

Very Low Bit-Rate Video Coding by Combining H.264/AVC Standard and 2-D Discrete Wavelet Transform

Ali Aghagolzadeh^{1,2}, Saeed Meshgini¹, Mehdi Nooshyar¹, Mehdi Aghagolzadeh¹

¹Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran

²Iranian Telecommunication Research Center (ITRC), Tehran, Iran

E-mail: aghagol@tabrizu.ac.ir, saeed_meshgini@tabrizu.ac.ir, nooshyar@tabrizu.ac.ir
Received October 25, 2009; revised November 11, 2009; accepted February 16, 2010

Abstract

In this paper, we propose a new method for very low bit-rate video coding that combines H.264/AVC standard and two-dimensional discrete wavelet transform. In this method, first a two dimensional wavelet transform is applied on each video frame independently to extract the low frequency components for each frame and then the low frequency parts of all frames are coded using H.264/AVC codec. On the other hand, the high frequency parts of the video frames are coded by Run Length Coding algorithm, after applying a threshold to neglect the low value coefficients. Experiments show that our proposed method can achieve better rate-distortion performance at very low bit-rate applications below 16 kbits/s compared to applying H.264/AVC standard directly to all frames. Applications of our proposed video coding technique include video telephony, video-conferencing, transmitting or receiving video over half-rate traffic channels of GSM networks.

Keywords: Video Coding, H.264/AVC Standard, Run Length Coding, Two-Dimensional Wavelet Transform

1. Introduction

The demands for video transmission and delivery over both high and low bandwidth channels have been accelerated. The high bandwidth applications include digital video by satellite (DVS) and high-definition television (HDTV). The low bandwidth applications are dominated by transmission over the Internet, where the majority of modems work at speeds below 56 kbits/s [1].

On the other hand, representing video material in a digital form requires a long number of bits. The volume of data generated by digitising a video signal is too large for the most transmission systems. This means that compression is essential for the most digital video applications. An efficient and well-designed video compression system gives very significant performance advantages for visual communication at both low and high transmission bandwidths. At low bandwidths, compression enables applications that would not otherwise be possible, such as basic-quality video telephony over a standard telephone connection. At high bandwidths, compression can support a much higher visual quality. Video compression and video codecs will therefore remain a vital part of the emerging multimedia applications for the foreseeable future, allowing designers to make the most

efficient use of the available transmission capacity. The development of video coding technology since 1980 has been bounded up with a series of international standards for video compression. Each of these standards supports a particular application of video coding (or a set of applications), such as videoconferencing and digital television [2].

H.264/AVC is the newest video coding standard of the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group. The goals of this standardization efforts were enhanced compression efficiency, network-friendly video representation for both interactive (video telephony) and non-interactive (broadcast, streaming, storage and video on demand) applications [3]. H.264/AVC has achieved a significant improvement in rate-distortion efficiency relative to the previous standards [4]. However, H.264/AVC standard, like the previous video coding standards, results in a number of unacceptable artifacts such as blockiness when operated at very low bit rates. Hence, there is a need for new techniques to improve the coding efficiency and produce acceptable quality of video at very low bit-rate applications.

In this paper, a new video compression method for very low bit-rate coding is proposed. The main goal of

this paper is enhancing the compression efficiency (rate-distortion performance) at very low bit-rate applications (such as video-conferencing and video telephony). This has been achieved by combining H.264/AVC standard and two-dimensional discrete wavelet transform.

Experiments show that H.264/AVC standard, like the other video coding standards, has a good capability in coding of the low frequency components (the general structure) in contents of video frames, but it has difficulties in encoding the details of objects in video streams, like boundaries and edges. Since the techniques employed in this standard use only the statistical dependencies in the video signal at a block level and do not consider the semantic content of the video, at very low bit rates (high quantization factors) artifacts are introduced at the block boundaries. Usually these block boundaries do not correspond to physical boundaries of the moving objects and hence, visually annoying artifacts are introduced [5]. This problem is emphasized when the objects in video frame are dislocated rapidly; *i.e.* when a fast motion in a video stream occurs. Depending on the number of quantization levels used in the coding procedure, some details of an object are eliminated. The more the number of quantization levels is decreased, the more the details are vanished. High and suddenly motions in a video stream can also lead into loss of some important information through a limited capacity channel. The supporting idea of this paper is to combat these problems by extracting the details from a video sequence and then coding them by another scheme instead of H.264/AVC standard.

This paper is organized as follows. At first in Section 2, we give some analytic discussion about wavelet transform. The architecture of the proposed video coding system is then presented in Section 3. In Section 4, comparisons are given between the experimental results obtained by the proposed method and the original H.264 codec. The possible advantages of our proposed method in different applications are discussed in this section. Conclusions are given in Section 5.

2. Wavelet Transform

Although the Fourier transform has been the mainstay of transform-based image and video processing since the late 1950s, a more recent transformation, called the wavelet transform, is now making it even easier to compress, transmit, and analyze many images and videos. Unlike the Fourier transform, whose basis functions are sinusoids, wavelet transforms are based on small waves, called wavelets, of varying frequency and limited duration.

The goal of the modern wavelet research is to create a set of basis functions (or general expansion functions) and transforms that will give an informative, efficient,

and useful description of a function or signal. Another central idea is that of multiresolution analysis where the decomposition of a signal is done in terms of the different resolutions of details.

Both the mathematics and the practical interpretations of the wavelet transform seem to be best served by using the concept of resolution to define the effects of changing scales. To do this, we will start with a scaling function $\varphi(x)$ rather than directly with the wavelet $\psi(x)$. After the scaling function is defined from the concept of resolution, the wavelet functions will be derived from it. Good reviews of the wavelet transform are given in [6] and [7]. In following, a short review and mathematical interpretations of the wavelet transform are given [6] and [7].

We define a set of scaling functions in terms of integer translates of the basic scaling function by

$$\varphi_k(x) = \varphi(x - k) \quad k \in \mathbb{Z} \quad \varphi \in L^2(\mathbb{R}). \quad (1)$$

The subspace of $L^2(\mathbb{R})$ spanned by these functions is defined as

$$V_0 = \text{Span}_k \{\varphi_k(x)\} \quad (2)$$

for all integers k , $k \in \mathbb{Z}$. This means that

$$f(x) = \sum_k a_k \varphi_k(x) \quad \text{for any } f(x) \in V_0. \quad (3)$$

One can generally increase the size of the subspace spanned by changing the spatial scale of the scaling functions. A two-dimensional family of functions is generated from the basic scaling function by scaling and translation by

$$\varphi_{j,k}(x) = 2^{j/2} \varphi(2^j x - k) \quad (4)$$

whose span over k is

$$V_j = \text{Span}_k \{\varphi_k(2^j x)\} = \text{Span}_k \{\varphi_{j,k}(x)\} \quad (5)$$

for all integers $k \in \mathbb{Z}$. This means that if $f(x) \in V_j$, then it can be expressed as

$$f(x) = \sum_k a_k \varphi(2^j x + k). \quad (6)$$

For $j > 0$, the span can be larger since $\varphi_{j,k}(x)$ becomes narrower and is translated into smaller steps. It, therefore, can represent finer details. For $j < 0$, $\varphi_{j,k}(x)$ is wider and is translated into larger steps. So these wider scaling functions can represent only coarse information, and the size of the space they span is smaller. In order to follow our intuitive ideas of scale or resolution, we formulate the basic requirements of multiresolution analysis (MRA) by requiring nested spanned spaces as

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2 \quad (7)$$

or

$$V_j \subset V_{j+1} \quad \text{for all } j \in \mathbb{Z} \quad (8)$$

with

$$V_{-\infty} = \{ \} , \quad V_{\infty} = L^2. \quad (9)$$

The space that contains high resolution signals also contains those of lower resolution.

Because of the definition of V_j , all spaces have to satisfy a natural spacing condition:

$$f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1} \quad (10)$$

which ensures that all elements in a space are simply scaled versions of the elements in the next space. This relationship of the spanned spaces is illustrated in **Figure 1**.

The nesting of the spans of $\varphi(2^j x - k)$, denoted by V_j and graphically illustrated in **Figure 1**, is achieved by requiring that $\varphi(x) \in V_1$. This means that if $\varphi(x)$ is in V_0 , it is also in V_1 , the space spanned by $\varphi(2x)$. This means that $\varphi(x)$ can be expressed in terms of a weighted sum of the shifted $\varphi(2x)$ as

$$\varphi(x) = \sum_n h(n) \sqrt{2} \varphi(2x - n), \quad n \in \mathbb{Z} \quad (11)$$

where the coefficients $h(n)$ are a sequence of real or possibly complex numbers called the scaling function coefficients (or the scaling filter or the scaling vector) and the $\sqrt{2}$ maintains the norm of the scaling function with the scale of two. This recursive equation is fundamental to the theory of the scaling functions and is referred by different names such as refinement equation, multiresolution analysis (MRA) Equation, or dilation Equation.

The Haar scaling function is the simple unit-width, unit-height pulse function $\varphi(x)$ shown in **Figure 2(a)**.

It is obvious that $\varphi(2x)$ can be used to construct $\varphi(x)$ by

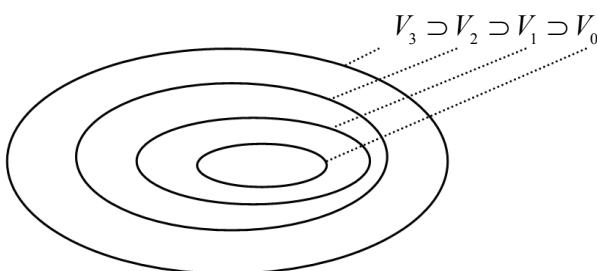


Figure 1. Nested vector spaces spanned by the scaling functions.

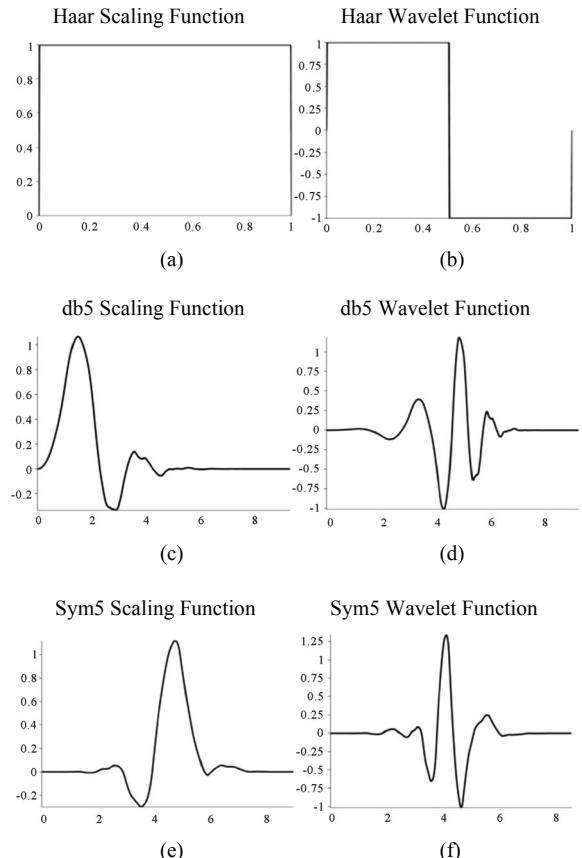


Figure 2. “Haar”, “db5”, and “Sym5” scaling and wavelet functions.

$$\varphi(x) = \varphi(2x) + \varphi(2x-1) \quad (12)$$

which means that relation (11) is satisfied for coefficients $h(0) = 1/\sqrt{2}$, $h(1) = 1/\sqrt{2}$. The fifth-order Daubechies scaling function shown in **Figure 2(c)**, satisfies relation (11) for $h(0) = 0.1601$, $h(1) = 0.6038$, $h(2) = 0.7243$, $h(3) = 0.1384$, ..., $h(9) = 0.0033$. Also, the fifth-order Symlet scaling function shown in **Figure 2(e)** satisfies Equation (11) for $h(0) = 0.0195$, $h(1) = -0.0211$, $h(2) = -0.1753$, $h(3) = 0.0166$, ..., $h(9) = 0.0273$. Indeed, the design of wavelet systems is how to choose the coefficients $h(n)$.

The important features of a signal can better be described or parameterized, not by using $\varphi_{j,k}(x)$ and increasing j to increase the size of the subspace spanned by the scaling functions, but by defining a slightly different set of functions $\varphi_{j,k}(x)$ that span the differences between the spaces spanned by the various scales of the scaling function. These functions called the wavelet functions. There are several advantages for requiring that the scaling and wavelet functions be or-

thogonal. Orthogonal basis functions allow simple calculation of expansion coefficients and also Parseval's theorem holds that allows partitioning of the signal's energy in the wavelet transform domain. The orthogonal complement of V_j in V_{j+1} is defined as W_j . This means that all members of V_j are orthogonal to all members of W_j . We require

$$\int \varphi_{j,k}(x) \psi_{j,l}(x) dx = 0 \quad (13)$$

for all corresponding $j, k, l \in \mathbb{Z}$. The relationship between the various subspaces can be seen from the following expansions. From (7), we may start at any V_j , say at $j = 0$, and write

$$V_0 \subset V_1 \subset V_2 \subset \dots \subset L^2. \quad (14)$$

We now define the wavelet spanned subspace W_0 such that

$$V_1 = V_0 \oplus W_0 \quad (15)$$

which extends to

$$V_2 = V_0 \oplus W_0 \oplus W_1. \quad (16)$$

In general, this gives

$$L^2 = V_0 \oplus W_0 \oplus W_1 \oplus \dots \quad (17)$$

when V_0 is the initial space spanned by the scaling function $\varphi(x-k)$. **Figure 3** pictorially shows the nesting of the scaling function spaces V_j for the different scales j and how the wavelet spaces are the disjoint differences (except for the zero element) or, the orthogonal complements.

The scale of the initial space is arbitrary and could be chosen at a higher resolution of, say, $j=10$ to give

$$L^2 = V_{10} \oplus W_{10} \oplus W_{11} \oplus \dots \quad (18)$$

or at a lower resolution such as $j=-5$ to give

$$L^2 = V_{-5} \oplus W_{-5} \oplus W_{-4} \oplus \dots \quad (19)$$

or at even $j=-\infty$ where (17) becomes

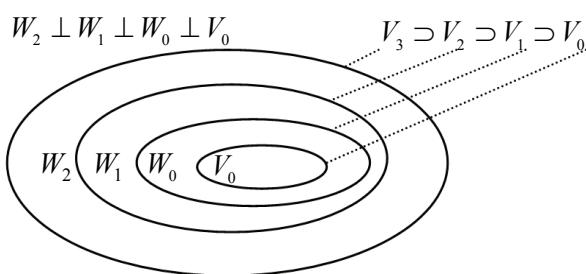


Figure 3. Scaling and wavelet functions vector spaces.

$$L^2 = \dots \oplus W_{-2} \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus W_2 \oplus \dots \quad (20)$$

eliminating the scaling space altogether.

Since these wavelets reside in the space spanned by the next narrower scaling function, $W_0 \subset V_1$, they can be represented by a weighted sum of the shifted scaling function $\varphi(2x)$ defined in (11) by

$$\psi(x) = \sum_n h_l(n) \sqrt{2} \varphi(2x-n), \quad n \in \mathbb{Z} \quad (21)$$

for some set of coefficients $h_l(n)$. From the requirement that the wavelets span the difference or orthogonal complement spaces, and the orthogonality of the integer translates of the wavelet (or scaling function), it can be shown that the wavelet coefficients (modulo translations by integer multiples of two) are required by orthogonality to be related to the scaling function coefficients by

$$h_l(n) = (-1)^n h(1-n). \quad (22)$$

The function generated by (21) gives the prototype or the mother wavelet $\psi(x)$ for a class of expansion functions of the form

$$\psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k) \quad (23)$$

where 2^j is the scaling of x , k is the translation in x , and $2^{j/2}$ maintains the L^2 norm of the wavelets for the different scales. The Haar wavelet function which is associated with the scaling function in **Figure 2(a)**, is shown in **Figure 2(b)**. For the Haar wavelet, the coefficients in (21) are $h_l(0) = 1/\sqrt{2}$, $h_l(1) = -1/\sqrt{2}$ which satisfy Equation (22). Daubechies and Symlet wavelet functions associated with the scaling functions in **Figures 2(c)** and **2(e)**, are shown in **Figures 2(d)** and **2(f)**, respectively.

We have now constructed a set of functions $\varphi_k(x)$ and $\psi_{j,k}(x)$ that could span all of $L^2(\mathbb{R})$. According to (17), any function $g(x) \in L^2(\mathbb{R})$ could be written

$$g(x) = \sum_{k=-\infty}^{\infty} c(k) \varphi_k(x) + \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} d(j, k) \psi_{j,k}(x) \quad (24)$$

as a series expansion in terms of the scaling function and wavelets. In this way, the first summation in (24) gives a function that is a low resolution or coarse approximation of $g(x)$. For each increasing index j in the second summation, a higher or finer resolution function is added, which leads to more details.

The 1-D Discrete Wavelet Transform: Since

$$L^2 = V_{j_0} \oplus W_{j_0} \oplus W_{j_0+1} \oplus \dots \quad (25)$$

by using (4) and (23), a more general statement for the expansion Equation (24) can be given by

$$\begin{aligned} g(x) &= \sum_k c_{j_0}(k) 2^{j_0/2} \varphi(2^{j_0}x - k) \\ &\quad + \sum_k \sum_{j=j_0}^{\infty} d_j(k) 2^{j/2} \psi(2^j x - k) \end{aligned} \quad (26)$$

or

$$\begin{aligned} g(x) &= \sum_k c_{j_0}(k) \varphi_{j_0,k}(x) \\ &\quad + \sum_k \sum_{j=j_0}^{\infty} d_j(k) \psi_{j,k}(x) \end{aligned} \quad (27)$$

where j_0 could be zero as in (17) and (24), it could be ten as in (18), or it could be negative infinity as in (20) where no scaling functions are used. The choice of j_0 sets the coarsest scale whose space is spanned by $\varphi_{j_0,k}(x)$. The rest of $L^2(\mathbb{R})$ is spanned by the wavelets which provide the high resolution details of the signal. The coefficients in this wavelet expansion are called the one-dimensional discrete wavelet transform (1-D DWT) of the signal $g(x)$. If the wavelet system is orthogonal, these coefficients can be calculated by inner products

$$c_j(k) = \int g(x) \varphi_{j,k}(x) dx \quad (28)$$

and

$$d_j(k) = \int g(x) \psi_{j,k}(x) dx \quad (29)$$

The DWT is similar to Fourier series but, in many ways, is much more flexible and informative. It can be made periodic like Fourier series to represent periodic signals efficiently. However, unlike Fourier series, it can be used directly on non-periodic transient signals with excellent results.

The 2-D Discrete Wavelet Transform: The one-dimensional transforms of the previous discussion are easily extended to two-dimensional functions like images. In two dimensions, a two-dimensional scaling function, $\varphi(x,y)$, and three two-dimensional wavelets, $\psi^H(x,y)$, $\psi^V(x,y)$, $\psi^D(x,y)$ are required. Each is the product of one-dimensional scaling function φ and corresponding wavelet ψ . Excluding products of functions with the same variable that produce one-dimensional results, like $\varphi(x)\varphi(y)$, the four possible products produce the separable scaling function

$$\varphi(x,y) = \varphi(x)\varphi(y) \quad (30)$$

and the separable directionally sensitive wavelets

$$\psi^H(x,y) = \varphi(x)\varphi(y) \quad (31)$$

$$\psi^V(x,y) = \varphi(x)\psi(y) \quad (32)$$

$$\psi^D(x,y) = \psi(x)\varphi(y). \quad (33)$$

These wavelets measure functional variations – intensity or gray-level variations for images – along the different directions: ψ^H measures the variations along columns (for example, horizontal edges), ψ^V responds to the variations along rows (like vertical edges), and ψ^D corresponds to the diagonals variations. The directional sensitivity is a natural consequence of the separability imposed by Equations (31) to (33); it does not increase the computational complexity of the two-dimensional transform.

Given separable two-dimensional scaling and wavelet functions, extension of the one-dimensional DWT to two-dimensions is straightforward. We first define the scaled and translated basis functions:

$$\varphi_{j,m,n}(x,y) = 2^{j/2} \varphi(2^j x - m, 2^j y - n) \quad (34)$$

$$\begin{aligned} \psi_{j,m,n}^i(x,y) &= 2^{j/2} \psi^i(2^j x - m, 2^j y - n), \\ i &= \{H, V, D\}. \end{aligned} \quad (35)$$

The discrete wavelet transform of function $g(x,y)$ of size $M \times N$ is then

$$\begin{aligned} c_{j_0}(m,n) \\ = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} g(x,y) \varphi_{j_0,m,n}(x,y) \end{aligned} \quad (36)$$

$$\begin{aligned} d_j^i(m,n) \\ = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} g(x,y) \psi_{j_0,m,n}^i(x,y), \\ i = \{H, V, D\}. \end{aligned} \quad (37)$$

As in the one-dimensional case, j_0 is an arbitrary starting scale and the $c_{j_0}(m,n)$ coefficients define an approximation of $g(x,y)$ at scale j_0 . The $d_j^i(m,n)$ coefficients add horizontal, vertical, and diagonal details for scales $j \geq j_0$. We normally let $j_0 = 0$ and select $N = M = 2^J$ so that $j = 0, 1, 2, \dots, J-1$ and $m, n = 0, 1, 2, \dots, 2^J-1$. Given the $c_{j_0}(m,n)$ and $d_j^i(m,n)$ of Equations (36) and (37), $g(x,y)$ is reconstructed via the inverse discrete wavelet transform

$$\begin{aligned} g(x, y) = & \frac{1}{\sqrt{MN}} \sum_m \sum_n c_{j_0}(m, n) \varphi_{j_0, m, n}(x, y) \\ & + \frac{1}{\sqrt{MN}} \sum_{i=H, V, D} \sum_{j=j_0}^{\infty} \sum_m \sum_n d_j^i(m, n) \psi_{j, m, n}^i(x, y). \end{aligned} \quad (38)$$

In the next section, we will apply the 2-D discrete wavelet transform to the frames of a video sequence independently to extract the low frequencies and the high frequencies components of each video frame.

3. Proposed Video Coding System

As mentioned before, the main idea of this paper is to decompose a given video stream into two separated parts such that one part includes low frequencies components (information about the main structures and the background of video frames) and the other part includes high frequencies components (information about edges, borders, and details of the video frames). The decomposition of the input video stream into two separated components is accomplished through the two-dimensional discrete wavelet transform.

As shown in the previous section, there are several well-known families of wavelets which can be used in image processing tasks such as Haar wavelets, Daubechies wavelets and Symlets (short form for symmetrical wavelets). Among the different families of wavelets, Haar wavelet transform is the simplest one and has very low complexity; for this reason it is used in many applications in signal and image processing. Hence, in our proposed method, we use two-dimensional Haar wavelet as default. Of course, in order to generalize our technique for other types of wavelets, we have tested our proposed scheme by the fifth-order two-dimensional Daubechies wavelet and the fifth-order two-dimensional Symlet. The results are given in Section 4.

Since H.264 codec is more compatible with coding the main structures of the objects and the low frequencies components in a video sequence, the proposed method utilizes two-dimensional wavelet transform to extract the low frequencies components from video sequence and encode them by H.264 codec. The visual quality of these components directly depends on the quantization factor and the other parameters of H.264 video codec. In our proposed method, the low frequencies part of each frame has comparatively very smaller dimensions. Quantizing these parts of the video with more bits and utilizing the efficient types of motion estimation for motion compensation will increase the quality of the reconstructed video.

The remaining parts of the frames in the video stream, which are the high frequencies components, should be encoded by a different way. Since a large number of very small quantities are produced during the decomposition process, they can be neglected by assigning zero values

after a thresholding procedure. So, a large number of zeros are the most repeated symbols in the high frequencies bands. When a specific symbol is repeated very frequently in a sequence, an optimum source coding procedure can be done by Run Length Coding (RLC). In a raw of "zero" repetitions, one "zero" symbol and the number of repetitions are encoded afterward. The more the symbol "zero" is repeated, the more the sequence is compressed [8]. By applying a proper threshold value, the enough number of zeros is produced, so the compression rate is increased. This hard threshold value (T) is simply applied on each transform coefficient value ($P_{i,j}$) of the high frequencies bands by the following decision equation:

$$P_{i,j} = \begin{cases} P_{i,j} & |P_{i,j}| > T \\ 0 & \text{otherwise} \end{cases} \quad (39)$$

Figure 4 shows the block diagram of the overall proposed system. First of all, the two-dimensional discrete wavelet transform is applied on the video source and the low frequencies part is encoded by H.264 codec and the remaining parts, which include information mostly about the video objects' edges and borders, are encoded using RLC algorithm.

To apply the two-dimensional wavelet transform on a given video sequence, it is applied on each frame of video sequence, independently. Since the video is QCIF formatted, each frame contains luminance (Y) and chrominance (Cb and Cr) layers; therefore the two-dimensional wavelet transform is applied three times for each frame. By recollecting the LL band of the luminance and chrominance values for each frame and combining them into a video with sequenced frames, a new video sequence is generated with very smaller dimensions, with the same structure as the original video sequence.

Figure 5 shows an example for two-dimensional wavelet transform. **Figure 5(a)** is a frame of "Suzie"

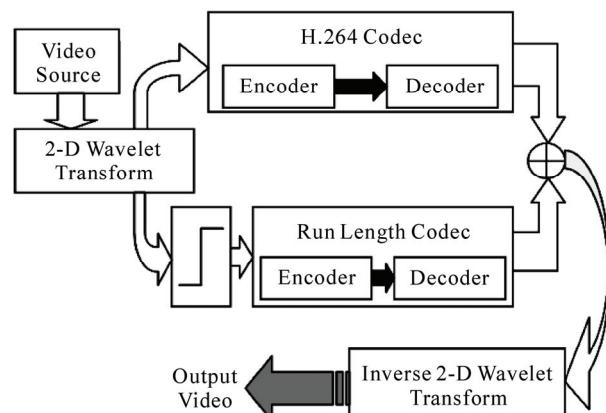


Figure 4. Block diagram of the proposed system.

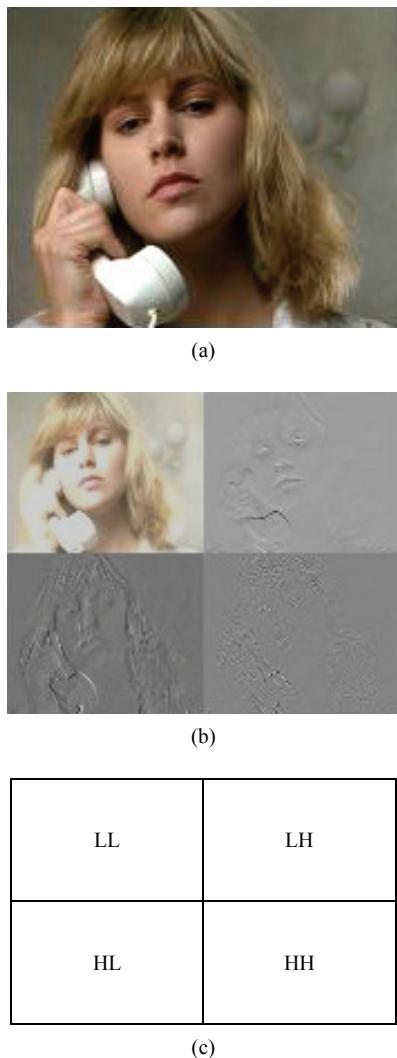


Figure 5. (a) a video frame; (b) two-dimensional Haar wavelet transform of the frame; (c) the corresponding bands.

video sequence. After applying a two dimensional Haar wavelet transform on it, **Figure 5(b)** is obtained. Finally, **Figure 5(c)** indicates the corresponding LL, LH, HL, and HH bands according to **Figure 5(b)**. Considering that the LH, HL, and HH bands show the disparity between the neighboring pixels, respectively, in the horizontal, vertical and oblique directions, these bands resemble the edges and borders in a frame of video. Therefore the corresponding regions in the frame which do not have edges and borders, produce zero or near zero values for these bands. Also applying the hard threshold value can simply increase the number of “zero” symbols. By increasing the threshold value, more “zero” symbols are produced and the compression rate is increased; therefore fewer bits are utilized for encoding by RLC algorithm. In other words, the amount of bits used to represent the high frequencies components of a frame is negligible when compared to the amount of bits produced by H.264 en-

coder to represent the low frequencies components of that frame [9].

4. Experimental Results

In this section, the results of the proposed method are compared with the results of H.264 default mode. At first, we need to choose a proper threshold value. A suitable value for the threshold can be chosen by cross-validation. The proposed method is applied on some famous test video samples like “Suzie” and “foreman” video sequences. Experiments on these video sequences show that by selecting the hard threshold value so that about 95 percent of the coefficients in the high frequencies bands are set to “zero”, the best rate-distortion performance can be achieved. It is noticeable that for achieving the equal compression rates for the LH, HL, and HH bands and also in different layers of the input video (luminance and chrominance layers), the different amount of threshold values must be applied for the different bands, since the required threshold value for the HH band is lower than the required threshold value for the LH and HL bands. In **Figure 6**, the hard threshold value is chosen so that about 95 percent of the quantities in any band, except the LL band, will be “zero”; therefore an equivalent compression is achieved for all three bands. It must be mentioned that the quantities produced by the two-dimensional wavelet transform for the LH, HL, or HH bands are either positive or negative values; therefore an absolute threshold value is applied by the decision Equation (39).

The rate-distortion plots of the proposed method and H.264 default mode are compared in **Figure 7** for “Suzie” video sequence. Rate-distortion plot presents the amount of PSNR over the different bit rates. PSNR for the default mode is computed by comparing the output video of H.264 decoder with the original video (input video) pixel-wise, where the dimensions of each frame are 176×144 pixels. The proposed method utilizes the two-dimensional Haar wavelet transform; therefore the dimensions of each input frame to H.264 encoder are

LL (H.264)	LH (95%)
HL (95%)	HH (95%)

Figure 6. The hard threshold values are chosen so that about 95 percent of the coefficients in the high frequency regional bands are set to “zero”.

88×72 pixels. Hence, the spatial resolution of the proposed method is 4 times smaller than the original H.264 mode, resulting in a very large compression rate; but PSNR is quite comparable for very low bit rates.

In order to test the performance of the proposed method for the other types of wavelets, we also test our proposed technique on the fifth-order Daubechies wavelet and the fifth-order Symlet wavelet. The rate-distortion plots of the proposed method by these wavelets are compared with Haar default wavelet and the original H.264 mode in **Figure 8**. As it shows, performance of our proposed system for these families of wavelets is comparable with Haar wavelet. This implies that we can easily generalize our proposed method for the other suitable types of wavelets.

In order to compare the visual quality of the decoded videos subjectively, we also show a sample frame of the

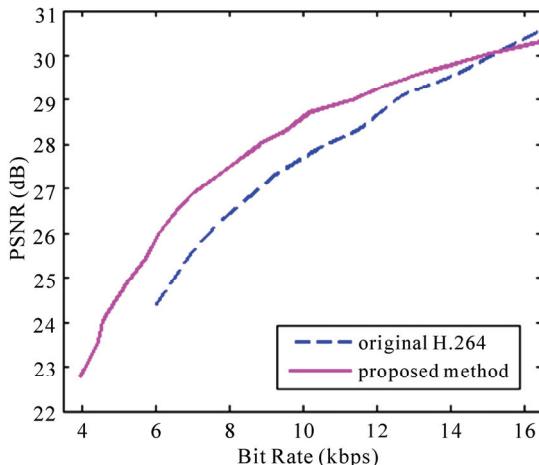


Figure 7. Comparison between rate-distortion plots of the original H.264 and the proposed method by “Haar” wavelet for “Suzie” video sequence.

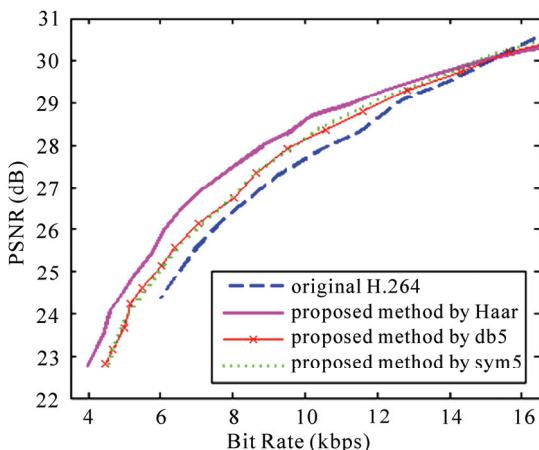


Figure 8. Comparison among rate-distortion plots of the original H.264 and the proposed method by “Haar”, “db5”, and “Sym5” wavelets for “Suzie” video sequence.

reconstructed videos for both the original H.264 codec and our proposed system in **Figure 9**. As we can see in **Figure 9**, the visual quality of the decoded video frame for the proposed scheme (right side pictures) is much better than the visual quality of the decoded frame for the original H.264 method (left side pictures). Although for the high bit rates, the proposed method can not achieve good results, but for very low bit rates, it shows superior results.

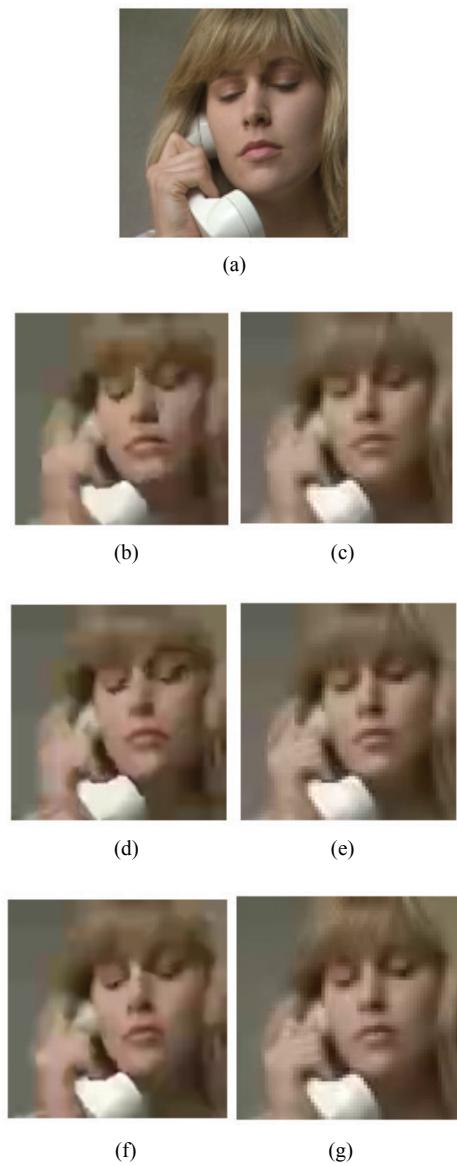


Figure 9. Subjective comparison between the visual qualities of the decoded videos for a sample frame of “Suzie” video sequence: (a) the original input frame; (b), (d), (f) the outputs for H.264 decoder at rates 10 kbps (PSNR=27.8), 11 kbps (PSNR=28.2), 13 kbps (PSNR=28.9), respectively; (c), (e), (g) the outputs for the proposed decoder at the rates 10 kbps (PSNR=28.7), 11 kbps (PSNR=29), 13 kbps (PSNR=29.7), respectively.

The main advantages of the proposed method are summarized as follows:

Advantage 1: For a bit rate between 4 to 16 kb/s (very low bit rates), PSNR of the proposed method is higher than PSNR of H.264 default mode. Since the most important information is lost during quantizing with high quantization factors, the proposed method avoid losing this part of information by separating them from the original video and then coding them using RLC algorithm. This property can highly be utilized in applications when very low bit rates are requested for video communication (such as videoconferencing and video telephony).

Advantage 2: The proposed method, compared with H.264 default mode, can achieve good performance for the much less bit rates. Therefore the proposed method can be utilized for sending video over very low capacity channels like the home-used dial-up connections. There is another case for very low capacity channels in which our proposed video coding system can be used effectively. In GSM (Global System for Mobile communication) networks, speech or other data are communicated between BTS (Base Transceiver Station) and MS (Mobile Station) mostly over a half-rate traffic channel at rate 11.4 kbytes/s. If we want to transmit or receive a video sequence over this very low capacity channel, It will be better to use the proposed video coding scheme of this paper since it provides much more acceptable basic-quality video in such a bit rate (11.4 kbytes/s) compared to the original H.264 codec as can be seen in **Figure 9**.

Advantage 3: The most challenging problem of H.264/AVC standard is its high computational complexity which has limited its usage in real-life applications. The computational complexity of H.264/AVC standard is directly related to the dimensions of the frames in the video sequences. Therefore reducing the spatial resolution to a quarter of the size of the original resolution would reduce the computational complexity dramatically. Since the computational complexity of the wavelet transform in comparison to the computational complexity of H.264 codec is almost negligible; therefore the proposed method is much faster than the case using just H.264 codec. This helps to improve the performance of H.264/AVC standard to be more compatible with the new emerging applications.

5. Conclusions

In this paper we described a novel video compression approach that combines H.264/AVC standard and two-dimensional discrete wavelet transform. The main goal of our proposed method is enhancing the performance of H.264/AVC standard to be more reliable for very low bit-rate applications. To do this, video information is decomposed into two parts, known as the low frequencies components and the high frequencies components,

which contain information about the objects' main structures and edges, respectively. To decompose this information, the two-dimensional discrete wavelet transform is applied on the sequenced frames. Then the low frequencies parts of all frames are encoded by H.264/AVC standard while the high frequencies parts of frames are encoded using RLC algorithm. As revealed by experiments, the main advantage of the proposed method compared to H.264 default mode is requiring lower bit rate for the same value of PSNR in case of very low bit rates. Also we showed that the proposed method is computationally more efficient than the ordinary H.264/AVC standard.

6. Acknowledgement

This research has been supported by Iran Telecommunication Research Center, Tehran, Iran, which is appreciated.

7. References

- [1] B. J. Kim, Z. Xiong and W. A. Pearlman, "Low Bit-Rate Scalable Video Coding with 3-D Set Partitioning in Hierarchical Trees (3-D SPIHT)," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10, No. 8, 2000, pp. 1374-1386.
- [2] I. E. G. Richardson, "Video Codec Design Developing Image and Video Compression Systems," John Wiley & Sons, 2002.
- [3] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer and T. Wedi, "Video Coding with H.264/AVC: Tools, Performance, and Complexity," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 4, No. 1, 2004, pp. 7-28.
- [4] T. Wiegand, G. J. Sullivan, G. Bjntegaard and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 7, 2003, pp. 560-576.
- [5] R. Talluri, K. Oehler, T. Bannon, J. D. Courtney, A. Das, and J. Liao, "A Robust, Scalable, Object-Based Video Compression Technique for Very Low Bit-Rate Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 7, No. 1, 1997, pp. 221-232.
- [6] C. S. Burrus, R. A. Gopinath and H. Guo, "Introduction to Wavelets and Wavelet Transforms: A Primer," Prentice Hall, 1998.
- [7] R. C. Gonzalez and R. E. Woods, "Digital Image Processing," 2nd Edition, Prentice Hall, 2002.
- [8] D. Salomon, "Data Compression: The Complete Reference," 4th Edition, Springer, Berlin, 2007.
- [9] A. Aghagolzadeh, S. Meshgini, M. Nooshyar and M. Aghagolzadeh, "A Novel Video Compression Technique for Very Low Bit-Rate Coding by Combining H.264/ AVC Standard and 2-D Wavelet Transform," *Proceedings of 9th International Conference on Signal Processing*, Beijing, 2008, pp. 1251-1254.