Efficient video coding using optimal compression plane and background modelling

M. Paul

School of Computing and Mathematics, Charles Sturt University, Bathurst, Australia
E-mail: mpaul@csu.edu.au

Abstract: All existing video coding standards consider a video as a temporal (along T-axis) collection of two-dimensional (2D) pictures (formed by XY-axes) and compress them by exploiting spatial and temporal redundancy in the pictures. A recent optimal compression plane (OCP) determination technique shows that better compression can be achieved by relaxing the physical meaning of axes by exploring information redundancy in a fuller extent where a video is considered as a 3D data cube. Spatial and temporal dimensions are determined based on the statistical redundancy along each axis. Treating a video as a 3D data cube revolutionises the traditional video features such as background, motion, object, zooming, panning etc. In this study, the authors apply dynamic background modelling to the OCP plane to exploit the newly generated background in the video for further improving the coding performance. The experimental results reveal that the proposed approach outperforms the existing state-of-the-art OCP technique as well as the H.264 video coding standard.

1 Introduction

Unlike any other multimedia component, a video requires a huge amount of raw data for storage and transmission to retain a pleasant quality. Thus, a proper compression technique (i.e. video coding) is inevitable to compress the huge amount of video data for practical applications with the stringent storage or bandwidth requirements. All existing video coding standards such as H.264 [1–3], MPEG-4 [4] etc. consider a video as a sequence of natural frames (formed by X- and Y-axes) and compress the frames by exploiting spatial redundancy (along X- and Y-axes within a frame) and temporal redundancy (along T-axis among the frames) separately. Recently an optimal compression plane (OCP) determination technique [5] considers a video as a three-dimensional (3D) data cube by relaxing the physical meaning of X-, Y- and T-axes of a video and achieves more compression through the exploration of the information redundancy in an extensive way.

A video \( (V_{XYT}) \) can be represented with X-, Y- and T-axes by considering any two axes to form a spatial frame and the rest axis to form temporal axis. Thus, a video can be represented as a collection of images \( I_{XY}, I_{TX} \) or \( I_{TY} \) formed by \( XY-, TX- \) or \( TY-plane \) with \( T-, Y-, X-axes \) as a temporal axis, respectively

\[
V_{XYT} = \{I_{XY}(t), \quad t = 1, 2, 3, \ldots\} \quad \text{or} \quad V_{XYT} = \{I_{TX}(y), \quad y = 1, 2, 3, \ldots\} \quad (1)
\]

or

\[
V_{XYT} = \{I_{TY}(x), \quad x = 1, 2, 3, \ldots\}
\]

where \( I_{XY} \) represents a natural image with axes of X and Y, \( I_{TX} \) represents a rearranged image with axes of T and X, and \( I_{TY} \) represents a rearranged image with axes of T and Y. One example is given in Fig. 1 where Fig. 1a shows natural images of the ‘Mobile’ video sequence, Fig. 1b shows images using TX-plane and Fig. 1c shows images using TY-plane. The OCP technique [5] first determines the statistical redundancy along an axis using average correlation coefficients (CCs) [6] between frames formed by the two remainder axes. Then the OCP selects the OPC formed by the two axes, which provide the largest CCs for intra-frame coding. A modified criterion for selecting the OPC is also proposed in [5] for inter-frame coding. The experimental results reveal that TX- or TY-plane rather than conventional XY-plane is selected as an OPC for a large number of common intermediate format (CIF), quarter CIF (QCIF) and standard definition (SD) video sequences including as ‘mobile’, ‘foreman’, ‘container’, ‘Tempete’ etc. An improved rate–distortion performance is achieved when JPEG2000 or H.264 coding technique is applied to the OCP.

Treating a video as a 3D data cube and rearranging the video frames in other two directions revolutionises the way to treat the traditional video features such as background, motion, object, panning, zooming etc. Owing to the rearrangement of the traditional XY-plane images into the TX- or TY-plane images, object and/or camera motions of a traditional video can be transformed into a simplified motions or simple background, for example, horizontal motions and vertical motions can be transformed into a static background in the TX- or TY-plane images, respectively, or a heterogeneous object can be transformed into a smooth object in TX- or TY-plane. Besides this, camera motions such as zooming and panning could not be
effectively estimated by the traditional translational motion estimation adopted into the H.264 [1]. Camera motions can also be transformed into a simplified motion/background in the TX- or TY-plane, thus any existing coding technique can encode them more efficiently using TX- or TY-plane.

On the other hand, to improve the coding efficiency in the case of repetitive motion, uncovered/occluded background, non-integer pixel displacement, lighting change etc. the H.264 uses multiple reference frames (MRFs) instead of single reference frame. When some information of a frame is used to encode other frame we call the first frame as a reference frame for the later frame. The number of reference frames in practical application is limited (typically five reference frames) because of the requirement of codes to identify the reference frame, computational time in motion estimation and compensation and memory buffer size to store reference frames for both encoder and decoder. Moreover, if the reference frames in MRFs does not cover those features properly, we will not obtain any coding improvement and a lot of computation time is wasted. A number of techniques including [7, 8] propose dual reference frames as a compromise in terms of rate – distortion performance, memory requirement and computational complexity between single and MRFs. The dual reference frames technique uses the immediate previous frame as a short-term reference (STR) frame and the other previously encoded frame as a long-term reference (LTR) frame while encoding the current frame. The basic assumption of the dual reference frames techniques is that the STR frame is found to be useful for moving areas and the LTR frame is useful for background areas of the image.

Recently a dynamic background frame termed as the most common frame in a scene (McFIS) [9] has been developed for video coding using the dynamic background modelling based on Gaussian mixture [10, 11]. The McFIS is used as a second reference frame for encoding the current frame assuming that the motion part of the current frame would be referenced using the immediate previous frame and the static background part would be referenced using the McFIS. The ultimate reference is selected at block and sub-block levels using the Lagrangian multiplier [3]. The McFIS is used as an LTR frame in the dual reference frames concept which is a special case (only two reference frames are used) of the concept of the MRFs. In this paper, we exploit the newly introduced background in the TX- or TY-plane through the McFIS. In the proposed method, we first determine the OCP and then encode the video towards the optimal plane with the McFIS as a second reference frame. The experimental results reveal that the proposed technique improves a significant video quality compared to the existing OCP technique and the H.264 video coding standard with comparable computational complexity. The detailed procedure, experimental results analyses and conclusions of the proposed technique are provided in the next sections.

2 Proposed technique

The proposed technique has two phases: (i) pre-processing phase to determine the optimal compression plane and (ii) actual coding phase to encode the frame with the help of McFIS modelling. In the pre-processing phase, the proposed technique determines the coding direction of a video using the statistical redundancy. The coding direction would be one of three possible directions such as $V_{XY}$, $V_{TX}$ or $V_{TY}$ where first two axes form the spatial image and the third axis forms the temporal image. In the actual coding, a frame is encoded using two reference frames; one is the immediate previous frame and the other one is the McFIS assuming that object areas and the background areas of the current frame use the immediate previous frame and the McFIS as the reference frame, respectively, based on
the Lagrangian multiplier. The McFIS (i.e. a dynamic background frame) is generated using the coded frames at the encoder and the decoder. As the same procedure and the same encoded frames are used at the encoder and the decoder, we do not need to transmit the McFIS from encoder to the decoder. The \( k \)th McFIS is used to encode the \((k + 1)\)th frame where \( k \) is the number of encoded frames are used to form the \( k \)th McFIS based on the Gaussian mixture modelling. The OCP formation and the McFIS generation are explained in detail at the next two subsections.

### 2.1 Optimal compression plane

As mentioned before a video can be treated as a 3D data cube and represented as one of \( V_{XYT} \), \( V_{TXY} \) or \( V_{TYX} \) formats. Obviously the most accurate determination of compression plane of a video is to apply the H.264 on each of the formats and then select the best one based on the joint rate–distortion performance. This brute-force method takes three times computational time compared to the standard coding schemes. To reduce the computational time, an OCP determination algorithm \([5]\) is proposed based on the statistical redundancy. The statistical redundancy along one axis can be estimated by the average CC formed by the other two axes. The bigger the average CC is, the more the statistical redundancy exists \([5]\). The CC is defined in \([6]\) as follows

\[
CC_{k+1} = \frac{\sum_{i,j} (I_k(x,y) - \bar{I}_k)(I_{k+1}(x,y) - \bar{I}_{k+1})}{\sqrt{\sum_{i,j} (I_k(x,y) - \bar{I}_k)^2} \sum_{i,j} (I_{k+1}(x,y) - \bar{I}_{k+1})^2} \tag{2}
\]

where \( I_k(x,y) \) and \( I_{k+1}(x,y) \) are the pixel intensities located at the \((x, y)\) position of \( k \)th and \((k + 1)\)th frames, respectively, and \( \bar{I}_k \) and \( \bar{I}_{k+1} \) are the average pixel intensities of \( k \) and \((k + 1)\) th frames, respectively.

Equation (2) is normalised using the number of frames used in the CC calculation to determine the \( C_T \) (along the \( T \)-axis), \( C_X \) (along the \( X \)-axis) and \( C_Y \) (along the \( Y \)-axis) using \( I_{XY}, I_{TY} \) and \( I_{TX} \) images, respectively, in the videos of \( V_{XYT}, V_{TXY} \) and \( V_{TYX} \), respectively. The optimal inter compression plane is determined \([5]\) as follows

\[
\text{OCP} = \begin{cases} 
    TX & \text{if } C_X \geq C_Y \text{ and } C_T - \max(C_X, C_Y) > T \\
    TY & \text{if } C_Y < C_X \text{ and } C_T - \max(C_X, C_Y) > T \\
    XY & \text{otherwise}
\end{cases} \tag{3}
\]

where \( T \) (chosen as \( 3.5 \times 10^{-2} \) in the current work) is a constant threshold which accounts for the advantage of block-based motion estimation in the \( XY \)-plane; this makes \( TX \)- or \( TY \)-plane only to be chosen when \( C_T \) is sufficiently high. When we apply inter- and intra-modes of the H.264 on any plane, the efficiency of the block-based motion estimation and the CC will dictate the plane selection for better compression. Thus, we need a threshold to predict the accuracy of the block-based motion estimation towards a direction. Experimentally, we have observed that if a video has motion in temporal \((T)\)-direction which can be successfully approximated by the block-based motion estimation, better compression can be achieved using \( XY \)-plane, otherwise, \( TX \)- or \( TY \)-plane provides better compression. It is very difficult to predict the motion estimation accuracy for a particular direction without exploring exhaustively for all directions. Thus, first we have coded 18 standard videos with different resolutions (QCIF, CIF and 4CIF formats) using all directions and then try to formulate the optimal plane using the correlations among \( C_T, C_X \) and \( C_Y \). The experimental results suggest that the compression results can be achieved if we use \( T = 0.035 \). As the calculation of \( C_T, C_X \) and \( C_Y \) is independent of frame size and resolution, the value \( T \) does not depend on the video size.

### 2.2 McFIS generation

When a previously (e.g. a few moments ago) occluded background is uncovered in a frame of a video because of the object movement, the conventional motion estimation and MRFs scheme adopted in the H.264 could not find the reference area in the reference frames because of the limited number of reference frames in the MRFs set. As a result, poor rate–distortion performance is achieved for the uncovered background areas. Motion estimation and compensation using a true background frame as a reference frame can provide better rate–distortion performance for the uncovered background area. Dynamic background modelling is required to generate a true background from the video with illumination changes, shadow, background changes, camera motions etc. Moreover, a true background could not be obtained without modelling as in the most cases videos are captured without explicit background scene.

Basic assumption of the background modelling is to consider a pixel position (i.e. a physical position of a pixel in an image) as a part of a background if the pixel intensity at that position does not change in a number of successive frames. A position may experience a background and a number of foregrounds during a time frame. To accommodate different experiences because of the changing of background and foreground, multiple Gaussian models are used for a pixel position where each model represents each experience (i.e. backgrounds or foregrounds). We assume that \( k \)th Gaussian at time \( t \) represents a pixel intensity with mean \( \mu_k \), standard deviation (STD) \( \sigma_k \), recent value \( \gamma_k \), and weight \( \omega_k \) such that \( \sum \omega_k = 1 \). The mean \( \mu_k \) represents the average pixel intensities of \( k \)th Gaussian model, the STD \( \sigma_k \) represents standard variations among the pixel intensities of \( k \)th Gaussian model, the recent value \( \gamma_k \) represents the instantaneous pixel intensity and the weight \( \omega_k \) represents the number of pixel intensities satisfy the \( k \)th Gaussian model. Obviously, the total weights of all Gaussian models should be normalised to make \( \sum \omega_k = 1 \).

The learning parameter \( \alpha \) is used to balance the contribution between the current and past values of parameters such as weight, STD, mean etc. After initialisation, for every new observation (i.e. instance) \( X_t \) (pixel intensity at time \( t \)) is first matched against the existing models in order to find one (e.g. \( k \)th model) such that \( |X_t - \mu_k| < 2.5 \sigma_k \). If such a model exists, update corresponding recent value parameter \( \gamma_k \) with \( X_t \). Other parameters are updated with the learning rate as

\[
\mu_k = (1 - \alpha)\mu_k^{t-1} + \alpha X_t \tag{4}
\]

\[
\sigma_k^2 = (1 - \alpha)\sigma_k^{t-2} + \alpha(X_t - \mu_k)^2(X_t - \mu_k) \tag{5}
\]

\[
\omega_k = (1 - \alpha)\omega_k^{t-1} + \alpha \tag{6}
\]
and the weights of the remaining Gaussians (i.e. \( l \neq k \)) are updated as \( \omega_l = (1 - \alpha) \omega_l \). Afterward, the weights are renormalised using \( \omega_l = \omega_l / \Gamma \) where \( \Gamma = \sum_{l=1}^{K} \omega_l \). If such a model does not exist, a new Gaussian model is introduced with \( \gamma = \mu = X', \sigma = 30 \) and \( \omega = 0.001 \) by evicting the \( K \)th (based on \( \omega / \sigma \) in descending order) model if it exists. Different approaches for background modelling and updating are mentioned in [10–13]. Stauffer and Grimson [10] used a threshold on the relative information of the amount of background and foreground in a video to model the dynamic background and Lee [11] also used two thresholds to represent the ratio of background and foreground of a video using sigmoid function to model the background. It is very difficult and sometimes impossible to obtain prior knowledge about the ratio of background and foreground in a video in prior. To overcome the predefined threshold problem, Haque et al. [12] and Paul et al. [13] used recent pixel value of the selected Gaussian model for object detection and video coding application, respectively.

The existing background modelling techniques [10–13] used original frames to model the background whereas Paul et al. [9] used coded frames to model the dynamic background frame in a same manner in the encoder and decoder for better rate–distortion performance as well as to avoid the transmission of the McFIS from encoder to decoder. To minimise the distortion effect because of the quantisation on the coded frames, Paul et al. [9] used neighbouring pixel information of the current coded frame and co-located pixel information of the previous McFIS for updating the current McFIS. The modelling and generating techniques used in [9] take more computational time (for processing neighbouring pixels) and memory (for storing additional McFIS in the buffer). In addition, the McFIS modelling scheme [9] using neighbouring pixel information and previous McFIS information could not perform well for the videos with high texture and camera motions (such as ‘Tempete’ video) because of the less correlation among the neighbouring pixels of the videos. To overcome the above mention problem, the proposed scheme obtains the

![Fig. 2 McFISes at different planes using silent and mobile video sequences where red marked arrows indicate uncovered background captured by the corresponding McFISes](image-url)
background pixel intensity by taking average of the ‘mean’ pixel intensity and ‘recent’ pixel value of the model that has the highest value of weight/STD among the Gaussian mixture models of a pixel. The recent pixel intensity of a model indicates the most recent intensity change of the pixel for the model whereas the mean pixel intensity of a model carries the history of the pixel position and provides the average trend of the pixel intensities of the model over the frames (i.e. times). Thus, the average of the recent and mean pixel intensities of the background model provides us a good tradeoff between the recent change and the history of the background pixel in terms of compression perspective for the typical videos as well as videos with texture and camera motion. For high-texture video such as Tempete, the proposed scheme reduces up to 60 bits per frame and 5% computational time compared to the scheme [9].

2.3 McFISes in XY-, TX- and TY-planes

Using the above mentioned McFIS generation procedure we have generated different McFISes using two popular videos namely ‘silent’ and ‘mobile’ of $V_{XY}$, $V_{TX}$ and $V_{TY}$, respectively. The first and third columns of Fig. 2 show original 50th frames for different planes ($XY$, $TX$ and $TY$) and second and fourth columns of figure show 50th McFISes of different planes. The second and fourth columns of Fig. 2 show some uncovered background areas indicated by the red marked arrows. If we compare the images of the $TX$- and $TY$-planes for silent and mobile sequences compared with those of $XY$-planes, we can easily differentiate them in terms of smoothness. As we mentioned in the third paragraph of Section 1, smoother objects or motion can be obtained after transforming the video into different planes. The McFISes of $TX$- and $TY$-planes also provide smoother images compared to the McFISes of $XY$-planes. This result also indicates that a better rate–distortion performance can be achieved using the smoother McFISes of $TX$- and $TY$-planes compared to the rough McFISes of the $XY$-plane as the primary goal of the McFIS is to capture more background.

3 Experimental results

To compare the performance of the proposed scheme using the McFIS as an LTR frame in the optimal compression plane, we have selected other two state-of-the-art methods such as the H.264 [1], which is the latest and the best coding standard to date and the OCP scheme [5], which is the most relevant scheme to the proposed scheme. As the proposed scheme uses two reference frames, that is, the immediate previous frame and the McFIS, we have also selected the H.264 and the OCP schemes with two reference frames, that is, immediate two previous reference frames. The proposed scheme, the H.264 [1] with two reference frames (H.264-2Ref), and the OCP scheme [5] with two reference frames (OCP-2Ref) schemes are implemented based on the H.264 recommendations adapted from JM 10.1 H.264/AVC reference software with 25 Hz, $\pm$ 15 as the search length with Quarter-pel accuracy and

![Computational Time Comparison](image1)

Fig. 3 Computational time comparisons between the proposed technique and the H.264 with two reference frames (H.264-2Ref) against the H.264 with single reference frame (H.264-1Ref)

![Rate–distortion performance comparison](image2)

Fig. 4 Rate–distortion performance comparison using (i) H.264 [1] (ii) OCP [5] (iii) LTR [7] (iv) Proposed scheme on OCP plane (here $TY$-plane) and (v) Proposed scheme on $XY$-plane on ‘Tempete’ video sequence to demonstrate the superiority of the proposed scheme against other relevant schemes
16 GOP size [14]. The frame format is IPPP [14] where the first frame is intra (I) frame and subsequence frames are predicted (P) frames. We have executed all schemes on a PC with Intel(R) Core(TM) 2 CPU 6600 at 2.40 GHz, 2.39 GHz and 3.50 GB of RAM.

The OCP-2Ref and the proposed techniques require some computational time to determine the optimal plane for a video. The proposed technique requires additional time to generate McFIS compared to the H.264-2Ref and the OCP-2Ref schemes. The McFIS generation time is not more than 5% of the overall encoding time of a frame with 15 motion search length for dual reference frames. For calculating the computational time of the proposed method, we have considered the computational time of (i) OPC determination, (ii) scene change and (iii) McFIS generation. In the proposed scheme, we need to check whether there is scene change. We need to reset background model at scene change. 

Fig. 3 shows the times of computational complexity requirements by the proposed technique and the H.264-2Ref scheme against the H.264 with single reference frame (H.264-1Ref) for six standard video sequences namely ‘tempete’, ‘mobile’, ‘Paris’, ‘silent’, ‘container’, ‘bus and waterfall’ of CIF (352 × 288) format. Note that the OCP-2Ref scheme requires roughly the same amount of time with that of the H.264-2Ref since the OPC determination in the OCP scheme only takes about 0.22% of the computation of the H.264 [5]. The experimental results reveal that the proposed scheme requires the least time for the videos with smooth motion videos such as ‘Paris’, ‘silent’ and ‘container’, and on the other hand, the proposed technique requires more time for the high motion videos such as ‘Tempete’, ‘mobile’ and ‘waterfall’ video sequences compared to the H.264-2Ref scheme. The main reason of the reduced computational time requirement of the proposed technique for smooth motion video sequences is that the proposed scheme produces more large block modes (16 × 16, 16 × 8, 8 × 16, 8 × 8 and skip modes etc.) using the McFIS as a reference frame for the smooth/ uncovered background areas compared to the H.264-2Ref. Thus, the proposed scheme saves computational time by

Fig. 5 Rate–distortion performance by the proposed, the optimal compression plane and the H.264 with two reference frames techniques using ten standard video sequences with different resolution (i.e. CIF-352 × 288, 4CIF-704 × 576 and HD-960 × 512) and different contents
terminating motion estimation at large block modes without exploring small block modes (i.e. $8 \times 4$, $4 \times 8$ and $4 \times 4$). For example, for silent video sequence, average larger blocks in the proposed scheme are around 90% compared with 87% in the H.264-2Ref scheme. From figure we may conclude that the proposed technique is comparable in terms of computational time requirement with the OCP-2Ref and the H.264-2Ref schemes. We note that an efficient version of the proposed technique can be formulated by reducing the search range for the McFIS referencing as the McFIS is only used to refer the background that has no or little motion. Moreover, the operations needed for the McFIS generation are constant against the motion search length, and thus, the computation of McFIS modelling is insignificant in comparison with a large motion search length (e.g. $\pm 32$ or more). We need to determine OPC for a number of frames (we have used 128 frames in the experiment), and then encode those frames using the OCP. Thus, for each 128 frames we need to determine OCP as well as new McFIS. If we lose any data for the bad transmission, McFIS may be a better solution to recover the data as the background model and the McFIS have the better history of the pixel intensities over the frames than any other single frame. Scene change detection can help to reset McFIS for a scene change situation [13].

We have compared the proposed scheme against a number of relevant existing schemes such as (i) H.264 [1], (ii) OCP [5], (iii) high-quality dual reference frames with an LTR frame (i.e. H.264-LTR) [7] and (iv) the proposed scheme on XY-plane to demonstrate the superiority of the proposed scheme against other contemporary and relevant schemes. Fig. 4 shows the rate–distortion performance of the schemes using Tempete video sequence. Since Tempete has camera motion and high texture, we have selected Tempete video sequence for comparison. The figure shows that the proposed scheme on XY-plane outperforms the H.264-2Ref scheme on XY-plane. The result demonstrates that McFIS is better second reference frame compared with other reference frame normally used in the H.264. The H.264-LTR scheme [7] outperforms the H.264-2Ref and the proposed scheme on XY-plane because of the high quality of an LTR frame used for the second reference frame. The OCP scheme outperforms all schemes except the proposed scheme on OCP. For the Tempete video sequence the TY-plane is selected as an optimal plane using (3). Finally, figure reveals that the proposed scheme on optimal plane outperforms all relevant schemes. Our hypothesis in this paper is that treating a video as a 3D data cube revolutionises the traditional video features such as background, motion, object, zooming and panning. When we rearrange Tempete video into other planes some of the features such as zooming, object movement and camera motion transformed into simpler background and smooth motion which are better approximated by the McFIS and inter/intra modes of the immediate previous reference frame, thus, we obtain better rate–distortion performance using the proposed scheme. Same trend of performance is also observed for other video sequences.

The rate–distortion improvement by the proposed scheme is because of the inclusion of the McFIS as a second reference frame for the optimal plane. Although, the McFIS is originally designed for the traditional XY-plane to capture uncovered background, McFIS has the capability to capture any uncovered background for any other directions. The main idea of McFIS modelling is that it considers background if a pixel position has non-variance nature of
its intensities over the time. Thus, it is equally applicable for any directions of coding; however, the benefit depends on the amount of background/uncovered background for a particular coding directions. As the correlation of the OCP is the best among all directions for a video, McFIS also provides more background for the OCP. We have provided more rate–distortion performance (see Fig. 5) comparison using different video sequences for better understanding of the proposed scheme against the H.264-2Ref and the OCP-2Ref algorithm. For better visibility of the curves, we do not include the rate–distortion performance using the H.264-LTR and the proposed scheme on XY-plane in Fig. 5, because the proposed scheme outperforms those schemes (see Fig. 4). We have selected ten standard video sequences with different motion patterns, different contents and different resolutions such as CIF (352 × 288), 4CIF (704 × 576) and HD-high definition (960 × 512) format. The videos are silent (CIF), Paris (CIF), Tempete (CIF), bus (CIF), container (CIF), mobile (CIF), Coastguard (CIF), waterfall (CIF), Tempete (4CIF) and padestrianarea (HD). For all video sequences except the ‘bus’ and ‘Coastguard’ sequences, the OCP scheme selects different plane (TX or TY) as the OPC instead of traditional XY-plane (which is normally used to encode a video). Thus, the rate–distortion performance of the OCP-2Ref and the H.264-2Ref are the same for bus and Coastguard video sequences. However, for all video sequences, the proposed scheme outperforms the OCP-2Ref and the H.264-2Ref schemes because of the exploitation of the newly generated background by the McFIS. Fig. 5 confirms that the proposed technique outperforms the OCP-2Ref and the H.264-2Ref schemes by 0.47 and 1.25 dB on average for ten video sequences, respectively. Fig. 5 also shows rate–distortion performance using same sequence (i.e. Tempate) with two different resolutions (i.e. CIF and 4CIF). The experimental data show that for a given quantisation parameter (QP) the peak signal-to-noise ratio (PSNR) difference between the proposed scheme and the H.264 is less for higher resolution. In the experiment, we have taken the same number of temporal frames for both cases. Thus, we can conclude that for a given QP and the same number of temporal frames, the image quality improvement by the proposed scheme diminishes with the resolutions compared to the H.264. Deng et al. [15] observed that H.264 performs better for higher-resolution videos if the block size and search length are also increased. In future, we will investigate the performance of the proposed scheme with higher resolutions using different larger block sizes, larger search lengths and different temporal frames. We have also observed that sometimes the OCP selects XY-plane for video sequences with high motion (e.g. bus and Coastguard). At present, the OCP determination algorithm could not always provide OPC as the brute-force approach, although significant improvement has been demonstrated. Sometimes, McFIS is not effective if a video sequence has lots of camera motions and frequent scene changes. Our future plan is to derive a suitable pre-processing OCP determination technique which agrees more with the brute-force approach so that more improvement can be yielded. We also investigate for improving the McFIS modelling technique in the camera and scene change scenarios by considering global motion estimation in future. We do not expect any significant performance difference for the proposed method when colour video is used instead of the monospectral video because the most of the coding techniques first process (e.g. motion estimation and motion compensation) a video based on the luminance component and then encode colour components based on the relationship with the luminance component. However, in future we also investigate performance of the proposed scheme with colour videos.

4 Conclusions

In this paper, we proposed a new video coding technique using the OPC and the McFIS to improve the coding performance. The OCP technique considers a video as a 3D data cube by relaxing the physical meaning of a video (which is normally considered as the collection of temporal images). In the proposed technique, an OCP (among XY-, TX- and TY-planes) is determined based on the statistical redundancy, and then a dynamic background modelling scheme is applied to the OCP to find a background frame to encode the current frame using the background frame and the immediate previous frame as two reference frames. Owing to the newly generated background and smooth areas through the rearranging the frames in the OCP scheme, the proposed technique improves the rate–distortion performance. The experimental results with different motion patterns and characteristics reveal that the proposed technique improves the video quality significantly compared to the relevant state-of-the-art methods.

5 References

1 ITU-T Recommendation H.264: advanced video coding for generic audiovisual services, March 2009