Pattern-based video coding with dynamic background modeling

Paul, Manoranjan; Lin, Weisi; Lau, Chiew Tong; Lee, Bu-Sung

2013

Paul, M., Lin, W., Lau, C. T., & Lee, B.-S. (2013). Pattern-based video coding with dynamic background modeling. EURASIP journal on advances in signal processing, 2013(138), 1-15.

https://hdl.handle.net/10356/101383

https://doi.org/10.1186/1687-6180-2013-138

EURASIP Journal on
Advances in Signal Processing
a SpringerOpen Journal

RESEARCH   **Open Access**

Pattern-based video coding with dynamic background modeling

Manoranjan Paul[1*], Weisi Lin[2], Chiew Tong Lau[2] and Bu-Sung Lee[2]

Abstract

The existing video coding standard H.264 could not provide expected *rate-distortion* (RD) performance for *macroblocks* (MBs) with both moving objects and static background and the MBs with uncovered background (previously occluded). The *pattern-based video coding* (PVC) technique partially addresses the first problem by separating and encoding moving area and skipping background area at block level using binary pattern templates. However, the existing PVC schemes could not outperform the H.264 with significant margin at high bit rates due to the least number of MBs classified using the pattern mode. Moreover, both H.264 and the PVC scheme could not provide the expected RD performance for the uncovered background areas due to the unavailability of the reference areas in the existing approaches. In this paper, we propose a new PVC technique which will use the *most common frame in a scene* (McFIS) as a reference frame to overcome the problems. Apart from the use of McFIS as a reference frame, we also introduce a content-dependent pattern generation strategy for better RD performance. The experimental results confirm the superiority of the proposed schemes in comparison with the existing PVC and the McFIS-based methods by achieving significant image quality gain at a wide range of bit rates.

Keywords: Video coding; Virtual reference frame; Long-term reference frame; Uncovered background; Multiple reference frames

1 Introduction

H.264, the latest video coding standard [1,2], outperforms its competitors such as H.263, MPEG-2, MPEG-4, etc. due to a number of innovative features in the intra- and inter-frame coding techniques. *Variable block size (VBS) motion estimation and motion compensation (ME&MC)* are the most prolific features. In the VBS scheme, a 16×16 pixel *macroblock* (MB) is partitioned into several small rectangular- or square-shaped blocks. ME&MC are carried out for all possible combinations, and the ultimate block size is selected based on the *Lagrangian optimization* [3-5] using the bits and distortions of the corresponding blocks. Real-world objects, by nature, may be in any arbitrary shapes, and ME&MC using only rectangular- and square-shaped blocks just approximate the real shape; thus, the coding gain would not be satisfactory. A number of research works are conducted by non-rectangular block partitioning [6-11]
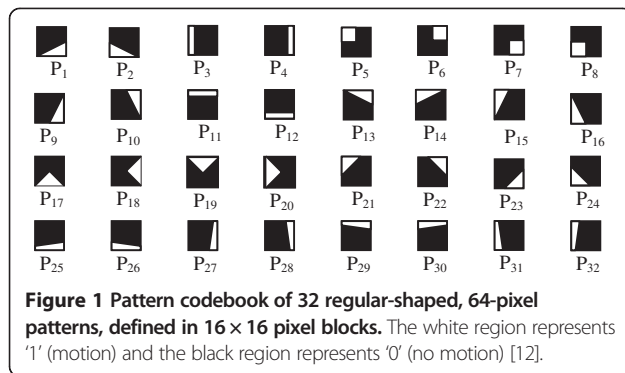
using geometric shape partitioning, motion-based implicit block partitioning, and L-shaped partitioning. The requirement of excessively high computational complexity in the segmentation process and the marginal improvement over the H.264 make them less effective for real-time applications [12]. Moreover, the requirement of valuable bits for encoding the area covering almost static background makes the abovementioned algorithms inefficient in terms of rate-distortion performance.

To exploit the non-rectangular block partitioning and partial block skipping by separating *moving regions* (MRs) from static background regions in an MB, the *pattern-based video coding* (PVC) [12,13] schemes partition the MBs via a simplified segmentation process that again avoids handling the exact shape of the moving objects so that the popular MB-based ME could be applied. The PVC algorithm focuses on the MRs of the MBs, through the use of a set of regular-shaped pre-defined 64-pixel pattern templates (see Figure 1). The MR is defined as the difference between the current MB and the collocated MB of the reference frame. The pattern templates were designed using '1' in 64-pixel positions and

* Correspondence: mpaul@csu.edu.au
[1]Centre for Research in Complex Systems (CRiCS), School of Computing and Mathematics, Charles Sturt University, Panorama Avenue, Bathurst NSW-2795, Australia
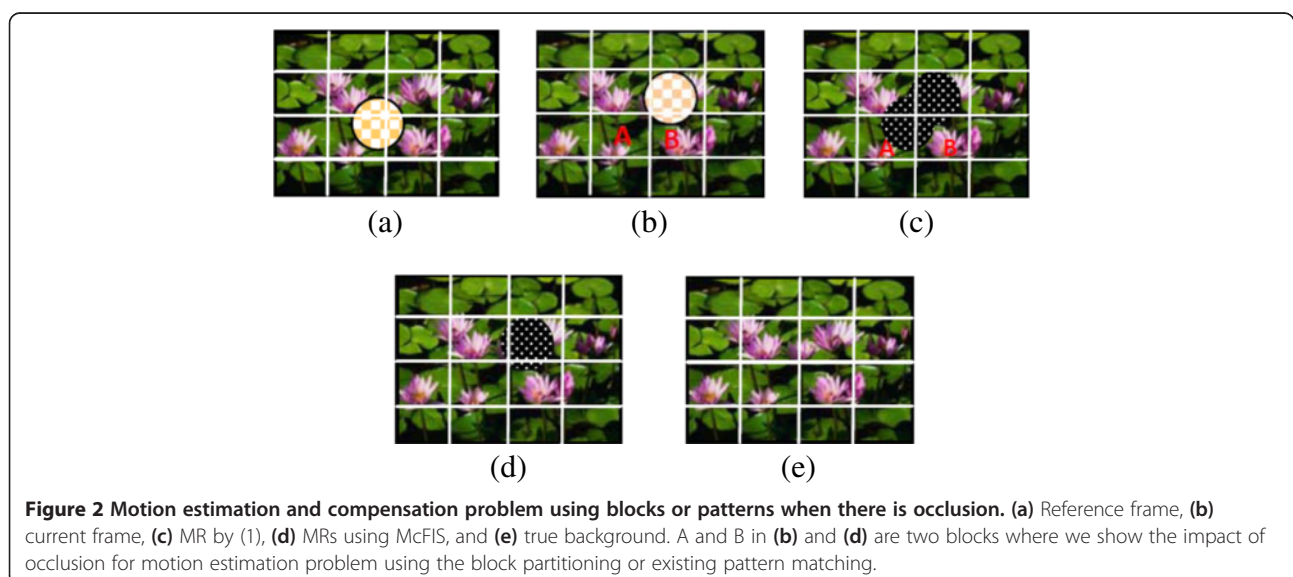Full list of author information is available at the end of the article

Figure 1 Pattern codebook of 32 regular-shaped, 64-pixel patterns, defined in 16 × 16 pixel blocks. The white region represents '1' (motion) and the black region represents '0' (no motion) [12].

'0' in the remaining 192-pixel positions in a 16 × 16 pixel MB. The MR of an MB is defined as a region comprising a collection of pixel positions where pixel intensity differs from its reference MB by a margin (e.g., [13] used 2). Using some similarity measures, if the MR of an MB is found to be well covered by a particular pattern (based on the white areas of a pattern template), then the MB can be classified as a *region-active MB* (RMB) and coded by considering only the 64 pixels of the pattern, with the remaining 192 pixels being skipped as static background. Embedding PVC in the H.264 standard as an extra mode provides higher compression for RMBs as a larger segment with static background is coded with the partially skipped mode [12]. Note that the pattern mode differs from any other existing block partitioning modes in the H.264 in terms of the encoding areas of an MB (i.e., the former encodes only one fourth of the areas of an MB by skipping the rest of the areas as background while the latter encodes whole areas of an MB).

The MR generated from the difference of the current MB and the collocated MB from the *traditional* reference frames (i.e., the immediate previous frame or any frame

which is previously encoded) may contain a moving object or uncovered background (detailed in Figure 2). The ME&MC using pattern-covered MR (i.e., covered by the white regions only of any pattern template (see Figure 1)) for uncovered background would not be accurate if there is no similar region in the reference frames. As a result, no coding gain can be achieved for the uncovered background using the PVC techniques. Similar issues occur for any other H.264 VBS modes due to the lack of a suitable matching region in the reference frames.

To address the abovementioned problem, we need a reference frame where we will find the uncovered background for the current MB once that region is evidenced. Only a *true* background of a scene can be the best choice to be the reference frame for uncovered background. Moreover, an MR generated from the true background against the current frame (instead of from the previous frame against the current frame) represents only the moving object instead of both the moving object and the uncovered background. Thus, the selection of the best matched pattern against the newly generated MR is the best approximation of the object/partial object in an MB because the MR does not have any uncovered background. The ME&MC using the best matched pattern carried out on the immediate previous frame will provide more accurate motion vector and thus minimize the residual errors for the object/partial object within the MB, while the rest of the area (which is not covered by the white regions of the pattern; see Figure 1) is copied from the true background frame. The immediate previous frame is used for ME&MC assuming that the object is visible in the immediate previous frame. The other modes of H.264 can also use true background as well as the immediate previous frame (in *multiple reference frames* (MRFs) technique [2]) as two separate

Figure 2 Motion estimation and compensation problem using blocks or patterns when there is occlusion. (a) Reference frame, **(b)** current frame, **(c)** MR by (1), **(d)** MRs using McFIS, and **(e)** true background. A and B in **(b)** and **(d)** are two blocks where we show the impact of occlusion for motion estimation problem using the block partitioning or existing pattern matching.

reference frames. The Lagrangian optimization will pick the optimal reference frame.

Recently, dynamic background modeling using a Gaussian mixture model [14-16] has been introduced for robust and real-time object detection from the so-called dynamic environment where true background is impossible due to illumination variation over time, camera displacement, shadow/reflection of foreground objects, and intrinsic background motions (e.g., waving tree leaves). The object can be detected more accurately by subtracting the background frame (generated from the background model) from the current frame. Some techniques such as sprite coding [17], golden frame generation [18], etc. are used to extract background, but they need computationally expensive and very sophisticated preprocessing steps including object segmentation process. Due to the dependency on block-based motion vectors and the lack of adaptability in multimodal backgrounds for dynamic environment, the background frame generation techniques in [19,20] could not perform well. Recently, a dynamic background frame termed as the *most common frame of a scene* (McFIS) [21] has been developed for video coding using dynamic background modeling. In this paper, the McFIS is to be used as another reference frame assuming that the background and foreground of the current frame will be referenced from the McFIS and the immediate previous frame, respectively, while only *dual* reference frames are used. To be more specific, we will use the McFIS to generate a new MR for the PVC technique and also use it as a reference frame for the pattern mode as well as other modes. The ultimate mode will be selected using the Lagrangian optimization.

As we have discussed in the first paragraph of the current section, real-world objects, by nature, may be in any arbitrary shape, and ME&MC using only rectangular-, square-, or even any regular-shaped blocks just approximate the real shape; thus, the coding gain would not be satisfactory. Thus, intuitively any attempt to encode a video by the PVC technique using content-based pattern templates generated through the adaptation of the MRs of the video will eventually provide a better coding performance [22]. The algorithms in [22] used a number of future frames to generate pattern templates, and thus, it may not be suitable for some applications where frame delay cannot be tolerated, such as real-time interactive communication. The algorithm tries to reduce frame delay up to the size of *group of picture* (GOP) using both previously generated patterns and current patterns by processing frames in a GOP. Moreover, due to the requirement of bits for shape coding of the pattern templates to maintain the same *pattern codebook* (PC) in the encoder and the decoder, the improvement of the rate-distortion performance is below the expected level.

In this paper, we introduce an efficient *arbitrary-shaped pattern-based video coding* (ASPVC) scheme using a content-based pattern generation strategy from decoded frames and the McFIS to avoid any frame delay and pattern shape coding (as both encoder and decoder use the same procedure and frames to generate pattern templates). The experimental results confirm that the proposed method outperforms the two recent and relevant algorithms by improving image quality significantly. The preliminary idea is published in [23]. This paper has been extended by including the following things: (1) generation of arbitrary-shaped content-based pattern templates from the decoded frames, (2) embedding the pattern mode into the H.264 framework by adjusting the corresponding bits and distortion, (3) new McFIS generation strategy based on the theoretical relationship between the distortion and quantization step size, (4) computational complexity comparison with other relevant existing techniques, and (5) more insight reasoning supported by data and analysis to show the superiority of the proposed algorithms compared to the algorithms.

The rest of the paper is organized as follows: Section 2 describes the motivation and detailed steps of the proposed pattern-based video coding scheme using the McFIS and pre-defined regular-shaped pattern templates, Section 3 explains details of another proposed pattern-based video coding scheme using dynamic and arbitrary-shaped pattern templates for better moving object approximation through real-time content-dependent generated patterns, Section 4 demonstrates the experimental setup and results and also analyzes and compares the proposed techniques with contemporary and relevant techniques [1,21,24,25], and Section 5 concludes the paper.

2 Proposed PVC scheme using regular-shaped pattern templates

Generally, ME&MC using more than one reference frame (i.e., MRFs) exhibit better rate-distortion performance compared to a single reference frame (i.e., using the immediate previous frame) in the expense of computational time [24-30]. The computational time in the MRFs increases almost proportionally with the number of reference frames for ME&MC. Dual reference frame techniques [24-26] represent a good compromise between single and MRFs in terms of computational time and rate-distortion performance. The proposed scheme is a pattern-based video coding under the H.264/AVC framework (wherein pattern mode is embedded) with dual reference frames. Between the dual frames, one is the immediate previous frame and the other is the McFIS, assuming that motion areas and normal/uncovered static areas will be referenced from the immediate previous frame and the McFIS, respectively, through Lagrangian optimization.

The McFIS is generated by dynamic background modeling using Gaussian mixture models [14-16]. It is constructed from the already encoded frames at the

encoder and decoder using the same technique so that the McFIS need not to be transmitted from the encoder to the decoder. When a frame is decoded at the encoder/decoder, the McFIS is updated using the newly decoded frame. The detailed procedure will be described in Subsection 2.3. To exploit the non-rectangular MB partitioning and partially skipped mode, a pattern mode is incorporated as an extra mode into the conventional H.264 video coding standard and is defined as the PVC scheme [12,13]. Figure 1 has shown the PC comprising 32 patterns which are used in the proposed scheme. Each pattern is a binary 16×16 pixel matrix, where the white region indicates 1 (i.e., to capture foreground) and the black region indicates 0 (i.e., to capture background). Actually, the pattern is used as a mask to segment out the foreground from the background within a 16×16 pixel MB.

We need to determine MR for the current MB using the MBs from the current and reference frames. Then to find the best matched pattern from the PC through a *similarity metric* [12], ME&MC are carried out using only a pattern-covered MR (i.e., covered by the white region in Figure 1). In the proposed scheme, we also introduce a new pattern matching scheme for ME&MC so that we can overcome the occlusion problem in the existing schemes by exploiting uncovered background.

If an MB has both moving object and background, then the block is normally encoded using a pattern mode. In the pattern mode, the moving object is approximated using the best matched pattern, and the rest of the area is treated as skipped area (i.e., it is not encoded). The existing PVC scheme determines the moving object using MR which is the difference between the current block and its collocated block in the reference frame (i.e., immediate previous frame). If we calculate the MR in this way, we get two moving regions: (1) one is due to uncovered background, i.e., the old place where the object was at the $(t-1)$th time, and (2) the other is due to the movement of the object in the new location, i.e., the new place where the object is at the tth time. Actually, the MR in the second case represents the object, and the MR in the first place represents the uncovered background. Both MRs are drawn in Figure 2c. As the McFIS comprises only the background, if we extract the MR by comparing the current block against the McFIS, we will get the MR which represents the true object rather than both uncovered background and object. Then, in the proposed scheme, we determine the best matched pattern for the pattern-covered areas and perform motion estimation and compensation using the immediate previous frame to find the best matched object. The rest of the region of the block is copied from the collocated block in the background frame, i.e., McFIS. The experimental results reveal that (see Figure 3) the proposed scheme encoded more RMBs compared to
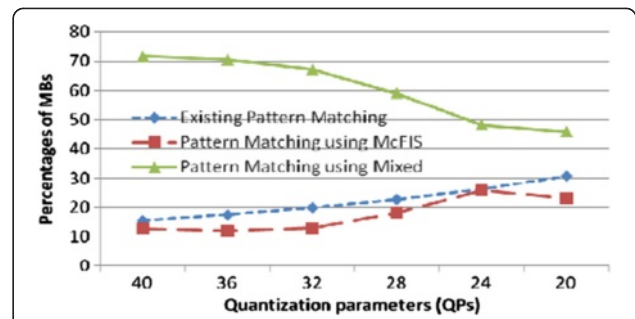

Figure 3 Average percentages of MBs selected by the three techniques. (1) The existing pattern matching [12], i.e., matching and ME&MC being carried out using immediate previous frame, (2) pattern matching and ME&MC being carried out using the McFIS, and (3) pattern matching using the McFIS but ME&MC being carried out with the immediate previous frame.

the existing schemes. Encoding more RMBs indicates more compression.

Thus, the new ideas are (1) extracting only moving object areas as MR rather than both object and background areas, (2) performing motion estimation and motion compensation using pattern-covered area (i.e., moving object areas) using immediate previous frame, and (3) treating the background area as a skipped area and coping from the McFIS. The detailed procedures will be explained in the following subsection.

2.1 New ME&MC for uncovered background areas

Let F_k^i and F_k^{i-1} be the kth MBs from the ith and $(i-1)$th frames, respectively. According to the PVC scheme [12], the MR M_k^i is defined as follows:

$$M_k^i(x, y) = \left| F_k^i(x, y) - F_k^{i-1}(x, y) \right|. \tag{1}$$

The similarity of a pattern $P_n \in PC$ with the MR in the kth MB is defined as

$$S_{k,n}^i(x, y) = \sum_{x=0}^{15} \sum_{y=0}^{15} \left| M_k^i(x, y) \times P_n(x, y) \right|. \tag{2}$$

The best matched pattern for an MR is then selected as

$$P_j = \arg \max_{\forall P_n \in PC} \left(S_{k,n}^i \right). \tag{3}$$

Figure 2b shows a current frame, Figure 2a shows a reference frame, the MR (marked as texture) according to Equation 1 is shown in Figure 2c, and a true background without an object (here a moving ball) is shown in Figure 2e. From the figure, we can easily observe that the second block of the third row (marked as block A in Figure 2b) has both a moving object (see Figure 2a) and an uncovered background (see Figure 2b). When ME&MC

are carried out for block A (i.e., uncovered background), there is no matched region for block A in the reference frame (i.e., in Figure 2a). Thus, the pattern mode as well as any other mode could not provide accurate ME&MC for blocks similar to A. This problem can be solved if we can generate a true background (Figure 2e) and if ME&MC are carried out using the background as a reference frame by any suitable H.264 mode or pattern mode (if the MR is best matched with any pattern). In this work, we use McFIS (actually a dynamic background frame; to be discussed in Subsection 2.3) for referencing the uncovered background.

When a pattern is matching against the MR (i.e., the part of the ball) in block B (Figure 2b), ideally, pattern 11, 14, or 30 (see Figure 1) would be the best matched pattern, but due to the MR generated by (1) (see Figure 2c) comprising both a moving object and an uncovered background, pattern 21 is the best matched pattern. However, ME&MC using pattern 21 do not find a proper reference region in any reference frames (i.e., Figure 2a or Figure 2e) and result in poor rate-distortion performance. To solve this problem, we need to generate a new MR using the McFIS and the current frame (see Figure 2d) and then use the immediate previous frame for ME&MC using the pattern-covered region, and the rest of the region of the MB is copied from the collocated background frame, i.e., McFIS. In this process, we need to replace F_k^{i-1} in (1) by kth MB from McFIS (i.e., $McFIS_k^i$) to find the object motion. However, we also use two other options using the existing pattern matching with Equation 1 and ME&MC, i.e., using the immediate previous frame (existing pattern matching) and using the McFIS (pattern matching using McFIS) to maximize the rate-distortion performance where an MR is not well matched with the best pattern.

Figure 3 compares the average percentages of MBs selected by the Lagrangian optimization as the reference MBs for three relevant techniques: (1) the existing pattern matching [12] (where the MR is determined based on the difference between the current block and the collocated block in the immediate previous frame), i.e., matching and ME&MC being carried out using the immediate previous frame, (2) pattern matching (where the MR is determined based on the difference between the current block and collocated block in the McFIS and then find the best matched pattern for the MR) and ME&MC being carried out using the McFIS, and (3) pattern matching using the McFIS but ME&MC being carried out with the immediate previous frame. Note that technique 3 is the newly introduced pattern matching and ME&MC approach in this work, while technique 2 is the existing MRF approach with the McFIS. Techniques 1 and 3 have a difference in generating the MR of the current MB (against either the McFIS or the immediate previous frame) to find the best

matched pattern at the encoder but have no difference in decoding point of view as both use the immediate previous frame as the reference frame. Thus, we accommodate all three techniques for the pattern mode. The first 300 frames of six standard video sequences, namely *Paris*, *Bridge Close*, *Silent*, *News*, *Salesman*, and *Hall Objects*, have been used for the evaluation. The figure shows that the proposed technique 3 selects the most number of MBs compared to the other two techniques. The higher percentage represents higher effectiveness for referencing. The results indicate that the proposed pattern matching and ME&MC technique are expected to perform better, and this will be further evidenced by the rate-distortion performance in Section 4.

Figure 3 shows that the percentages of RMBs are from 45% to 70% in the proposed scheme, whereas they are from 10% to 30% in the existing schemes. The rest of the MBs are encoded as the traditional H.264 MBs. At a low bit rate (when *quantization parameter* (QP) is high), the percentage of RMB (i.e., MBs selected by the pattern mode) is larger, and it gradually decreases with the bit rates in the proposed scheme. The decreasing trend of the RMBs with bit rates is quite understandable. The rationality is that at low bit rates, if an MR of an MB is not completely covered by the best matched pattern, the MB can be still encoded using the pattern mode (i.e., as RMB) as the distortion due to the unmatched area might not be significant compared to bit rate saving in calculating the Lagrangian cost function. However, at high bit rates, the distortion might be significant compared to bit saving in calculating the cost function to select other modes compared to the pattern mode.

2.2 Embedding pattern mode within the H.264 framework

Due to the object's shape, motion characteristics, prediction accuracy, and ratio of foreground and background in an MB, a certain mode might not always have a specific ratio-distortion (R-D) characteristics; however, it is a general trend that when the Lagrangian multiplier is relatively high (at low bit rates), more emphasis is given to the bit rates compared to the distortion; on the other hand, when the Lagrangian multiplier is relatively low (at high bit rates), more emphasis is given to the distortion compared to the bit rates. Thus, for a given block (16 × 16), larger modes (such as 16 × 16, 16 × 8, and 8 × 16) might be chosen at low bit rates, whereas smaller modes (8 × 8, 8 × 4, 4 × 8, and 4 × 4) might be chosen at high bit rates. The general tendency of the pattern mode is that it provides less bits and high *mean square error* (MSE) compared to the other modes because we only consider pattern-covered areas for bits but overall areas for calculating MSE (non-pattern-matched area contributing in higher MSE). Figure 3 also shows the same tendency as

the number of MBs encoded as the pattern mode is decreasing with the bit rates.

In the proposed method, we have added a pattern mode and kept all other H.264 modes including 4×4, 4×8, and 8×4. Thus, if the 8×8 block mode is selected, then 8×8 blocks are further decomposed into smaller modes, but we do not use any smaller size pattern mode (e.g., 16-pixel patterns) in the decomposition for the proposed scheme. The pattern size is 64; thus, if we let parts (i.e., 64 pixels) of the 16×16 block to be inter-predicted and the others are skipped, the setting can be used to approximate any of the patterns. The main difference is that in the pattern mode, a 16×16 MB (i.e., 256-pixel block) is represented by a smaller block, i.e., one of the 64-pixel patterns; on the other hand, H.264 treats the MB as a 256-pixel block by signaling zero motion vector and zero residual errors for the skipped areas. In the PVC scheme, we need to send some bits for the pattern index; on the other hand, the H.264 needs to send some bits to signal the zero motion vectors and zero residual errors for the skipped areas. The experimental results reveal that ultimately the pattern mode is the winner for significant times through Lagrangian optimization.

As the size of a pattern (to capture and encode MR) is one fourth (i.e., 64 pixels among 256 pixels) of an MB, ME&MC using pattern-covered areas generally provide less bits (due to the coding of one fourth of the areas) and more MSE (due to the mismatch between a pattern and the MRs) compared to the other modes such as 16×16, 16×8, 8×16, and 8×8. After analyzing a number of video sequences, we have observed that the average bits required by the 16×16, 16×8, 8×16, and 8×8 modes against the pattern mode are 2.61, 2.78, 2.71, and 2.93 times, respectively. The corresponding MSE ratios are 0.91, 0.89, 0.89, and 0.86. Thus, using conventional *Lagrangian multiplier* (LM) recommended in the H.264, i.e., $\lambda = 0.85 \times 2^{(QP-12)/3}$, the pattern-based video coding scheme encodes a large number of MBs as RMBs, which results in low bit rates with low peak signal-to-noise ratio (PSNR) compared to the H.264 for a similar QP. This may be a problem for the existing rate-control mechanisms as the relationship of the QPs and the rate-distortion may be different. To address this problem, a comprehensive R-D analysis is given by Paul and Murshed in [12], where a pattern mode has been embedded into the H.264 coding framework by modifying LM. The Lagrangian multiplier (after embedding the pattern mode) is relatively smaller compared to the H.264-recommended LM. Paul and Murshed [12] recommended a new LM: $\lambda^{PVC} = 0.4 \times 2^{(QP-12)/3}$. If a video sequence has a few numbers of RMBs (for example, very high motion video sequence), the amendment of LM may cause a problem in the existing rate-control mechanism.

An intuitive solution is to change the Lagrangian cost function (by adding distortion with the product of bits and the LM) of the pattern mode and then allow this mode into the competition with the other modes under the H.264 optimization framework. Thus, the existing rate-control mechanism based on the QP-rate-distortion relationship does not affect so much. As the pattern mode yields less bits with more MSE compared to the other H.264 modes, we can adjust the cost function of the pattern mode by adjusting MSE and bits. Figure 4 shows (by dotted lines) bits and MSE ratios between different conventional modes (i.e., 16×16, 16×8, 8×16, and 8×8) and the pattern mode against different QPs before adjustment of the bits and MSE generated by the pattern mode when the pattern mode is selected by the H.264-recommended LM. It shows that, on average, 2.76 times of bits is required by the conventional modes and, on average, 0.89 for the MSE ratio is obtained by the other modes compared to the pattern mode. We adjust (i.e., by reducing) MSE with the pattern mode by generating high-quality MRs (i.e., pattern-covered MRs) using finer quantization compared to the other modes. We also adjust bits by multiplying the corresponding bits by a factor, β (>1), in cost function determination to restrict some MBs to be classified as RMBs where the MBs are poorly matched with the best pattern.

Obviously, making the MSE ratio towards 1 (i.e., the MSE for the pattern mode is the same as that of other modes) by finer quantization and keeping the bit requirement at its lowest level by multiplying with β for the pattern mode would be the desirable case to improve overall rate-distortion performance. Figure 4 shows (solid lines) bits and MSE ratios between different conventional modes (i.e., 16×16, 16×8, 8×16, and 8×8) and the pattern mode against different QPs after adjustment ($QP_{pvcmode} = QP_{othermode} - 2$ and $\beta = 1.5$) of the bits and MSE. It shows that, on average, 2.37 (instead of 2.76 before adjustment) times of bits is required by the conventional modes and, on average, 0.95 (instead of 0.89 before adjustment) for the MSE ratio is generated by the other modes compared to the pattern mode. Note that this adjustment makes sure that for a given QP, the PSNR of the PVC is comparable with the H.264, although the corresponding bit rate of the PVC is much lower compared to that of the H.264, so that the PVC exhibits better overall rate-distortion performance compared to the H.264. To make the coding performance uniform for a wide range of bit rates and low to high motion video sequences, we have adjusted the bits and distortion for the pattern mode based on the experimental results.

2.3 New McFIS generation technique

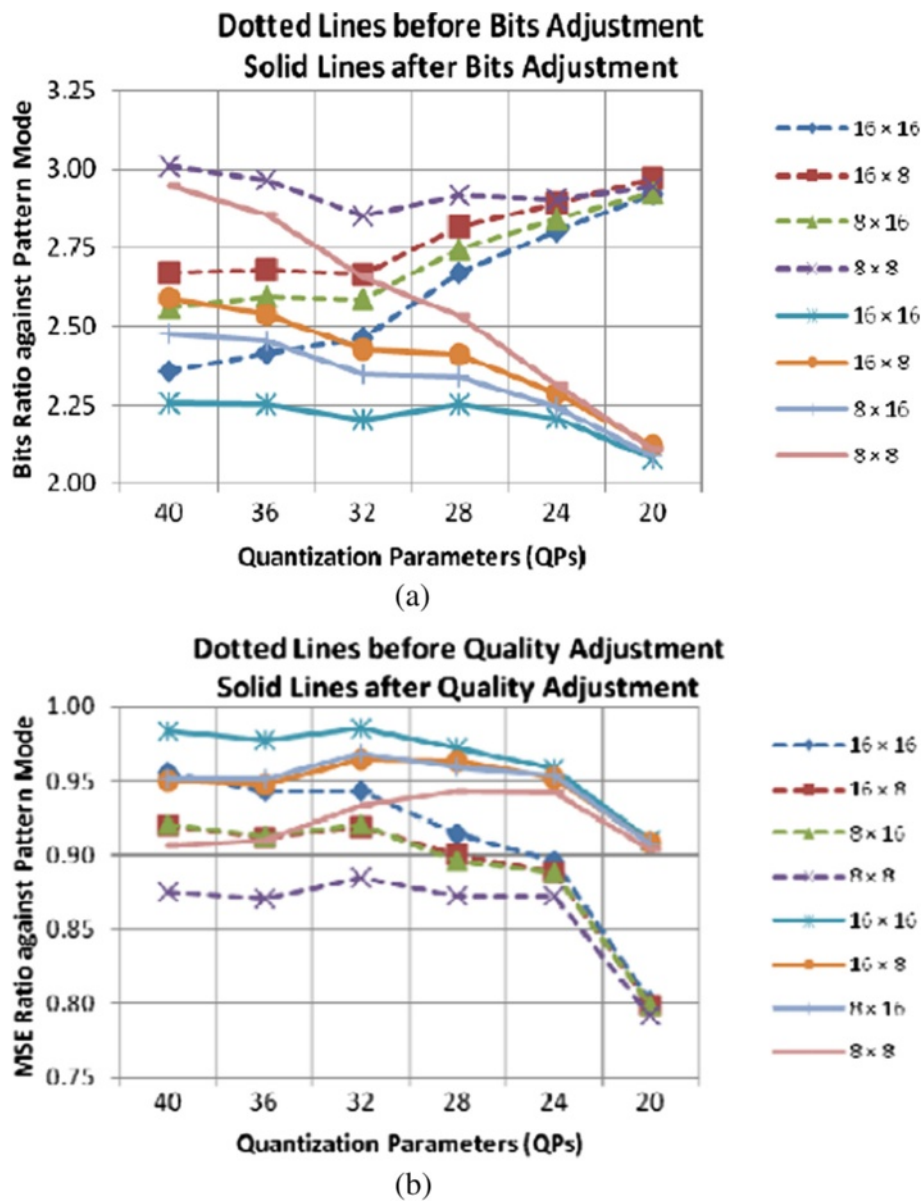In a video scene, a pixel may be a part of different objects and backgrounds over the time (i.e., in the temporal

Figure 4 Comparison of bits (a) and MSE (b) by different modes against the pattern mode. The dotted lines and solid lines indicate before and after quality or bit adjustment, respectively. Different modes are 16 × 16, 16 × 8, 8 × 16, and 8 × 8.

frames). Each part can be represented by a Gaussian model expressed by pixel intensity *variance*, *mean*, and *weight* [14-16]. Thus, to model a pixel over the time, Gaussian mixture models are used. Intuitively, if a model has large weight and low variance, then most probably the model represents the most stable background. A mean value of the best background model is taken as background pixel intensity for that pixel. In this way, an entire background frame (i.e., McFIS) is constructed. Instead of the mean value, the last satisfied pixel intensity (preserved when a pixel satisfies a model) may be taken as the background pixel intensity to avoid an artificial mean value [16]. As mentioned in [21], background generation using pixel

mean (or pixel recent value) is not very effective in video coding applications as the McFIS is generated from the distorted image (i.e., the decoded frame); neighboring pixel intensities within the McFIS (i.e., spatial correlation) are therefore used to generate better McFIS [21]. We have also observed that there is pixel intensity similarity among neighboring pixels. This relationship is also observed by the other researchers, and thus, pre-/post-filtering techniques were introduced by exploiting neighboring pixels to reduce pixel intensity discrepancy in decoded frames due to the quantization and/or block-based ME&MC [31,32]. Paul et al. in [33] generated McFIS with modified decoded frames using neighboring pixel intensities of the decoded

frames. By further investigation, we have found that temporal correlation exploitation is also crucial to construct McFIS along with the spatial correlation. Thus, we modify the existing McFIS [33] as follows, assuming D^i and D^{i-1} as the ith and $(i-1)$th McFISs, respectively:

$$D^i(x,y) = \begin{cases} \tau D^i(x,y) + (1-\tau)D^{i-1}(x,y) & if \ \left|D^i(x,y) - D^{i-1}(x,y)\right| < T_p, \\ D^i(x,y), \text{otherwise} \end{cases},$$

$$(4)$$

where τ ($0 < \tau < 1$) and T_p are the weighting factor and threshold, respectively. It is obvious that there should be a strong correlation between consecutive McFISs especially in the stable region (i.e., background). A small difference (i.e., T_p) may be due to the quantization error instead of different environments. Thus, to rectify this variance, a weighted average is formulated for the current McFIS from the previous McFIS. A large value of τ means that we give more emphasis to the current McFIS.

Although the current pixel and the pixel at the collocated position of the previous McFIS are similar, the abovementioned T_p adjustment is due to the quantization error but not environment changes (i.e., not for object movements in the background areas), and we have investigated the distortion due to quantization. As the quantization error varies with the quantization step, T_p should vary with QP, since a large QP creates high distortion while low QP creates low distortion. To find the relationship of T_p with QP, we need to know the relationship of the distortion and quantization step size. There is a theoretical derivation of the relationship of the distortion and the quantization in [34] where it is shown that the mean square quantization error varies with $\Delta^2/12$, with Δ being the quantization step size. This approximation is fairly accurate when the quantization step size is smaller than the signal standard deviation [34], at the middle range of quantization (but applicable for the entire range of quantization). The relationship (when distortion is defined as the mean quantization error) is plotted in Figure 5. We have also investigated this relationship using actual frames and with their reconstructed frames of a number of video sequences such as Paris, Silent, Salesman, and News (average result is plotted in Figure 5). The figure shows that the experimental value is smaller than the theoretical one due to the other factors (more accurate ME&MC using VBS, significant amount of static regions, etc. while the H.264 is used). We have fixed T_p (plotted in Figure 5) as two times of the experimental value as both the current McFIS and the previous McFIS can suffer from quantization error. To minimize the quantization error on background areas for improving the quality of the McFIS, T_p can be approximated as $T_p = 0.6513e^{0.0861 \times QP}$.

Figure 5 Proposed threshold T_p as a function of QPs. With theoretical and experimental distortion (mean error) using a number of video sequences.
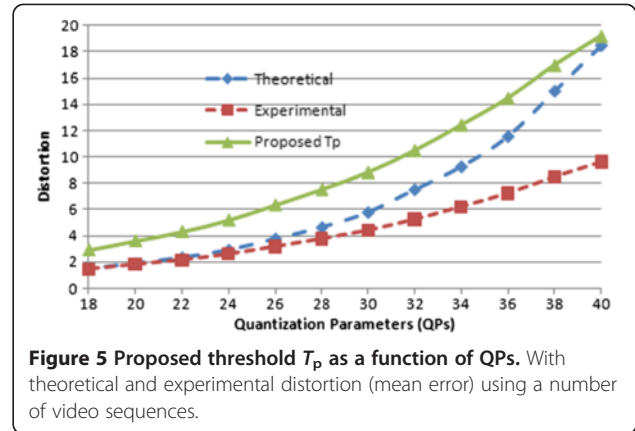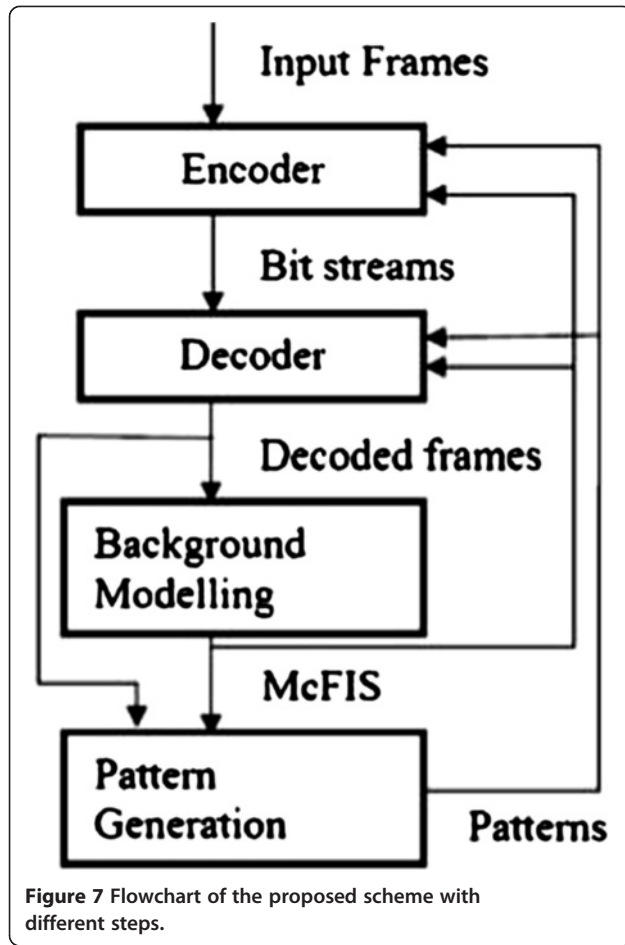
2.4 Encoding and decoding of the proposed scheme

In the proposed scheme, the first frame of video is encoded as an intra-frame, and the subsequent frames are encoded as inter-frames until a scene change [35] occurs. When a frame is encoded and decoded at the encoder, the McFIS is updated using the most recent decoded frame through background modeling. When a scene change occurs, the modeling parameters are reset and a new McFIS will be generated. As the McFIS contains a *stable* portion of a scene, the *sum of absolute difference* (SAD) between the current frame and the McFIS is a good indicator for scene change. Obviously, an automatic (not hand-made) cut of the scene (i.e., scene change) cannot be consistently defined and clearly confirmed. In this scheme, a scene change is detected in two different ways: (1) based on the ratio of SAD_i and SAD_{i-1} where SAD_i is calculated between McFIS and the ith frame (i.e., the current frame) and SAD_{i-1} between the McFIS and the $(i-1)$th frame (i.e., the previous frame) and (2) based on the percentage of the McFIS references in encoding. As the McFIS contains a stable portion of a scene, SAD_i between the current frame and the McFIS is a good indicator for scene change. In the proposed scheme, we consider scene change if $\frac{SAD_i}{SAD_{i-1}} > 1.7$. Paul et al. [36] mentioned that the percentage of McFIS referencing is a good indication to test the relevance of the current McFIS as a reference frame. Thus, we also generate a new McFIS if the percentage of the McFIS reference is below a threshold (e.g., for the current implementation, we use 3%). For each MB, we have examined all modes including the pattern mode using two reference frames, and then the ultimate mode is selected based on the LM. In the pattern mode, we only conduct ME&MC using regions covered by the best pattern (using Equation 3).

To avoid more than four 4×4 DCT transformations for a pattern (as 64 1's in a pattern), we need to rearrange the residual errors covered by the pattern into one 8×8 block. For arranging residual errors, we scan the 16×16 residual block row-wise from top to bottom and left to right and

put them into an 8 × 8 block when we find 1 in the corresponding matched pattern. Arrangement of residual errors for a pattern is shown in Figure 6. A pattern is shown in Figure 6a, numbering of residual errors according to the position of 1 in a pattern is shown in Figure 6b, and arrangement of residual errors into an 8 × 8 block according to the numbering is shown in Figure 6c. The inverse arrangement is also used in the decoder to get back the original shape of the pattern-covered residual error for block reconstruction.

3 Proposed PVC scheme with arbitrary-shaped pattern templates using McFIS

Obviously, the content-based PVC in [22] outperforms the PVC with pre-defined regular-shaped patterns [12,13] due to the better moving-region shape approximation. The limitations of the content-based PVC approach are its frame delay due to pattern generation using future frames (and then encode the frames using the generated patterns) and requirement of bits for encoding patterns themselves for transmission to the decoder (to make the same PC available at the encoder and decoder). Intuitively, by processing a smaller number of future frames provides better object shape approximation (by the generated patterns) with reduced frame delay but requires more bits to encode patterns themselves (after each time of pattern generation). On the other hand, processing a larger number of future frames provides poorer shape approximation (and fewer bits for encoding patterns themselves) with increased frame delay for pattern generation and encoding frames. Thus, in the proposed ASPVC scheme, we use a small number of decoded frames to generate patterns and encode future frames. As we do not use any future frame in the pattern generation process, the proposed scheme does not bring about any frame delay. Moreover, the proposed ASPVC scheme uses the same procedure to generate pattern templates at the encoder and the decoder, and thus, we do not need to encode the shape of pattern templates. As a result, better rate-distortion performance can be achieved due to the saving of bits which are previously used for pattern-shaped coding.

Figure 7 shows the flowchart of the proposed scheme with different steps. Each frame is encoded using the pattern mode with other relevant H.264 modes with extra reference frame, i.e., McFIS. The McFIS is generated from decoded frames. The content-based patterns are generated using decoded frames and the McFIS.

The proposed ASPVC technique needs to generate MRs from already decoded frames, create a PC comprising a number of pattern templates from those MRs using a suitable algorithm (such as [22]), and then encode frames using all modes including the pattern mode with the generated PC. Note that for the first time (when no arbitrary-shaped patterns are available), the proposed ASPVC uses

Figure 6 Rearrangement of the residual errors covered by a pattern into an 8 × 8 block. (a) An example of a pattern. **(b)** Numbering the positions according to row-wise scan order (i.e., top to bottom and left to right). **(c)** Positioning the residual errors according to the numbering into an 8 × 8 block.

pre-defined patterns. As we use the same technique and decoded frames at the encoder and decoder, we do not need to transmit patterns themselves to the decoder. The same arrangement (left to right and top to bottom) and re-arrangement procedure is also applied to the residual errors (covered by the pattern) at the encoder and decoder to avoid multiple 8 × 8 blocks (see Subsection 2.4). The following subsections discuss the detailed procedures of MR generations, content-based pattern generations, and other issues related to the proposed ASPVC scheme.

3.1 Moving region detection

To generate a PC, we need MRs from all MBs of the participated frames F^{i-1} to F^{i-n} for encoding the i to $(i+n)$th

Figure 7 Flowchart of the proposed scheme with different steps.

frames where $n > 0$. To capture MRs created by the moving object (but not uncovered background) only, we use Equation 1 for MRs, M^i, by replacing F^{i-1} with the $(i-1)$th McFIS i.e., $McFIS^{i-1}$. The selection of n has an impact on overall performance, i.e., rate-distortion, memory requirement, and computational time of the proposed technique. Setting $n = 1$ requires PC generation for every frame coding. Thus, it requires more computational time due to the PC generation overhead but less memory requirement (i.e., needing to store only one decoded frame) and better rate-distortion performance. Setting $n > 1$ requires more memory (to store n decoded frames), least computational time (due to less PC generation overhead), and poorer rate-distortion performance. In our experiment, we have used $n = 3$ as the balance among memory requirement, computational time, and rate-distortion performance.

3.2 PC generation

After collection of all MRs of the participated frames, we divide them into α clusters using a clustering algorithm such as *fuzzy C-means* [37,38] or *K-means* [39] to cluster them into α classes based on the *gravitational center* (GC)

[22] of MRs. The GC of an MR is a weighted average of all coordinates of non-zero errors where the corresponding weights are their absolute error values. The GC is defined as follows:

$$
GC(M_k^i) = \left(\frac{\sum_{x=0}^{15}\sum_{y=0}^{15} xM_k^i(x,y)}{\sum_{x=0}^{15}\sum_{y=0}^{15} M_k^i(x,y)}, \frac{\sum_{x=0}^{15}\sum_{y=0}^{15} yM_k^i(x,y)}{\sum_{x=0}^{15}\sum_{y=0}^{15} M_k^i(x,y)} \right),
$$
(5)

where M_k^i is the MR of the kth macroblock in the ith frame and (x, y) is the coordinate of a position.

Figure 8 shows the distribution of GCs of a number of MRs and their corresponding cluster using K-mean clustering algorithm while encoding the Salesman video sequence at QP = 30. When we cluster the MR, we try to keep the closest (in terms of distance based on the GC) MRs into one cluster. A simplistic greedy heuristic is used to form a pattern from each cluster [22]. We add all MRs of a cluster using matrix addition to get the cumulative MRs for a cluster. The cumulative MRs (see Figure 9a,d) give us an indication of important areas within the 16×16 block for a cluster. We then select the highest magnitude 64-position (see Figure 9b,e) among them and assign 1 to them and 0 to the rest of the positions to form a 64-pixel binary pattern. This procedure provides a *local* optimal pattern generation for a given cluster [22]. We use the same pattern generation technique used in [22] but with a different MR which contains different errors rather than only those with 1 and 0.

Figure 9 shows the steps of the pattern generation technique using the Salesman video sequence while encoding at QP = 30 for the first-time pattern generation using decoded frames. The cumulative MRs (after adding all corresponding errors within a cluster) of two clusters
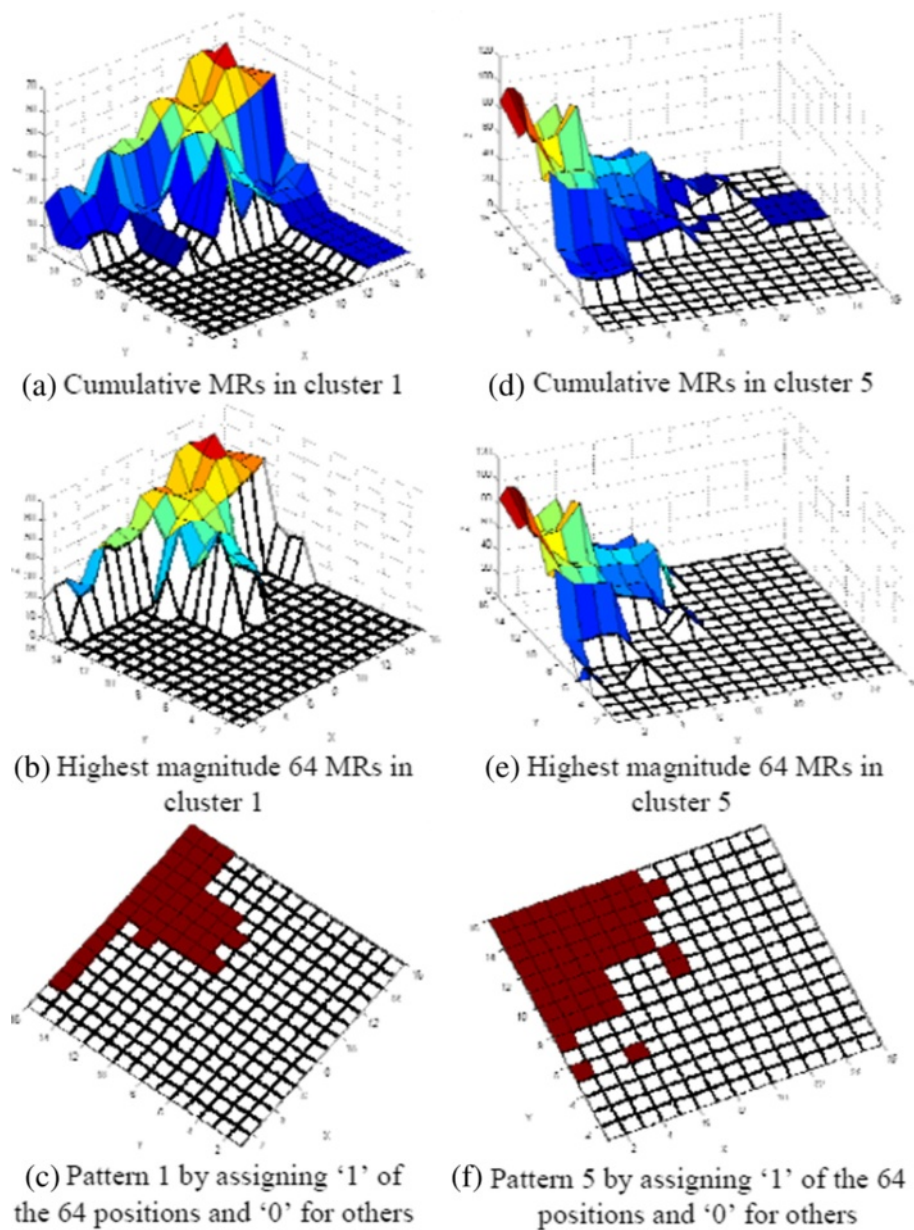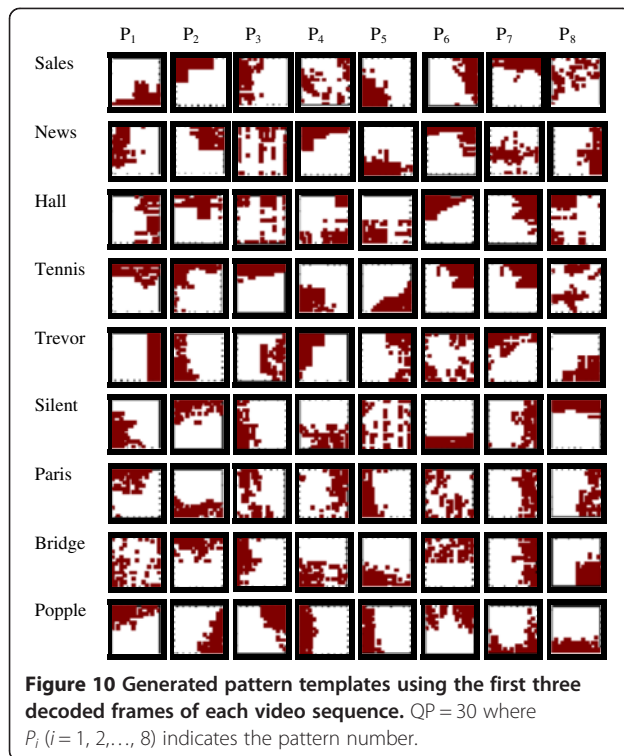

Figure 8 Example of MR clustering based on GCs of the MRs using *K*-mean clustering technique. (a) GCs of MRs with their grouping into one of eight clusters and **(b)** eight clusters (connected by lines among the centers of MRs) based on the GCs.
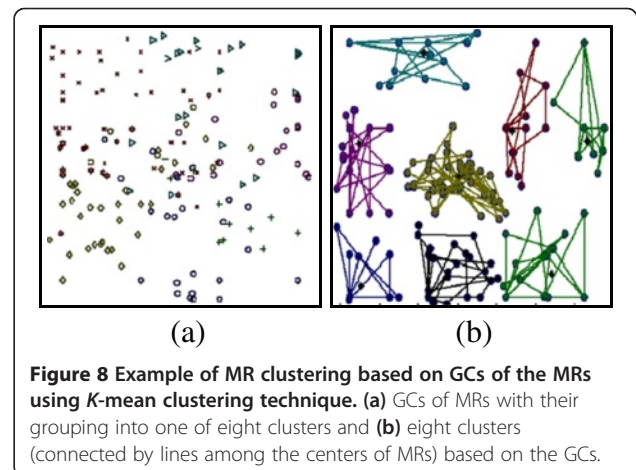
Figure 9 Content-based pattern generation steps through pictorial diagram. Two examples of pattern generation steps are shown using the Salesman video sequence. **(a)** Cumulative MRs in cluster 1, **(b)** highest magnitude 64 MRs in cluster 1, and **(c)** pattern 1 by assigning 1 of the 64 positions and 0 for others. **(d)** Cumulative MRs in cluster 5, **(e)** highest magnitude 64 MRs in cluster 5, and **(f)** pattern 5 by assigning 1 of the 64 positions and 0 for others.

(namely clusters 1 and 5) are shown in Figure 9a,d, respectively. Figure 9b,e shows the corresponding highest magnitude 64 values, while Figure 9c,f shows patterns 1 and 5 after assigning 1 to the highest magnitude 64 position and 0 to the rest of the positions. The final pattern sets from different video sequences are listed in Figure 10 for the first instance (i.e., using the first three decoded frames) while encoding at QP = 30.

3.3 Impact of number of patterns and size of patterns

Obviously, a large number of patterns can approximate the different shapes of the MRs well but require more bits to identify the pattern itself. For example, 32 patterns require 5 bits to identify an individual pattern if we use a fixed length code. Using some sophisticated approach, we may reduce the identification code size, but we have observed that more than 32 pre-defined patterns are not

Figure 10 Generated pattern templates using the first three decoded frames of each video sequence. QP = 30 where P_i ($i = 1, 2, ..., 8$) indicates the pattern number.

suitable for any video sequences. In the case of content-dependent patterns, we have observed that generally eight patterns are suitable for all videos, although for some cases (where the number of MBs classified by the pattern mode is high for low motion videos), slightly better performance can be observed using 16 patterns.

All the pre-defined patterns are with 1′s in 64-pixel positions that covered MRs, and ME&MC are carried out using those positions only in the pattern mode. If the pattern mode wins the competition with the other modes based on the Lagrangian multiplier, theoretically, it may provide four times for compression (actually, 2.7 times; see Figure 4) compared to the other modes due to the one-fourth size of the pattern against a 16×16 block.

Figure 10 shows eight generated patterns using the technique above for each of the nine video sequences using the first three decoded frames. We may note that the shapes of the patterns are irregular and different from each other compared to the pre-defined patterns (see Figure 1). As the patterns are generated from the content of the video, we can expect better rate-distortion performance if we encode the video using these patterns. Obviously, the patterns generated from the next instance (i.e., using different frames) would be different from those in Figure 10 due to the different MRs.
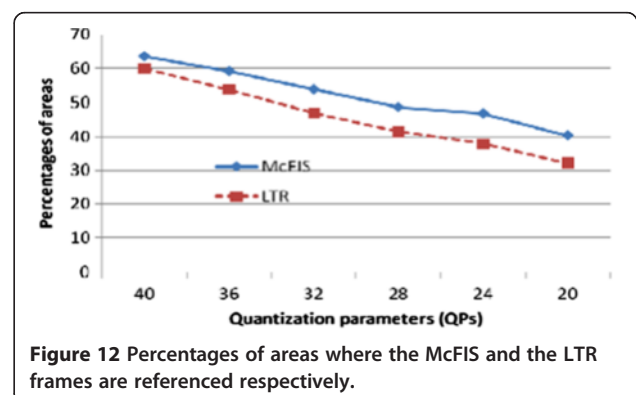
4 Overall experimental results

Apart from the experimental results reported in the previous sections to provide the ground for the idea of both

Figure 11 Computational comparisons of the proposed methods against the H.264 with single reference frame. Proposed methods: McFIS-PVC and McFIS-ASPVC, H.264-5Refs, McFIS-D, and LTR-PVC.
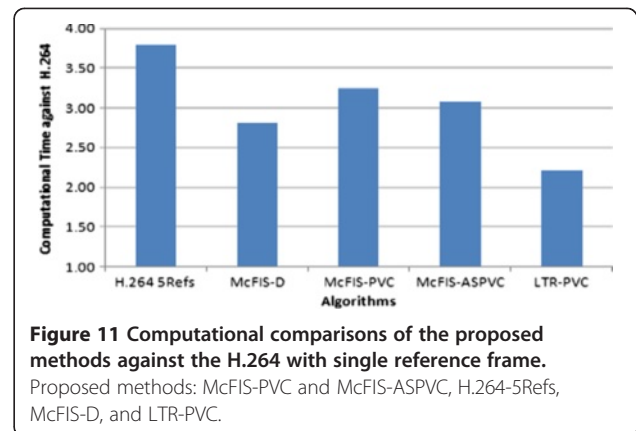
regular- and arbitrary-shaped pattern-based coding, overall experiments are also performed using nine standard video sequences (Salesman, News, Hall Objects, Tennis, Trevor, Silent, Paris, Bridge Close, and Popple) with QCIF, CIF, and 4CIF resolutions toward effectiveness of referencing, computational time, and rate-distortion.

All sequences are encoded at 25 frames per second and 32 frames as the GOP size. Full-search quarter-pel ME with ±15 as the search length is used. We have used the IPPPP… format. We have proposed two schemes: one is pattern-based video coding with pre-defined regular-shaped pattern templates where we used McFIS as the second reference frame and termed the technique as McFIS-PVC, and the other is dynamic pattern-based video coding with content-dependent arbitrary-shaped pattern templates where we also used McFIS as the second reference frame and termed the technique as McFIS-ASPVC. We have compared the proposed schemes (i.e., McFIS-PVC and McFIS-ASPVC) with a number of algorithms to demonstrate their strength. The technique we have selected for comparisons are the following:

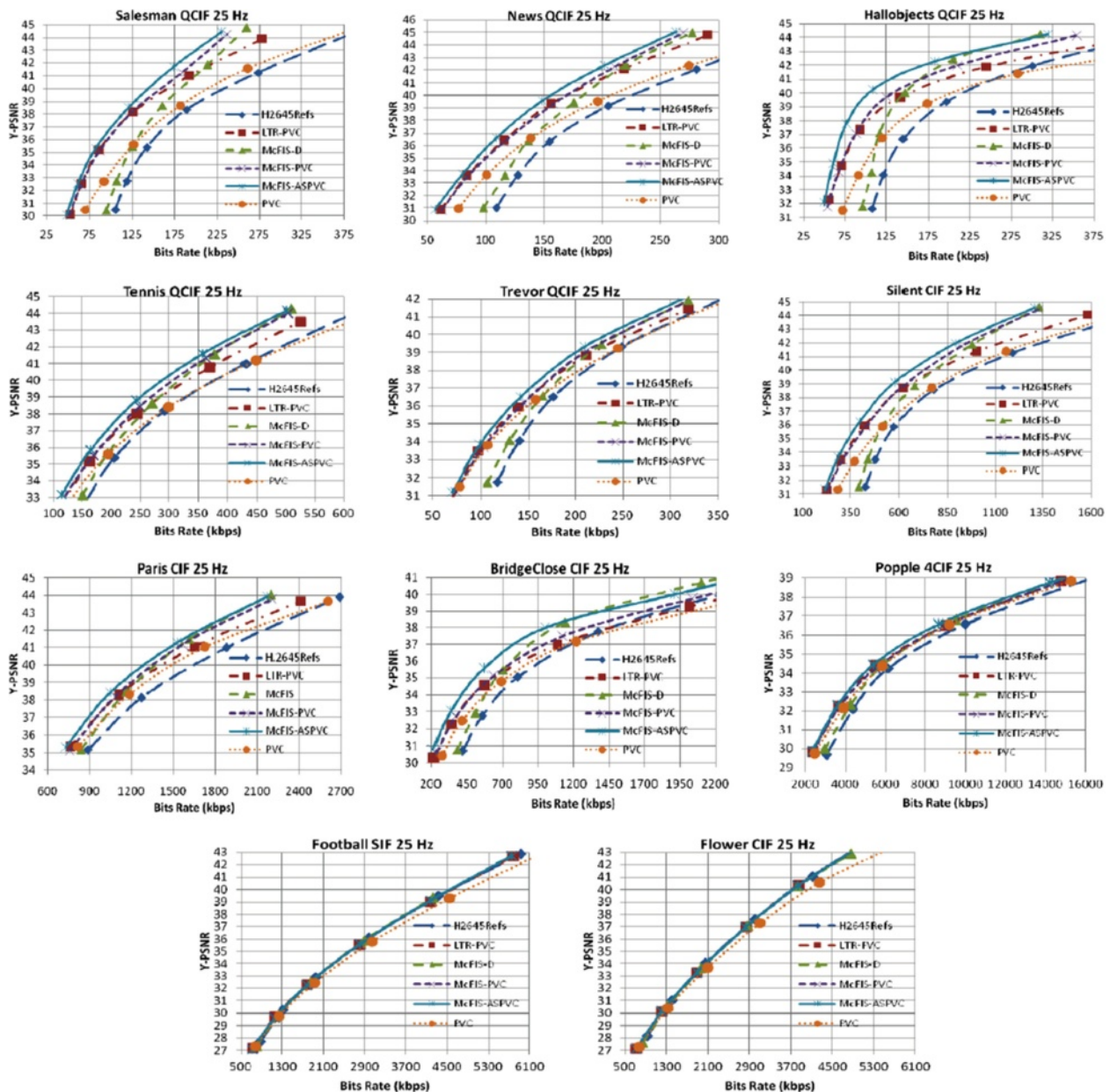- *H.264-5Refs.* The latest video coding standard H.264 [1] with five reference frames to see the

Figure 12 Percentages of areas where the McFIS and the LTR frames are referenced respectively.

Figure 13 Rate-distortion performance of the proposed and conventional HQLTR frame techniques. Rate-distortion performance of the proposed (McFIS-PVC and McFIS-ASPVC), the conventional HQLTR frame (LTR-PVC) [24,25] embedding the PVC, the PVC [12], the H.264 with five reference frames (H.264-5Refs), and dual-frame reference frames using the most common frame in a scene (McFIS-D) [21] for nine standard video sequences.

performance of the proposed schemes as this technique is the general state of the art in video coding techniques.

- *PVC.* The pattern-based video coding in [12] is the best algorithm in terms of rate-distortion performance among existing PVC algorithms. Thus, we have compared the proposed approach with this algorithm.
- *LTR-PVC.* The long-term reference (LTR) frame [24,25] is a good competitor of the McFIS (i.e.,

dynamic frame) for a coding scheme using dual reference frames. Thus, we apply the PVC technique using the LTR frame and select for comparison as this comparison will tell how effective the McFIS over the LTR frame is when both use the PVC technique.

- *McFIS-D.* The algorithm in [21] where the McFIS (generated from decoded frames) is used as the second reference frame, but no pattern mode is used. The

algorithm in [21] also differs from the proposed scheme in the McFIS generation where spatial neighboring pixels were used to modify the McFIS (i.e., unlike in Equation 4 where the previous McFIS is used).

In our implementation, we use high-quality LTR (HQLTR) and high-quality intra-(I)-frame for better performance. To ensure this, we set the QPs for HQLTR and the I-frame as QP(I) = QP(HQLTR) = QP(P)-4, where QP(.) represents the corresponding QP in the inter-frame.

4.1 Computational time
To compare the computational complexity, we have calculated the computational time of the proposed methods, i.e., McFIS-PVC and McFIS-ASPVC, and other methods such as H.264-5Refs, McFIS-D, and LTR-PVC against the H.264 with a single reference frame. The results are shown in Figure 11. The figure reveals that the proposed methods require more computational time compared to the McFIS-D and LTR-PVC methods. The McFIS-PVC requires extra time to process a large number of patterns (i.e., 32 patterns), while the McFIS-ASPVC requires extra time to generate patterns from the video frames. However, it is noted that the McFIS-ASPVC requires less computational time compared to the McFIS-PVC due to the small number of patterns (i.e., 8 instead of 32). However, the proposed techniques are faster compared to the H.264-5Refs technique as the H.264-5Refs requires more time for five reference frames (instead of two reference frames for the proposed techniques).

4.2 Rate-distortion performance
Figure 12 shows the average percentages of referencing with the McFIS and the LTR frames, respectively (where the remaining portions are referenced using the immediate previous frame). The results indicate that the McFIS captures more background areas compared to the conventional LTR frame. This translates into the improving rate-distortion performance by the proposed scheme compared to that of the LTR frame.

Figure 13 further shows the rate-distortion performance using the proposed McFIS-PVC and McFIS-ASPVC techniques, the conventional HQLTR frame technique LTR-PVC, the McFIS-D, the PVC [12], and the H.264-5Refs algorithms for nine standard video sequences. The figure confirms that the proposed methods consistently outperform the relevant four existing algorithms by 0.3 ~ 1.5 dB. The existing PVC algorithms could not outperform the standard H.264 with good margin at very high bit rates as the number of RMBs decreases with the bit rates. However, the proposed PVC with McFIS comprehensively outperforms the H.264 at any bit rates.

The proposed methods outperform the relevant state-of-the-art methods for the fixed and/or moderate camera motion video sequences (e.g., Tennis and Trevor). However, the proposed techniques in their current state could not provide better rate-distortion performance compared to the H.264 for the videos with high activities (i.e., camera/object motions) as the McFIS is least relevant for the referencing of the high camera motion videos and the number of RMBs of the high object motion videos (e.g., Football and Flower) is insignificant (around 3%) to improve the R-D performance of the proposed methods. Note that Figure 13 also confirms that unlike the PVC scheme [12], the R-D performance of the proposed schemes is similar to that of the existing relevant schemes for high-activity videos. This establishes our hypothesis on the adjustment of the bits and distortion in the Lagrangian cost function determination for the pattern mode while embedding the pattern mode into the existing H.264 framework.

The contributions of the paper are as follows: (1) to overcome the pattern matching limitation in the existing algorithms [12,22] for the occlusion scenario, a new MR detection technique is proposed using a true background frame, i.e., McFIS, which is also used for the pattern generation process to capture only the object-generated MR; (2) to avoid the performance degradation of the existing algorithms [12,22] at high bit rates, a new technique is proposed for embedding the pattern mode into the H.264 using the quality and bit adjustment for the pattern mode; (3) to avoid the quantization errors among adjacent McFISs, a new McFIS generation technique is proposed based on the theoretical relationship between quantization and distortion; and (4) to avoid the frame delay of the existing algorithm [22], a new pattern generation technique is proposed using decoded frames which improves the rate-distortion performance by saving pattern-shaped codes as the same pattern generation technique is used in the encoder and decoder.

5 Conclusions
In this paper, we have proposed a new pattern-based video coding idea using (1) indexing patterns (of both regular and arbitrary shapes) for motion estimation and compensation and (2) a dynamically updated background frame (i.e., McFIS) as the long-term reference frame to overcome the inaccurate motion estimation and compensation problem in the uncovered background areas by a new pattern matching and referencing technique. We have also devised a scheme for generating content-dependent arbitrary-shaped pattern templates where the McFIS is also used as the second reference frame. The extensive experimental results gave insight of the proposed idea and showed that the proposed techniques outperform the four most relevant existing algorithms by improving 0.3 ~ 1.5 dB in coded image quality.

Author details
[1]Centre for Research in Complex Systems (CRiCS), School of Computing and Mathematics, Charles Sturt University, Panorama Avenue, Bathurst NSW-2795, Australia. [2]School of Computer Engineering, Nanyang Technological University, Singapore, Singapore.

References
1. Joint Video Team (JVT) of ISO MPEG, ITU-T VCEG, and JVT-G050, *Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H264/ISO/IEC 14496–10 AVC)* (Joint Video Team (JVT) of ISO MPEG, ITU-T VCEG, and JVT-G050 8th Meeting, Geneva, 2003)
2. T Wiegand, GJ Sullivan, G Bjøntegaard, A Luthra, Overview of the H.264/AVC video coding standard. IEEE Trans Circ Sys Video Technol **13**(7), 560–576 (2003)
3. G Sullivan, T Wiegand, Rate-distortion optimization for video compression. IEEE Signal Process Mag **15**(6), 74–90 (1998)
4. T Wiegand, B Girod, Lagrange multiplier selection in hybrid video coder control. IEEE Int Conf Image Process **1**, 542–545 (2001)
5. T Weigand, H Schwarz, A Joch, F Kossentini, Rate-constrained coder control and comparison of video coding standards. IEEE Trans Circ Syst Video Technol **13**(7), 688–702 (2003)
6. OD Escoda, P Yin, C Dai, X Li, Geometry-adaptive block partitioning for video coding. IEEE Int Conf Acoust Speech Signal Process (ICASSP-07) **1**, 657–660 (2007)
7. OD Escoda, P Yin, C Gomila, *Hierarchical B-frame results on geometry-adaptive block partitioning* (Paper presented at the VCEG-AH16 Proposal, ITU/SG16/Q6/VCEG, Antalya, Turkey, 2008)
8. J Chen, S Lee, KH Lee, WJ Han, *Object boundary based motion partition for video coding* (Paper presented at the Picture Coding Symposium, Lisbon, Portugal, 2007)
9. JH Kim, A Ortega, P Yin, P Pandit, C Gomila, *Motion compensation based on implicit block segmentation* (IEEE International Conference on Image Processing (ICIP-08), San Diego, 2008), pp. 2452–2455. 12–15 Oct 2008. (IEEE, Piscataway, 2008)
10. S Chen, Q Sun, X Wu, L Yu, *L-shaped segmentations in motion-compensated prediction of H.264* (IEEE International Conference on Circuits and Systems (ISCAS-08), Seattle, 2008), pp. 1620–1623. 18–21 May 2008. (IEEE, Piscataway, 2008)
11. T Fukuhara, K Asai, T Murakami, Very low bit-rate video coding with block partitioning and adaptive selection of two time-differential frame memories. IEEE Trans Circ Syst Video Technol **7**, 212–220 (1997)
12. M Paul, M Murshed, Video coding focusing on block partitioning and occlusions. IEEE Trans Image Process **19**(3), 691–701 (2010)
13. K-W Wong, K-M Lam, W-C Siu, An efficient low bit-rate video-coding algorithm focusing on moving regions. IEEE Trans Circ Syst Video Technol **11**(10), 1128–1134 (2001)
14. C Stauffer, WEL Grimson, Adaptive background mixture models for real-time tracking. IEEE Conf CVPR **2**, 246–252 (1999)
15. D-S Lee, Effective Gaussian mixture learning for video background subtraction. IEEE Trans Pattern Anal Mach Intell **27**(5), 827–832 (2005)
16. M Haque, M Murshed, M Paul, *On stable dynamic background generation technique using Gaussian mixture models for robust object detection* (5th IEEE International Conference on Advanced Video Signal Based Surveillance, Santa Fe, 2008), pp. 41–48. 1–3 Sept 2008. (IEEE, Piscataway, 2008)
17. A Krutz, A Glantz, T Sikora, *Background modelling for video coding: from sprites to global motion temporal filtering* (IEEE International Symposium on Circuits and Systems, Paris, 2010), pp. 2179–2182. 30 May–2 June 2010. (IEEE, Piscataway, 2010)
18. P Wilkins, J Bankoski, Y Xu, *System and method for video encoding using constructed reference frame* (Patent 20100061461 A1, U.S, 2010). 11 Mar 2010
19. R Ding, Q Dai, W Xu, D Zhu, H Yin, Background-frame based motion compensation for video compression. IEEE Int Conf Multimed Expo (ICME) **2**, 1487–1490 (2004)
20. D Hepper, Efficiency analysis and application of uncovered background prediction in a low bit rate image coder. IEEE Trans Commun **38**(9), 1578–1584 (1990)
21. M Paul, W Lin, CT Lau, BS Lee, *Video coding using the most common frame in scene* (IEEE International Conference on Acoustic Speech Signal Process, Dallas, 2010), pp. 734–737. 14–19 Mar 2010. (IEEE, Piscataway, 2010)
22. M Paul, M Murshed, An optimal content-based pattern generation algorithm. IEEE Signal Process Letters **14**(12), 904–907 (2007)
23. M Paul, W Lin, CT Lau, BS Lee, *Pattern based video coding with uncovered background* (IEEE International Conference on Image Processing, Hong Kong, 2010), pp. 2065–2068. 26–29 Sept 2010 (IEEE, Piscataway, 2010)
24. V Chellappa, PC Cosman, GM Voelker, Dual frame motion compensation with uneven quality assignment. IEEE Trans Circ Syst Video Technol **18**(2), 249–256 (2008)
25. M Tiwari, PC Cosman, Selection of long-term reference frames in dual-frame video coding using simulated annealing. IEEE Signal Process Letters **15**, 249–252 (2008)
26. D Liu, D Zhao, X Ji, W Gao, Dual frame motion compensation with optimal long term reference frame selection and bit allocation. IEEE Trans Circ Syst Video Technol **20**(3), 325–339 (2010)
27. YW Huang, BY Hsieh, SY Chien, SY Ma, LG Chen, Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC. IEEE Trans Circ Syst Video Tech **16**(4), 507–522 (2006)
28. L Shen, Z Liu, Z Zhang, G Wang, An adaptive and fast multi frame selection algorithm for H.264 video coding. IEEE Signal Process Letters **14**(11), 836–839 (2007)
29. TY Kuo, HJ Lu, Efficient reference frame selector for H.264. IEEE Trans Circ Syst Video Technol **18**(3), 400–405 (2008)
30. K Hachicha, D Faura, O Romain, P Garda, Accelerating the multiple reference frames compensation in the H.264 video coder. J Real Time Image Process **4**(1), 55–65 (2009)
31. P List, A Joch, J Lainema, G Bjøntegaard, M Karczewicz, Adaptive deblocking filter. IEEE Trans Circ Syst Video Technol **13**(7), 614–619 (2003)
32. H Kimata, Y Yashima, N Kobuyashi, Edge preserving pre-post filtering for low bitrate video coding. IEEE Int Conf Image Process **3**, 554–557 (2001)
33. M Paul, W Lin, CT Lau, BS Lee, Video coding with dynamic background. EURASIP J Adv Signal Process (2013). doi:10.1186/1687-6180-2013-11
34. HM Hang, JJ Chen, Source model for transform video coder and its application—part I: fundamental theory. IEEE Trans Circ Syst Video Technol **7**(2), 287–298 (1997)
35. JR Ding, JF Yang, Adaptive group-of-pictures and scene change detection methods based on existing H.264 advanced video coding information. IET Image Process **2**(2), 85–94 (2008)
36. M Paul, W Lin, CT Lau, BS Lee, Explore and model better I-frame for video coding. IEEE Trans Circ Syst Video Technol **21**(9), 1242–1254 (2011)
37. JC Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J Cybern **3**, 32–57 (1973)
38. JC Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms* (Plenum, New York, 1981)
39. JB MacQueen, *Some methods for classification and analysis of multivariate observations, Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1 (University of California Press, Berkeley, 1967), pp. 281–297