A Long Term Reference Frame for Hierarchical B-Picture based Video Coding

Manoranjan Paul, *Senior Member*, *IEEE*, Weisi Lin, *Senior Member*, *IEEE*, Chiew Tong Lau, *Member*, *IEEE*, and Bu-Sung Lee, *Member*, *IEEE*

***Abstract*- Generally, H.264/AVC video coding standard with *hierarchical bi-predictive picture* (HBP) structure outperforms the classical prediction structures such as 'IPPP...' and 'IBBP...' through better exploitation of data correlation using reference frames and unequal quantization setting among frames. However, *multiple reference frames* (MRFs) techniques are not fully exploited in the HBP scheme due to the computational requirement for B-frames, unavailability of adjacent reference frames, and with no explicit sorting of the reference frames for foreground or background being used. To exploit MRFs fully and explicitly in background referencing, we observe that not a single frame of a video is appropriate to be the reference frame as no one covers adequate background of a video. To overcome the problems, we propose a new coding scheme with the HBP which uses the *most common frame in scene* (McFIS), generated by background modeling, as a *long term reference* (LTR) frame for the third uni-predictive reference frame, so that foreground and background areas are expected to be referenced from the two frames in the HBP structure and the McFIS respectively. There are two approaches to generate McFIS under the proposed methodology. In the first approach, we generate a McFIS using a number of original frames of a scene in a video and then encode it as an I-frame with higher quality. For the rest of the scene this generated I-frame is used as an LTR frame. In the second approach, we generate a McFIS from the decoded frames and then use it as an LTR frame, without the need to encode the McFIS. The first and the second approaches are suitable for a video with static background and dynamic background respectively. In general, the second approach requires more computational time compared to the first approach. The experiments confirm that the proposed scheme outperforms three state-of-the-art algorithms by improving image quality significantly with reduced computational time.**

***Index Terms*—** Most common frame in scene (McFIS), video coding, Long term reference frame, and uncovered background

I. INTRODUCTION

H.264/AVC video coding standard improves *rate-distortion* (RD) performance significantly compared to its predecessors and competitors due to its state-of-the-art techniques in *Intra* (I)- and Inter- (i.e., *predictive* (P) and *bidirectional* (B)) frame coding [1]-[4]. Among them variable block size *motion estimation* (ME) with fractional pixel accuracy and *multiple reference frames* (MRFs) are the most important techniques in Inter-frame coding. However, this improvement comes at the expense of huge ME computational time. According to the analysis conducted by Huang *et al*. [5], ME consumes around 50~90% of overall encoding time. Obviously ME computational time also varies with the number of reference frames, precision of ME, etc. A comprehensive performance and complexity analysis on a tool-by-tool basis is provided in [6]. The MRFs technique facilitates better predictions than using one reference frame, targeting at video with features such as repetitive motion, uncovered background, non-integer pixel displacement, lighting change, etc. Wiegand *et al*. [7] propose a long-term memory scheme which used up to 50 previously decoded frames to determine the best motion vector. The number of reference frames in practical applications is limited due to (i) the requirement to identify the reference frames, (ii) computational time in ME which increases almost linearly with the number of reference frames unless fast motion search algorithms are used, and (iii) memory buffer size to store decoded frames in both encoder and decoder. Typically the number of reference frames varies from 1 to 5. If the cycle of features (i.e., repetitive motions, uncovered background, etc.) exceeds the number of reference frames used in the MRFs where reference frames are consecutive, we may not get any coding improvement and therefore much of the computation with MRFs is wasted.

This work is partially supported by the SINGAPORE MINISTRY OF EDUCATION Academic Research Fund (AcRF) Tier 2, Grant Number: T208B1218 and Faculty Compact Funding, Charles Sturt University.

M. Paul is with the School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW-2795, Australia (phone: +61-2-6338 4260; Fax: +61-2- 6338 4260; e-mail: mpaul@csu.edu.au.

W. Lin is with the School of Computer Engineering, Nanyang Technological University, Singapore (phone: +65- 6790 6651; e-mail: WSLIN@ntu.edu.sg. C. T. Lau is with the School of Computer Engineering, Nanyang Technological University, Singapore (phone: +65- 6790 5047; e-mail: ASCTLAU@ntu.edu.sg. B-S Lee is with the School of Computer Engineering, Nanyang Technological University, Singapore (phone: +65- 6790 5371 Fax: +65- 6792 6559; e-mail: ebslee@ntu.edu.sg.

Sufficiently widely spaced MRFs can cover the whole cycle of features, but this approach is not practically feasible as with the increase in the number of reference frames, the memory requirement and the motion searching complexity increases significantly [11]; in addition, the proper MRFs in such cases are not easy nor practical to determine.

A number of techniques including [5][8]-[10] reduce computational time associated with MRFs without significantly sacrificing image quality. They achieve this by avoiding unnecessary searching for various reference frames based on some assumptions about the correlation between the current frame and the previous frame [5][9], the homogeneity among the areas of the same object [8], and the accuracy of moving pixel segmentation [10].

To reduce computational time associated with MRFs, a number of techniques [11]-[14] use only two reference frames: a *long term reference* (LTR) frame and a *short term reference* (STR) frame. When the *n*th frame is being encoded, the (*n*-1)th frame is used as the STR and the (*n*-*N*)th frame (where *N*>1) is used as the LTR for *N* frames. The LTR then jumps forward by *N* frames and again remains the same for encoding the next *N* frames. In this technique for *N* frames, each encoded frame is used as the STR and the *N*th frame is used as the LTR. This process also allows high quality coding for the LTR by allocating more bits [12][13] to improve the overall performance. The performance improvement is heavily influenced by the *jumping* parameter and the bit allocation strategy [11]. Liu *et al*. [11] propose an adaptive jump updating scheme with optimal LTR selection based on the RD relationship.

The basic assumption of the dual reference frames for encoding the current frame is that the STR would be referenced for local motion i.e., moving areas, and the LTR would be referenced for still regions i.e., background areas of the frame [13]. Thus, it would be an implicit background (from the LTR frame) and foreground (from the STR frame) referencing system. Our hypothesis is that the better LTR frame should have maximum correlation (i.e., similarity) with the other frames and/or maximum background areas compared to the other frames, so better RD performance can be achieved with less computation.

Recently a video codec popularly known as VP8 [15]-[18] also recommends dual reference frames using a virtual (or constructed) reference frame called a *golden frame* for background referencing, which retains a frame´s worth of decompressed data from the arbitrarily distant past (i.e., equivalent to an LTR frame). The codec can update any part of that frame at any future point in time. The golden frame or virtual reference frame is one kind of LTR frame which needs to be updated after a regular/irregular interval with higher quality based on (i) the speed of motion, (ii) how well each frame predicts the next one, and (iii) how frequently the golden frame is selected as the best choice reference frame for encoding *macroblocks* (MBs). A golden frame can be regarded as a better LTR frame because an LTR frame is unaltered once decided while a golden frame may be updated to maintain its relevancy.

Wilkins *et al*. [18] propose a construction method of the golden frame based on local and global motion vectors of the MB-level. The proposed golden frame developed using image super resolution may have a larger size than the video frame size. Even before [18], techniques in [19]-[22] construct *simple* golden frames such as background-frame or background memory for efficient video coding based on block-based motion information. Mukawa and Kuroda [23] introduce background replenishment and the updating technique at the pixel level using the frame-difference signal to a certain threshold (against cumulative differences for a block) to decide whether a picture element belongs to a static or nonstatic region. For dynamic background adaptation, the background pixel value is changed by ±1 out of 256 toward the new reconstructed signal value. Hepper [24] modifies the technique by evaluating the change information of successive pictures to adapt new background information quickly. Due to the dependency on block-based motion vectors [19]-[22] (or block-based processing [23] [24]) and lack of adaptability in multi-modal backgrounds for dynamic environments, the background frame generation techniques in [19]-[24] could not perform well. Uncovered background can also be efficiently encoded using sprite/multiple-sprite [25] coding through object segmentation. Most of the video coding applications could not tolerate inaccurate video/object segmentations and expensive computational complexity incurred by segmentation algorithms. Wilkins *et al.*'s approach [18] suffers from the same drawbacks of the sprite/multiple sprite coding approach and the approaches in [19]-[24], as it is regarded as the amalgamation of those approaches. Moreover, updating any part of the golden frame based on the newly uncovered background is not effective due to the limitation of the accuracy and computational requirement for detecting the newly exposed background.

In general, H.264 with HBP [26][27] outperforms the conventional prediction structure (such as 'IPPP...' or 'IBBP...') due to its better exploitation of bi-predictive data correlation through preceding and succeeding frame referencing (i.e., two reference frames) together with unequal quantization/bit allocation among frames at different levels. To maintain bi-predictive referencing, the HBP may improve RD performance for motion areas using shorter-distance reference frames and for background areas using longer-distance reference frames. Moreover, to refer static and uncovered background areas, B-frame is not the right option as static and uncovered background areas have no motion. Thus, due to the unavailability of the adjacent reference frames or with no explicit background referencing being used, the HBP scheme cannot take full advantage of the MRFs (i.e., more than 2 reference frames) benefits. Note that the golden/LTR frame can be regarded as MRFs.

A *ground truth background* of a scene in a video sequence can be a better choice for a golden frame compared to a random frame to conform to the implicit background

/foreground referencing assumption (i.e., the LTR is for static region reference and the STR is for moving region reference). In [28][29], the concept of McFIS (*most common frame in scene*) was introduced as a reference frame for normal/uncovered background regions of the current frame, based upon *dynamic background modelling* (DBM) [30]-[32]. Zhang *et al*. proposed a number of algorithms [38][39] using background modelling for surveillance video coding with a number of video sequences captured using stationary cameras. Zhang *et al*. [38] introduced two kinds of frames (namely background frame and difference frame) for input frames to represent the foreground/background. The background frame is modelled and encoded with very high quality. The difference frame is encoded using 9-bit H.264/AVC encoding. The experimental results show that the algorithm [38] outperforms H.264 for surveillance videos captured by stationary cameras by a significant margin. As the algorithm in [38] does not include any *scene change detection* (SCD) and *adaptive group of pictures* (AGOP) strategy, it is not suitable for video sequences with scene change and camera motion in its current form. The algorithm (in its current form) [38] needs many bits to encode high quality background frame at each scene change point. The requirement of bits for high quality background is sometimes higher compared to the saving of bits using the high quality of background frame in subsequent frames due to the short length of scene.

In this paper, we propose a new HBP scheme using the McFIS as a golden frame to be the third reference frame so that the background regions can be referenced from the McFIS. We observe that an I-frame requires 2~10 times more bits (depends on the content of the videos and operational bit rates) compared to an inter (i.e., P or B)-frame for the same image quality. Generally, if a sequence does not contain any scene changes or extremely high motion compared to the adjacent frames, insertion of I-frames degrades the coding performance, although insertion of I-frame facilitates texture refreshing, error propagation control, and random access support. Therefore, we need to insert I-frames based on the AGOP determination and SCD algorithm. In the HBP prediction structure, AGOP determination based on SCD is quite challenging, due to the fact that the existing SCD determination algorithms [33]-[35] are based on the correlation of the temporal order of the frames in a video whereas the HBP does not maintain the temporal order of the frames while encoding. Therefore, we also devise an AGOP determination approach using adaptive SCD thresholding based on the same McFIS as the McFIS is capable of representing the stable region of a scene.

The prior works [28][29][40] introduced the McFIS concept; however, the generation process and application of the McFIS are different in the proposed scheme. In the prior works, the modeling of the McFIS uses frames (either reconstructed or original) in sequential time order (i.e., temporally in one direction). In the proposed method, we model the McFIS according to the hierarchical order (i.e., not sequential order) and use the McFIS for referencing purpose in hierarchical order. The experimental results indicate that using the McFIS as a third reference frame improves the RD performance significantly compared to the HBP with a decoded frame as the third reference frame. The results also reveal that the proposed schemes save up to 28% and 11% computational time respectively compared to the HBP with a third reference frame. As we know that multi-view video coding scheme H.264/MVC [41] uses HBP referencing approach for coding efficiency, thus, the approach adopted in the proposed scheme will open new research avenues to improve the coding efficiency of the multi-view video coding scheme by incorporating the McFIS in H.264/MVC.

The rest of the paper is organized as follows: Section II illustrates the motivation of the proposed work by analyzing frame similarity and percentage of background in video, and then describes an intuitive HBP structure with three reference frames. Section III discusses the proposed new HBP prediction structure with the McFIS, which is generated using original frames and used as an I-frame for the third reference frame. Section IV describes how to apply the proposed concept with the McFIS, which is generated from coded frames and used as the golden frame in the HBP structure. Section V compares the computational time with relevant existing schemes. Section VI describes the experimental setup and analyzes the overall experimental results, confirming that the new scheme outperforms the conventional HBP and the HBP using LTR as the third reference frame. Finally Section VII concludes the paper.

II. RELATED DISCUSSION ON VIDEO CODING WITH HBP

A. Frame Similarity and Percentage of Background

The basic assumption of the dual reference frames for encoding the current frame is that the STR would be referenced for local motion i.e., moving areas, and the LTR would be referenced for still i.e., background areas of the frame [13]. Based on this assumption, our proposition is that a frame would be a better LTR frame if it provides better similarity and/or more background with other frames. The frame *similarity* is defined as the inverse of the *mean absolute difference* (MAD) between co-located pixels of two frames. Note that we do not consider global motion estimation in the similarity measurement as the proposed background modeling does not incorporate global motion. A pixel is defined as a background pixel if the pixel intensity difference between two co-located pixels is within one gray level. If the first 50 frames of two standard video sequences *Paris* and *Silent* are used, Fig. 1 shows the average (i.e., using all pixels in a frame) similarity and percentage of background with respect to a frame (curves for the first 25 frames are shown in Fig. 1 (a) & (b)) with the rest of the frames in the video (without comparing itself). For example, a pixel in frame *t* is defined to be a background pixel if the difference between its value and the co-located value in all frames of the 50 frames is within one gray level. The curves correspond to different frames. Frame 15 provides the maximum similarity (based on the smallest MAD) for the other

frames with *Paris* (see Fig. 1(c) which is derived from Fig. 1(a) after averaging). On the other hand, Frame 19 has the maximum background regions compared to the other frames for the *Paris* sequence (see Fig. 1 (d) which is derived from Fig. 1(b) after averaging). As the difference is not significant (this is the reason that a good LTR is not easy to find from video frames, since, more often than not, it does not exist in natural video!), we need to propose a new method in this paper for finding a better LTR frame i.e., golden frame compared to the existing LTR frames (see the next paragraph for more details).

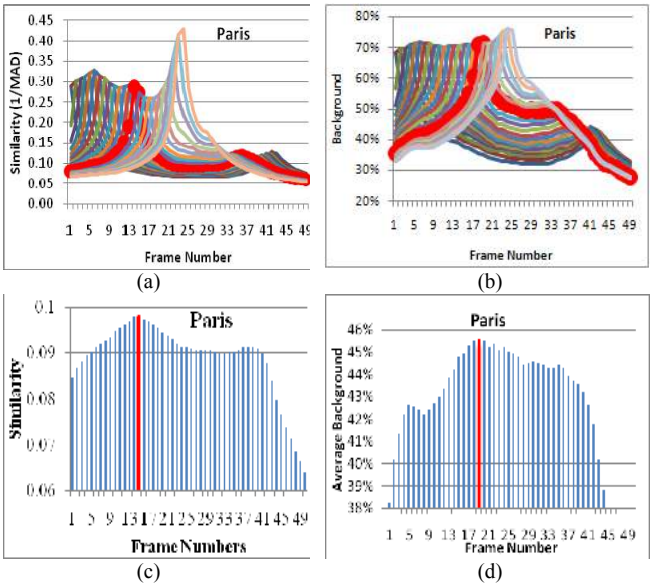


Fig. 1: Frame similarity and percentage of background with respect to other frames: average similarity and percentage of background among frames of Paris video sequence: (a) similarity; (b) percentage of background among frames; (c) average similarity of (a); (d) average background of (b); and the curves with bold red markers denoting the frames corresponding to the maximum similarity or the maximum background among the first 25 frames.

Although the aforementioned frames (with bold red markers in the figure) represent the best (apparently!) frames (among 25 frames) to be the LTR, the similarity or the amount of background of these frames is not significantly distinguishable from the other frames (i.e., via the comparison of the area under each curve). Therefore, we explore the use of McFIS instead in the rest of this paper. The golden frame [18] can provide better performance due to updating (based on block-motion, prediction accuracy, etc.) compared to the static LTR frame.

A. The HBP Prediction Structure

H.264/AVC has the flexibility to decouple the coding and display order of frames. Moreover, any frame can be marked as a reference frame and used for prediction of the following frames independently of the corresponding slice types [26]. This flexibility and referencing are explored in the HBP prediction structure. Fig. 2 (a) shows a typical HBP structure [26] with encoding image types, coding and display order of a GOP (comprising 16 frames in a GOP for this case). To get the better coding performance of the HBP structure, different *quantization parameters* (QPs) are used for different hierarchy levels. Normally, finer quantization is applied to frames which are more frequently used as reference frames for the other frames directly or indirectly. For example, Frame 9 (according to the display order) in Fig. 2 (a) is used more frequently (6 times directly for Frames 5, 7, 8, 10, 11 & 13, and 8 times indirectly for Frames 2, 3, 4, 6, 12, 14, 15 & 16) compared to any other frame (except the first frame) as a reference frame. Note that the first frame is used 4 times directly and 11 times indirectly as a reference frame.

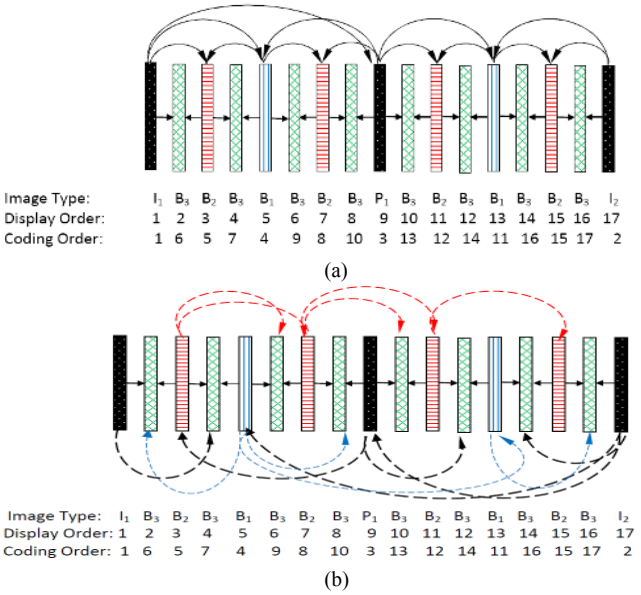


Fig. 2: Referencing of hierarchical B-picture prediction structure where 16 frames are used as a GOP, (a) hierarchical B-picture prediction structure using two reference frames (i.e., bidirectional) and (b) the extra i.e., third reference frame (hierarchical B-picture prediction structure is similar to (a), thus they are not included in (b)).

Due to the great flexibility of H.264/AVC on coding order, display order, and referencing, the MRFs techniques can be applied to the HBP. Intuitively, we make a three-reference HBP technique (HBP-3Ref) which is shown in Fig. 2 (b) based on the availability and closeness of the frames for referencing. Note that the HBP-3Ref has also the similar HBP prediction structure shown in Fig. 2(a), thus the HBP references are not included in Fig. 2(b). The HBP-3Refs scheme uses the two closest neighboring frames (subject to availability in terms of coding order) from two different directions (i.e., from List_0 and List_1) for bi-predictive ME and *motion compensation* (MC). The scheme also uses the next closest neighboring frame for single directional ME and MC. Actually the third reference frame can be added to List_0 or List_1, however, we do not consider the third frame for bi-directional referencing with any other reference frame to reduce the computational complexity. The ultimate decision is taken for mode selection based on Lagrangian optimization [42] between the results of the two motion compensated techniques. For a frame in Fig. 2 (b),

frames with outgoing arrows are the third reference frame. Take Frame 5 (according to the display order) as an example. Frame 1 and Frame 9 are used as two bi-predictive frames and Frame 17 is used as a third reference frame. On the other hand, to encode Frame 2, Frame 1 and 3 are used as bi-predictive frames and Frame 5 is used as the third reference frame.

III. HBP STRUCTURE WITH MCFIS AS AN I-FRAME

Due to the limited effectiveness of the MRFs technique in the HBP structure and to exploit the implicit background/foreground referencing, we propose a new coding scheme with HBP (named McFIS-I) using the McFIS (which is encoded as an I-frame at the beginning of a scene in a video) as a golden frame to be used as a third reference frame. In this scheme, the moving region of the current frame is expected to be referenced from the two bi-predictive reference frames, while the static/uncovered background area is expected to be referenced from the McFIS.

When we select from three reference frames, our hypothesis is that the third reference frame (McFIS, LTR, or the farthest reference frame) is mainly used as a reference frame for static background or uncovered background areas. As the McFIS is used as a reference frame for static background areas, reference areas should be co-located with the current frames for static cameras, and the motion vector length should be zero. To alleviate computational complexity, we use the third reference frame as a normal reference frame (i.e., uni-predictive ME and MC using this reference frame instead of bi-predictive using two reference frames) and then used the Lagrangian Multiplier [42] to select the better one between this and the bi- predictive ME and MC using the first and the second reference frames.

Note that the McFIS (like the concept of a golden frame [15]-[18]) is not displayable at the decoder. The McFIS is generated using a number of frames of a scene and encoded as an I-frame with high quality. To avoid multiple McFISes within a scene of a video sequence we also introduce an AGOP determination technique using a new SCD algorithm. Note that the existing SCD algorithm is not effective in the HBP structure as those techniques are derived based on the temporal ordering of frames.

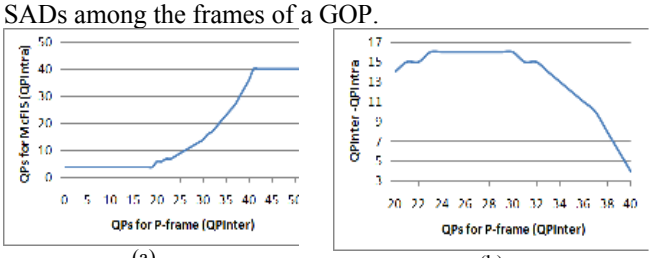
B. Structure of the Proposed Predictive Scheme with McFIS and HBP

The proposed scheme (McFIS-I) is based on the HBP structure where we first generate a McFIS from a number of frames of a scene in a video (the generation process will be described in Section III.B). Then we encode it as an I-frame. All frames of a scene are encoded as either B-frames or P-frames with an extra reference frame (i.e., McFIS) unless a scene change occurs. If there is a scene change, we generate a new McFIS and encode it as an I-frame, and all frames of that scene will be encoded either as B- or P- frames. We note that the first frame of a GOP (according to display order) is encoded as a P-frame using the McFIS in the proposed scheme whereas in the conventional scheme (see Fig. 2), it is encoded as an I-frame.

As the McFIS plays an important role in the proposed scheme, we encode it with relatively finer quantization compared to the inter frames. We compute the QP for the I-frame (QP_{Intra}) as follows

$$QP_{Intra}=\begin{cases}40 & if\ QP_{Inter}>40\\ \left\lfloor e^{0.09\times QP_{Inter}}\right\rfloor & if\ 20\le QP_{Inter}\le 40\\ 4 & if\ QP_{Inter}<20\end{cases} \quad (1)$$

where QP_{Inter} is the QP of the *P*-frame, and $\lfloor . \rfloor$ denotes the floor operation. QP_{Intra} is also plotted against QP_{Inter} in Fig. 3(a) for the entire range of QPs. At high bit rates (i.e., around QP_{Inter} = 20) the quality of the inter-frames is already higher so we need a higher quality McFIS to improve overall RD performance. To ensure this we use finer quantization for the I-frame at the high bit rates and coarser quantization at the low bit rates through the formulation of Equation (1). Fig. 3(b) shows the difference between QP_{Inter} and QP_{Intra}, for the middle range of QPs. There is little effect in the RD performance of the proposed method based on our experimental results if we use variable QP_{Intra} for QP_{Inter}>40 and QP_{Inter}<20 to encode the McFIS. For P or B frames, we maintain the same quantization variations mentioned in [26] for different hierarchy levels. Based on a given quantization parameter for key pictures QP_0, the remaining quantization parameters for pictures of a given temporal level $k > 0$ are determined by $QP_k = QP_{k-1} + (k = 1 ? 4 : 1)$. For SCD we propose an adaptive thresholding scheme (to be described in Section III.D) based on the ratio of the SADs among the frames of a GOP.



(a) (b)

Fig. 3: The relationship between QPs of Intra-frame (i.e., McFIS) and QPs of Inter-frame (P-frame) of the proposed scheme; (a) QP_{Inter} vs. QP_{Intra} for entire range of QP_{Inter} (also shown mathematically in Equation (1)), and (b) difference between QP_{Inter} and QP_{Intra} against QP_{Inter} for middle range of QPs (which are mostly used in typical video coding applications).

C. McFIS Generation

Each pixel position of a scene is modeled independently by a mixture of *K* Gaussian distributions [29]-[32]. A pixel position may be occupied by different objects and backgrounds in different frames. Each Gaussian model represents the intensity distribution of one of the different components e.g., objects, background, shadow, illumination, surround changes (like clouds in an outdoor scene), etc. observed by the pixel position in different frames. A Gaussian model is represented by the recent pixel intensity, *mean* of pixel intensity, pixel intensity variance, and weight (i.e., how many times this model is satisfied by the incoming pixel intensity). The system starts with an empty set of models and initial parameters. If the maximum number of models allowable for a pixel is three, we

can get a maximum of $3\times H\times W$ models for each video scene where $H\times W$ is the resolution of a frame.

For a pixel, the McFIS generation algorithm is shown in Fig. 4 which takes the recent pixel intensity (i.e, pixel intensity at the current time, X^t) and existing model for that pixel position (if any) as inputs and returns background pixel intensity (i.e., McFIS) and the background model. For every new observation X^t at the current time t, it is first matched against the existing models in order to find one such that the difference between the newly arrived pixel-intensity and the *mean* of the model is within 2.5 times of the *standard deviation* (STD) of that model. If such a model exists, its associated parameters are updated with a learning rate parameter. The recent pixel intensity of the model is replaced by the newly arrived pixel intensity. If such a model does not exist, a new Gaussian is introduced with the intensity as a mean (μ), a high STD (σ), recent pixel value (γ), and a low weight (ω), and the least probable model is evicted. The least probable model is determined based on the lowest value of ω/σ among the models. We have fixed the initial parameters in this implementation as follows: maximum number of models for a pixel K = 3, learning rate α = 0.1, weight ω = 0.001, and variance σ = *3*0 as mentioned in [28]-[32]. Obviously, there is influence of the initial parameters in the background modeling and eventually in coding performance. For example, if we use K =2 instead of 3 we may miss the stable background for a scenario where two foregrounds (e.g., objects and clouds) appear after a stable background for a given time. If we use α = 0.01 instead of α = 0.1, the model takes longer to update the current background/foreground. A small weight makes sure that when a new model is introduced, it could not be selected as a background model immediately after its introduction. Thus, a model could not be selected as a background model when the model is introduced due to a noise. The variance controls the range of pixel intensities which can be covered by a model. In the proposed scheme, we tried different initial values, and then fixed the values for better performance using different video sequences. Note that we use all frames from the first two GOPs for McFIS generation and then used it to encode all frames (i.e., all GOPs) of the scene. We do not change the McFIS unless a scene change occurs. A new McFIS is generated, encoded, and then used for a new scene. Of course, this is just a possibility of configuration for encoding.

We assume that the k-th Gaussian at time t represents a pixel intensity with mean μ_k^t, STD σ_k^t, the recent value γ_k^t, and the weight ω_k^t such that $\sum_{\forall k}\omega_k^t = 1$. The learning parameter α is used to balance the contribution between the current and past values of parameters such as weight, STD, mean, etc. After initialization, for every new observation X_t at the current time t, it (i.e., X_t) is first matched against the existing models in order to find one (e.g., kth model) such that $|X^t - \mu_k^{t-1}| \le 2.5\,\sigma_k^{t-1}$. If such a model exists, update the corresponding recent value parameter γ_k^t with X^t. Other associated parameters are updated with learning rates as follows [29]:

$$\mu_k^t = (1-\alpha)\mu_k^{t-1} + \alpha X^t; \quad (2)$$

$$\sigma_k^{t\,2} = (1-\alpha)\sigma_k^{t-1\,2} + \alpha(X^t - \mu_k^t)^T(X^t - \mu_k^t); \quad (3)$$

$$\omega_k^t = (1-\alpha)\omega_k^{t-1} + \alpha, \quad (4a)$$

and the weights of the remaining Gaussians (i.e., l where $l \neq k$) are updated as

$$\omega_l^t = (1-\alpha)\omega_l^{t-1}. \quad (4b)$$

The weights are then renormalized. If such a model does not exist, a new model is introduced with $\gamma = \mu = X^t$, σ = 30, and ω = 0.001 by evicting the K-th (i.e., the third model based on w/σ in descending order) model if it exists.

Algorithm [Ψ^t, Ω^t] = McFISgeneration (X^t, Ω^{t-1})
Parameters: X^t is the pixel intensity at time t; Ω is the structure of K Gaussian mixture models at time t-1 where each model contains mean, STD, weight, and recent value i.e., $\{\mu_k^{t-1}, \sigma_k^{t-1}, \omega_k^{t-1}, \gamma_k^{t-1}\}$; Ψ^t is the background pixel intensity i.e., McFIS at time t.

For the first time $\Omega_1^t = \{X^t, 30, 0.001, X^t\}$; $\Psi^t = X^t$;
otherwise
  IF ($|X^t - \mu_k^{t-1}| \le 2.5\,\sigma_k^{t-1}$ for any $k \le$ K
    Update μ_k^t, $\sigma_k^{t\,2}$, ω_k^t, according to Equations (2), (3), and (4); $\gamma_k^t = X^t$;
  ELSE
    Find the maximum number of models, τ in Ω;
    IF (τ <K)
      $\mu_{\tau+1}^t = X^t$; $\sigma_{\tau+1}^t = 30$; $\omega_{\tau+1}^t = 0.001$;
      $\gamma_{\tau+1}^t = X^t$;
    ELSE
      $\mu_\tau^t = X^t$; $\sigma_\tau^t = 30$; $\omega_\tau^t = 0.001$; $\gamma_\tau^t = X^t$;
    ENDIF
  ENDIF
  Normalized all ω_k^t so that $\sum_{\forall k}\omega_k^t = 1$;
  $\Omega_k^t = \{\mu_k^t, \sigma_k^t, \omega_k^t, \gamma_k^t\}$ for all k;
  Sort Ω_k^t based on ω^t/σ^t in descending order;
  $\Psi^t = \mu_1^t$;

Fig. 4: Pseudo code for McFIS Generation algorithm.

To get the background pixel intensity from the DBM technique above for a particular pixel, we take the *mean* value of the background model that has the highest value of ω/σ. In this way we can make a background frame (comprising background pixels) as the McFIS. Examples are shown in Fig. 5 using the first 50 original frames of *Hall Monitor* video

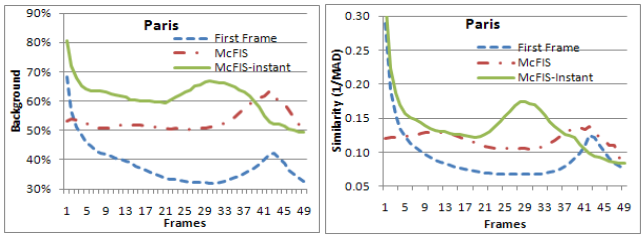
sequences. Fig. 5 (a) shows the 50th frame of videos, and Fig. 5 (b) show corresponding McFISes. The oval in (b) indicates the uncovered background captured by the corresponding McFISes. We also create a background frame shown in Fig. 5(c) using a motion vector-based technique [19]. This background does not capture uncovered background (i.e., no background at the man's position (black regions) due to the non-zero motion vectors for those regions). Thus, this background frame is not suitable for efficient coding compared to the background generated using DBM. To capture the uncovered background by any single frame is impossible unless this uncovered background is visible for one frame and that frame is used as an LTR frame in the relevant existing approaches (this is practically very difficult to ensure). Thus, the McFIS is more suitable as an LTR frame than any single pre-encoded frame.

(a) 50th frame of *Hall monitor* (b) 50th McFIS of *Hall Monitor* (c)Background frame generated by [19]

Fig. 5: Examples of McFIS and uncovered background (inside the ovals) using *Hall Monitor* video sequence, (a) original 50th frame, (b) corresponding McFIS, and (c) background-frame generated from *Hall Monitor* video using [19].

D. McFIS as the Third Reference Frame in the HBP Structure

To verify the quality of McFIS to be an LTR frame, we analyze the similarity (in terms of 1/MAD) of frames and the percentages of background with respect to McFIS and the first frame (used often as a reference frame in the conventional scheme). We use two kinds of McFISes: McFIS and McFIS-Instant. For the former, we use all 50 original frames of a video sequence to generate McFIS and then the final McFIS at the 50th frame is used to calculate the similarity and percentage of background of those 50 frames individually. For the latter, the



(a) (b)

Fig. 6: Correlation between the first frame, McFIS, and McFIS-Instant (generated up to that frame where background (or similarity) is calculated) and other frames; (a) percentages of backgrounds of the 50 frames against the first frame, McFIS, and McFIS-Instant respectively, and (b) similarity (expressed in 1/MAD), McFIS, and McFIS-Instant respectively.

1st to the (i-1)-th frames are used to generate the (i-1)-th McFIS and then the McFIS is used to calculate similarity and percentage of background for the i-th frame; for example, to get similarity and percentage of background for the 20th frame, we generate the McFIS using the first 19 frames. Note that for the 50th frame, the similarity and percentage of background for the McFIS and McFIS-Instant are the same as both use the 50th McFIS. Also note that, in the implementation, the calculation of similarity and percentage of background is not needed. The results are shown in Fig. 6 (note that the curves for the first frame in all 2 subfigures are the same as the corresponding ones in Fig. 1). The figure shows that both McFIS and McFIS-Instant exhibit better correlation compared to the first frame. This figure shows that McFIS would be the better choice to be the LTR frame as it provides more coverage of the background (static and uncovered background). In addition, it has demonstrated that McFIS-Instant performs generally better than McFIS because of its closeness to the current frame for which similarity and amount of background are determined. However, in the actual implementation of lossy encoder/decoder, we cannot use McFIS-Instant as the decoder does not have error-free decoded frames (in terms of quantization and channel error). Thus, we develop a modified McFIS-Instant approach (named McFIS-D) using decoded frames (instead of original frames) which will be described in Section IV.

E. SCD and Adaptive GOP Determination

Insertion of an I-frame without significant scene change exhibits poor RD performance, although it has other advantages such as allowing random access into the sequence and cutting off error propagation for noisy channels. We need to determine SCD and then AGOP for optimum RD performance. The conventional SCD algorithms [33]-[35] for AGOP determination need temporal frame correlation whereas the HBP scheme does not maintain temporal frame order in its coding order. However, modified conventional SCD algorithms can be applied in HBP cases by determining the temporal order and creating scene change tags at the corresponding frames. Moreover, as the McFIS has inherent characteristics of representing a scene by retaining its stable portion (i.e., background); it can be effectively used in the SCD and AGOP determination. We can determine the adaptive threshold based on the SAD ratio of the first few consecutive frames (which are also used to generate the McFIS) of a scene against the McFIS and then use it to detect scene change for the rest of the frames. When scene change occurs, we need to regenerate a new McFIS and a threshold. Obviously, if we first detect the scene change and then insert an I-frame at the appropriate place, the RD performance for the frames after scene change will be higher. Then we need to maintain two McFISes, one for the frames before and another for the frames after a scene change in a GOP. This will be more complicated if there are multiple scene changes within a GOP. Therefore, we just use a single McFIS for a GOP in the proposed scheme, to demonstrate the basic idea.

The scene changes when there is a significant change (i.e., compared to a predefined threshold) in terms of the SAD between the McFIS and the current frame and between the McFIS and the previous frame. Motion of frames in a scene

may not be the same, and thus it is very difficult to determine the threshold based on a small number of frames and then use it for all frames that follow for scene change determination. To determine a better threshold we encode different video sequences using different bit rates by H.264, and observe that the PSNRs calculated using a pair of coded adjacent frames in a scene is within a certain range. The lowest PSNR level that we find by analysing a number of coded video sequences is 18 dB under a wide range of bitrates. Without actual coding, we generate distorted frames where PSNRs are around 18 dB. We add noise to the frames of a video randomly in such a way that the PSNRs of any two adjacent frames are approximately 18 dB. Then we find the maximum SAD among them, which is used for SCD threshold determination. The resultant threshold indicates the possible inter-frame difference bound within a scene (i.e., without scene change), and inter-frame difference with scene change should be above this threshold. Of course, this is just one way to determine the threshold.

TABLE 1

A MIXED VIDEO SEQUENCE FOR SCD AND AGOP

Mixed (QCIF)	Frames	Frames in Mixed Sequence	Scene Change at Frame
Akiyo	150	1~150	~
Miss America	100	151~250	151
Claire	100	251~350	251
Car phone	100	351~450	351
Hall Monitor	150	451~600	451
News	100	601~700	601
Salesman	150	701~850	701
Grandma	100	851~950	851
Mother	100	951~1050	951
Trevor	150	1051~1200	1051, 1111

We calculate SAD using the noisy frames against the McFIS and find the maximum ratio. The maximum ratio indicates the maximum frame variation of the frames with respect to McFIS. Clearly, 18dB distortion from noise is different from 18dB distortion from natural video signal, however, we use it as an approximation and to avoid false SCD in a frame with a rapid object movement, we use a higher threshold i.e., twice the maximum ratio. To test SCD and AGOP we create a mixed video sequence (see Table 1) comprising 1200 frames from ten different video sequences. There are ten scene changes (at Frame 151, 251, 351, 451, 601, 701, 851, 951, 1051, and 1111) where 9 scene changes occur between every two individual video sequences and one scene change occurs within a video sequence (i.e., in the *Trevor* video sequence).

An SCD is detected if the ratio of SAD is greater than the adaptive threshold determined above. Unlike the existing SCD and AGOP determination algorithms, we also use percentage of reference to detect SCD. It is obvious that if the percentages of reference from the third frame (i.e., McFIS) is very low (i.e., zero or near zero), then the McFIS no longer represents the background of the frame. Thus, we assume that when a SCD occurs, a new McFIS needs to be generated. Fig. 7 shows SCD using the proposed (McFIS-I) approach where eight scene changes are detected by the adaptive threshold (indicated by solid circles) and two scene changes are detected by low percentages of reference. In our implementation, we assume that if the percentage of reference is below 2%, then SCD occurs. Scene change detection is always a challenging issue as the definition of scene change can be varied such as abrupt scene change, camera zooming, camera panning, and scene dissolving, etc. In our experiments we combine two metrics to determine a scene change. The first one is based on the difference between two frames, which is quite successful in the case of abrupt scene change, and the second one is based on the percentage of reference which can be successful in the cases of zooming, panning, and scene dissolving scenario.

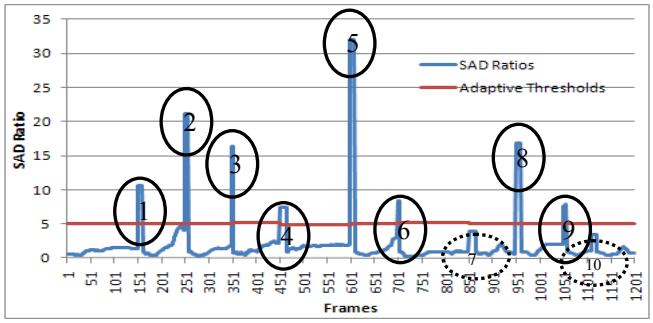


Fig. 7: Scene change detection using the proposed method for mixed video sequence with a total of 10 scene changes in the 1200 frames where solid circle and dotted circle indicate scene change detection by the adaptive threshold based on SAD ratio and the threshold based on the percentages of references respectively.

IV. HBP STRUCTURE WITH MCFIS GENERATED FROM DECODED FRAMES

In the proposed McFIS-I algorithm (described in Section III), McFIS is generated using a number of original frames of a scene and then encoded as an I-frame to be used as a third reference frame for the entire scene. There are several problems with the McFIS-I algorithm, and one of them is the need for transmission of McFIS in each scene. In addition, McFIS can also lose its similarity with the scene if more uncovered background regions are exposed later in the scene. Moreover, it is not easy to determine the best possible number of frames to participate in the McFIS generation process due to the diversified nature of the actual videos.

A. McFIS with decoded frame (McFIS-D)

To address the above mentioned issues we also propose a modified algorithm (named McFIS-D) where the McFIS is generated and updated (in both encoder and decoder) using decoded frames so that no McFIS needs to be encoded for use in the decoder (as mentioned in Section III.C). We note that McFIS-Instant and McFIS-D are equivalent if lossless encoding is used. Obviously the McFIS-D algorithm requires more computational time (both in the encoder and the decoder) compared to the McFIS-I algorithm as it requires updating the background model and generating the background frame after encoding each frame. However, unlike the McFIS-I algorithm, it does not require any extra frames (i.e., McFIS) to encode and transmit. In the McFIS-D scheme the first frame of a scene is

encoded as an I-frame. Note that the McFIS-I scheme is more similar to the LTR-based scheme because after generation it is not modified over time, and on the other hand, the McFIS-D scheme is closer to the golden frame-concept because it is continuously modified using newly coded frames.

Obviously, the background frame generation procedure of the McFIS-I (using original frames) and the McFIS-D (using decoded frames, involving error due to quantization) algorithms are not the same. We need to minimize the distortion of the decoded frames for McFIS generation. Let M_{t-1} and $M_t^{'}$ be the final (t-1)-th McFIS and t-th McFIS after incorporating the (t-1)-th decoded frame respectively. We generate the final t-th McFIS as follows:

$$M_t(x,y) = \begin{cases} \tau M_t^{'}(x,y) + (1-\tau) M_{t-1}(x,y) & \text{if } \left| M_t^{'}(x,y) - M_{t-1}(x,y) \right| < T_p \\ M_t^{'}(x,y), & \text{otherwise} \end{cases} \quad (5)$$

where τ and T_p are the weighting factor and threshold respectively. In our experiment we used $\tau = 0.5$ and $T_p = 5$. If the difference between two co-located pixel intensities in adjacent McFISes is less than T_p, we assume that this happens due to quantization effects rather than foreground/background changes, since the difference is very small. Thus, T_p is used to minimize the distortion due to the quantization and results in RD performance improvements. Since quantization errors of two co-located pixels of subsequent frames would not be the same, the weighted average of the two intensities reduces the distortion. Due to the different quantization values used in different kinds of frames (e.g., I, P, or B) and different hierarchical levels of frames (e.g., B_1 to B_3 in Fig 3), co-located pixels may have different quantization error.

Instead of using the mean value (of the model) as the background pixel's intensity (refer to the fourth paragraph in Section III.B), the McFIS-D approach uses the *recent* pixel value: when a pixel satisfies a model (see the condition in Line 9 of Fig. 4), the pixel intensity is stored as the recent value of the model [32]. In the McFIS-D algorithm, the McFIS is always updated with the latest decoded frame and used as a third reference frame, and thus, the McFIS generated by the recent value provides more reference compared to that of the McFIS-I algorithm due to the closeness with the current frame. This ensures better RD performance. It is evidenced by the McFIS-Instant as demonstrated in Fig. 6. On the other hand, if the McFIS-I algorithm uses the recent value to generate the McFIS, it could not perform well for those frames at the end of a long scene. Due to the longer frame delay for the encoder and decoder, we could not use all frames of a long scene for background modeling and the McFIS generation for the McFIS-I scheme. As we use a small number of frames of a scene to generate McFIS in the McFIS-I scheme, the recent pixel values of the frames loses its relevancy to the frames towards the end of a long scene if there are small changes in the background. Moreover, as the McFIS-I scheme generates background from the original frames, the fluctuation of the background pixel intensity for a pixel position is limited (because there is no quantization distortion), and thus, the mean value works better in the McFIS-I case.

B. Similarity and Difference among McFISes

In this paper we mention three kinds of McFISes: McFIS-I, McFIS-D, and McFIS-Instant.

McFIS-I: In this scheme, we use a number of original frames of a scene to generate McFIS and then McFIS is encoded as an I-frame with finer quantization and used as a third reference frame for the rest of the frames (except the first frame) of the scene. Thus, one extra frame needs to be coded for the McFIS-I scheme. In this scheme we code the first frame of a video scene as a P-frame using the McFIS frame as a reference frame. To generate the McFIS, we use the mean value of the background model.

McFIS-D: In this scheme, the McFIS is generated from the decoded frames. We use the (i-1)-th McFIS as a third reference frame to encode the i-th frame. Unlike the McFIS-I scheme, we do not need to encode it because the same mechanism is used to generate it in the encoder and decoder. In this scheme we encode the first frame of a video scene as an I-frame. To generate the McFIS, we use the recent value of the background model and Equation (5).

McFIS-Instant: McFIS-instant is equivalent to McFIS-D if lossless decoded frames (in terms of channel error and quantization error) are available at the encoder and decoder. It has been demonstrated through experimental results that McFIS-Instant performs generally better than the McFIS because of its closeness to the current frame for which similarity and amount of background is determined. However, in actual implementation of lossy encoder/decoder, we cannot use McFIS-Instant, as the decoder does not have error-free decoded frames.

V. COMPUTATIONAL COMPLEXITY

The proposed techniques (both McFIS-I and McFIS-D) require extra operations to generate and encode (or update) the McFIS compared to the original state-of-the-art scheme HBP, the scheme HBP with three reference frames (HBP-3Ref) or the scheme HBP with LTR frame (HBP-LTR). In McFIS-I, as we generate and encode the McFIS once for a scene using the first few frames, the impact of extra operations is negligible if the scene is sufficiently long; the proposed McFIS-D scheme requires more computation as it needs to update the McFIS after encoding each frame. Note that the McFIS-D scheme requires computational time in the decoder as we do not send any McFIS.

The proposed techniques aim to reference the current frame for its background or more stable regions, and thus, small search range ME should be sufficient to find the best match for the background region. The small non-zero search length addresses camera jerking and/or the need for fractional ME. Thus, if we use small search length ME for the third reference frame, the proposed techniques need comparable or less computation compared to the HBP-3Ref scheme. For comparison, we also use another scheme (HBP-FirstFrame) which is the same as the proposed McFIS-I scheme but replacing McFIS with the first frame of a scene. The computational time requirements by different algorithms are

shown in Fig. 8 against the HBP scheme and the HBP-3Ref scheme. Note that we consider full search length for all types of reference frames in the HBP and HBP-3Ref schemes, however, we consider full search length for B-type reference frames and short search length for the third reference frame (i.e., first, LTR, or McFIS) in the HBP-FirstFrame, HBP-LTR, and both proposed schemes (i.e., McFIS-I and McFIS-D) because the third reference is only used for referencing the static/occluded background areas. In the experiment we use 15 and 2 for the full search length and short search length respectively.

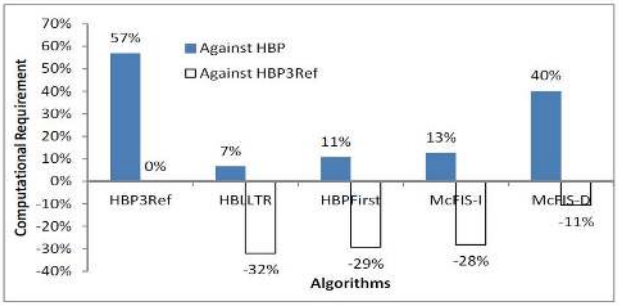


Fig. 8: Computational complexity of the proposed techniques (McFIS-I and McFIS-D), the HBP-LTR, the HBP-FirstFrame, the HBP-3Ref schemes against HBP and HBP-3Ref schemes.

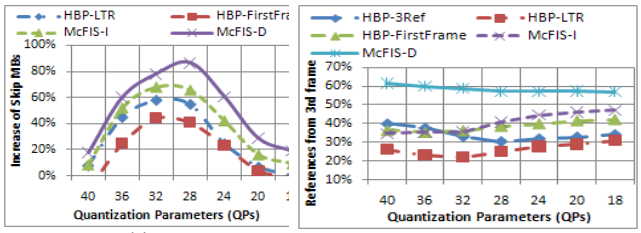
The figure reveals that the extra computational time requirements for HBP-3Ref, HBP-LTR, HBP-FirstFrame, and two proposed (i.e., McFIS-I and McFIS-D) techniques are 57%, 7%, 11%, 13%, and 40% compared to the HBP scheme respectively. On the other hand, the computational time saving for HBP-LTR, HBP-FirstFrame, and two proposed (i.e., McFIS-I and McFIS-D) techniques are 32%, 29%, 28%, 11% compared to the HBP3Ref scheme. The algorithms were executed on a personal computer (Intel® Core™ 2 CPU 6600 @ 2.4GHz, 3.5 GB RAM). Note that there is no significant loss (i.e., bit rate increases less than 1% and PSNR decreases less than 0.02 dB) of RD performance by the proposed techniques while small search lengths are used, while the other scheme such as HBP-3Ref increases bit rates and decreases quality more as it does not only refer to background regions. Thus, we could not use short search lengths for the scheme.

The proposed techniques need extra time in the encoder and decoder compared to the other relevant techniques due to the background modeling. The background modeling time is fixed and does not depend on the search length. The experimental result shows that extra 0.64% and 2% of encoding times are needed using McFIS-I and McFIS-D methods respectively compared to the encoding time of 100 frames with 15 search length. For the McFIS-I scheme we do not need extra computational time for background modeling in the decoder as we send encoded McFIS for the decoder. However, we need extra time for background modeling in the decoder for the proposed McFIS-D scheme. In the scheme, the decoder requires the same time as the encoder for background modeling due to the same procedure is applied in the encoder and the decoder.

VI. OVERALL EXPERIMENTAL RESULTS

A. Experimental Setup:

To compare the performance of the proposed schemes (i.e., McFIS-I and McFIS-D), we implement H.264 with HBP prediction structure, as well as the aforementioned HBP-3Ref, HBP-LTR, and HBP-FirstFrame. All algorithms are implemented based on H.264 (adapted from the JM 18 H.264/AVC reference software) recommendations with 25 Hz, ±15 as the search length with quarter-pel accuracy. Experimental results show that relative performance differences among the schemes do not vary significantly if we increase the search length to 32 or 64. We use QCIF (176 by 144 pixels) and CIF (352 by 288 pixels) size video sequences. H.264-HBP and HBP-3Ref have 16 GOP size i.e., they have I-frames at regular intervals (i.e., at the first frame of each GOP) whereas the proposed methods and the HBP-FirstFrame method have I-frames only when scene change occurs. The proposed techniques have three reference frames and the HBP structure has two reference frames. Thus, for fair comparison we select the HBP-3Ref and HBP-LTR schemes for comparison. The same QP for high quality LTR frame and McFIS has been used for fair comparison. We use Lagrangian multiplier as $0.85 \times 2^{(QP-12)/3}$ where QP varies for different types of frames. We use a number of QPs from 20 to 40 for P-frames and QPs for other types of frames are adjusted based on the recommended values in HBP scheme (see discussion in Section II.A). In the HBP-LTR scheme, we use the *jumping* parameter as 32 and select the high quality LTR frame as the first I-frame of an even number GOP in a video. We also use other jumping parameters (e.g., 16) and other frames (e.g., the middle frame of a GOP) as LTR frame, but the above mentioned combination is the best. Note that all frames of the first two GOPs in a scene are used to generate the McFIS in the proposed McFIS-I scheme.

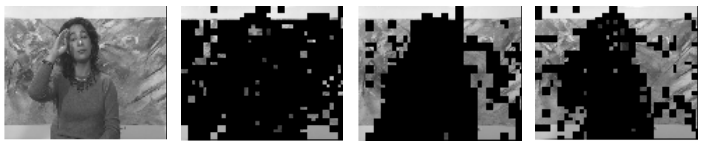


(a) (b)

Fig. 9: (a) Increase (%) of skip MBs against the HBP scheme with three reference frames (HBP-3Ref) by HBP-LTR, HBP-firstFrame, and the proposed (McFIS-I and McFIS-D) algorithms for mixed video sequence; and (b) the percentages of references from the 3rd reference frame by HBP-3Ref, HBP-LTR, HBP-FirstFrame, and proposed (McFIS-I and McFIS-D) schemes.

The schemes use two closest neighboring frames (subject to availability in terms of coding order) from two different directions (i.e., List_0 and List_1) for bi-directional motion estimation and compensation. The schemes also use the third frame (the third closest, the first frame, the LTR frame, and the McFIS by the HBP-3Refs, HBP-FirstFrame, HBP-LTR, and

the proposed schemes respectively) for uni-predicitve ME & MC. The ultimate decision is taken for mode selection based on the Lagrangian optimization [42] between the results of the two motion compensated techniques. A total of 288 frames are used for all video sequence (except for the Popple, Football, and Mixed video sequences where 112, 112, and 1200 frames are used).



(a) (b) (c) (d)

Fig. 10: Reference regions (non black blocks) using HBPLTR, McFIS-I, and McFIS-D methods; (a) original image, (b), (c), and (d) reference blocks by HBPLTR, McFIS-I, and by McFIS-D methods respectively.

B. Performance Comparisons

As the McFIS represents the *stable* part of a scene (i.e., static/uncovered background), the proposed schemes have more skip MBs compared to other algorithms. Fig. 9 (a) shows 20~85% and 8~70% increase (in percentages) of skip MBs using the proposed McFIS-D and McFIS-I schemes compared to the HBP-3Ref scheme. Fig. 9 (b) shows percentages of references coming from the third frame by the relevant algorithms. The proposed McFIS-I and McFIS-D schemes select more areas (i.e., more blocks) from the third reference frame (i.e., McFIS) compared to other schemes while encoding a frame. This indicates that the McFIS can capture more background compared to other third reference frame.

TABLE 2
PERFORMANCE COMPARISON USING BD-PSNR [36][37] AGAINST HBP SCHEME

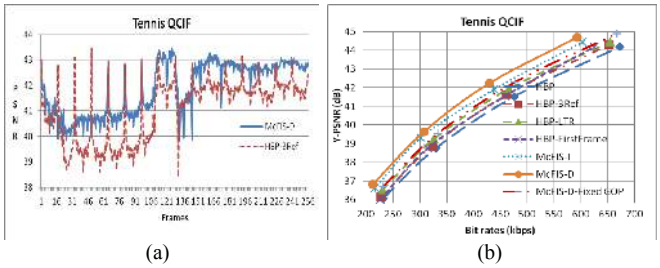
Video Sequence	3Ref	LTR	First Frame	McFIS-I	McFIS-D
News	0.02	0.45	1.99	3.29	3.83
Hall Objects	0.04	0.13	1.60	2.95	3.61
Salesman	0.00	0.24	2.30	4.03	4.30
Tennis	0.28	0.54	0.16	1.05	1.40
Trevor	0.17	0.24	0.12	0.39	0.61
Mixed	0.35	0.42	0.69	1.30	1.67
Silent	0.12	0.36	1.93	3.07	3.34
Paris	0.16	0.27	1.04	1.40	2.00
Bridge Close	0.12	0.18	0.73	1.80	1.77
Popple	0.30	0.21	0.43	0.68	0.82
Average	**0.16**	**0.30**	**1.10**	**2.00**	**2.34**

Fig. 10 shows evidence where various schemes encode a block using different reference frames. In the figure we show an original frame (Frame 23 of *Silent*) and its referencing scenario by either B reference frames or LTR frame (i.e., high quality LTR frame for HBPLTR scheme and McFIS for the proposed McFIS-I and McFIS-D schemes). While coding the corresponding blocks, the HBP-LTR and the proposed methods use the LTR frame or McFIS as a reference frame for the normal areas (i.e., non black blocks) of Fig. 10(b), (c), and (d) and B- frames (according to the hierarchy structure) as reference frames for the *black* blocks of Fig. 10(b), (c), and (d). The figure confirms that areas of McFIS reference (see Fig. 10 (c), and (d)) are larger and more aligned to the background areas compared to the areas of the LTR frame (see Fig. 10 (b)).

TABLE 3
PERFORMANCE COMPARISON USING BD-BIT RATE (%) [36][37] AGAINST HBP SCHEME

Video Sequence	3Ref	LTR	First Frame	McFIS-I	McFIS-D
News	-0.23	-4.17	-15.45	-24.76	-32.19
Hall Objects	-1.79	-2.68	-20.93	-41.32	-48.70
Salesman	-0.22	-2.44	-19.99	-31.06	-31.38
Tennis	-3.95	-6.73	-4.76	-12.86	-16.76
Trevor	-2.34	-2.92	-2.47	-5.12	-7.89
Mixed	-5.01	-5.38	-9.34	-16.71	-21.67
Silent	-1.84	-4.98	-21.81	-32.99	-33.84
Paris	-1.94	-2.51	-12.51	-16.78	-23.22
Bridge Close	-2.64	-3.28	-11.96	-29.66	-30.13
Popple	-4.43	-3.21	-6.76	-10.16	-11.88
Average	**-2.44**	**-3.83**	**-12.60**	**-22.14**	**-25.77**

BD-PSNR and BD-Bitrate are two measures to see the average difference between two RD curves [36][37]. In our case we consider four points (for PSNRs and Bit rates) which are covered by all algorithms using different QPs. The PSNR range varies in different video sequences; however, normally we cover PSNR range from 33.0 dB to 44.0 dB. Table 2 and Table 3 show a summary of the RD performance in terms of BD-PSNR and BD-Bit rate [36][37] for a wide range of bit rates against HBP scheme. Table 2 reveals that the proposed techniques (i.e., McFIS-I and McFIS-D) outperform the HBP-3Ref scheme on average by 1.84 dB and 2.18 dB respectively when 10 standard video sequences are used. Table 3 demonstrates that the proposed techniques reduce the bit rates on average by 20% and 23% respectively on average compared to the HBP-3Ref scheme when the same video sequences are used.



(a) (b)

Fig. 11: (a) Frame-by-frame PSNR comparisons for the Tennis video sequence using the proposed McFIS-D and HBP-3Ref techniques under the same bit rate; (b) RD performance of the proposed schemes (McFIS-I, McFIS-D, and McFIS-D with fixed GOP) against existing algorithms.

Fig. 11(a) shows frame-by-frame PSNR comparisons for the *Tennis* QCIF video sequence using the proposed McFIS-D scheme against the HBP-3Ref scheme under the same bit rate (i.e., 440 kbps) with the same QP_{Intra} but different QP_{Inter}. Note that the figure is drawn based on the display order of frames rather than encoding order. The proposed McFIS-D scheme detects 6 scene changes with 256 frames. The proposed

schemes such as McFIS-I and McFIS-D use flexible GOP based on the scene change. However, if we allow fixed GOP (i.e., by inserting I-frame at regular interval) in the proposed scheme, the proposed scheme still outperforms the relevant existing schemes (see Fig. 11(b)).

Fig. 12 shows the overall RD performance for a wide range of bit rates using a number of video sequences, inclusive of a mixed video sequence comprising 1200 frames from ten different video sequences to test the effectiveness of the SCD and AGOP on RD performance by the proposed techniques. We select four video sequences (*tennis and Tempete*) with complex object motion and camera motion with scene changes to investigate the effectiveness of the proposed techniques in camera motion and scene change situations.

The figure confirms that the proposed techniques outperform the relevant state-of-the-art algorithms, namely HBP, HBP-3Ref, and HBP-LTR. The RD performance in Fig. 12 reveals that the proposed technique also outperforms the relevant techniques for a wide range of bit rates for Tennis, Mixed video, and Silent video sequences. Actually, the overall PSNR improvement came from good reconstruction of background by keeping the good reconstruction foreground. Moreover, a smart video encoder/decoder can be designed to encode foreground with better quality by allocating more bits which are saved from background coding. As expected, the performance gain of the proposed techniques depended on the amount of background in a video. For example, *Silent* has more background area compared to *Paris* and hence the performance improvements of the proposed techniques are more significant.

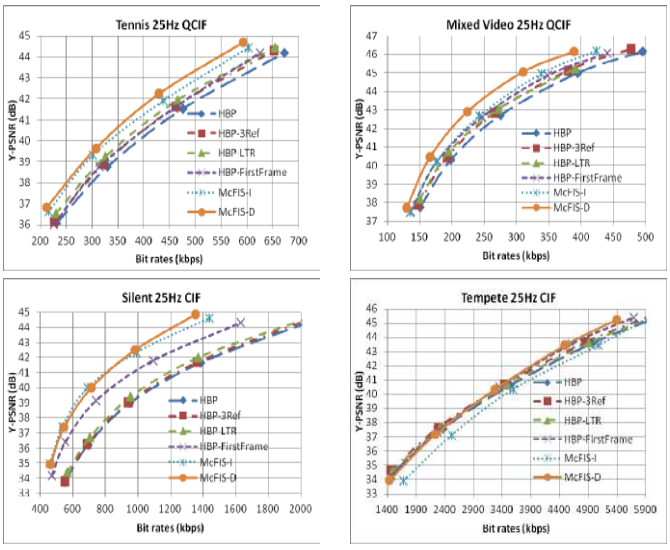


Fig. 12: RD performance by the proposed (McFIS-I and McFIS-D), HBP, HBP-3Ref, HBP-LTR, and HBP-FirstFrame algorithms using a wide range of standard video sequences.

Overall, as can be seen from Table 2, Table 3, and Fig. 12, the proposed methodology outperforms the existing schemes consistently in a wide range of situations (visual content, bitrates and resolutions). If we compare the RD performance between the McFIS-I and HBP-FirstFrame techniques in Fig. 12, we can see the gain due to introduction of the McFIS. For example, it is around 1.0 dB for *Silent* at 1200 kbps. If we compare the RD performance among the McFIS-I, HBP-FirstFrame and HBP-3Ref techniques in Fig. 12, we can see the gain due to introduction of AGOP. For example, it is also around 1.0 dB for *Silent* at 1200 kbps. For the McFIS-I scheme we cannot use McFIS as a reference frame until it is generated. We used all frames from the first two GOPs to generate the McFIS and then we can use it for subsequent frames. Thus, we need two GOPs frame delay in the McFIS-I scheme. To avoid frame delay, we can continue coding without McFIS until it is generated. The proposed McFIS-D has no frame delay problem as the McFIS can be used from the very beginning, although the performance gain due to McFIS in the McFIS-D scheme for the initial few frames may not be significant.

We test our algorithms with camera motion video sequence such as *Tennis*, *Trevor*, *Garden*, and *Tempete*. The experimental results show that the proposed technique (McFIS-D and McFIS-I) outperforms the other relevant algorithms where camera motion is relatively low (e.g., *Tennis* and *Trevor*). However, the proposed McFIS-I technique in its current state could not provide better RD performance compared to the other methods for the video sequences with high camera motion e.g., *Tempete, Garden,* and *Football* as the McFIS-I technique encodes a number of I-frames (one at each scene change) due to the low percentage of McFIS references (see the last paragraph of Section III.D). This occurred due to significant zooming in the *Tempete* video and huge motion activities in *Football* sequence. It is interesting to observe that the proposed McFIS-D technique outperforms the other relevant algorithms at high bit rates for the *Tempete, Football, and Garden* sequence with small margin. This occurred due to two reasons (i) unlike the McFIS-I scheme the McFIS-D scheme does not encode any I-frames at the scene change points (thus saving bits), (ii) the McFIS-D technique updates its McFIS dynamically to make it relevant for referencing future frames. We conduct the experiment using all algorithms for *Foreman, Calendar, and Bus* video sequences to see the effect of camera panning. The result shows that the McFIS-I method determines a number of scene changes and does not provide coding gain compared to others. Thus, the McFIS-I scheme is suitable for long scene with static background and the McFIS-D is suitable for both short and long scene with dynamic background. In general, the McFIS-D scheme requires more computational time compared to the McFIS-I scheme. The bottom line is that the proposed McFIS-D scheme does not deteriorate the RD performance compared to other schemes even in camera motion scenarios.

To avoid frame delay at either a scene change point or at the starting point of a video, the first few frames of a scene are encoded without using the latest McFIS in the proposed McFIS-I scheme as it needs a number of frames for the McFIS modeling. On the other hand, for the proposed McFIS-D scheme, we can use McFIS for encoding frames at the beginning as the background modeling and generation and using of the McFIS are simultaneous. The proposed McFIS-D

scheme does not introduce any frame delay. Thus, in both proposed methods, we can avoid frame delay. We need memory in both encoder and decoder to store a McFIS (equivalent to a frame) for referencing purpose. Except for the HBP scheme, all other schemes need extra memory to store the third reference frame, LTR frame, or McFIS for the encoder and the decoder. In addition, the McFIS-I scheme needs memory for storing the background model in the encoder but not in the decoder. However, the McFIS-D scheme needs the same amount of memory for storing the background model in both encoder and decoder. We need to store background model for each pixel. Each pixel may have maximum three models; however, for the most cases there is one (background) model, for some cases two (background and foreground), and for rare cases three (two background and one foreground or one background and two foreground). Each model has four fields (i) mean (0~255), (ii) variance (0~900), (iii) weight (0~1), and (iv) recent value (0~255).

VII. CONCLUSIONS

In this paper, we propose a new methodology for hierarchical bi-predictive picture-based video coding techniques using the dynamic background frame (i.e., McFIS) as the third reference frame to overcome multi-frame referencing difficulties in conventional techniques. The proposed techniques outperform the existing LTR frame techniques by better exploiting uncovered background through implicit background/foreground referencing, and can be regarded as a method for finding a so called golden frame. There are two approaches to generate the McFIS under the proposed methodology. In the first approach, we generate a McFIS using a number of frames of a scene in a video and then encode it as an I-frame with higher quality. For the rest of the scene that I-frame is used as a reference frame (i.e., an LTR frame). In the second approach, we generated McFIS from the decoded frames and then used it as an LTR frame, without the need to encode a McFIS (which therefore becomes a "hidden layer" in both an encoder and a decoder).

The overall experimental results show that the proposed technique improves PSNR from 0.4 dB to 4.2dB in comparison with the relevant existing methods over a wide range of bit rates and for a large number of standard test videos (with different visual content/motion/resolution) and their combinations. At the same time, the experimental results indicate that the proposed techniques use 28% and 11% less computational time compared to the other relevant existing HBP techniques.

The second proposed approach (i.e., McFIS constructed from decoded frames) is better in terms of RD performance because of its adaptability towards the end of a scene with background regions, but with higher computational complexity (nevertheless, still 11% more efficient than the relevant existing HBP schemes).

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560-576, 2003,

[2] ITU-T Recommendation H.264: Advanced video coding for generic audiovisual services, 03/2009.

[3] M. Paul and M. Murshed, "Video Coding focusing on block partitioning and occlusions," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 691-701, 2010.

[4] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103 - 1120, 2007.

[5] Y. –W. Huang, B. –Y. Hsieh, S. –Y. Chien, S. –Y. Ma, and L. –G. Chen, "Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 16, no. 4, pp. 507-522, 2006.

[6] S. Saponara, C. Blanch, K. Denolf, and J. Bormans, "The JVT Advanced Video Coding Standard: Complexity and Performance Analysis on a Tool-by-Tool Basis", *IMEC*, 2003.

[7] T. Wiegand, X. Zhang, and B. Girod, "Motion-compensating Long-term memory prediction,", *IEEE Int. Conference on Image Processing* (ICIP), vol. 2, pp. 53-56, 1997.

[8] L. Shen, Z. Liu, Z. Zhang, and G. Wang, "An Adaptive and Fast Multiframe Selection Algorithm for H.264 Video Coding," *IEEE Signal Processing Letters*, vol. 14, No. 11, pp. 836-839, 2007.

[9] T. –Y. Kuo, H. –J. Lu, "Efficient Reference Frame Selector for H.264," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 3, pp. 400-405, 2008,

[10] K. Hachicha, D. Faura, O. Romain, and P. Garda, "Accelerating the multiple reference frames compensation in the H.264 video coder," *Journal of Real-Time Image Processing, Springer*, Vol. 4, No. 1, pp. 55-65, 2009.

[11] D. Liu, D. Zhao, X. Ji, and W. Gao, "Dual Frame Motion Compensation With Optimal Long-Term Reference Frame Selection and Bit Allocation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 3, pp. 325 - 339, 2010.

[12] V. Chellappa, P. C. Cosman, and G. M. Voelker, "Dual Frame Motion Compensation With Uneven Quality Assignment," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 2, pp. 249 - 256, 2007.

[13] M. Tiwari and P. C. Cosman, "Selection of Long-Term Reference Frames in Dual-Frame Video Coding Using Simulated Annealing," *IEEE Signal Processing Letter*, vol. 15, pp. 249-252, 2008.

[14] A. Mavlankar and B. Girod, "Background extraction and long-term memory motion-compensated prediction for spatial-random-access-enable video coding," *International Picture coding Symposium*, 2009.

[15] "Golden frame", http://www.on2.com/index.php?602, retrieved at 24th May 2010.

[16] J. Bankoski, "The VP8 video codec: High compression + low complexity," http://www.dspdesignline.com/howto/214303691, retrieved at 24th May 2010.

[17] P. Wilkins, "The On2 VP6 codec: how it works," http://www.dspdesignline.com/211100053?printableArticle=true, retrieved at 24th May 2010.

[18] P. Wilkins, J. Bankoski, and Y. Xu, "System and method for video encoding using constructed reference frame," patent number: 20100061461, 2010.

[19] R. Ding, Q. Dai, W. Xu, D. Zhu, and H. Yin, "Background-frame based motion compensation for video compression," *IEEE Int. Con. on Multimedia and Expo* (ICME), vol. 2, pp. 1487-1490, 2004.

[20] K. Zhang and J. Kittler, "Using background memory for efficient video coding," *IEEE Int. Conference on Image Processing*, pp. 944-947, 1998.

[21] K. Zhang and J. Kittler, "A background memory update scheme for H.263 video codec," In *EUSIPCO-98*, Island of Rhodes, Greece, 1998.

[22] S.-Y. Chien, S.-Y. Ma, and L.-G. Chen, "Efficient moving object segmentation algorithm using background registration technique", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 7, pp. 577 - 586, 2002.

[23] **N.** Mukawa and H. Kuroda, "Uncovered background prediction in interframe coding," *IEEE Transactions on Communications*, vol. 33, no. 11, pp. 1227-1231, 1985.
[24] D. Hepper, "Efficiency analysis and application of uncovered background prediction in a low bit rate image coder," *IEEE Transaction on Communication*, vol. 38, no. 9, pp. 1578-1584, 1990.
[25] A. Krutz, A. Glantz, and T. Sikora, "Background Modelling for Video Coding: From Sprites to Global Motion Temporal Filtering," *IEEE International Symposium on Circuits and Systems* (ISCAS-10), pp. 2179-2182, 2010.
[26] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B-pictures and MCTF," *IEEE International Conference on Multimedia and Expo* (IEEE ICME-06), pp. 1929-1932, 2006.
[27] M. Flierl and B. Girod, "Generalized B Pictures and the Draft H.264/AVC Video Compression Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7), 587 – 597, 2003.
[28] M. Paul, W. Lin, C. T. Lau, and B. –S. Lee, "Video coding using the most common frame in scene," *IEEE Int. conference on Acoustics, Speech, and Signal processing* (ICASSP-10), pp. 734-737, 2010.
[29] M. Paul, W. Lin, C. T. Lau, and B. –S. Lee, "Explore and model better I-frame for video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 9, pp. 1242-1254, 2011.
[30] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 246–252, 1999.
[31] D.-S. Lee, "Effective Gaussian mixture learning for video background subtraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 27, no. 5, pp. 827–832, May 2005.
[32] M. Haque, M. Murshed, and M. Paul, "Improved Gaussian mixtures for robust object detection by adaptive multi-background generation," *IEEE Int. Conference on Pattern Recognition,* pp. 1-4, 2008.
[33] J. –R. Ding and J. –F. Yang, "Adaptive group-of-pictures and scene change detection methods based on existing H.264 advanced video coding information," *IET Image Proc.*, vol 2, no. 2, pp. 85-94, 2008.
[34] A. Dimou, O. Nemethova and M. Rupp, "Scene change detection for H.264 using dynamic threshold techniques," *Proceedings of 5th EURASIP Conference on Speech and Image Processing, Multimedia Communications and Service*, 2005.
[35] G. Rascioni, S. Spinsante, and E. Gambi, "An Optimized Dynamic Scene Change Detection Algorithm for H.264/AVC Encoded Video Sequences," *International Journal of Digital Multimedia Broadcasting*, Article ID 864123, 2010.
[36] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," VCEG-M33, ITU-T Q.6/SG16 VCEG, 2001.
[37] S. Kondo and H. Sasai, "Motion-compensated video coding using sliced blocks," *Systems and Computers in Japan*, 38 (7), 12-22, 2007.
[38] X. Zhang, L. Liang, Q. Huang, Y. Liu, T. Huang, and W. Gao, "An efficient coding scheme for surveillance videos captured by stationary cameras," *Proc. of the SPIE*, Volume 7744, article id. 77442A, 2010.
[39] X. Zhang, Y. Tian, T. Huang, and W. Gao, "Low-complexity and high-efficiency background modeling for surveillance video coding," *IEEE International Conference on Visual Communications and Image Processing* (VCIP), pp. 1-6, 2012.
[40] M. Paul, W. Lin, C. T. Lau, and B.-S. Lee, "McFIS: Better I-frame for video coding," IEEE International Symposium on Circuits and Systems (ISCAS), pp. 2171-2174, 2010.
[41] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," *Proceedings of the IEEE*, 99(4), 626 - 642, 2011.
[42] M. Paul and W. Lin, C. T. Lau, and B. –S. Lee, "Direct Inter-Mode Selection for H.264 Video Coding using Phase Correlation," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 461–473, 2011.

Manoranjan Paul (M'03 SM'13) received PhD from Monash University in 2005. He has worked as a Research Fellow in the University of New South Wales in 2005~2006, Monash University in 2006~2009, and Nanyang Technological University in 2009~2011. He is currently working as a Faculty Member in the School of Computing and Mathematics, Charles Sturt University (2011~). His major research interests are in the fields of image/video coding/compression, EEG signal analysis, and computer vision. He has published more than 65 refereed papers in international journals and conferences. Dr. Paul regularly published journal articles in the IEEE Trans. on Image Processing, IEEE Trans. on Circuits and Systems for Video Technology, and IEEE Trans. on Multimedia, which are considered as the top ranked journals in the image processing, video technology, and multimedia fields respectively. He is a keynote speaker on "Vision friendly video coding" in IEEE ICCIT 2010 and tutorial speaker on "Multiview video coding using cuboid data compression" in DICTA 2013.

Dr Paul is a senior member of the IEEE and ACS. Dr. Paul has served as a guest editor of Journal of Multimedia from 2008 to 2013. Currently Dr. Paul is an Associate Editor of EURASIP Journal on Advances in Signal Processing.

Weisi Lin received his Ph.D. from King's College London. He was the Lab Head and Acting Department Manager for Media Processing, in Institute for Infocomm Research, Singapore. Currently, he is the Associate Chair (Graduate Studies) in Computer Engineering, Nanyang Technological University, Singapore. His research areas include image processing, perceptual multimedia modeling and evaluation, and video compression. He published 250+ refereed papers in international journals and conferences.

He is on the editorial boards of IEEE Trans. on Multimedia, IEEE SIGNAL PROCESSING LETTERS and *Journal of Visual Communication and Image Representation*. He chairs the IEEE MMTC IG on Quality-of-Experience. He has been elected as an APSIPA Distinguished Lecturer (2012/3). He is the Lead Technical-Program Chair for Pacific-Rim Conference on Multimedia (PCM) 2012 and International Workshop on Quality of Multimedia Experience (QoMEX) 2014, and a Technical-Program Chair for *IEEE International Conference on Multimedia and Expo (*ICME) 2013. He is a fellow of Institution of Engineering Technology, and an Honorary Fellow, Singapore Institute of Engineering Technologists.

Chiew-Tong Lau received the B.Eng. degree from Lakehead University, Canada, in 1983, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of British Columbia, Canada, in 1985 and 1990 respectively. He is currently an Associate Professor in the School of Computer Engineering, Nanyang Technological University, Singapore. His research interests include wireless communications systems and signal processing.

Bu Sung Lee received his B.Sc. (Hons) and PhD from the Electrical and Electronics Department, Loughborough University of Technology, UK in 1982 and 1987 respectively. He is currently an Associate Professor with the Nanyang Technological University, Singapore. Bu Sung Lee, holds a Joint appointment as Director, Service Platform Lab, HP Labs. Singapore from July 2010 till end-June 2012.

Bu Sung Lee, has been very active in the area of establishing a Research and Education Network locally as well as globally. He is the founding President and current President(2011-) of Singapore Advanced Research and Education Network(SingAREN) and Chair of the TransEurasia Information Network Cooperation Center(TEIN*CC) governors.

Bu Sung Lee has published over 200 peer reviewed papers. His research covers Cloud Computing, Network and Video coding. His particular interest are in data replication, scheduling, network Qos (wired and wireless), and ad hoc network.