EXPLOITING TEMPORAL CORRELATION WITH ADAPTIVE BLOCK-SIZE MOTION ALIGNMENT FOR 3D WAVELET CODING

Ruiqin Xiong[1], Feng Wu[2], Shipeng Li[2], Zixiang Xiong[3], Ya-Qin Zhang[2]
1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080
2. Microsoft Research Asia, #49 ZhiChun RD, Haidian, Beijing, 100080
3. Dept. of Electrical Engineering, Texas A&M University, College Station, TX 77843

ABSTRACT

This paper proposes an adaptive block-size motion alignment technique in 3D wavelet coding to further exploit temporal correlations across pictures. Similar to B picture in traditional video coding, each macroblock can motion align from forward and/or backward for temporal wavelet de-composition. In each direction, a macroblock may select its partition from one of seven modes – 16x16, 8x16, 16x8, 8x8, 8x4, 4x8 and 4x4 – to allow accurate motion alignment. Furthermore, the rate-distortion optimization criterions are proposed to select motion mode, motion vectors and partition mode. Although the proposed technique greatly improves the accuracy of motion alignment, it does not directly bring the coding efficiency gain because of smaller block size and more block boundaries. Therefore, an overlapped block motion alignment is further proposed to cope with block boundaries and to suppress spatial high-frequency components. The experimental results show the proposed adaptive block-size motion alignment with the overlapped block motion alignment can achieve up to 1.0 dB gain in 3D wavelet video coding. Our 3D wavelet coder outperforms the MC-EZBC for most sequences by 1~2dB and we are doing up to 1.5 dB better than H.264.

Keywords: 3D wavelet coding, lifting-based wavelet transform, adaptive block-size motion alignment, overlapped block motion alignment, rate-distortion optimization

1. INTRODUCTION

In video coding, besides taking advantage of spatial correlation within a picture by transform and quantization, the most important issue is how to fully exploit the strong temporal correlation across pictures. This issue has been extensively investigated in past decades. Generally, the proposed approaches for utilizing temporal correlation among pictures can be divided into two categories: motion compensation (MC) and temporal transform.

Motion compensation is an indispensable part in traditional predicted DCT video coding schemes, which prevails in most existing video coding standards. A predicted picture is generated from the previously reconstructed picture and/or the future reconstructed picture with estimated motion parameters and then is subtracted from the original picture in motion compensation. The obtained residual picture contains less energy than the original picture, thus costs less bits to code. The early motion compensation technique adopted in MPEG-1 is very simple -- each 16x16 macroblock only has an integer-pixel shifting associated with its reference in horizontal and vertical axes, respectively [1]. For the sake of accurate motion compensation, MPEG-2 extends the estimated shifting accuracy to half-pixel [2]. MPEG-4 not only further extends the estimated shift accuracy up to quarter-pixel but also allows 8x8 block as a unit of motion compensation [3]. To more accurately describe and compensate local motion among pictures, the latest video coding standard H.264/AVC adopts the adaptive block-size motion compensation technique [4]. The unit for motion compensation in a macroblock can be selected from one of seven modes: 16x16, 8x16, 16x8, 8x8, 8x4, 4x8 and 4x4. With rate-distortion optimization (RDO), testing results from the latest H.264/AVC reference software show the technique can improve coding efficiency about 1.0dB at quarter-pixel accuracy.

On the contrary, 3D wavelet or DCT video coding schemes do not use traditional motion compensation. Instead, correlations across pictures are exploited by temporal transform. However, due to global and local motion that typically coexists across pictures, pixels at the same position of adjacent pictures usually correspond to different parts of an object. Directly applying temporal transform on spatially co-located pixels across pictures may result in low coding efficiency because of misalignment. Similar to motion compensation used in traditional predicted DCT coding, motion alignment is needed in 3D video coding to improve energy compaction of temporal transforms. Many studies have been conducted on motion alignment. Taubman *et al.* [5] pre-distorted video sequence by translating pictures relative to one another before wavelet transform so as to compensate for camera pan motion. Wang *et al.* [6] used the mosaic technique to warp each video picture into a common coordinate system and then applied a shape-adaptive 3D wavelet transform on the warped video. Both schemes assume a global motion model, which may be inadequate for video sequences with local motion as well.

The idea of block matching in motion compensation was also applied to motion alignment. Ohm [7] incorporated a 16x16 macroblock matching technique similar to that in traditional predicted DCT video coding into temporal wavelet transform. But Ohm's scheme cannot guarantee perfect reconstruction beyond full-pixel motion alignment. Xu *et al.* [8] utilized the same motion technique to get the integer motion of each 16x16 macroblock. Then, pixels along the same motion trajectory are linked to form a single thread according to the motion vectors of the macroblocks they belong to. This process is called as motion threading. Luo et al. [10] introduced lifting-based wavelet transform to motion threading to support sub-pixel motion alignment. But the unit of motion alignment in [10] is still a 16x16 macroblock. In the MC-EZBC scheme proposed by Chen and Woods [9], motion alignment is performed by splitting a picture into smaller blocks first and then forming hierarchical variable size block structure at sub-pixel precision.

It is not reasonable to assume that all pixels in a macroblock always have the same motion because of irregular object shapes and its motion, the adaptive block-size motion compensation technique employed in H.264/ MPEG-4 AVC has found to be more efficient. Therefore, in this paper we propose an adaptive block-size motion alignment technique in 3D wavelet coding to further exploit temporal correlations across pictures. The motion threading 3D wavelet coding scheme is selected as the baseline in this paper because it better deals with the long-term correlations in video sequence. Similar to B picture in traditional video coding, each macroblock can motion align from forward and/or backward. But in each direction, a macroblock may select its partition from one of seven modes –16x16, 8x16, 16x8, 8x8, 8x4, 4x8 and 4x4 – to allow finer motion alignment, where each sub-block has its own motion vector. Furthermore, the rate-distortion optimization criterions based on the Lagrangian cost function are proposed to select the best motion vector and partition mode for each macroblock.

However, enabling the adaptive block-size motion alignment can not directly bring significant coding efficiency gain in the 3D wavelet coding, although the distortions on the generated temporal high-pass pictures are reduced greatly. This issue is investigated in this paper as well. The reason is that the temporal high-pass pictures may contain more sharp edges and blocking artifacts caused by the adaptive block-size motion alignment. They would result in large magnitude coefficients in the spatial high-frequency subbands when the temporal high-pass pictures are further decomposed spatially by wavelet transform. Therefore, the overlapped block motion alignment (OBMA) is proposed in this paper to suppress the spatial high-frequency components in the temporal high-pass pictures.

The rest of the paper is organized as follows. Section 2 gives a brief overview of the 3D wavelet motion threading scheme. Section 3 describes the proposed adaptive block-size motion alignment technique. Section 4 proposes the overlapped motion alignment to suppress the spatially high-frequency components in the temporal high-pass pictures. Experimental results are reported in section 5. Section 6 concludes this paper.

2. OVERVIEW OF THE MOTION THREADING TECHNIQUE

Figure 1 shows the motion threading 3D wavelet video coding with the lifting-based implementation of temporal 5/3 wavelet transform. There is only a layer of temporal de-composition exemplified here. The input pictures are at the bottom. The output temporal low-pass pictures (e.g., L_0, L_1) and high-pass pictures (e.g., H_0, H_1) are at the top. Low-pass and high-pass pictures are updated alternatively. In general, the temporal de-composition consists of two lifting steps and each elementary of lifting step involves only three pictures as shown by the dashed circle in Figure 1.

For instance, in the first lifting step, the picture of Frame_1 is updated to the high-pass picture H_0 by a pair of its neighboring pictures of Frame_0 and Frame_2. Due to the existence of global and local motion among pictures, pixels located at the same position in different pictures may be out of alignment so that the temporal wavelet de-composition cannot effectively achieve high energy compaction. Therefore, Frame_1 is first performed with bi-directional motion estimation. And then, pixels in Frame_1 with the corresponding pixels in Frame_0 and/or Frame_2 identified by their motion modes and vectors are performed with wavelet filtering. This process is similar to calculating B picture residues in traditional predicted DCT coding schemes. In the second lifting step, the picture of Frame_0 is updated to the low-pass picture L_0 by the available neighboring high-pass picture. The motion modes and vectors from L_0 and H_0 are derived inversely from that from Frame_1 to Frame_0 obtained in the first lifting step. The above process is repeated to calculate H_1 and L_1, H_2 and L_2, until the last picture.

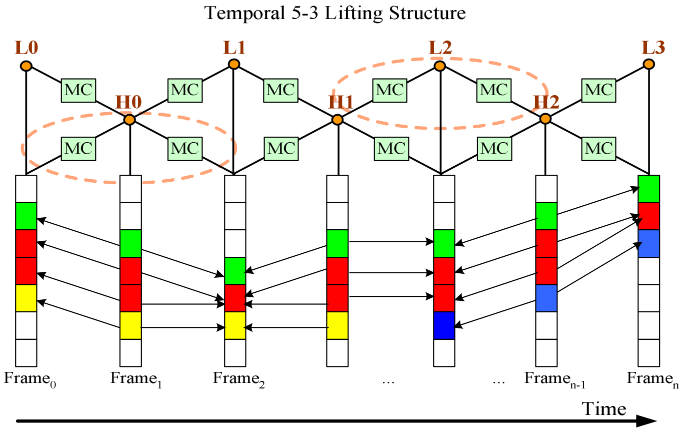

Figure 1: The motion threading 3D wavelet coding with the lifting-based implementation of temporal 5/3 wavelet transform.

Obviously, there are two special cases in the lifting-based motion threading 3D wavelet coding scheme. The one is that multiple pixels in the odd picture are mapped to the same pixel in the neighboring picture because of block-based motion alignment. In this case, all of them can be regarded as connecting with the pixel and use its value to calculate the high-pass values. When the pixel is updated to a low-pass value, only the first connection in row scan order is used at both encoder and decoder for perfect reconstruction. Another is that pixels have no matching ones in the neighboring pictures. In this case, a macroblock is marked as a terminating one when it contains many such pixels. It is similar to an Intra macroblock in traditional predicted DCT coding schemes. For individual pixels without matching, they are linked forcibly on both sides using the motion vectors of adjacent pixels.

In order to further exploit the temporal correlations among the low-pass pictures, they are continuously performed with temporal de-composition again. In general, there are three or four layers of temporal de-composition. For example, a four-layer temporal de-composition will produce five temporal subbands including the four high-pass subbands, i.e., H, LH, LLH, LLLH, and the final low-pass subband LLLL. All these temporal subbands are further decomposed by spatial wavelet transform before all wavelet coefficients are coded by 3D ESCOT [8].

3. PROPOSED ADAPTIVE BLOCK-SIZE MOTION ALIGNMENT

This section proposes the adaptive block-size motion alignment technique in 3D wavelet video coding and the criterions on selection of motion mode, motion vectors and macroblock partition. The rate-distortion curves show the proposed technique can greatly reduce the distortion of high-pass pictures.

3.1 Adaptive block-size motion alignment (ABSMA)

As discussed above, motion estimation is performed on each macroblock of odd picture to obtain its forward and backward motion parameters. Similar to B pictures in traditional predicted DCT coding schemes, when motion parameters are applied in wavelet filtering, either bi-directional alignment or single directional alignment should be allowed. In addition, the strong correlation between forward and backward motion vectors should be exploited to code them efficiently. Four correlated motion estimation (CME) modes i.e., DirInv, Bid, Fwd and Bwd, are proposed in Luo's scheme [10]. In DirInv mode, the motion is assumed to be stationary, and the forward and backward motion vectors have the same absolute value with opposite sign. Only one motion vector needs to be transmitted to the decoder. In Fwd or Bwd mode, only the forward or backward motion vector is transmitted. In Bid mode, both the forward and the backward motion vectors are transmitted. As a matter of factor, the motion vectors among neighboring macroblocks also have the strong correlation. A skip mode is proposed in this paper, where no motion vectors need to be transmitted. The forward and backward motion vectors are derived from the coded neighboring macroblocks. In Figure 2, the arrows with solid lines indicate the related motion parameters need to be coded and transmitted.

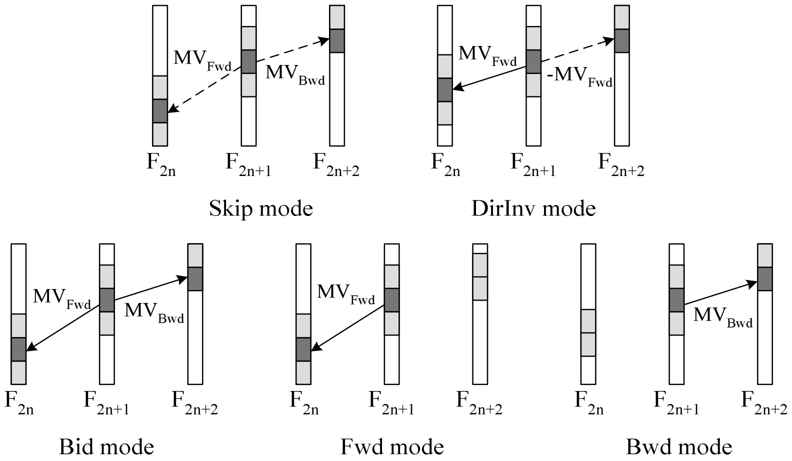

Figure 2: Five modes for motion coding and motion alignment.

In the previous works [8] and [10], each macroblock in one direction only has a motion vector. The proposed adaptive block-size motion alignment allows each macroblock with more than a motion vector. There are seven different partitions in the proposed technique – 16x16, 8x16, 16x8, 8x8, 8x4, 4x8 and 4x4 - as shown in Figure 3. Each sub-block has its own motion vector, that is, pixels in a macroblock may use different motion vectors for alignment. For example, when a macroblock selects the partition of 8x8, it has four different motion vectors for every block. There are two constrains in the proposed adaptive block-size motion alignment: (1) when a macroblock is coded with Skip mode, its partition is allowed as 16x16 only. (2) When a macroblock is coded with DirInv mode, the partition in forward and backward directions should be same.

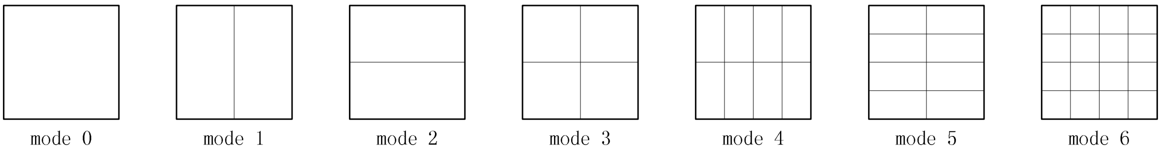

Figure 3: Partition modes of a macroblock in the proposed adaptive block-size motion alignment.

3.2. Rate-distortion optimized selection of motion parameters

The better motion mode, partition mode and motion vectors for a macroblock are selected with rate-distortion optimization. The greedy algorithm is used in this paper for reducing the searching space. The detail description is given as follows.

Firstly, each of the seven partition modes selects a set of best motion vectors with each motion vector corresponding to a sub-block. Traditional full search or any fast motion estimation algorithms can be used here to find the motion vector of each subblock. Considering the cost of coding motion vector, the Lagrangian cost function is defined as

$$J_{MOTION} = SAD_{block}(s - c(m)) + \lambda_{MOTION} \cdot R_{mv}(m - p) \tag{1}$$

The first term in equation (1) is the sum of absolute difference between the original sub-block s and its prediction $c(m)$ from the neighboring picture. For single directional modes (e.g., Fwd and Bwd), $c(m)$ is the single directional prediction decided by the motion vector. For the modes (e.g., Skip, DirInv and Bid), $c(m)$ is the bi-directional prediction. The second term is the motion cost, where m is the motion vector of the current sub-block and p is the motion vector prediction. $R_{mv}()$ is the needed bits for coding the predicted motion vector obtained by looking up an UVLC table. λ_{MOTION} is the Lagrange multiplier for searching motion vectors.

Secondly, when the motion vectors for all sub-blocks at a certain partition are available, the best partition mode is determined by minimizing

$$J_{MODE} = SSD_{MB}(s - c(m_i, MODE)) + \lambda_{MODE} \cdot (\sum R_{mv}(m_i - p_i) + R_{MODE}) \quad (2)$$

where $\sum R_{mv}(m_i - p_i)$ and R_{MODE} represent bits for coding all predicted motion vectors in a macroblock and its partition mode, respectively. $SSD_{MB}()$ is the sum square difference between the original macroblock and its prediction at the partition mode.

Finally, when the best partition mode and motion vectors are determined for each of five motion modes, the one with the smallest cost is selected for each macroblock based on another rate-distortion criterion $Cost = \eta \cdot SAD + \lambda \cdot R_{motion}$. The SAD value measures the matching correctness of the forward and backward predictions. R_{motion} represents the bits for coding motion mode as well as macroblock partition mode and motion vectors. Since the wavelet synthesis error on the boundary is larger than that on non-boundary, η is set as follows to add a penalty to the Fwd and Bwd modes.

$$\eta = \begin{cases} 1 & Skip, DirInv, Bid \\ 1.5 & Fwd, Bwd \end{cases} \quad (3)$$

λ is the Lagrange multiplier that trades off rate and distortion. Since the slope of the R-D curve is steeper at low bit rates than at high bit rates, λ should have a larger value at low bit rates. In our experiments, λ is empirically set as 16.

3.3. The R-D curves with the adaptive block-size motion alignment

After enabling the adaptive block-size motion alignment, we can perform a more accurate motion alignment during the lifting steps. The R-D curves of all the temporal sub-bands t-H and t-LH are given in Figure 4. The x-axis is average bits to code each wavelet coefficient and the y-axis is the corresponding average distortion for each pixel measured in mean square error. At the bit rate of zero, that is to say, when no bits are transmitted for wavelet coefficients, the distortion for each band is just the energy of predicted error of motion alignment. Obviously, the adaptive block-size motion alignment achieves better prediction. However, as the bit rate increases, the benefit of the proposed technique decreases quickly. At a bit rate close to or more than 0.15 bpp, the distortions of both schemes become almost the same. In other words, it is difficult to code the temporal high-band pictures generated by the proposed technique. The experimental results in Section 5 also show that using the adaptive block-size motion threading technique alone has no remarkable gain especially at high bit rates. The reason is that the temporal high-pass pictures may contain more sharp edges and blocking artifacts brought by the adaptive block-size motion alignment.

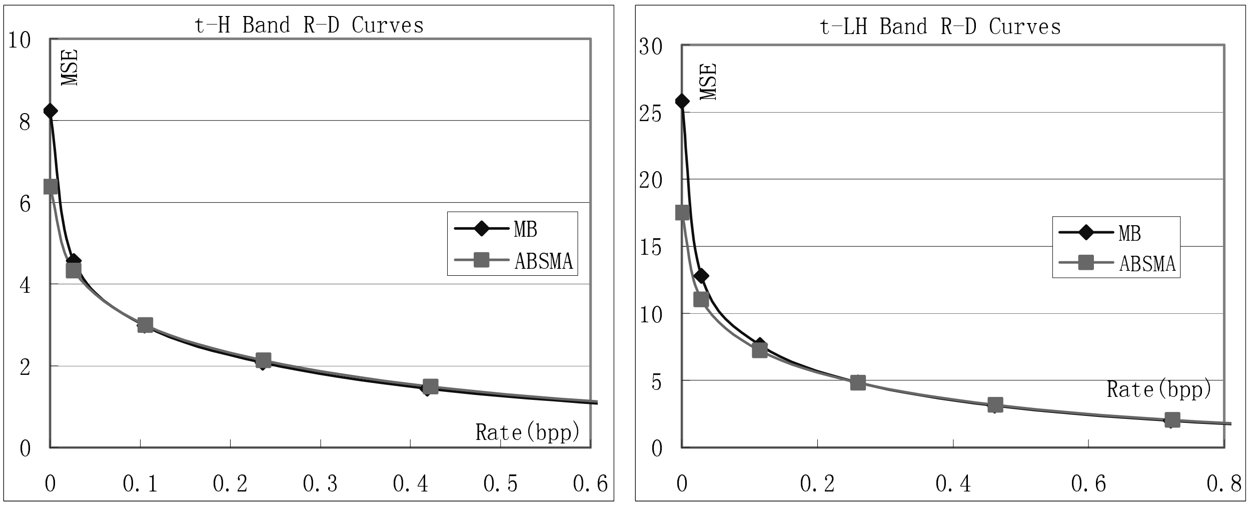

Figure 4: The R-D curves of the temporal high-pass subbands t-H and t-LH with the proposed adaptive block-size motion alignment.

4. PROPOSED OVERLAPPED BLOCK MOTION ALIGNMENT

4.1. Problem statement

In order to analyze the above problem, we use $\psi_p(\mathbf{x})$ and $\psi_r(\mathbf{x})$ to represent the predicted picture for wavelet de-composition and the reference picture, respectively. In block based motion alignment, the prediction for a macroblock is generated by copying pixels of a macroblock from the interpolated reference, which can be formulated as

$$\psi_p(\mathbf{x}) = \psi_r(\mathbf{x} + \mathbf{d}_m), \qquad \mathbf{x} \in \mathcal{B}_m \tag{4}$$

where $\mathbf{d}_m$ represents the motion vector of the block $\mathcal{B}_m$. Every pixel corresponds to a single (either integer or fractional) pixel in the reference frame as illustrated in Figure 5. Using the adaptive block-size motion alignment allows macroblocks to be split into some smaller blocks when complex local motion exists. The average prediction error for the macroblock is reduced. However, since each sub-block has its own motion vector and the neighboring motion vectors may be not always the same, some sharp edges and blocking effects at the block boundaries arise in the prediction picture, which would likewise result in the edges and blocking artifacts in temporal high-pass pictures. It's well-known that wavelet transform is a kind of global transform. The artificially blocking artifacts can affect coding efficiency significantly because a large mount of signal energy is spread in the spatial high-frequency subbands when the temporal high-pass pictures are further decomposed spatially by wavelet filters.

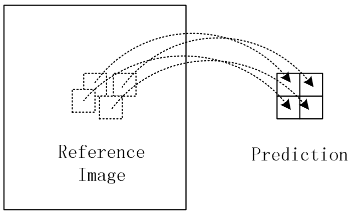

Figure 5: The generated prediction for temporal wavelet de-composition.

4.2. Overlapped block motion alignment (OBMA)

In the block-based motion model, pictures are assumed to undergo block-wise pure translation. However, the actual motion field in real-world video sequences usually varies smoothly within the same moving object while discontinues at boundaries of different moving objects. The boundaries of moving object are most likely inconsistent to the block boundaries. Therefore, considering the influence of motion vectors of neighboring blocks, Orchard et al proposed the overlapped block motion compensation (OBMC) in [13], where the prediction for a pixel is a weighted combination of several prediction generated by motion vectors of itself and its neighboring blocks:

$$\psi_p(\mathbf{x}) = \sum_{k \in \mathcal{K}} h_k(\mathbf{x}) \cdot \psi_r(\mathbf{x} + \mathbf{d}_{m,k}), \qquad \mathbf{x} \in \mathcal{B}_m \tag{5}$$

In this paper, we propose a blocking-effect free motion alignment referred to as the overlapped block motion alignment (OBMA). In the temporal lifting steps, each pixel has multiple corresponding pixels in the neighboring picture, according to the motion vector of its own block and motion vectors of neighboring blocks. A weighted combination of these pixel values are used as the prediction during lifting.

		3	6	6	3		
		9	12	12	9		
3	9	14	17	17	14	9	3
6	12	17	20	20	17	12	6
6	12	17	20	20	17	12	6
3	9	14	17	17	14	9	3
		9	12	12	9		
		3	6	6	3		

Figure 6: The window function in the proposed OBMA.

When the adaptive block-size motion alignment is enabled, seven different block-sizes can be used. The smallest block size is of 4x4. Therefore, 4x4 size is used as the basic unit in OBMA for simplicity. Another reason for using 4x4 as the

OBMA unit is that the bi-orthogonal 9/7 filter is used in the spatial de-composition and each pixel is only related to four neighboring pixels on each side. An overlapped region of size 4 in OBMA is usually enough for reducing the influence of blocking effects on wavelet transform. A raised cosine like window function is used in the proposed scheme as shown in Figure 6. For example, the pixel at the center of a block is calculated by $p = (20 \cdot p_1 + 6 \cdot p_2 + 6 \cdot p_3)/32$.

Figure 7 shows the R-D curves for the temporal high-pass sub-bands t-H and t-LH with OBMA. It shows that after we enabled the OBMA technique, the mean square prediction error has no remarkable change at the bit rate of zero but the R-D performance is improved greatly in the large range of bit rates due to the suppression of spatial high-frequency component.

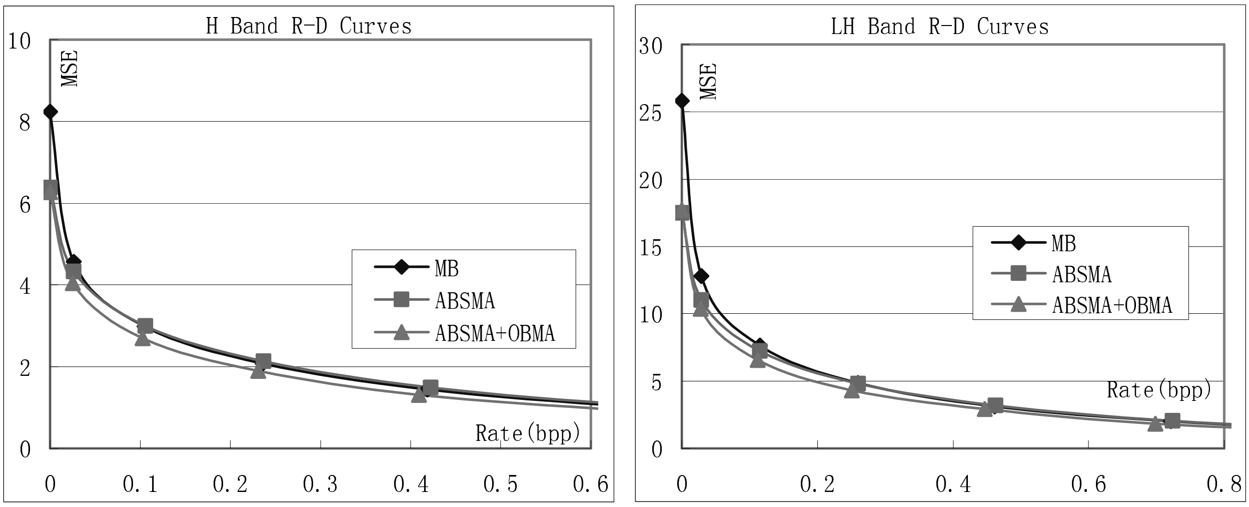

Figure 7: The R-D curves for the temporal highpass subbands t-H and t-LH with the proposed OBMA.

5. EXPERIMENTAL RESULTS

We have conducted several experiments to analyze the performance of each proposed techniques in this paper. Six MPEG standard test sequences are used: *Foreman, Mobile, Silence, Stefan, Coastguard* and *Table tennis*. All of them are in CIF format of 300 frames at a frame rate of 30Hz. In all of the tests, the entire sequence is temporally de-composed by a four-layer lifting structure into five temporal subbands. Each temporal subband is further spatially decomposed by a 3-level Spacl spatial wavelet transform. The resulted wavelet coefficients are entropy coded by 3-D ESCOT, as done in [8]. Motion estimation is performed on each layer at quarter-pixel accuracy and the motion search range at each layer is set as 32, 64, 128 and 128, respectively.

In the first experiment, we check the performance of the adaptive block-size motion alignment technique in improving the accuracy of motion alignment so as to improve the quality of bi-directional prediction and reduce the energy in temporal high-pass picture. The proposed OBMA are not turned on yet in these tests. The test results are shown in Table1. We can observe several facts from these results. First, the quality of forward prediction pictures is almost similar to that of the backward prediction pictures, while the bi-directional predictions are much better. The MSE (mean square error) of bi-directional prediction is about half of the MSE of single-directional prediction. It can be interpreted as that if the forward prediction error and backward prediction error are independent random variables, using the average of both predictions can reduce the MSE of prediction by 50%. Second, in the multi-level temporal decomposition, it is more difficult to achieve a good motion alignment at higher level due to the larger temporal distance between the pictures to be aligned. Therefore, among the four temporal highbands: t-H, t-LH, t-LLH and t-LLLH, the lowest temporal subband t-H contains least energy in each frame while the highest temporal subband t-LLLH contains the most energy.

The quality of prediction images generated by the adaptive block-size motion alignment is compared with those generated by the macroblock-based motion alignment. Due to the complex local motion exists in sequences, using the block-size adaptive motion alignment allows more accurate description of motion, thus improving the quality of motion aligned prediction, especially for sequences with sharp and irregular motion such as *Stefan* and *Foreman*. It is shown in

Table1 that it improves prediction quality by 2.5~3.0dB for *Stefan* and *Foreman* while the improvement is about 1.4dB for *Coastguard*. The energy contained in temporal highbands, which is the prediction error of each level, is reduced by 25%~50%.

Table 1: prediction improvement by block-size adaptive motion threading

		Level0			Level1			Level2			Level3		
		Left	Right	Bid	Left	Right	Bid	Left	Right	Bid	Left	Right	Bid
MB	MSE	24.6	26.7	13.9	44.6	47.8	27.2	69.9	75.3	46.0	110.4	124.3	80.7
	PSNR (dB)	34.8	34.6	37.4	32.2	32.0	34.4	30.3	29.9	32.0	28.2	27.9	29.6
ABSMA	MSE	13.1	14.6	8.2	22.0	23.8	14.1	36.1	36.7	23.5	58.4	65.3	42.2
	PSNR (dB)	37.3	37.0	39.5	35.2	35.0	37.2	33.3	33.2	35.1	31.2	31.0	32.6
improvement	MSE	11.5	12.0	5.7	22.7	24.0	13.1	33.8	38.6	22.6	52.0	59.1	38.5
	MSE (%)	46.7	45.1	41.2	50.8	50.2	48.0	48.4	51.2	49.0	47.1	47.5	47.7
	PSNR (dB)	2.5	2.4	2.1	3.0	3.0	2.8	3.0	3.3	3.1	3.0	3.0	3.0

Foreman

		Level0			Level1			Level2			Level3		
		Left	Right	Bid	Left	Right	Bid	Left	Right	Bid	Left	Right	Bid
MB	MSE	15.5	15.5	8.5	26.3	26.0	16.1	39.4	39.6	26.4	53.1	59.8	40.5
	PSNR (dB)	36.8	36.8	39.4	34.4	34.4	36.5	32.6	32.6	34.3	31.2	30.6	32.2
ABSMA	MSE	9.7	9.8	5.3	17.6	17.4	10.8	28.4	28.2	19.0	38.3	44.1	29.6
	PSNR (dB)	38.7	38.7	41.3	36.1	36.1	38.2	34.0	34.1	35.8	32.6	31.9	33.6
improvement	MSE	5.8	5.8	3.2	8.7	8.6	5.3	11.0	11.5	7.4	14.9	15.7	10.9
	MSE (%)	37.6	37.3	37.3	33.0	33.0	32.8	27.9	28.9	28.0	28.0	26.3	26.9
	PSNR (dB)	1.9	1.9	1.9	1.7	1.8	1.7	1.4	1.5	1.5	1.5	1.3	1.4

Silence

		Level0			Level1			Level2			Level3		
		Left	Right	Bid	Left	Right	Bid	Left	Right	Bid	Left	Right	Bid
MB	MSE	84.8	83.0	47.3	134.5	127.4	82.7	200.2	174.9	119.8	301.7	244.7	180.9
	PSNR (dB)	28.9	29.0	31.5	26.9	27.1	29.0	25.2	25.8	27.4	23.4	24.3	25.6
ABSMA	MSE	57.7	56.3	32.3	85.9	80.4	53.4	113.7	101.4	68.9	162.3	137.9	98.8
	PSNR (dB)	30.7	30.7	33.2	28.9	29.1	30.9	27.6	28.1	29.8	26.1	26.8	28.2
improvement	MSE	27.0	26.7	15.0	48.7	47.0	29.3	86.5	73.5	50.9	139.3	106.8	82.1
	MSE (%)	31.9	32.2	31.8	36.2	36.9	35.4	43.2	42.0	42.5	46.2	43.6	45.4
	PSNR (dB)	1.8	1.7	1.8	2.0	2.0	1.9	2.5	2.4	2.4	2.7	2.5	2.6

Mobile

		Level0			Level1			Level2			Level3		
		Left	Right	Bid	Left	Right	Bid	Left	Right	Bid	Left	Right	Bid
MB	MSE	34.0	25.5	16.4	52.1	55.3	31.5	110.6	71.7	56.5	178.2	116.1	97.8
	PSNR (dB)	34.1	34.5	36.8	31.6	31.9	33.9	29.1	29.9	31.3	27.2	28.1	29.0
ABSMA	MSE	21.8	16.5	10.7	31.1	35.2	19.7	72.0	44.7	36.7	114.7	71.4	61.7
	PSNR (dB)	36.1	36.4	38.7	33.7	33.8	35.9	31.0	31.9	33.2	29.0	29.9	30.9
improvement	MSE	12.2	9.0	5.7	21.0	20.1	11.8	38.6	27.0	19.8	63.5	44.7	36.1
	MSE (%)	35.8	35.2	34.9	40.3	36.3	37.3	34.9	37.6	35.0	35.7	38.5	36.9
	PSNR (dB)	2.0	1.9	1.9	2.1	2.0	2.0	1.9	2.0	1.9	1.9	1.9	1.9

Table

		Level0			Level1			Level2			Level3		
		Left	Right	Bid	Left	Right	Bid	Left	Right	Bid	Left	Right	Bid
MB	MSE	152.3	153.2	88.3	231.2	241.0	147.3	310.0	312.1	197.1	498.0	487.1	344.5
	PSNR (dB)	27.1	27.2	29.7	25.1	25.0	27.2	23.6	23.6	25.6	21.6	21.8	23.4
ABSMA	MSE	71.8	76.3	44.7	118.7	118.6	77.4	162.5	159.6	104.9	262.7	254.6	179.1
	PSNR (dB)	30.3	30.4	32.6	28.1	28.2	30.1	26.5	26.6	28.4	24.4	24.7	26.2
improvement	MSE	80.5	76.9	43.6	112.6	122.4	69.9	147.5	152.5	92.3	235.3	232.4	165.4
	MSE (%)	52.9	50.2	49.4	48.7	50.8	47.4	47.6	48.9	46.8	47.2	47.7	48.0
	PSNR (dB)	3.2	3.2	3.0	3.1	3.1	2.9	2.9	3.0	2.8	2.8	2.9	2.8

Stefan

		Level0			Level1			Level2			Level3		
		Left	Right	Bid	Left	Right	Bid	Left	Right	Bid	Left	Right	Bid
MB	MSE	44.8	42.7	21.1	90.1	89.7	50.1	146.2	147.1	95.7	198.6	194.1	139.9
	PSNR (dB)	31.8	31.9	35.1	28.7	28.7	31.2	26.6	26.5	28.4	25.2	25.3	26.7
ABSMA	MSE	31.7	31.3	14.4	67.5	66.5	36.9	109.7	107.3	71.7	145.2	141.4	102.8
	PSNR (dB)	33.3	33.3	36.7	30.0	30.0	32.6	27.9	27.9	29.7	26.6	26.7	28.1
improvement	MSE	13.1	11.5	6.7	22.6	23.2	13.2	36.5	39.8	24.0	53.4	52.7	37.1
	MSE (%)	29.2	26.8	31.9	25.0	25.8	26.4	24.9	27.1	25.1	26.9	27.2	26.5
	PSNR (dB)	1.5	1.4	1.6	1.3	1.3	1.4	1.3	1.4	1.3	1.4	1.4	1.4

Coastguard

In the second experiment, we compare the performance of overlapped block motion alignment (OBMA) technique with that of macroblock-based motion threading scheme, where the adaptive block-size motion alignment is already turned on in these tests. Figure 8 shows that the OBMA technique can achieve up to 0.5dB gain in the final coding efficiency by suppressing blocking effects in temporal high-pass pictures.

Finally, we compare the proposed 3D advanced wavelet motion alignment scheme with two benchmark coders: MC-EZBC [9] and H.264 JM6.1e [15]. The result of MC-EZBC is quoted from [14]. For H.264[15], we try to set the test conditions as follows to obtain its best performance. The GOP is set as a whole sequence with only one I frame; and two B pictures are inserted between each two P pictures; the quantization parameters for these three picture types satisfy $QP_I=QP_P-1=QP_B-2$; motion estimation is performed at quarter-pixel accuracy with a search range of 32; five references are allowed for both the P frames and B frames; and CABAC and R-D optimization are also turned on. The testing conditions for our proposal are the same as described above. From the results shown in Figure 9, we can see that the proposed scheme outperforms MC-EZBC for all the six sequences by 1~2dB. The proposed scheme is also compared with the best results of H.264. For *Silence*, *Foreman* and *Stefan* sequences, the loss of our coder is about 0.4 to 1.4 dB. For *Table tennis* and *Mobile* sequences, the performance of our coder catches up with that of H.264. Furthermore, for *Coastguard* sequence, our coder even outperforms the best result of H.264 by up to 1.1dB. Considering that the results of H.264 are of single layer bitstreams which are optimized for each QP, our proposed scalable coder is very competitive with H.264.

6. CONCLUSIONS

To fully exploit the temporal correlation across pictures, this paper first proposes the adaptive block-size motion alignment scheme, which can improve the quality of predictions and reduce the energy of temporal high-pass pictures greatly. Similar to B picture in traditional video coding, each macroblock can motion align from forward and/or backward for temporal wavelet de-composition. In each direction, a macroblock may select its partition from one of seven modes – 16x16, 8x16, 16x8, 8x8, 8x4, 4x8 and 4x4 – to allow accurate motion alignment. Furthermore, the rate-distortion optimization criterions are proposed to select motion mode, motion vectors and partition mode. In addition, an overlapped block motion alignment is proposed to suppress the energy in spatial high-frequency subbands and facilitate the coding of temporal high–pass residual pictures. The experimental results show the proposed techniques can improve coding efficiency up to 1.0dB in 3D wavelet video coding. Our new 3D wavelet video scheme outperforms MC-EZBC by 1~2dB for most of test sequences and is very competitive with the best results of H.264. It even outperforms H.264 by 1.1dB for test sequence such as *Coastguard*.

In the future works, we try to conduct some researches to further optimize the current motion alignment scheme. The selection of motion mode, motion vectors and partition mode at each macroblock may be optimized by finding a better measurement of prediction error. In addition, adaptive temporal and spatial de-composition is another way to efficiently present the characteristic of local signal. The number of de-composition layers, the wavelets filters used at each layer and even the lifting structure at each pixel can be adjusted accordingly.

ACKNOWLEDGEMENTS

The proposed techniques in this paper are based on the previous works from Lin Luo and Jizheng Xu. The authors would like to thank them for providing their source codes and sharing many heuristic discussions on 3D wavelet video coding.

REFERENCES

[1] ISO/MPEG, “Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbps”, ISO/IEC 11172-2,1992

[2] ISO/MPEG, “Generic coding of moving pictures and associated audio information”, ISO/IEC 13818-2,2000

[3] “MPEG-4 Video Verification Model Version 13.0”, ISO/IEC JTC 1/SC29/WG11 N2687, March,1999

[4] JVT, “Joint Final Committee Draft (JFCD) of Joint Video Specification”, document JVT-D157, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Klagenfurt, July 2002.

[5] D. Taubman and A. Zakhor, “Multirate 3-D subband coding of video”, IEEE Trans. Image Processing, vol. 3, no. 5, pp. 572-588, Sept. 1994.

[6] A. Wang; Z. Xiong; P.A. Chou.; S. Mehrotra, “Three-dimensional wavelet coding of video with global motion compensation”, Proceedings. DCC '99, 1999. Page(s): 404 –413.

[7] J.-R. Ohm, “Three Dimensional Subband Coding with Motion Compensation,” IEEE Trans. on Image Processing, vol. 3, no. 5, pp. 559-571, Sept. 1994.

[8] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, “Three-dimensional embedded subband coding with optimized truncation (3D ESCOT)”, Applied and Computational Harmonic Analysis10, pp.290-315, 2001.

[9] P. Chen, and J. W. Woods, “Improved MC-EZBC with quarter-pixel motion vectors”, JVT proposal, ISO/IEC JTC1/SC29/WG11, MPEG2002/M8366, Fairfax, VA, May 2002.

[10] L. Luo, F. Wu, S. Li, Z. Zhuang, “Advanced lifting-based motion threading technique for 3D wavelet video coding,” Proc of SPIE VCIP2003, vol.5150, pp.707-718, Jul.2003.

[11] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, ``Memory-constrained 3-D wavelet transform for video coding without boundary effects," IEEE Trans. Circuits and Systems for Video Tech., vol. 12, pp. 812-818, September 2002.

[12] I. Daubechies and W. Sweldens, “Factoring wavelet transforms into lifting steps”, J. Fourier Anal. Appl., vol. 4, pp. 247-269.

[13] M.T. Orchard and G.J. Sullivan, “Overlapped block motion compensation: an estimation-theoretic approach,” IEEE trans. Image Processing, vol.3, pp.693-699, Sep. 1994.

[14] P.Chen, and J.W.Woods, “Exploration Experimental Results and Software”, JVT proposal, ISO/IEC JTC/SC29/ WG11, MPEG2002/M8524, Klagenfurt, AT, July 2002.

[15] ITU-Telecom Standardization Sector VCEG, “H.26L Test Model Long Term Number 7 (TML-7) dratf0”,VCEG-M81, May 2001.

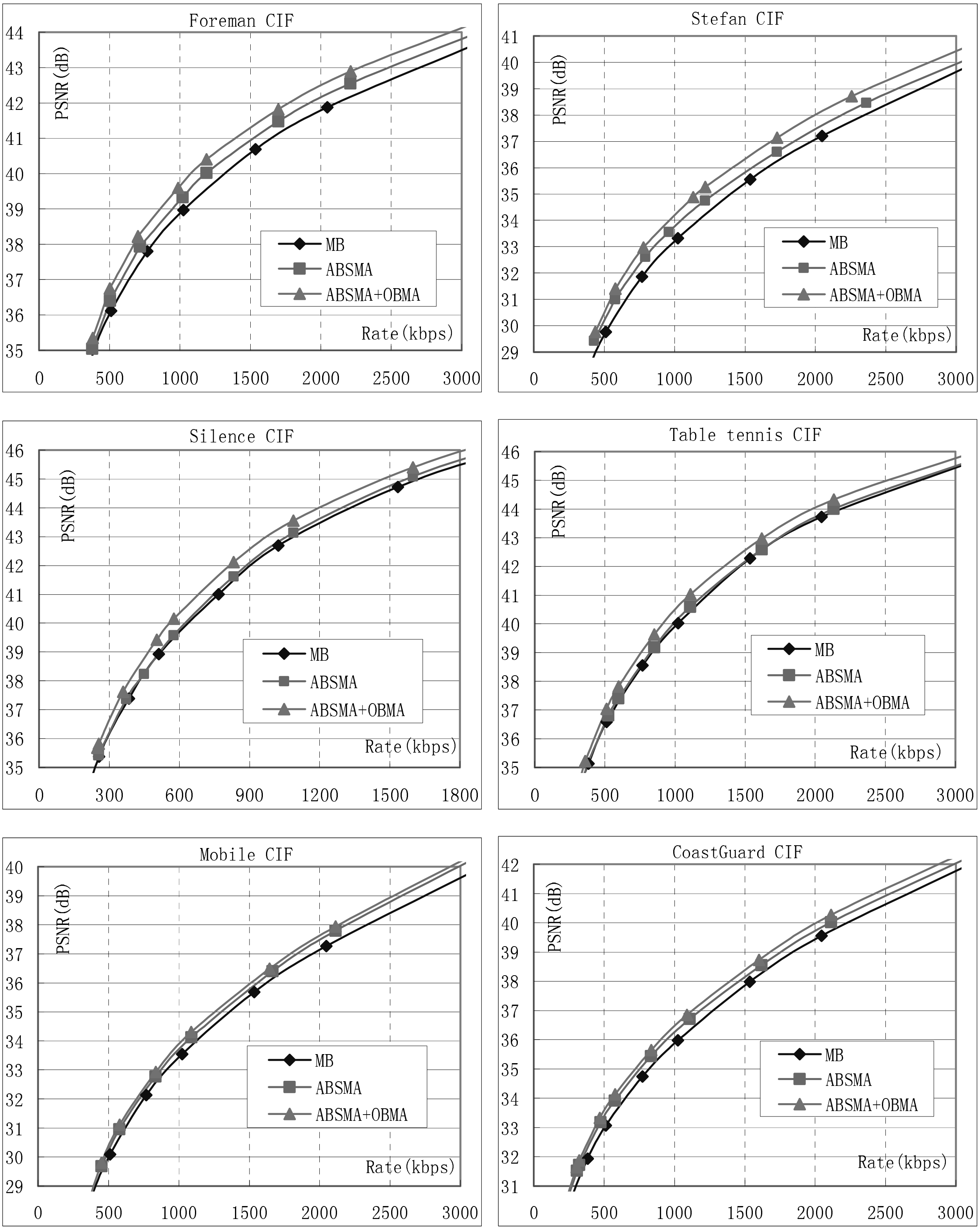

Figure 8: The performance evaluations of each proposed technique.

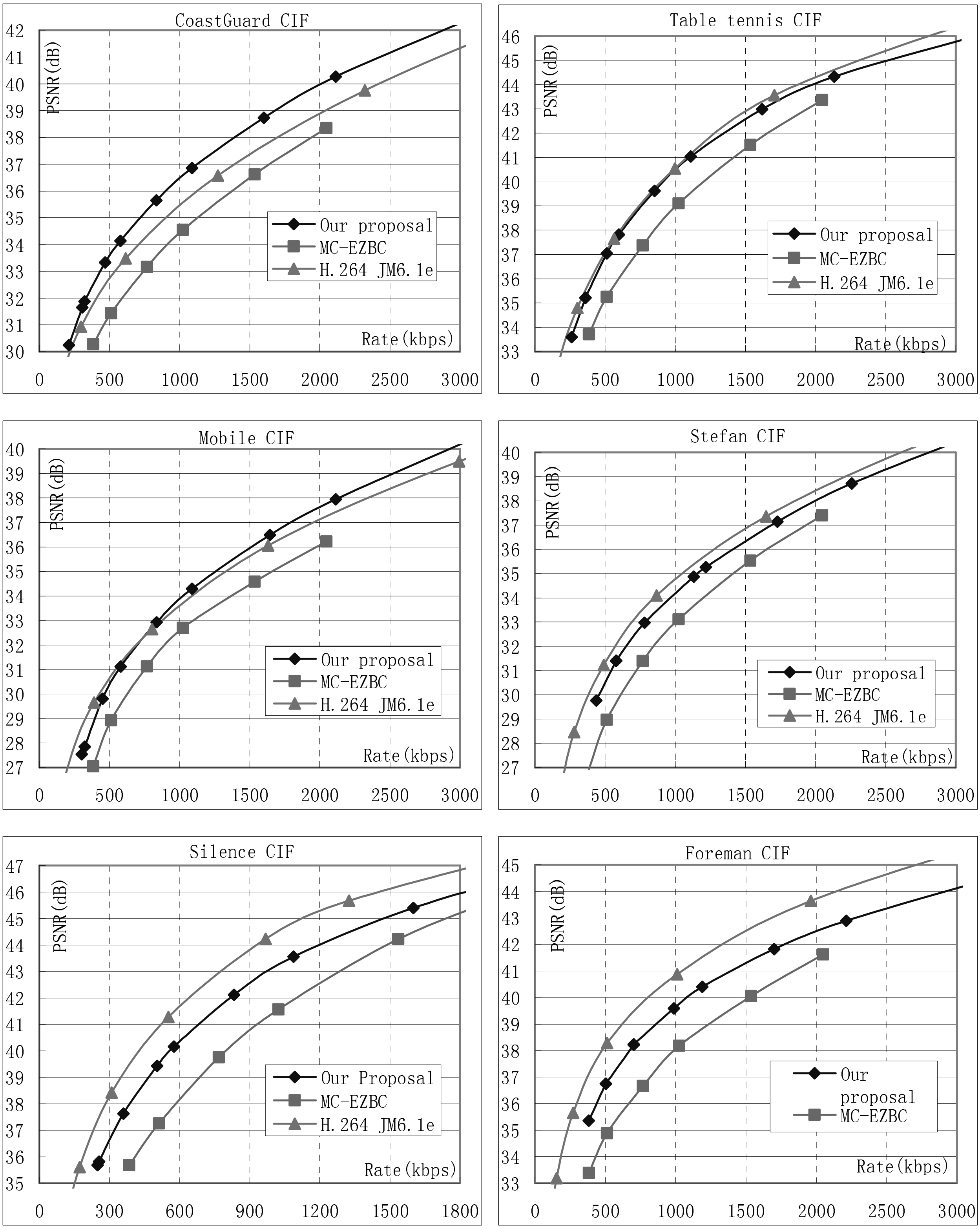

Figure 9: The performance comparisons between MC-EZBC, H.264 JM6.1e and the proposed scheme.