
City	0:18:53	0:35:53	0:51:14	1:00:38
Crew	0:10:51	0:23:16	0:42:35	0:57:12
Harbour	0:20:11	0:42:21	1:03:07	1:14:59
Ice	0:12:58	0:24:11	0:34:26	0:40:33
Soccer	0:19:52	0:38:20	0:54:15	1:05:21

fluctuations of the PSNR values are respectively the highest and the lowest.

The particular values of the average coding time for other values of the tPL parameter are shown in Tab. II. Changing the value of the SR parameter from 8 to 16 contributes to an increase in the encoding time by an average of between 64% and 106%, depending on the examined sequence and on the values of the remaining parameters. These values for each sequence are as follows:

- Basket – 106%,
- City – 78%,
- Crew – 98%,
- Harbour – 85%,
- Ice – 64%,
- Soccer – 89%.

TABLE III
AVERAGE CODING TIMES DEPENDING ON SR PARAMETER VALUE

Sequence name	Average coding time for SR parameter [h:m:s]	
	8	16
Basket	0:36:42	1:15:38
City	0:29:59	0:53:20
Crew	0:22:29	0:44:28
Harbour	0:25:13	1:05:05
Ice	0:21:12	0:34:51
Soccer	0:30:44	0:58:10

TABLE IV
AVERAGE CODING TIMES DEPENDING ON MV PARAMETER VALUE

Sequence name	Average coding time for MV parameter [h:m:s]		
	1/2	1/4	1/8
Basket	0:53:58	0:55:30	0:59:01
City	0:40:00	0:41:26	0:43:32
Crew	0:31:28	0:32:41	0:36:17
Harbour	0:49:07	0:49:46	0:51:36
Ice	0:26:16	0:27:17	0:30:33
Soccer	0:41:52	0:43:43	0:47:45

The particular values of the average coding time for other values of the SR parameter are shown in Tab. III. Changing the values of the remaining coding parameters does not have a big influence on the coding time. For the MV parameter, changing

TABLE V
AVERAGE CODING TIMES DEPENDING ON MB PARAMETER VALUE

Sequence name	Average coding time for MB parameter [h:m:s]		
	3	3	5
Basket	0:51:22	0:56:27	1:00:41
City	0:39:08	0:41:57	0:43:53
Crew	0:31:54	0:33:03	0:35:28
Harbour	0:46:23	0:50:57	0:53:08
Ice	0:26:39	0:27:56	0:29:31
Soccer	0:41:35	0:44:18	0:47:29

TABLE VI
OPTIMAL VALUES OF CODER PARAMETERS FOR CODING TIME EQUAL TO 1/2 T

Sequence name	tPL	MV	MB	SR
Basket	5	1/4	16x16	8
City	4	1/4	8x8	8
Crew	5	1/4	16x16	8
Harbour	4	1/4	8x8	8
Ice	4	1/4	8x8	8
Soccer	4	1/4	8x8	8

its value from 1/2 to 1/8 causes an increase in the coding time by an average of 11%. For the MB parameter, changing its value from 16x16 to 4x4 causes an increase in the coding time by an average of 13%. Particular values of the mean coding time for the MV parameter, obtained in the experiments are shown in Tab. IV. Particular values of the mean coding time for the MB parameter, obtained in the experiments are shown in Tab. V.

The next step was a detailed analysis of the obtained results and selecting such parameters of the coder that provide the smallest drop in the coding effectiveness for an assumed coding time. The optimal values of the parameters (for the sake of PSNR values), for a reduced coding time equal to 1/2T, obtained on the basis of the experiments are shown in Tab. VI.

As follows from the results shown in Tab. VI, in most cases for parameters equal to: tPL=4, MV=1/4, MB=8x8 and SR=8, coding times are reduced by 50% compared to the longest coding times. There are simultaneously such values of the coding parameters for which the biggest values of the PSNR (for coding time equal to 1/2T) have been recorded. In

TABLE VII
PSNR VALUES FOR CODING TIME EQUAL TO T AND 1/2T;
BITRATE 512KBPS

Sequence name	PSNR [dB] time=T	PSNR [dB] time=1/2T	PSNR difference [dB]
Basket	26.64	26.54	0.10
City	38.06	37.22	0.84
Crew	34.33	34.29	0.04
Harbour	31.09	30.97	0.12
Ice	39.20	39.11	0.09
Soccer	34.25	33.81	0.44

TABLE VIII
 PSNR VALUES FOR CODING TIME EQUAL TO T AND 1/2T;
 BITRATE 768KBPS

Sequence name	PSNR [dB] time=T	PSNR [dB] time=1/2T	PSNR difference [dB]
Basket	28.93	28.48	0.45
City	39.94	39.44	0.50
Crew	35.89	35.83	0.06
Harbour	32.69	32.67	0.02
Ice	41.68	41.62	0.06
Soccer	36.33	35.93	0.40

TABLE IX
 PSNR VALUES FOR CODING TIME EQUAL TO T AND 1/2T;
 BITRATE 1024KBPS

Sequence name	PSNR [dB] time=T	PSNR [dB] time=1/2T	PSNR difference [dB]
Basket	30.50	30.18	0.32
City	41.26	40.78	0.48
Crew	37.16	37.09	0.07
Harbour	33.89	33.85	0.04
Ice	43.64	43.61	0.03
Soccer	38.04	37.66	0.38

Table VII, the PSNR values are shown: for all the examined video sequences, for the coding time equal to T and 1/2T, for a bitrate equal to 512kbps and for the parameters as in Tab. VI. In each considered case, a decrease in the PSNR values was observed. The biggest drop in the PSNR value was recorded for the City sequence – 0,84dB, whereas the smallest drop in the PSNR value – for the Ice sequence – 0.09dB, by an average of 0.27dB. However, with the exception of the City sequence, the drop in the PSNR values is relatively small when you consider the fact that the coding time is reduced by 50%. For a higher transmission speed: 768kbps (Tab. VIII) and 1024kbps (Tab. IX), a similar trend was observed, though the drop in the PSNR was slightly smaller, by an average of 0.25dB at 768kbps, and by an average of 0.22dB at 1024kbps. It is worth to notice that for higher bitrate values, the drop in the PSNR values is smaller.

In the case of a fourfold reduction of the coding time, equal to 1/4T, similar trends were observed as in the time equal to 1/2T. The optimal values of the parameters (for the sake of PSNR values), for a reduced coding time equal to 1/4T, obtained on the basis of the experiments are shown in Tab. X. There are simultaneously such values of coding parameters for which the biggest values of the PSNR (for coding time equal to 1/4T) have been recorded. As follows from the results shown in Tab. X, in most cases for the parameters equal to: tPL=3, MV=1/4, MB=8x8, SR=8, coding times are reduced by 75% compared with the longest coding times. In Table XI, the PSNR values are shown: for all the examined video sequences, for the coding time equal to T and 1/4T, for a bitrate equal to 512kbps and for the parameters as in Tab. X.

Similarly to 1/2T, in each considered case, a decrease in the PSNR values was observed. The biggest drop in the PSNR value was recorded for the City sequence – 3.04dB, the small-

TABLE X
 OPTIMAL VALUES OF CODER PARAMETERS FOR CODING TIME
 EQUAL TO 1/4 T

Sequence name	tPL	MV	MB	SR
Basket	3	1/4	8x8	8
City	3	1/4	16x16	8
Crew	3	1/8	8x8	8
Harbour	3	1/4	8x8	8
Ice	2	1/8	8x8	8
Soccer	3	1/2	8x8	8

TABLE XI
 PSNR VALUES FOR CODING TIME EQUAL TO T AND 1/4T;
 BITRATE 512KBPS

Sequence name	PSNR [dB] time=T	PSNR [dB] time=1/4T	PSNR difference [dB]
Basket	26.64	26.18	0.46
City	38.06	35.02	3.04
Crew	34.33	34.13	0.20
Harbour	31.09	30.26	0.83
Ice	39.20	38.42	0.78
Soccer	34.25	33.45	0.80

TABLE XII
 PSNR VALUES FOR CODING TIME EQUAL TO T AND 1/4T;
 BITRATE 768KBPS

Sequence name	PSNR [dB] time=T	PSNR [dB] time=1/4T	PSNR difference [dB]
Basket	28.93	28.51	0.42
City	39.94	37.53	2.41
Crew	35.89	35.81	0.08
Harbour	32.69	32.16	0.53
Ice	41.68	41.21	0.47
Soccer	36.33	35.49	0.84

TABLE XIII
 PSNR VALUES FOR CODING TIME EQUAL TO T AND 1/4T;
 BITRATE 1024KBPS

Sequence name	PSNR [dB] time=T	PSNR [dB] time=1/4T	PSNR difference [dB]
Basket	30.50	30.23	0.27
City	41.26	39.41	1.85
Crew	37.16	37.11	0.05
Harbour	33.89	33.57	0.32
Ice	43.64	43.23	0.41
Soccer	38.04	37.12	0.92

est drop in the PSNR value for the Crew sequence – 0.20dB, by an average of 1.02dB. However, with the exception of the City sequence, the drop in the PSNR value is relatively small and does not exceed 1dB. So the drop in the PSNR value can be regarded as relatively small and acceptable when you consider the fact that the encoding time was shortened by 75%.

For a higher transmission speed: 768kbps (Tab. XII) and 1024kbps (Tab. XIII) a similar trend was observed, though the drop in the PSNR was slightly smaller, by an average of 0.79dB at 768kbps and by an average of 0.64dB at 1024kbps.

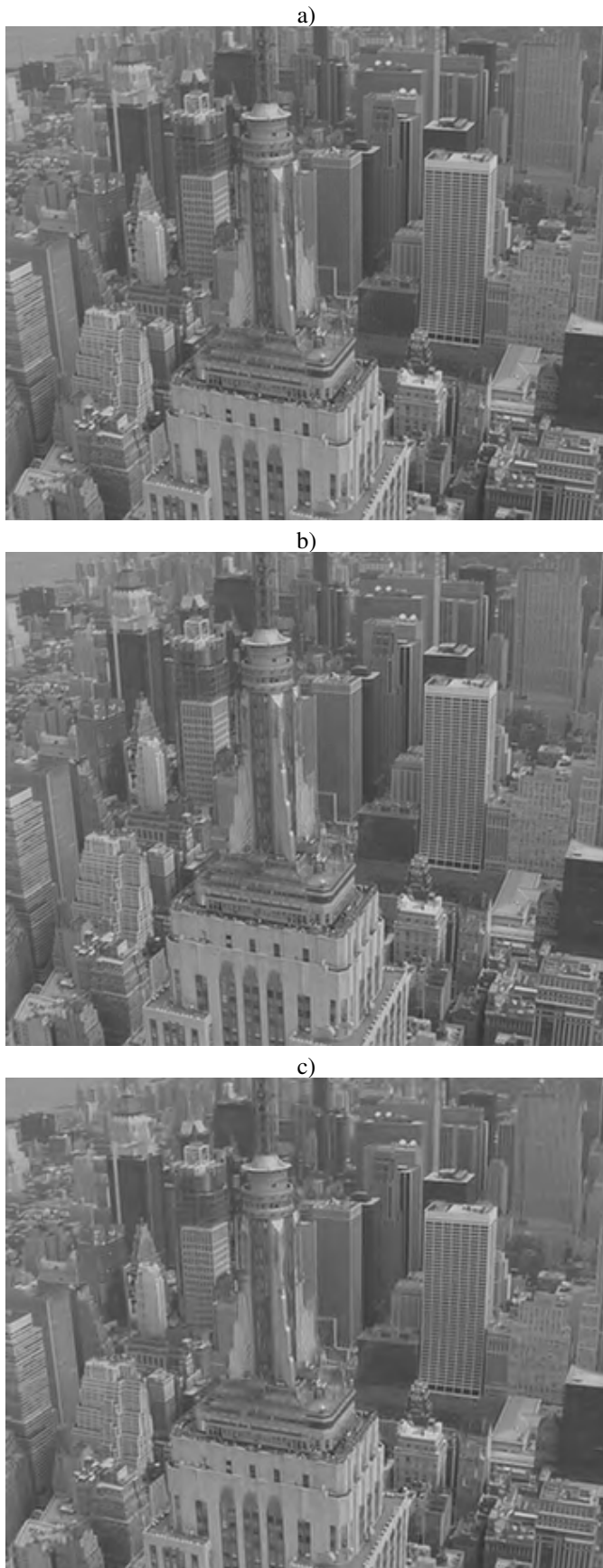


Fig. 8. Frames of the City video sequence: a) for the coding time equal to T, PSNR=38.06dB, b) for the coding time equal to 1/2T, PSNR=37.22dB, c) for the coding time equal to 1/4T, PSNR=35.02dB; bitrate equal to 512kbps.

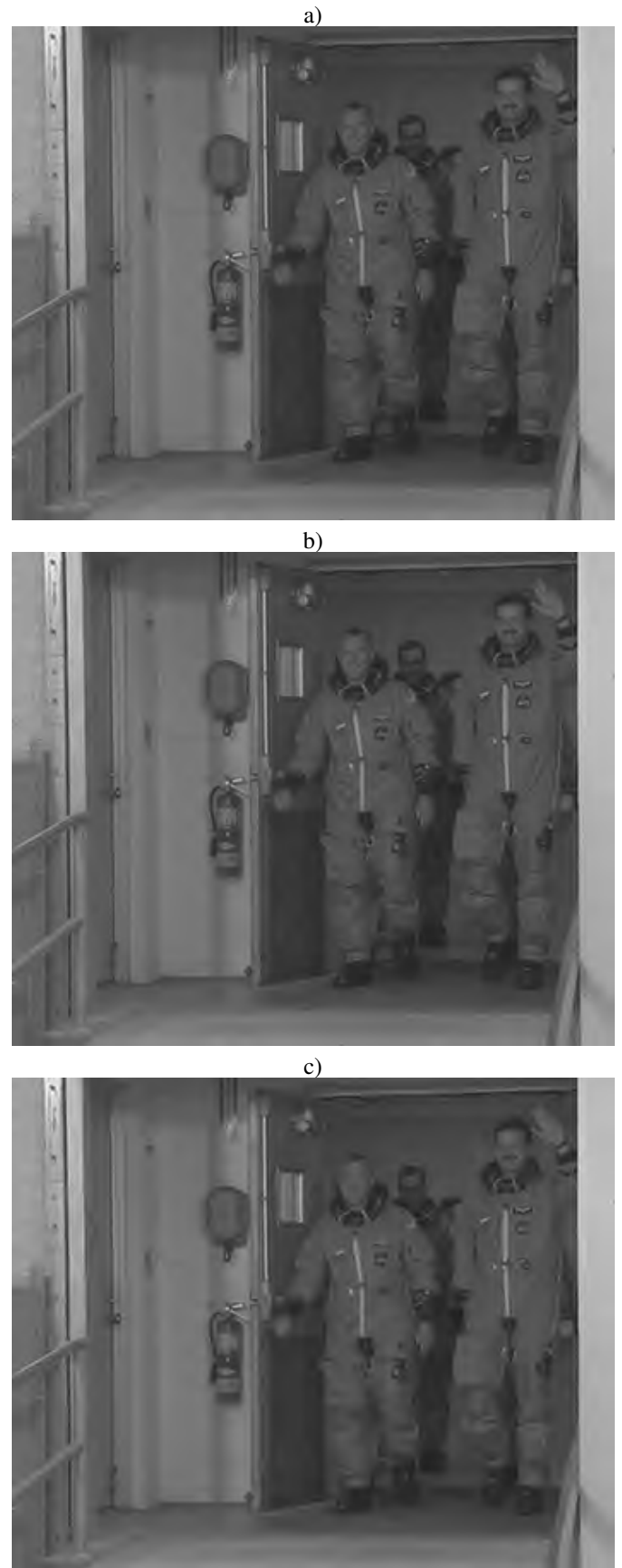


Fig. 9. Frames of the Crew video sequence: a) for the coding time equal to T, PSNR=34.33dB, b) for the coding time equal to 1/2T, PSNR=34.29dB, c) for the coding time equal to 1/4T, PSNR=34.13dB; bitrate equal to 512kbps.

It is worth to notice that for higher values of bitrates, the drop in the PSNR values is smaller. To sum up, for the coding time equal to $1/4T$, the drop in the PSNR is much larger than for the time $1/2T$ and in some applications it may not be acceptable.

In Figs. 8 and 9, a view of the City and Crew sequences are presented for times: T , $1/2T$ and $1/4T$, for a bitrate equal to 512kbps and an optimal selection of the parameters of the wavelet video coder. These are the two sequences where the biggest and the smallest drop, respectively, can be observed of coding effectiveness along with the limitation of the coding time.

V. CONCLUSION

In the paper, results of experimental research on a wavelet video codec are presented. The author has put emphasis on an optimal selection of the coder parameters to optimize the coding time. A set of coder parameters which guarantees the biggest possible PSNR values at a reduced coding time has been shown. The results prove that it is possible to reduce the coding time with a relative small drop in the coding effectiveness which is measured by the PSNR.

The research results facilitate the application of wavelet video coders in a recording unit fitted with less efficient processing units. Tests were carried out using a representative group of video test sequences. The selected sequences are commonly used to assess the effectiveness of video sequence compression. The higher the observed drop in the compression effectiveness, the lower the bitrate of the video test sequence. It also strongly depends on the content of the sequence processed by the coder.

REFERENCES

- [1] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *IEEE Trans. on Computer*, pp. 90–93, 1974.
- [2] P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice Hall, 1993.
- [3] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [4] J. W. Woods, *Subband Image Coding*. Kluwer Academic Publisher, 1991.
- [5] D. Wu, Y. Hou, and Y. Zhang, "Scalable video coding and transport over broad-band wireless network," *Proc. of the IEEE*, vol. 89, pp. 6–20, 2001.
- [6] A. Poplawski and M. Domanski, "Optimization of low-delay wavelet video codecs," *Proc. of Picture Coding Symposium*, Nov. 2007.
- [7] J.-R. Ohm, "Temporal domain subband video coding with motion compensation," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 3, pp. 229–232, 1992.
- [8] —, "Three-dimensional motion-compensated subband coding," *Proceedings International Symposium on Video Communications and Fiber Optic Services*, vol. 1977, pp. 188–197, 1993.
- [9] —, "Three-dimensional subband coding with motion compensation," *IEEE Transactions Image Processing*, vol. 3, pp. 559–571, 1994.
- [10] S. J. Choi and J. W. Woods, "Motion-compensated 3-D subband coding of video," *IEEE Transactions on Image Processing*, vol. 8, pp. 155–167, 1999.
- [11] P. Chen, "Fully scalable subband/wavelet coding," Ph.D. dissertation, Rensselaer Polytechnic Institute, 2003.
- [12] I. Daubechies and W. Sweldens, *Factoring wavelet transforms into lifting steps*. Bell Laboratories, Lucent Technologies, 1996.
- [13] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186–200, 1996.
- [14] R. L. Claypoole, G. M. Davis, W. Sweldens, and G. Baraniuk, "Non-linear wavelet transforms for image coding via lifting," *IEEE Trans. Image Process.*, vol. 12, no. 1449–1459, 2003.
- [15] B. Pesquet-Popescu and V. Botreau, "Three-dimensional lifting schemes for motion compensated video compression," *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal*, 2001.
- [16] D. LeGall and A. Tabatabai, "Subband coding of digital images using symmetric short kernel filters and arithmetic coding techniques," *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 761–765, 1988.
- [17] A. Antonini, M. Barlaud, and I. Daubechies, "Image coding using wavelet transform," *IEEE Transaction Image Processing*, vol. 1, pp. 205–220, 1992.
- [18] A. Cohen, I. Daubechies, and J. C. Feauveau, "Bi-orthogonal bases for compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. XLV, pp. 485–560, 1992.
- [19] G. Pau, B. Pesquet-Popescu, M. van der Schaar, and J. Viéron, "Delay-performance trade offs in motion-compensated scalable subband video compression," doc. m11084, Redmond MPEG meeting, July 2004.
- [20] G. Pau, J. Viéron, and B. Pesquet-Popescu, "Video coding with Flexible MCTF Structures for Low End-to-End Delay," *ICIP*, vol. 3, pp. 241–244, 2005.
- [21] ITU-R, *ITU-R Rec. BT.470-3 – Television systems*, 1994.
- [22] —, *ITU-R Rec. BT.500-6 – Methodology for the subjective assessment of the quality of television pictures*, 1994.
- [23] —, *ITU-R Rec. BT.813 – Methods for objective picture quality assessment in relation to impairments from digital coding of television signal*, 1992.