Optimal Compression Plane for Efficient Video Coding

Anmin Liu, Weisi Lin*, Manoranjan Paul, Fan Zhang, Chenwei Deng

***Abstract*—All existing video coding standards developed so far deem video as a sequence of natural frames (formed in the XY plane), and treat spatial redundancy (redundancy along X and Y direction) and temporal redundancy (redundancy along T direction) differently and separately. In this paper, we investigate into a new compression (redundancy reduction) method for video in which the frames are allowed to be formed in a non-XY plane. We are to exploit fuller extent of video redundancy, and propose an adaptive optimal compression plane (OCP) determination process to be used as a preprocessing step prior to any standard video coding scheme. The essence of the scheme is to form the frames in the plane formed by two axes (among X, Y and T) corresponding to signal correlation evaluation, which enables better prediction (therefore better compression). In spite of the simplicity of the proposed method, it can be used for both lossless and lossy compression, and with and without inter-frame prediction. Extensive experimental results show that the new coding method improves the performance of the video coding, for a number of coding methods (inclusive of lossless and near-lossless Motion JPEG-LS, Motion JPEG, Motion JPG2K, H.264 intra-only profile and H.264) and videos with different visual content.**

***Index Terms*—Optimal Compression Plane (OCP), video coding, video compression, preprocessing, lossless compression.**

I. Introduction

Digital video signals in raw formats require excessive storage capacity and huge transmission bandwidth, and therefore there is a need to reduce the data rate (i.e., via coding) before digital video can be integrated into the existing and emerging communication systems. Besides the requirement of real-time processing, the goal of video coding is to ensure good video quality within the provision of transmission and storage. Complexity, distortion and bitrate are factors that measure success of a video coding scheme. These factors are usually measured by computational time, PSNR (Peak Signal Noise Ratio) and bpp (bits per pixel), respectively.

If V_{XYT} represents a video sequence with axes of X, Y and T, it is usually deemed as successive natural images as:

$$V_{XYT} = \{ I_{XY}(t),\ t = 1, 2, 3\ldots \} \quad (1)$$

where I_{XY} represents a natural image with axes of X and Y. Therefore, video coding is an extension of image coding. There are two types of techniques for such extension:

- *With inter-frame prediction:* This type of extension is usually by reducing the temporal redundancy among successive natural frames prior to intra-frame coding with image coding techniques. The most commonly used technique for inter-frame prediction is Block based Motion Estimation (BME) [17-20]. In BME, each frame is divided into blocks of pixels (e.g. macro-blocks of 16×16 pixels in MPEG) and each block in the current frame is predicted from a block of equal size in the reference frame. The offset between the two blocks is known as a motion vector. The error between the current block and the similar block in the reference frame is encoded and transmitted along with the motion vector for the block. To exploit the redundancy between neighboring block vectors (e.g. for a single moving object covered by multiple blocks), it is common to encode only the difference between the current and previous motion vectors in the bit-stream.
- *Without inter-frame prediction:* In some cases, to save the power of the cell or when the processor's computational resource is limited, BME is impossible due to its high complexity (e.g., the computation complexity of BME varies from 50% to 90% of a typical video coding system [21-22].) Therefore, each frame would be coded independently by using image coding techniques [23-28]. The lack of use of inter-frame prediction results in reduction of compression capability, but robustness to error. It is often used in mobile appliances because it is with low processing requirement, ease of implementation, and broad compatibility; it is also used in the case when the zero-delay feature is required.

Both of the two types of extension techniques above have considered the physical meaning of each axis, they treat X and Y axes equally with each other (as spatial axes) and differently with T axis (as temporal axis). In this research, we propose a novel framework of preprocessing for video coding which is different with the existing paradigms by exploring the information redundancy in a fuller extent. Rather than explicitly distinguishing T axis as temporal axis, our scheme ignores the physical meaning of X, Y and T axes (somewhat similar with 3D transform [29-30]) and focuses on the amount

Manuscript received October 13, 2009; revised September 3, 2010 and March 6, 2011. This work was supported in part by MoE AcRF Tire 2 Grant (T208B1218), Singapore.

The authors (*corresponding author) are with School of Computer Engineering, Nanyang Technological University, Singapore, 639798 (e-mail: liua0002@ntu.edu.sg; wslin@ntu.edu.sg; M_PAUL@ntu.edu.sg; zhangfan.hust@gmail.com; CWDENG@ntu.edu.sg).

of video redundancy along each axis (more specifically, on the correlation coefficient (CC) along each axis). The key part of the proposed framework is an adaptive Optimal Compression Plane (OCP) determination preprocessing module. Different to the traditional XY compression plane, we form frames in the adaptively determined OCP; then, a standard coding scheme is used to better remove the redundancy. In our initial work [1], we have shown with initial result that OCP can be used with JPEG-LS for lossless video coding. In this paper, we will further justify the scheme and extend the concept of OCP to many lossy video coding techniques. We also study the distribution of the prediction error in different compression planes and for different compression techniques. The major characteristics of the research reported in this paper are: 1) a new coding concept based upon the automatic OCP decision is demonstrated; 2) consistent improvement of Rate-Distortion (RD) performance can be achieved by using OCP; 3) the proposed scheme can be used with the existing standard video compression codecs (encoders and decoders) since the required preprocessing is independent of the video coding scheme to be used; 4) the additional computational complexity of this scheme is minor because the required operation is only the calculation of the correlation coefficients (CCs), and as shown later, CCs can be calculated with sampled frames. With experimental results, we confirm that the proposed framework can achieve better RD performance. Moreover, evaluations on the performance of various coding techniques provide more insight into the behavior of the proposed framework.

Although there have been some preprocessing methods [39-41] for image/video coding, none of these existing methods has explored the similar problem of this work. In [39, 40], down-sampling and pre-filtering methods for image coding were proposed based on the correlation/characteristics (e.g., direction of edge) within the image frame. In [41], the perceptual redundancy of videos has been explored by modifying the motion compensated residuals to reduce their variation (for better compression) under the constraint of JND (Just-Noticeable-Distortion).

The rest of this paper is organized as follows. In Section II, we analyze video redundancy in terms of CCs. We compare the redundancy along different axes (X, Y and T), for different visual content. Section III describes the details of the proposed OCP based video coding scheme. In Section IV, we discuss how to determine the OCP with and without inter-frame prediction. The experimental results with further discussion are given in Section V. We validate our OCP framework with different visual content. Consistent improvement is achieved compared with standard Motion JPEG-LS [3-6], Motion JPEG [7-9, 23-24], Motion JPG2K [10-12, 25-26], H.264 intra-only profile [27-28] and H.264 [13-16] (inclusive of the cases with B pictures and two reference frames). Finally, conclusions are drawn in Section VI.

II. Video Redundancy Analysis

A video sequence can be described as a 3D cube with axes of X, Y, and T. The amount of statistical redundancy along one axis can be estimated by the average of CCs between frames formed by the two remainder axes. The bigger the average CC is, the more the statistical redundancy exists. Let us use $p_k(i, j)$ to represent the pixel to be encoded, located at (i, j) in the k^{th} frame, then the inter-frame CC between the k^{th} and the $(k+l)^{th}$ frames can be calculated as [2]:

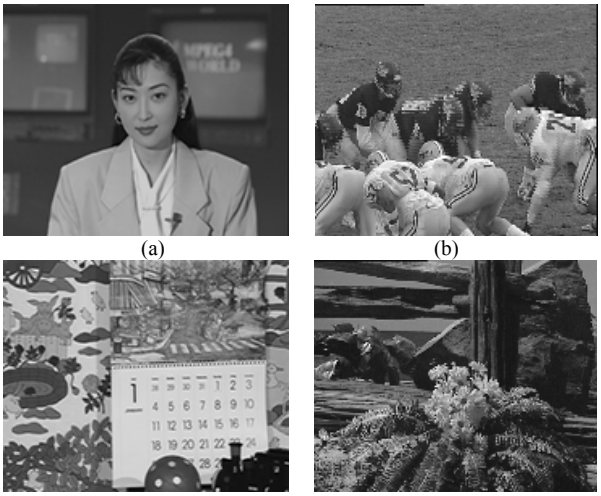

Fig. 1. Four video sequences with different typical motion characteristics. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

$$CC_{k,k+l} = \frac{\sum_{i,j}((p_k(i,j)-\overline{p_k})\cdot(p_{k+l}(i,j)-\overline{p_{k+l}}))}{\sqrt{\sum_{i,j}(p_k(i,j)-\overline{p_k})^2\cdot\sum_{i,j}(p_{k+l}(i,j)-\overline{p_{k+l}})^2}} \quad (2)$$

where l =0, 1, 2…, and $\overline{p_k}$ and $\overline{p_{k+l}}$ are the average values of the pixels in the k^{th} and the $(k+l)^{th}$ frames, respectively.

A. Temporal Redundancy

Experiments showed that temporal redundancy usually decreases slowly with time (i.e., redundancy along T axis), being still significant on average even between frames separated by ten or more frames [31] (except for the occurrence of scene change).

Fig. 1 has shown four publicly available video sequences with different typical motion characteristics: "Akiyo" (a talking head), "Football" (fast motion), "Mobile" (horizontal, vertical and rotational object motion coupled with camera motion), and "Tempete" (zooming out). All of them are with luminance only and QCIF (Quarter Common Intermediate Format) resolution (176×144), and from the 1st to the 128th frames. Fig. 2 illustrates the inter-frame CCs for them. Fig. 2(a) shows the CCs between the k^{th} and the $(k+1)^{th}$ frames and Fig. 2(b) denotes the CCs between the 1st and the kth frames. It is observed from Fig. (a) that all the sequences except for the "Football" sequence have very high CC (close to 1) between two adjacent frames while the "Football" sequence has relatively lower (around 0.5 for most of the frames) inter-frame

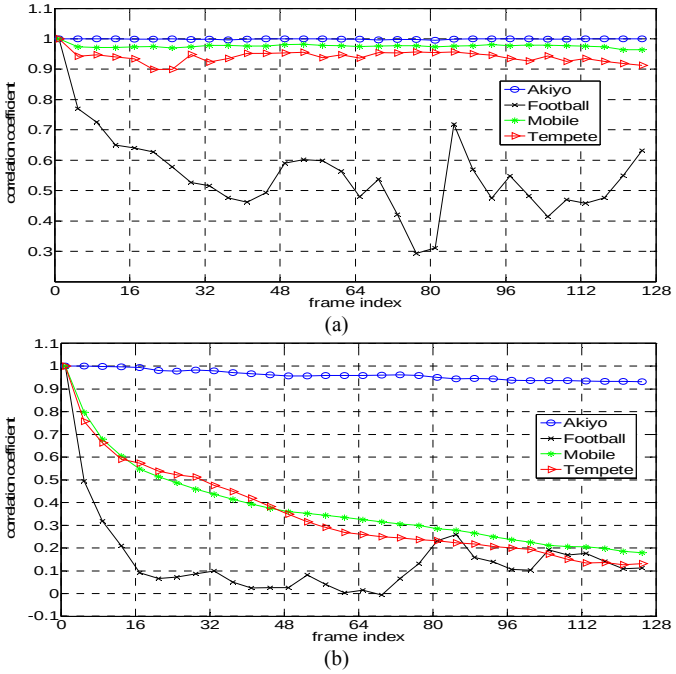

Fig. 2. Inter-frame correlation coefficients of four typical sequences. (a) Correlation coefficients between successive frames, (b) Correlation coefficients between frames separated by multiple frames.

CCs. It can be noticed from Fig. (b) that CC between frames separated by ten frames (i.e., the 1st and the 11th frames) are still very high (i.e., higher than 0.6 for all the sequences except for the "Football" sequence). It is apparent that there is a potential gain if more than one past reference frames are used for prediction, and this is the ground for video coding with multiple reference frames [35-37].

B. Comparison of Redundancy along Different Axes

We have discussed the redundancy along the T axis (i.e., temporal redundancy) in the previous subsection, and now we consider the redundancy along X and Y axes. As shown in (1), pixels in a video sequence are traditionally grouped into frames in the XY plane as I_{XY} at first, and then frames are grouped into a 3D matrix along the T axis. However, a video sequence can be also grouped as frames in the TX plane (as in (3) below) or the TY plane (as in (4) below):

$$\{ I_{TX}(y),\ y=1, 2, 3\ldots\} \tag{3}$$

$$\{ I_{TY}(x),\ x=1, 2, 3\ldots\} \tag{4}$$

C_T, C_X and C_Y are used to represent the amount of correlation (and therefore redundancy) measured along T, X and Y axes; they can be estimated by averaging inter-frame (formed in the other axes) CCs, and mathematically described as:

$$C_d = \sum_{k=2,3,\ldots L^d} CC^d_{(k-1),k} / L^d \tag{5a}$$

where $d \in \{T, X, Y\}$, $CC^d_{(k-1),k}$ and L^d is $CC_{(k-1),k}$ and the number of frames when formed in the axes other than d.

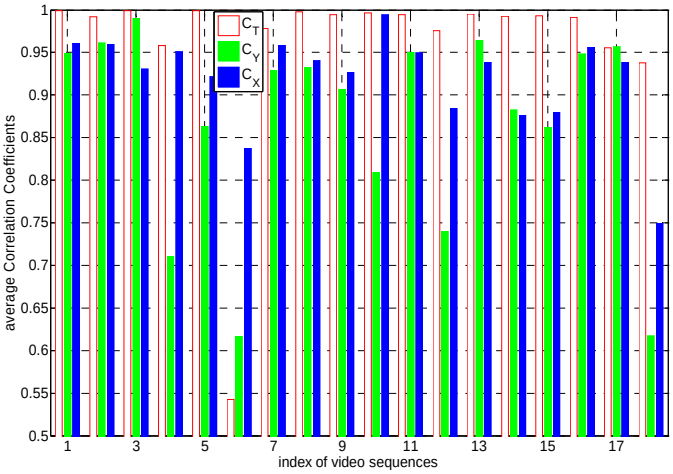

Fig. 3. Average inter-frame correlation coefficients along T, Y and X axes.

Table I
The names and indices of the video sequences

Index	Sequence name	Index	Sequence name
1	Akiyo	10	Highway
2	Carphone	11	Miss-America
3	Claire	12	Mobile
4	Coastguard	13	Mother-daughter
5	Container	14	News
6	Football	15	Salesman
7	Foreman	16	Silent
8	Grandma	17	Suzie
9	Hall	18	Tempete

Table II
Relationship among C_d

Label	Relationship	Label	Relationship
R_1	$C_T \geq C_X \geq C_Y$	R_4	$C_X \geq C_Y \geq C_T$
R_2	$C_T \geq C_Y \geq C_X$	R_5	$C_Y \geq C_X \geq C_T$
R_3	$C_X \geq C_T \geq C_Y$	R_6	$C_Y \geq C_T \geq C_X$

Fig. 3 has shown C_d for 18 sequences with QCIF resolution (their names and indices are shown in Table I). All the sequences are with luminance only, and from the 1st to the 128th frames. Note that Fig. 2 (a) is the CCs between successive frames for four of the sequences when $d = T$, and Fig. 3 gives the average CCs along different axes for each sequence. It can be observed from Fig. 3 that for all the sequences except for "Football" (with Index 6) the redundancy along T axis are much more than that along X and Y axes.

C. C_d Calculation using Sampled Frames

The calculation of C_d (as given in (2) and (5a)) takes some time since all the pixels in the video are involved in the calculation, and to reduce the computational complexity, we explore the possibility of using sampled frames to calculate C_d (C_T, C_X and C_Y). There are 6 types of relationship among C_d, and R_i (i=1, 2…6) are used to represent different relationship of C_d, as shown in Table II.

Table III
Relationship of C_d with different S conditions (shaded: S conditions with different calculated relationship of CCs compared with that of S=1)

Videos sequence	S			
	1	1/2	1/4	1/8
1	R_1	R_1		R_2
2	R_2	R_2		
3	R_2	R_2		
4	R_1	R_1		
5	R_1	R_1		
6	R_4	R_4		
7	R_1	R_1		
8	R_1	R_1	R_2	R_1
9	R_1	R_1		
10	R_1	R_1		
11	R_2	R_1		
12	R_1	R_1		
13	R_2	R_1		
14	R_2	R_1		
15	R_1	R_1		
16	R_1	R_1	R_2	
17	R_2	R_2		
18	R_1	R_1		

C_d for sampled frames is calculated as:

$$C_d = \sum_{k=\frac{1}{S},\frac{2}{S},\cdots,\frac{L_S^d}{S}} CC_{(k-1),k}^d / L_S^d \qquad (5b)$$

The frame sampling ratio S is defined as:

$$S = L_S^d / L^d \qquad (6)$$

where L_S^d is the number of sampled frames within a Pre-Processing Unit (PPU, which is the collection of frames being considered; more details and discussion to be presented in Section III) when formed in the axes other than d.

The experimental results of the relationship of C_d under different S conditions are shown in Table III. In the table, the sampling in most cases (13 out of 18) yields the same result with the exhausted evaluation (i.e., when S=1). From Table III, we see that sampled frames can be used to determine the relationship of C_d for a sequence, because the results with frame sampling (S<1) are largely the same as those without sampling (S=1). Note that even when S=1/8 (the last column of Table III), only 4 out 18 sequences obtain different relationship compared with that of S=1 (the second column of Table III), and the four mismatched video sequences (indexed as 1, 11, 14, 16) have very similar C_X and C_Y; so the mismatch between R_1 and R_2 would not affect the performance too much in terms of compression efficiency since the compression performance is closely related to C_d (as to be discussed in Section III.A).

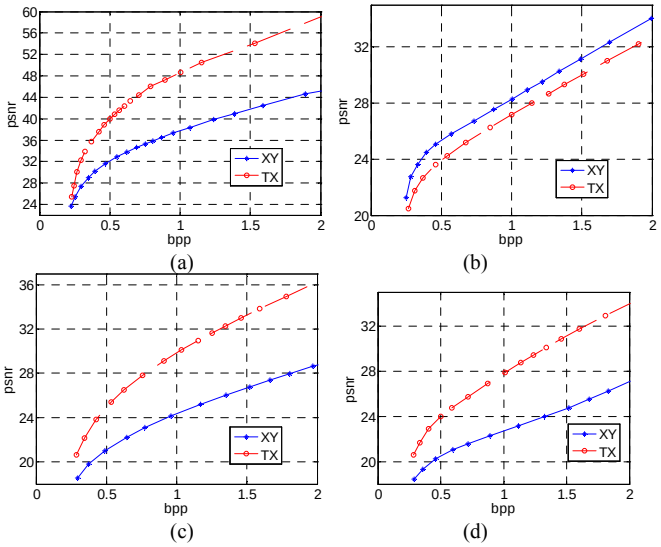

Fig. 4. Rate-Distortion performance for sequences with different frame formation. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

III. Proposed Framework for Efficient Video Coding

A. Optimal Coding Plane and C_d

The intra-frame redundancy can be measured by C_A and C_B for a frame formed in AB plane ($A,B = X,Y,T$): the larger the values of C_A and C_B are, the higher the intra-frame redundancy exists; as to be discussed hereafter, the higher the RD performance will be with AB plane without inter-frame prediction. In Fig. 4, we have demonstrated the RD curves for video sequences with different visual content and frame formation. Each frame is coded independently by JPEG. From Fig. 3 and Fig. 4 we can see that the RD performance is in accordance with the C_d criteria (the larger C_A and C_B are, the higher the RD performance is achieved when forming frames in AB plane). For example, for "Mobile" (with Index 12) video, the RD performance in the TX plane is better than that of the XY plane since C_T is significantly bigger than C_Y for this sequence (as shown in Fig. 3). Therefore, we can form frames in a non-XY plane before we compress each frame because C_X and C_Y may not the largest two among the three values of C_d

B. Proposed Framework

As shown in Fig. 5, our OCP-based video coding frame work consists of selection of appropriate Pre-Processing Unit (PPU, its size, i.e., number of frames in the XY plane is denoted as N), adaptive OCP decision, and video coding with a standard compression method. In each PPU, we may form frames in the XY, TX or TY plane. It is obvious that we need 2 bits of overhead to represent the three possible coding planes for each PPU, and this bit overhead is included in our rate calculation throughout this work.

The insight observation for this proposed framework is that the sampling processing is as usual but we ignore the physical meaning of a video during the video coding processing for better coder performance. People deem video as successive

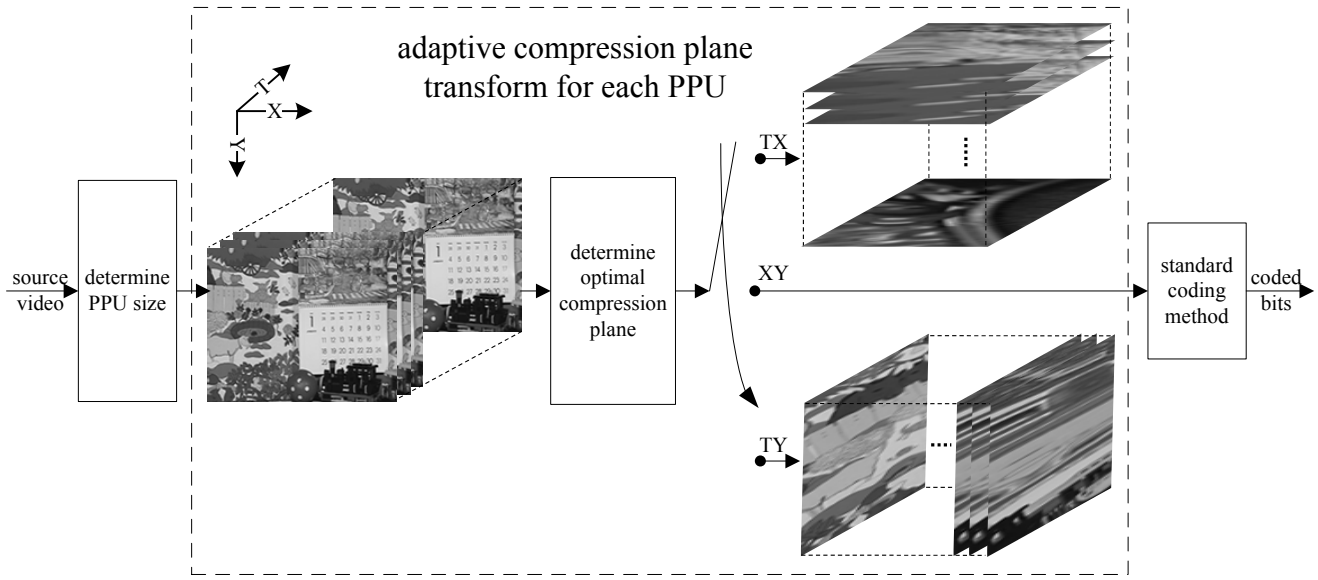

Fig. 5. Block diagram of the proposed scheme (illustrated with XY and non-XY frames of "Mobile" video sequence for better visual impression).

natural image frames, which are formed in the spatial domain and with clear physical meaning (e.g., a natural scene). However, in the sense of data structure, video is nothing more than a three dimensional data matrix, and the distinction among X (a spatial dimension), Y (the other spatial dimension), and T (the temporal dimension) is not absolutely necessary, in the viewpoint of compression. The adaptive plane selection according to video content makes sense and the coding system benefit, as will be demonstrated in this paper.

C. Impact of PPU Size

With the PPU in our design, there is an initial coding delay since we can only start coding after the first PPU rather than the first frame, and this is somewhat like the method with hierarchical B pictures [32] where the encoding delay is the GoP (Group of Pictures) size minus one. Therefore, our OCP based coding method works well for video transmission or storage but can not be used for interactive applications. Larger PPUs could be more efficient for exploring the temporal redundancy (this point will be demonstrated in Fig. 9 and discussed toward Fig. 10~Fig. 12 in Section V.C); while the drawbacks of larger frame memory and longer initial coding delay will surface. A larger PPU is also more likely to encounter scene change among the frames. One possible good solution is to conduct scene detection on the video sequence and set each scene as a PPU.

D. Computational Complexity Analysis

Since the proposed OCP is a preprocessing step prior to the actual coding, some additional computational complexity is needed to do the pre-analysis. However, this computational overhead is very small, since what we need to do is just to calculate C_d by using some sampled frames (as already presented in Section II.C). For instance, the computational complexity overhead is only about 2.58% and 0.22% compared with Motion JPEG-LS and H.264 when sampling ratio S is 1/8, on a PC with 2.40 GHz Intel Core2 CPU and 2 GB of RAM.

Table IV
Bits per pixel (bpp) for lossless compressed videos under different frame formation

Index	OCP	bpp			Saving (%)	
		XY	TX	TY	P_{min}	OCP
1	TX	3.205	1.598	1.622	50.1	
2	TY	3.619	3.455	3.537	4.5	2.3
3	TY	2.479	1.926	1.900	23.4	
4	TX	5.058	4.556	5.388	9.9	
5	TX	4.234	2.334	2.507	44.9	
6	XY	5.233	5.561	6.501	0.0	
7	TX	4.492	4.126	4.281	8.2	
8	TX	3.922	2.637	2.608	33.5	32.8
9	TX	3.899	3.136	3.156	19.6	
10	TX	3.518	3.239	3.590	7.9	
11	TY	2.698	2.572	2.550	5.5	
12	TX	5.925	5.040	5.263	14.9	
13	TY	3.490	2.592	2.564	26.5	
14	TY	3.845	2.150	2.149	44.1	
15	TX	4.487	2.567	2.861	42.8	
16	TX	4.413	2.597	2.566	41.9	41.1
17	TY	3.613	3.450	3.265	9.6	
18	TX	6.220	5.550	5.845	10.8	
mean	--	4.130	3.283	3.453	22.1	21.9

IV. OCP Determination

The most accurate determination of coding plane should adopt a brute-force method (i.e., to form frames in all three possible coding planes, code the video and select the best coding plane according to a joint rate-distortion criterion), but its computational complexity is about three times of the standard coding schemes. Therefore, we are to propose effective and efficient OCP determination methods.

A. OCP without Inter-frame Prediction

When there is no inter-frame prediction (i.e., coding each frame independently when the computational resource is limited), redundancy can be only reduced by intra-frame coding [23-28], the axes with bigger redundancy (i.e., larger C_d) should be the intra-frame directions, and therefore, the OCP is determined as:

$$OCP = \begin{cases} XY & if\ \min\{C_T, C_X, C_Y\} = C_T \\ TX & \min\{C_T, C_X, C_Y\} = C_Y \\ TY & \min\{C_T, C_X, C_Y\} = C_X \end{cases} \quad (7)$$

In Table IV, we have shown the OCP determined by using (7) for video sequences in Table I.

To demonstrate the effectiveness of (7), we have also shown the bpps for lossless compressed videos with different frame formation (XY, TX, and TY); each frame is coded independently by JPEG-LS [1, 3, 6]; P_{min} represents the plane with minimal bpp. The saving of bits is defined as:

$$\text{Saving} = (\text{bpp(XY)}-\text{bpp}(q))/\text{bpp(XY)} * 100\% \quad (8)$$

where q = OCP or P_{min}. Apparently, the ideal OCP is P_{min}. From Table IV we can see that there are only three out of the 18 sequences having OCP different from P_{min}; note that the difference between bpp(P_{min}) and bpp(OCP) for these three sequences are less than 2.5%, and the average saving of bits for P_{min} and OCP is very close (being 22.1% and 21.9%, respectively). Therefore, (7) is effective for OCP determination.

Fig. 6 and Fig. 7 further demonstrate the effect of compression for the OCP, with the histogram of the intra-frame prediction residues (by JPEG-LS) and the histogram of the DCT (Discrete Cosine Transformation) coefficients (by JPEG) for the four typical sequences in XY plane and TX one (note that the OCPs for the four typical sequences except for "Football" are all TX), respectively. We can observe from these two figures that, for the test sequences with TX as its OCP, the number of residues/coefficients with large magnitude decreases for TX plane, while the number of residues/coefficients with zero or small magnitude increases (i.e., the histogram for TX plane is sharper than that for the XY plane). For JPEG-LS, smaller prediction residues mean that we need fewer bits to code them; for DCT based image/video coding, smaller magnitude coefficients mean more zeros after quantization (with approximately the same quantization level and quantization error for different compression plane), and fewer bits needed for run length coding and entropy coding.

B. OCP with Inter-frame Prediction

With the XY plane for inter coding (i.e., the standard H.264), BME is very effective and most frames are inter-coded. For a non-XY plane for inter coding, BME still brings about some benefit (although BME does not carry its original physical meaning in this case) but is not as effective as for the XY plane; consequently, most "frames" are intra-coded. To confirm this point, we have shown the percentage of intra modes for P frames when coded in XY, TX and TY planes for "Akiyo" sequence in Fig. 8. From the figure we can see that the percentage for intra mode is more than 50% for non-XY planes.

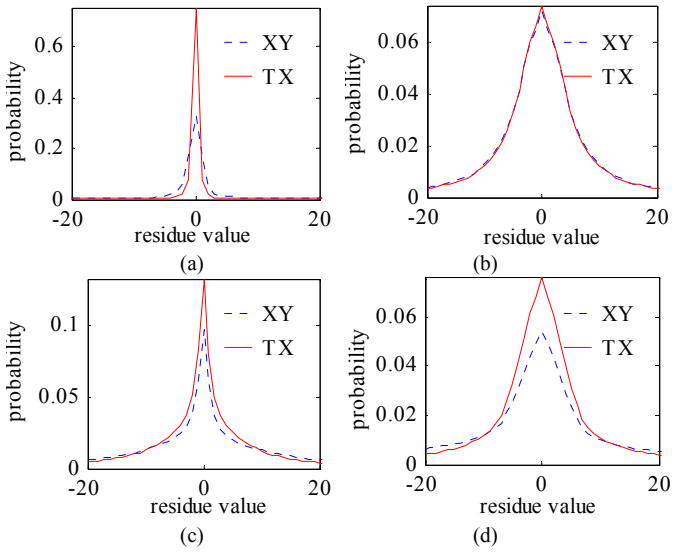

Fig. 6. Distribution of the intra-frame prediction (JPEG-LS) residues. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

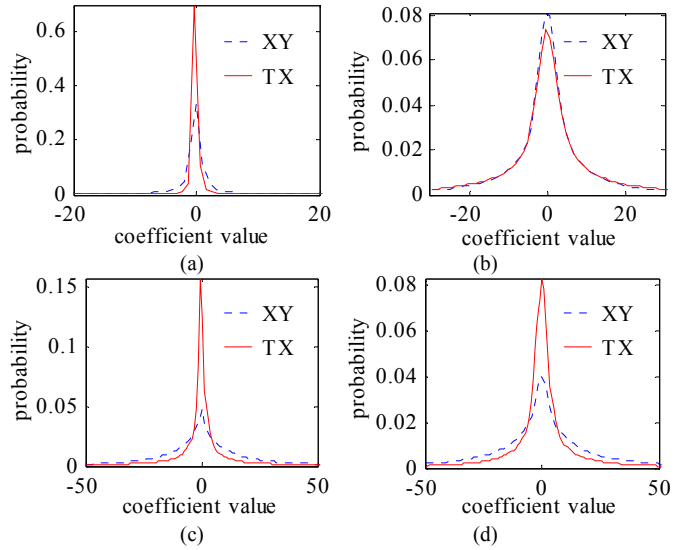

Fig. 7. Distribution of the DCT (Motion JPEG) coefficients. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

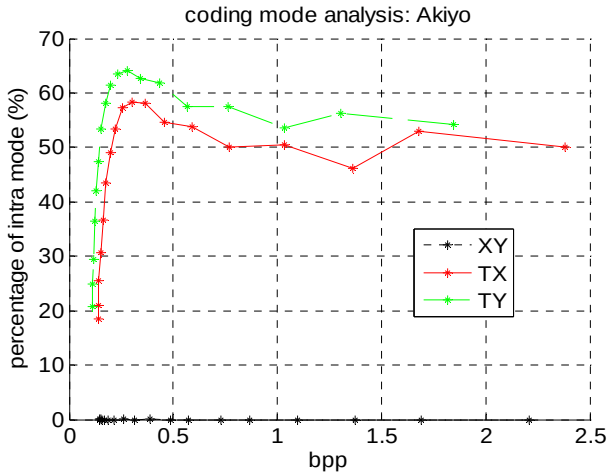

Fig. 8. Percentage of intra modes for H.264 coding in XY, TX and TY planes.

Therefore, two major factors to determine the overall performance comparison of the two schemes are the coding efficiency of BME in the XY plane and the coding efficiency of intra-coding in a non-XY plane. The former is good when motion vectors can be estimated well, while the latter is usually good (as shown in the results of intra-only video coding) and even better with video of low motion (see the comparison of Akiyo against Football in Fig. 4).

We modified (7) and add $C_T - \max\{C_X, C_Y\} > T_C$ into the criteria of selection a non-XY plane as OCP. Here T_C (chosen as 3.5×10^{-2} in the current work) is a constant threshold which accounts for the advantage of BME in the XY plane; this makes TX or TY plane only to be chosen when C_T is sufficiently high. Therefore, the OCP with inter-frame prediction is formulated as:

$$OCP = \begin{cases} TX, \text{ if } C_X \geq C_Y \text{ and } C_T - \max\{C_X, C_Y\} > T_C \\ TY, \text{ if } C_X < C_Y \text{ and } C_T - \max\{C_X, C_Y\} > T_C \\ XY, \text{ otherwise} \end{cases} \quad (9)$$

As can be seen from (9), there is a higher chance for the OCP to choose a non-XY plane when T_C is smaller.

V. Overall Experiments and Results

We have already discussed the determination of OCP with and without inter-frame prediction in Section IV. In this section, we will provide extensive experimental results to evaluate the overall effectiveness of OCP with a number of video coding techniques ranging from lossless to lossy coding.

In Part A of this section, we show the results for OCPs with and without inter-frame prediction. In Part B, the performance of OCP with Motion JPEG-LS for lossless and near-lossless video coding is firstly shown. For low complexity lossy video coding, the results of Motion JPEG [23-24], Motion JPG2K [25, 26] and H.264 intra-only profile [27, 28] with OCP are shown in Part C. Finally, we will present the results of OCP with H.264 in Part D. The same video database with 18 sequences as introduced in Table I is used to demonstrate the effectiveness and efficiency of allowing TX and TY plane options, with an addition of the "Tempete" sequence in SD resolution in Part D. For all the video sequences to be used in this section, only the first 128 luminance frames (XY plane) are used.

A. OCP under Different Conditions

With aforementioned video sequences, there is only one PPU for each sequence if N =128 and four PPUs if N =32 (PPU1 to PPU4 represent the first to the fourth PPUs, respectively). In Table V, we show the OCP for each PPU of all 18 sequences (as listed in Table I) with and without inter-frame prediction.

As we can see from Table V, the OCP for different PPUs (when N =32) is largely the same, and this represents the cases where there is no obvious scene change. The only exception is "Miss-America" (with Index 11), and the slightly different OCP results are caused by the different motion in PPU1/PPU2/PPU4 (only with facial movement) and PPU3 (there being also some shoulder movement).

Table V
OCPs without (with) inter-frame prediction

Index	N =32				N =128
	PPU1	PPU2	PPU3	PPU4	
1	TX (TX)				TX (TX)
2	TY (XY)				TY (XY)
3	TY (XY)				TY (XY)
4	TX (XY)				TX (XY)
5	TX (TX)				TX (TX)
6	XY (XY)				XY (XY)
7	TX (XY)				TX (XY)
8	TX (TX)				TX (TX)
9	TX (TX)				TX (TX)
10	TX (XY)				TX (XY)
11	TY (TY)	TY (TY)	TY (TY)	TY (TY)	TY (TY)
12	TX (TX)				TX (TX)
13	TY (TY)				TY (TY)
14	TY (TY)				TY (TY)
15	TX (TX)				TX (TX)
16	TX (TX)				TX (TX)
17	TY (XY)				TY (XY)
18	TX (TX)				TX (TX)

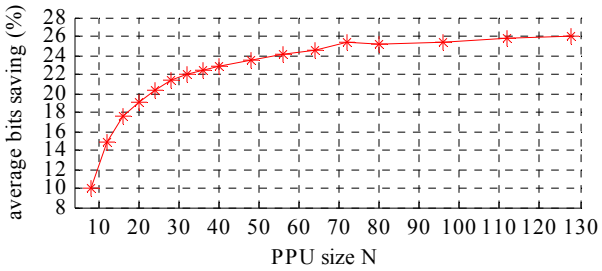

Fig. 9. Average saving of bits vs. Pre-Processing Unit (PPU) size.

Nearly all the video sequences have a non-XY OCP for the case without inter-frame prediction since temporal redundancy can be explored when coding performs in a non-XY plane. For the case with inter-frame prediction, OCPs for high motion sequences switched to the traditional XY plane since BME is more efficient to reduce the temporal redundancy than intra-frame prediction in TX (or TY) plane, as already analyzed in Section IV.B.

B. Performance Comparison of OCP with Motion JPEG-LS

In Section IV we have shown with Table IV the performance of lossless coding in OCP (in terms of bpp saving of the proposed method with respect to Motion JPEG-LS), and it can be seen that the proposed scheme outperforms Motion JPEG-LS by 21.9% on average.

Fig. 9 has shown the average (over the same 18 sequences) saving of bits versus the PPU size [1]. From the figure, we can see that PPU size N =32 is a good choice considering the trade-off among compression efficiency, memory requirement and initial coding delay, since the RD performance improvement saturates when N >32.

Now we come to near-lossless video coding using Motion JPEG-LS. The results shown in Table VI present the coding performance (N =32) over each video sequence. "Near" is one input parameter for JPEG-LS, representing the allowed maximal distortion for each pixel (e.g., Near=1 means the maximal distortion for each pixel is 1). The definition of the

saving of bits is the same as defined in (8), and the PSNR difference dP of the OCP is defined as:

$$dP = (PSNR_{OCP}-PSNR_{XY})/ PSNR_{XY} * 100\% \quad (10)$$

From the table we can see that when coding in OCP, we can save bits on average by 29.1%, 33.0%, 35.0% for the distortion level "Near=1", "Near=2", "Near=3", respectively, with almost the same PSNR (dP is only 0.3%, 0.6% and 0.8%, respectively). The results demonstrate that our OCP method works well with Motion JPEG-LS for near-lossless video coding.

Table VI
Results for Motion JPEG-LS (near-lossless) for OCP

Index	Near=1		Near=2		Near=3	
	dP	saving	dP	saving	dP	saving
1	-0.2	53.3	-0.3	54.1	-0.5	54.7
2	0.2	5.9	0.4	10.4	0.6	13.5
3	1.0	44.2	1.3	49.6	1.2	51.6
4	0.0	13.6	0.0	15.2	0.1	14.4
5	0.5	52.6	1.0	56.2	1.0	57.9
6	0.0	0.0	0.0	0.0	0.0	0.0
7	0.1	10.2	0.3	12.1	0.3	13.4
8	0.6	46.8	1.5	52.1	1.7	54.1
9	0.0	27.9	0.9	39.5	1.7	46.4
10	0.1	11.3	0.4	17.2	1.0	25.4
11	0.3	19.9	1.3	29.9	1.3	32.1
12	0.1	20.2	0.2	24.2	0.2	26.6
13	0.7	40.3	1.3	43.5	1.4	44.5
14	0.1	48.7	0.1	50.7	0.7	51.7
15	0.5	50.0	1.3	53.2	1.5	54.3
16	0.5	46.0	1.0	46.9	1.2	47.1
17	0.5	17.8	0.8	19.3	0.7	19.3
18	0.0	14.8	0.0	19.3	0.1	22.5
mean	0.3	29.1	0.6	33.0	0.8	35.0

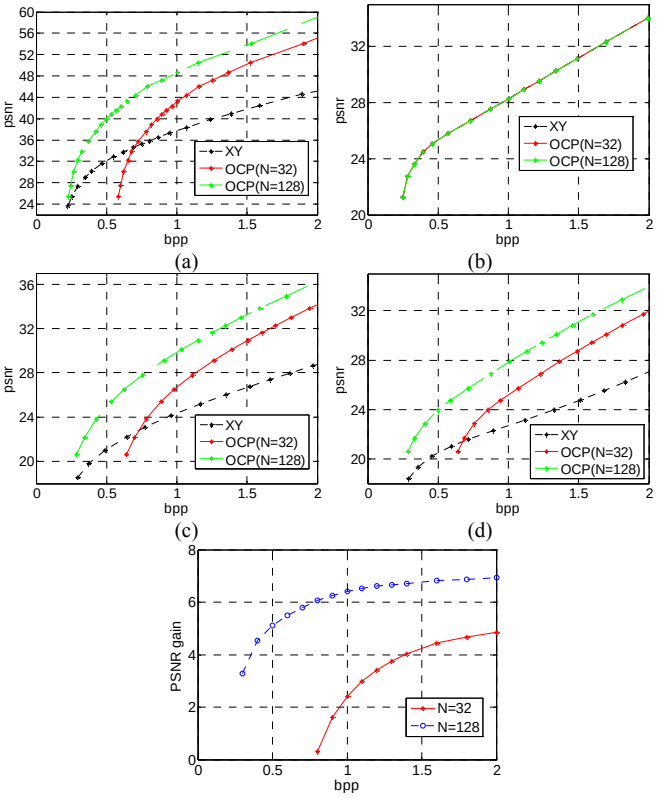

Fig. 10. Results for Motion JPEG. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

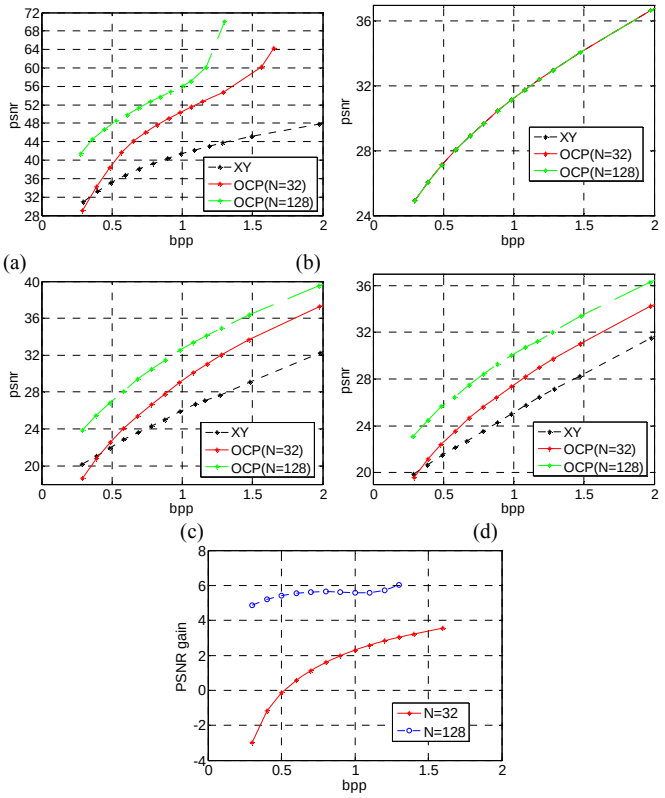

Fig. 11. Results for Motion JPEG 2000. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

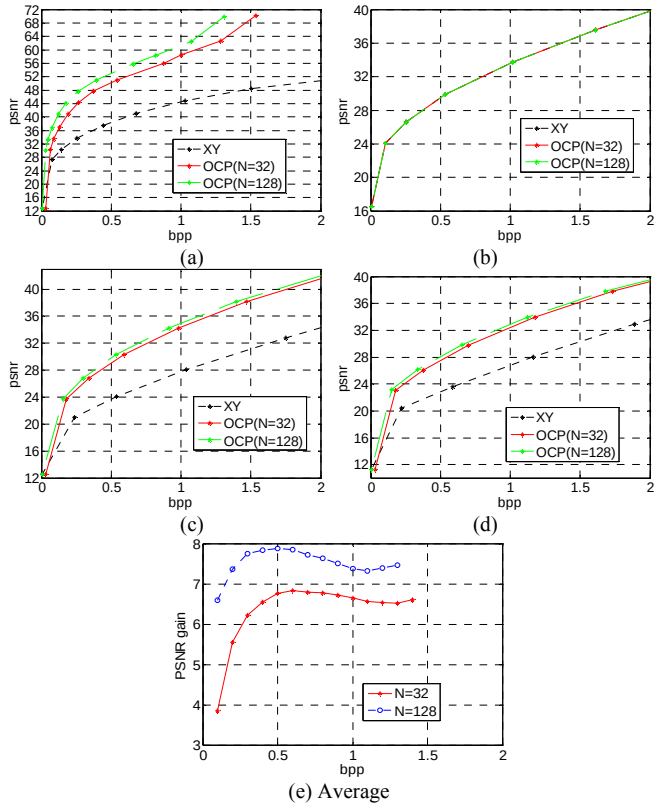

Fig. 12. Results for H.264 intra-only profile. (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

C. Performance Comparison of OCP with Motion JPEG, Motion JPEG 2000 and H.264 Intra-only Profile

Motion JPEG codes each frame of a digital video sequence separately as a JPEG image, and Matlab function [imwrite] with appropriate parameter is used for JPEG image compression; similar to Motion JPEG, Motion JPEG 2000 (JPG2K) codes each frame separately as a JPG2K image. In our experiment, each frame was coded by Jasper [10], which implements the codec specified in the JPG2K Part-1 standard (i.e., ISO/IEC 15444-1 [12]); MPEG and ITU-T have recently added four intra-only profiles to AVC/H.264 to respond to strong demands for high-quality video applications [27-28]. Therefore, we will also show the performance comparison of coding in OCP with H.264 intra-only profile.

To discuss the impact of PPU size (as continuation of Section III.C), two sets of PPU size (i.e., N=32 and N=128) were used, and there is only one PPU for each sequence if N=128 and there are four PPUs if N=32. Fig. 10 to Fig. 12 have given the RD curves (XY; OCP with N=32 and OCP with N=128) for the four typical sequences and the curve of average PSNR gain (i.e., $PSNR_{OCP}-PSNR_{XY}$), for the three coding techniques (namely Motion JPEG, Motion JPG2K and H.264 intra-only profile), respectively.

From Fig. 10 to Fig. 12, we can see that the RD performance with PPU size being 128 is always better than that of PPU size being 32 for non-XY planes (note that the OCP for "Football" (with Index 6) is the traditional XY plane and therefore have the same RD performance for different PPU sizes); this is expected since more temporal redundancy is explored when we use a larger PPU size.

From Fig. 10 (a) (c) and (d) we can also see that when the PPU size of 32 is used, the RD curve of a non-XY OCP may fall below that of the traditional XY plane for the cases of large QPs. The two reasons are as follows. (a) Smaller PPUs are usually less efficient for exploring the temporal redundancy. Fig. 10~12 have shown the RD performance for two PPU sizes (i.e., 32 and 128) for different videos and coding schemes, and it is demonstrated that a smaller PPU size is with poorer RD performance. (b) With a smaller PPU size, the resolution for a non-XY plane is smaller than that of the XY plane, and this leads to larger overhead bitrate (bpp) for the frame header (the size of the frame header increases slightly with the increase of frame resolution; therefore, the smaller the frame resolution is, the larger the overhead bitrate becomes for the frame header). On the other hand, with large QPs (i.e., for low bitrate coding), the overhead bitrate due to the frame header becomes significant in comparison with that for the visual content itself, and this is the other reason why the RD curve of a non-XY plane at the PPU size of 32 may fall below that of the traditional XY plane for a large QP. In the case of Fig. 10, the bpp for the intersection point (for the RD curve in the XY plane and that in a non-XY plane) with PPU size of 32 is just slightly larger than the overhead bitrate for a non-XY plane (i.e., about 0.6 bpp); so in this case, nearly all bits are used for the frame header and low RD performance is obtained for a non-XY plane.

Table VII
PSNR gain of OCP at 0.8 bpp against the three existing techniques

Index	Motion JPEG	Motion JPG2K	I264
1	10.3	13.7	15.8
2	2.7	2.0	2.6
3	10.0	3.7	5.9
4	3.1	2.7	3.6
5	12.9	12.2	13.8
6	0.0	0.0	0.0
7	2.5	2.3	2.2
8	9.1	9.8	11.4
9	8.6	8.3	9.2
10	4.1	1.2	3.1
11	3.7	0.6	2.4
12	4.9	6.2	6.9
13	7.0	3.1	8.6
14	8.8	10.8	14.8
15	8.6	11.0	15.5
16	6.6	8.6	12.8
17	1.8	0.6	2.8
18	4.4	4.9	5.9
mean	6.1	5.7	7.6

To discuss the impact of video content toward the PSNR gain, we list the detailed PSNR gain for different video content and coding technique in Table VII (N=128, bit budget is 0.8 bpp and I264 represent the H.264 intra-only profile). It can be noticed that OCP without inter-frame prediction works better for video having slow object moving speed and less moving areas (i.e., there are more stationary areas), since TX and TY options are adopted in these situations, as expected. The proposed scheme achieves the maximum bitrate saving for the video sequence "Akiyo" among the eighteen sequences, which is the most static one and has almost unchanged background (high redundancy along T axis). On the other hand, "Football" is the sequence with no bitrate saving due to the fact that it contains relatively less temporal redundancy.

D. Performance Comparison of OCP with H.264

When inter-frame prediction is allowed (i.e., in H.264), coding in a non-XY plane still has benefits as analyzed in Section IV.B. By using (9), we find that 11 out of 18 test sequences are with non-XY OCPs, and all of them are of low motion or with camera zooming. In our simulation, PPU size 128 is used and the configuration (denoted as IP) for H.264 is: the high profile of JM 16.0, encoding in grayscale, GoP size being 16 and with one reference frame, Simplified UMHexagon Search with search range of 16, no B frame, and all the other parameters being the default setting of high profile (e.g., high complexity RDO and CABAC coding).

In Fig. 13 we have shown the results for the four videos. Fig.13 (a) is with low motion, so a non-XY plane coding performs better, while in Fig. 13 (b) where the video is of high but predictable motion, the non-XY plane coding does not show its advantage since BME performs effectively in the XY plane. The videos in (c) and (d) share something in common: high and complex motion (objects move along different directions and the video is with camera zooming); since BME does not perform effectively in the XY plane with these two cases, the non-XY plane coding demonstrates its advantage.

For a comprehensive evaluation of the proposed scheme, the BD-PSNR (i.e., average PSNR differences between RD curves) and the BD-Bitrate (i.e., bitrate increase rate) [38] of the proposed scheme against H.264 are calculated for eighteen videos and given in the second and third columns of Table VIII. It has shown that coding in OCP has better or equal RD performance in comparison with the traditional XY plane except for the sequence "Miss-America". The degradation of RD performance for "Miss-America" is due to the prediction error by using (9). The average PSNR improvement (i.e., BD-PSNR) for all 18 sequences is 0.89 dB (equivalently, 11.01% in average bitrate saving).

Fig. 14 has shown the three RD curves for XY, TX, and TY coding planes for the sequence "Tempete" (a sequence with flowers, falling leaves, and stones) of SD (standard definition; 720x486 in fame size). The figure confirmed that the traditional XY coding plane does not always achieve the best RD performance for SD video sequence, and the proposed coding method (coding in a non-XY plane) can be still used for SD resolution for performance improvement.

The simulation results for other configurations of H.264 are also provided as in the last two columns of Table VIII, where IBP represents the configuration with B-pictures and S2R represents the configuration with two reference frames, and the other parameter settings in both configurations are the same as those in IP. From the table we can see that there are 0.47dB and 0.79dB BD-PSNR improvement on average compared with H.264 for the configuration with B-pictures and that with two reference frames, respectively.

VI. Conclusion

Since the compressibility of visual signal depends upon the associated signal redundancy, this study firstly investigates into the signal correlation with different compression planes. The analysis shows that coding in the traditional XY plane are not always optimal, and therefore we presented a novel framework for video coding via adaptive optimal compression plane (OCP) selection. An OCP is auto-detected by comparing the correlation coefficients (CCs) along different axes, and we have also shown that sparse frame samples are sufficient for an efficient process.

The proposed scheme is used as a preprocessor with any existing image and video coding standards and with low computational complexity overhead. Extensive experiments have been conducted with different visual content, and the effectiveness of the proposed OCP-based scheme has been confirmed. The proposed method improves the RD performance significantly for various coding schemes, such as Motion JPEG-LS, Motion JPEG, Motion JPEG, Motion JPG2K, H.264 intra-only profile, and H.264.

Acknowledgment

The authors are grateful to the Associate Editor Professor Xiaolin Wu and the anonymous reviewers for their encouraging and valuable advice that has led to this improved version and clearer presentation of the technical content.

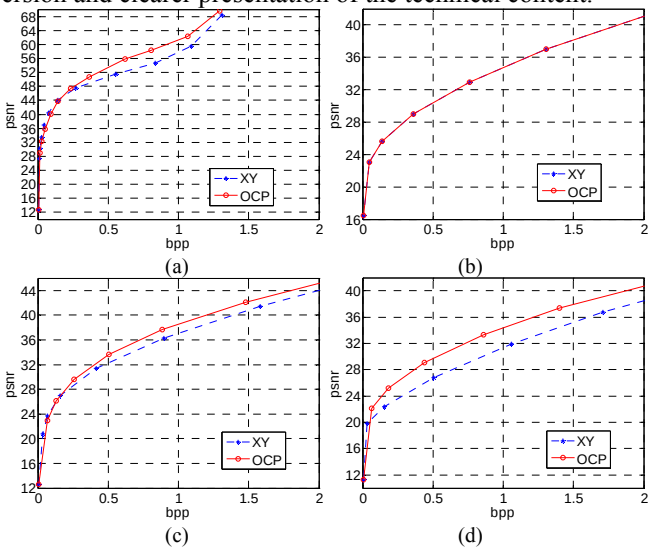

Fig. 13. Results for the comparison of the OCP and XY plane coding (i.e. H.264). (a) Akiyo, (b) Football, (c) Mobile, (d) Tempete.

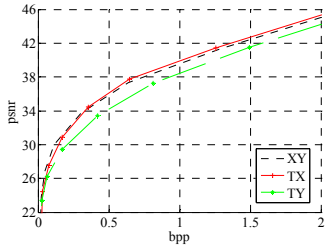

Fig. 14. Simulation result for the sequence "Tempete" (a 720x486 sequence with flowers, falling leaves, and stones).

Table VIII
Comparison of RD performance of the proposed scheme against H.264 under the IP configuration (IP) and the configurations with B pictures (IBP) and that with two reference frames (S2R) (for 0.05~2 bpp)

Video	BD-Bitrate (%) (for IP)	BD-PSNR(dB)		
		IP	IBP	S2R
Akiyo	-19.17	2.32	1.66	2.11
Carphone	0.00	0.00	0.00	0.00
Claire	0.00	0.00	0.00	0.00
Coastguard	0.00	0.00	0.00	0.00
Container	-35.40	2.54	1.12	2.31
Football	0.00	0.00	0.00	0.00
Foreman	0.00	0.00	0.00	0.00
Grandma	-20.21	1.24	0.76	1.12
Hall	-29.17	1.69	1.37	1.39
Highway	0.00	0.00	0.00	0.00
Miss-America	15.39	-0.68	-0.87	-0.80
Mobile	-15.52	1.11	-0.95	0.74
Mother-daughter	0.00	0.00	0.00	0.00
News	-16.74	1.72	1.19	1.66
Salesman	-30.73	2.44	2.00	2.07
Silent	-9.68	0.75	0.75	0.61
Suzie	0.00	0.00	0.00	0.00
Tempete	-37.02	2.92	1.43	3.03
Average	-11.01	0.89	0.47	0.79

References

[1] A. Liu, W. Lin, F. Zhang, "Lossless video compression with optimal compression plane determination," in *Proc. Int. Conf. Multimedia and Expo*, pp. 173–176, 2009.

[2] Y. Li, K. Sayood, "Lossless video sequence compression using adaptive prediction," *IEEE Trans. Image Process.*, vol. 16(4), pp. 997–1007, April 2007.
[3] Information Technology-Lossless and near-lossless compression of continuous-tone images-Baseline. International Telecommunication Union (ITU-T Recommendation T.87). ISO/IEC 14495-1, 1998.
[4] X. Li, X. Chen, X. Xie, G. Li, L. Zhang, C. Zhang, Z. Wang, "A Low Power, Fully Pipelined JPEG-LS Encoder for Lossless Image Compression," in *Proc. Int. Conf. Multimedia and Expo*, pp. 1906–1909, 2007.
[5] S. Miaou, F. Ke, S. Chen, "A Lossless Compression Method for Medical Image Sequences Using JPEG-LS and Interframe Coding," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13(5), pp. 811–821, Sept. 2009.
[6] M. J. Weinberger, and G. Seroussi, "From LOCO-I to the JPEG-LS Standard", Computer Systems Laboratory, HP Laboratories Palo Alto, HPL-1999-3, January 1999. http://www.hpl.hp.com/loco/.
[7] Information technology - Digital compression and coding of continuous-tone still images - Part 1: Requirements and guidelines - First Edition; Corrigendum 1:2005. ISO/IEC 14495-1, 1994.
[8] J. Byrne, S. Ierodiaconou, D. Bull, D. Redmill, P. Hill, "Unsupervised image compression-by-synthesis within a JPEG framework," in *Proc. Int. Conf. Image Processing*, pp. 2892–2895, Oct. 2008.
[9] R. Neelamani, R. Queiroz, Z. Fan; S. Dash, R. Baraniuk, "JPEG compression history estimation for color images," *IEEE Trans. Image Process.*, vol. 15(6), pp. 1365–1378, June 2006.
[10] International Organization for Standardization and International Electrotechnical Commission. ISO/IEC 15444-1:2000, Information technology—JPEG 2000 image coding system—Part 1: Core coding system.
[11] G. Baruffa, P. Micanti, F. Frescura, "Error Protection and Interleaving for Wireless Transmission of JPEG 2000 Images and Video," *IEEE Trans. Image Process.*, vol. 18(2), pp. 346–356, Feb. 2009.
[12] M. Adams, R. Ward, "Jasper: A portable flexible open-source software tool kit for image coding/processing," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 241–244, 2004.
[13] Advanced video coding for generic audiovisual services. International Telecommunication Union (ITU-T Recommendation H.264). ISO/IEC 14496-10, 2009.
[14] T. Wiegand, G. Sullivan, G. Bjontegaard, A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13(7), pp. 560–576, July 2003.
[15] P. Lambert, W. Neve, P. Neve, I. Moerman, P. Demeester, R. Walle, "Rate-distortion performance of H.264/AVC compared to state-of-the-art video codecs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16(1), pp. 134–140, Jan. 2006.
[16] "JM Software," http://iphome.hhi.de/suehring/tml/.
[17] Ka. Ng, L. Po, K. Wong, Chi. Ting, K. Cheung, "A Search Patterns Switching Algorithm for Block Motion Estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19(5), pp. 753–759, May 2009.
[18] M. Sarwer, Q. Wu, "Adaptive Variable Block-Size Early Motion Estimation Termination Algorithm for H.264/AVC Video Coding Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19(8), pp. 1196–1201, May 2009.
[19] L. Po, C. Ting, K. Wong, K. Ng, "Novel Point-Oriented Inner Searches for Fast Block Motion Estimation," *IEEE Trans. Multimedia*, vol. 9(1), pp. 9–15, Jan. 2007.
[20] V. Nguyen, Y. Tan, "Efficient block-matching motion estimation based on Integral frame attributes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16(3), pp. 375–385, Mar. 2006.
[21] A. Bahari, T. Arslan, A. Erdogan, "Low-Power H.264 Video Compression Architectures for Mobile Communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19(9), pp. 1251–1261, Sept. 2009.
[22] C. Chen, S. Chien, Y. Huang, T. Chen, T. Wang, L. Chen, "Analysis and Architecture Design of Variable Block-Size Motion Estimation for H.264/AVC," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol.53(3), pp. 578–593, Mar. 2006.
[23] D. T. Vo, T. Q. Nguyen, "Quality Enhancement for Motion JPEG Using Temporal Redundancies," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18(5), pp. 609–619, May 2008.
[24] M. Fujiyoshi, K. Kuroiwa, H. Kiya, "A scrambling method for Motion JPEG videos enabling moving objects detection from scrambled videos," in *Proc. Int. Conf. Image Processing*, pp. 773–776, 2008.
[25] J. Meessen, C. Parisot, X. Desurmont, J. F. Delaigle, "Scene analysis for reducing motion JPEG 2000 video surveillance delivery bandwidth and complexity," in *Proc. Int. Conf. Image Processing*, pp. 577–580, 2005.
[26] T. Tillo, M. Grangetto, G. Olmo, "Multiple description coding with error correction capabilities: an application to motion JPEG 2000," in *Proc. Int. Conf. Image Processing*, pp. 3129–3132, 2004.
[27] G. Sullivan, T. Wiegand, H. Yu, "New Profiles for Professional Applications," *JVT document JVT-V204*, Jan. 2007.
[28] H. Yu, "New MPEG-4-AVC/H.264 Profiles for Professional and High-Quality Video Applications," *SMPTE Motion Imaging Journal*, pp. 45–49, Jan. 2009.
[29] Y. Liu, F. Wu, K. N. Ngan, "3-D Object-Based Scalable Wavelet Video Coding With Boundary Effect Suppression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17(5), pp. 639–644, May 2007.
[30] V. Seran, L.P. Kondi, "New scaling coefficients for bi-orthogonal filter to control distortion variation in 3D wavelet based video coding," in *Proc. Int. Conf. Image Processing*, pp. 1873–1876, 2006.
[31] E. Carotti, J. Martin, A. Meo, "Low-complexity lossless video coding via adaptive spatio-temporal prediction," in *Proc. Int. Conf. Image Processing*, pp. 197–200, 2003.
[32] H. Schwarz, D. Marpe, T. Wiegand, "Analysis of Hierarchical B Pictures and MCTF," in *Proc. Int. Conf. Multimedia and Expo*, pp. 1929–1932, 2006.
[33] S. Yu, C. Chrysafis, "New Intra Prediction using Intra-Macroblock Motion Compensation", JVT-C 15 1rl-L.doc, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG Meeting, May 2002.
[34] K. Tang, K. Ngan, "Enhancement Techniques for Intra Block Matching," in *Proc. Int. Conf. Multimedia and Expo*, pp. 420–423, 2007.
[35] S. Kim, J. Han, J. Kim, "An efficient scheme for motion estimation using multireference frames in H.264/AVC," *IEEE Trans. Multimedia*, vol. 8(3), pp. 457–466, June 2006.
[36] C. Chiang, S. Lai, "Fast multi-reference motion estimation via statistical learning for H.264/AVC," in *Proc. Int. Conf. Multimedia and Expo*, pp. 61–64, 2009.
[37] P. Wu, C. Xiao, "An adaptive fast multiple reference frames selection algorithm for H.264/AVC," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 1017–1020, 2008.
[38] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," VCEG Contribution VCEG-M33, April 2001.
[39] X. Wu, X. Zhang, X. Wang, "Low Bit-Rate Image Compression via Adaptive Down-Sampling and Constrained Least Squares Upconversion," *IEEE Trans. Image Process.*, vol. 18(3), pp. 552–561, 2009.
[40] Y. Tanaka, M. Hasegawa, S. Kato, M. Ikehara, T.Q. Nguyen, "Adaptive Directional Wavelet Transform Based on Directional Prefiltering," *IEEE Trans. Image Process.*, vol. 19(4), pp. 934–945, 2010.
[41] X. Yang, W. Lin, Z. Lu, E. Ong, S. Yao, "Motion-compensated residue preprocessing in video coding based on just-noticeable-distortion profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15(6), pp. 742–752, 2005.