

Efficient Pattern Index coding Using Syndrome Coding and Side Information

*¹Manoranjan Paul and ²Manzur Murshed

^{*1}*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.*

²*Gippsland School of Information Technology, Monash University, Churchill, Vic 3842, Australia (phone: +61-3-9902-6467; fax: +61-3-9902-7137; e-mail: Manzur.Murshed@infotech.monash.edu.au*

Abstract

Pattern-based video coding focusing on moving regions has already established its superiority over the H.264 at very low bit rate. Up to a certain limit, the larger the number of pattern templates, the better the approximation to the moving regions. However, beyond that limit no coding gain is observed due to the need of an excessive number of pattern identification bits. Recently, distributed video coding schemes have used syndrome coding to predict the original information in the decoder using side information. In this paper a pattern identification scheme is proposed which predicts the pattern from the syndrome codes and side information in the decoder so that the actual pattern identification code is not needed. The experimental results confirm that the new scheme improves the rate-distortion performance compared to the existing pattern-based video coding and compared with the H.264 standard. The proposed new scheme will also present opportunities for further syndrome coding application.

Keywords: Video Coding, Pattern Recognition, Side Information, Distributed Video Coding, H.264, Low Bit Rate.

1. Introduction

Applications such as video conferences, video calls, and Internet video streaming through limited bandwidth channels require *very low bit rate* (VLBR) video coding techniques. The recent block-based video coding standard H.264/AVC [1] has introduced *tree-structured variable block sizes* (TVBS) (i.e., 16×16, 16×8, 8×16, 8×8, 8×4, 4×8, and 4×4 pixels) *motion estimation* (ME) and *motion compensation* (MC) strategies to capture various motions within a 16×16 pixels *macroblock* (MB). Motion vector and residual error are needed to be encoded for each block or sub-block together with the type of partitions. In this approach, the large block (e.g., 16×16 pixels) requires less number of bits for motion vectors and type of partitions but more bits for residual errors due to the coarse motion approximation to encode a 16×16-pixel block [2]. On the other hand, to encode the same size of block using a small block (e.g., 4×4) requires more bits for motion vectors (i.e., each sub-block is associated with motion vector) and type of partitions but less bits for residual error due to the fine motion approximation. An optimization cost function is used (i.e., Lagrangian multiplier [3][4]) to select the final block partitioning among them for a MB. This approach provides a significant improvement of the rate-distortion performance of the H.264 against the other video coding standard such as H.263, MPEG-4 etc.

However, these strategies failed to exploit intra-MB temporal redundancy (ITR) (see **Figure 1**) which is static in successive frames. There is no mechanism in the TVBS to encode “partial skip” MB to exploit the ITR. Indeed, few bits are used to signal zero residual error and zero motion vector for ITR to decoder, which is significant when a sequence is encoded at VLBR. Moreover, ME and MC using the rectangular or square shaped block partitioning using the TBVS sometimes could not capture motion within the MBs properly due to the poor *moving region* (MR) approximation as the real-world

moving region can have non-rectangular and non square shape. We also empirically observed in [5] that while coding head-and-shoulder type video sequences at low bit-rate, more than 70% MBs were never partitioned by the H.264 that would otherwise be at very high bit-rate. It can be easily observed that the possibility of choosing smaller block sizes diminishes as the target bit-rate is lowered. Consequently, coding efficiency improvement due to TVBS can no longer be realized for a low bit-rate target as larger blocks have to be chosen in most cases to keep the bit-rate in check but at the expense of inferior shape approximation.

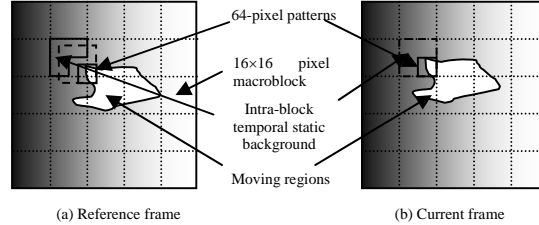


Figure 1: An example on how the PVC scheme exploits the intra-MB temporal redundancy (ITR) to improve coding efficiency [4].

Recently, many researchers [6]-[10] successfully introduced other forms of block partitioning to approximate the shape of an MR even more closely to improve the compression efficiency. But none of these techniques, including the H.264 standard, allows for encoding a block-partitioned segment by skipping ME and MC. Consequently they use unnecessary bits to encode almost zero-length motion vector with perceptually insignificant residual errors for the background segment. These bits are quite valuable at low bit-rate that could otherwise be spent wisely for encoding residual errors in perceptually significant segments. Moreover, the required computational complexity is too high for some real time applications using low powered devices.

Pattern-based video coding (PVC) [11]-[14] has been used to exploit the ITR using binary patterns (see **Figure 1**) in partitioning the *suitable* MB via a simplified segmentation process that has avoided handling the exact shape of the moving objects, so that popular MB-based ME&MC techniques could be applied. ME and MC are carried out using only pattern covered region. The PVC scheme focuses on the MRs of the MBs, through the use of a set of regular 64-pixel pattern templates (see **Figure 2**). The pattern templates were designed using ‘1’s in 64 pixel positions and ‘0’s in the remaining 192 pixel positions in a 16×16-pixel MB. The MR of a MB is defined as a region comprising a collection of pixel positions where pixel intensity differs from its reference MB. Using some similarity measures, if the MR of a MB is found well covered by a particular pattern, 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 the larger segment with high ITR is coded with no motion vector and residual error [4].

The MR of an MB has different shapes [14] as it depends on the part of object contained in that MB. Clearly, a large number of pattern templates approximate the MR more accurately than a smaller number but, after a certain limit, no coding gain is observed due to the increased number of *pattern identification codes* (PIC). Variable length PICs using pattern frequency information and a pattern co-occurrence matrix [13] used 4.62 and 4.15 bits per pattern representing a RMB instead of 5 bits for 32 patterns respectively. Hence, any strategy that reduces the PIC size concomitantly improves the overall encoding performance by classifying more RMBs.

Computational complexity of the H.264 coding standard is a major bottleneck for some applications using low powered battery operated electronic devices and limited processing powered devices [2][3][15][16]. To overcome the problem, very recently, computational complexity distribution of the existing video coding standards modeled with a heavy encoder and light decoder is challenged by a stark contrast *distributed video coding* (DVC) architecture [17]-[23] which is modeled with a light encoder and a heavy decoder. The DVC exploits the joint source-channel coding theorems developed

by Slepian and Wolf [22] and Wyner and Ziv [23]. The main idea is to divide the source data into a finite number of sets, popularly known as *cosets*, and the original data can be retrieved in the decoder from the coset index and predicted data by applying the nearest neighborhood technique. The predicted data are generated from the side information, for example, previously decoded data. If the maximum distance between actual and predicted data is E_{max} and the minimum distance of each coset at level i is $d_{min} = 2^{i+1}$, then the coset theorem states that the actual data will be retrieved without error if $E_{max} < d_{min} / 2$ is satisfied. In this case we need to divide the data set into $i+1$ cosets. Compression of the scheme stems from transmitting only the coset index (syndrome) instead of the actual value.

Although the compression ratio of the DVC can theoretically be as efficient as that of the conventional heavy encoders, this upper bound can only be achieved with the accurate estimation of inter-frame correlation structures. In real videos however the correlation structures are highly variant spatially, potentially unbounded in magnitude, and imprecisely known, leading to significant imperfection. Moreover, being a block-based codec, this new technique is obviously unable to exploit the intra-block temporal redundancy as alluded for the existing standards. So far no distributed video coding schemes are comparable to the conventional coding in terms of rate-distortion performance.

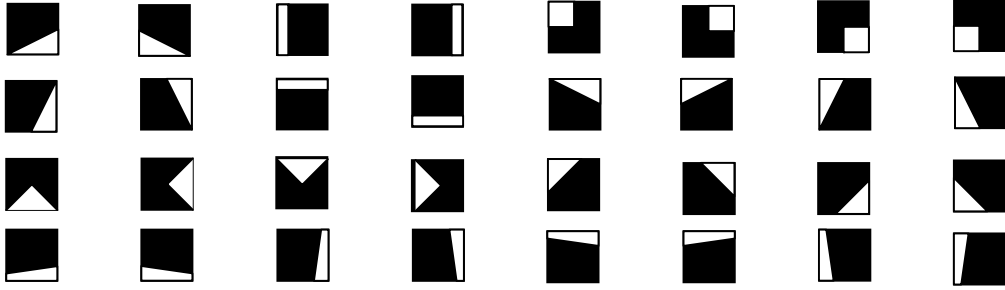


Figure 2: The pattern codebook of 32, 64-pixel patterns (numbering from left to right and top to bottom), defined in 16×16 blocks, where the white (moving) region represents 1 and black (static) region represents [4].

In this paper, we describe the use of a distributed coding concept to encode PICs of 32 patterns. If we applied this concept straightforward to encode 32 patterns, we would not achieve any compression as the maximum prediction error is 16 for 32 patterns which require 32 cosets. Obviously smaller number of cosets will provide compression for those RMBs where accurate pattern retrieval is possible. Figure 2 shows all 32 patterns. Let $G(A)$ denote the gravitational centre of the 16×16 binary matrix A , such that

$$G(A) = \left(\frac{1}{256} \sum_{x=0}^{15} \sum_{y=0}^{15} x \cdot A(x, y), \frac{1}{256} \sum_{x=0}^{15} \sum_{y=0}^{15} y \cdot A(x, y) \right) \quad (1)$$

It is interesting to observe that the *gravitational centres* (GCs) of the MR of all patterns are almost uniformly distributed in 360° with respect to the centre. We will exploit the angular distance for coset formation because the dissimilarity between two patterns is directly proportional to the angular difference between them. We will use side information from the previously decoded frames as we experimentally observed that there is a strong correlation between two successive co-located MBs in terms of pattern similarity. As the same predicted system can be designed in the both encoder and decoder, we will not lose any information if we discard those RMBs where pattern could not be retrieved accurately.

In the paper a novel pattern index scheme is proposed which successfully exploited distributed video coding concepts for retrieving the original pattern in a decoder without knowledge of its index. Using this approach a significant reduction of the PIC size is achieved ($5 - \log_2(\text{number of cosets})$) per

RMB and as a result the rate-distortion performance is improved compared to the existing pattern-based coding algorithm, as well as the H.264 video coding standard. The proposed system is a conventional coding system and very similar to current PVC technique with the sole exception that it is applied with distributed coding concepts to retrieve the pattern index at the decoder. The preliminary version of this paper has been published in [25].

This paper is organized as follows: Section 2 describes coset formation using the angular distances. Section 3 describes the side information generation. Section 4 illustrates pattern retrieval. Section 5 analyses simulations set up and experimental results and Section 6 offers conclusions.

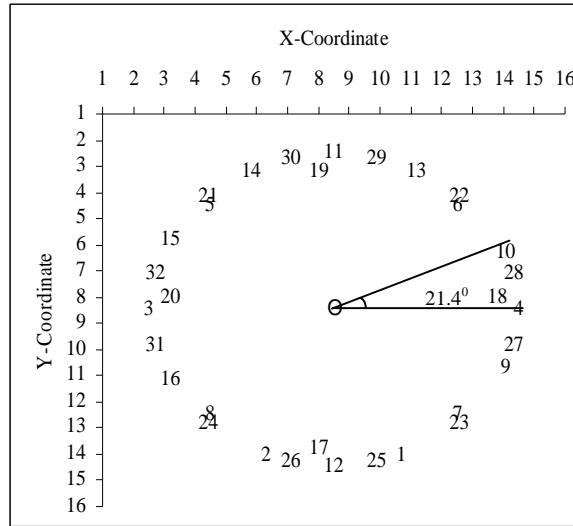


Figure 3: Positions of gravitational centers of patterns in 16×16 block and the numbers indicate the patterns.

2. Coset Formation with 32 Patterns

The main target in coset formation is to maximize the distance among the patterns within a coset and minimize the distance among the cosets. A small number of cosets provide better compression as they require less number of bits (syndrome codes) to encode a coset index; however the probability of getting exact patterns will be decreased because the distance among the patterns in a coset is less. On the other hand, if the number of cosets is large, the syndrome code size will be large but the probability of getting exact pattern will be high. There are four ways we can form cosets using 32 patterns such as 2, 4, 8, and 16 cosets containing 16, 8, 4, and 2 patterns in each coset respectively. By considering all 64 '1's in a pattern will make the coset formation better. But this strategy is an intractable problem as we need to consider all $^{256}C_{64}$ combinations. Thus, a coset formation technique needs to be converted into a simple problem where the gravitational center of a pattern will represent a corresponding whole pattern. By doing this we could not utilize all available information in coset formation, but it simplifies our coset formation strategy.

Figure 3 shows the positions of GCs of 32 patterns. The GCs of patterns (considering only white moving region) are almost symmetrical around 360^0 (see Figure 3). Patterns could be ranked based on the angular distance of their GCs from the positive X-axis where the middle point of MB is considered the origin. For example, patterns 4, 18, 28, 10, and 6 are ranked as 1, 2, 3, 4, and 5. In order to form $n=2^k$ cosets of 32/n patterns, every n th ranked pattern will be grouped into one coset where k is an integer. The various sizes of cosets and their minimum distances (d_{min}) using angular distance are shown in Table 1. As we mention earlier, if the predicted pattern is within $d_{min}/2$ distance of the correct pattern then a syndrome coding technique will find the actual pattern.

Table 1: Various size of cosets using angular distance and corresponding minimum distance among the cosets

Cosets	Patterns	Distance
2	[19,14,5,32,3,16,24,26,12,1,7,27,18,10,22,29] [30,21,15,20,31,8,2,17,25,23,9,4,28,6,13,11]	13 ⁰
4	[19,5,3,24,12,7,18,22] [30,14,31,2, 25,9,28,13] [14,32,16,26,1,27,10,29] [21,20,8,17,23,4,6,11]	37 ⁰
8	[19,3,12,18] [30,31,25,28] [14,16,1,10] [21,8, 23, 6] [5,24,7,22] [15,2,9,13] [32,26,27, 29] [20, 17,4,11]	85 ⁰
16	[19,12] [30,25] [14,1] [21,23] [5,7] [15,9] [32, 27] [20,4] [3,18] [31,28] [16,10] [8,6] [24,22] [2,13] [26,29] [17,11]	175 ⁰

Table 1 shows the minimum angular distances 13⁰, 37⁰, 85⁰, and 175⁰ using 2, 4, 8, and 16 cosets respectively. Theoretically the distances would be 22.5⁰, 45⁰, 90⁰, and 180⁰, but already designed patterns are not fully symmetrical around the circle.

3. Side Information Generation

If $C_k(x,y)$ and $R_k(x,y)$ denote the k^{th} MB of the current and reference frames respectively of a video sequence, where $0 \leq x, y \leq 15$, the moving region $M_k(x, y)$ of the k^{th} MB in the current frame is obtained by:

$$M_k(x, y) = T(|C_k(x, y) \bullet B - R_k(x, y) \bullet B|) \quad (2)$$

where B is a 3×3 unit matrix for the morphological closing operation \bullet [24], which is applied to reduce noise, and the thresholding function $T(v) = 1$ if $v > 2$, and is 0 otherwise. If we generate a binary matrix for the whole current frame with respect to the reference frame, we will get the moving region for the entire current frame. **Figure 4** shows two moving regions between frame 1 & 3 in (a) and frame 3 & 5 in (b). If we carefully observe these two moving regions, we will see the similarity between them in terms of shape and size in MB level. From this observation we argue that side information can be generated from previously decoder frames. Note that we consider alternative frames as we use 15 frames per second at rate-distortion calculation. Predictably, more similarity will be observed between two adjacent frames.



Figure 4: The moving region between frame 1 and 3 (a) and frame 3 and 5 (b) for Miss America video sequence using (2).

This co-relation can be quantified using pattern similarity. Let $|Q|_\ell$ be the total number of ℓ 's in the matrix Q . The dissimilarity of a pattern $P_n \in PC$ with the MR in the k^{th} MB can be defined as:

$$D_{k,n} = |M_k|_1 - |M_k \wedge P_n|_1. \quad (3)$$

Clearly, the higher the similarity lower will be the value of $D_{k,n}$. The best similar pattern P_i will be selected for the MR M_k as follows:

$$P_i = \arg \min_{\forall P_n \in PC} (D_{k,n}). \quad (4)$$

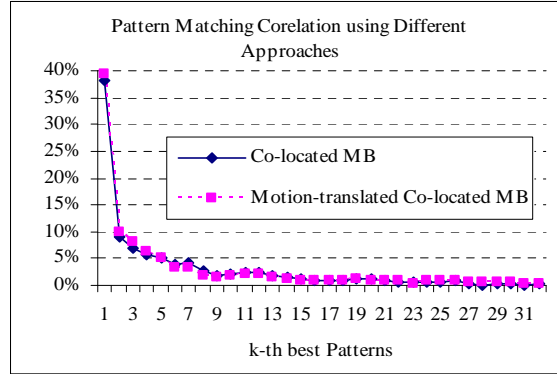


Figure 5: Pattern prediction by k -th best pattern of current MB using co-located MB and motion-translated co-located MB respectively.

To find the relationship between the best pattern of current MB and the pattern for co-located MB in the previously decoded frame, we investigated the rank (in terms of similarity) of the current best pattern for the co-located MBs. **Figure 5** shows the normalized ranking. It revealed that in 40% of cases the same pattern was selected as the best pattern for both the current and co-located MBs. Instead of co-located MB, we also investigated the MB which is actually referenced using the motion vector of the current MB (known as motion-translated co-located MB). The experimental results confirmed that the later approach provides slightly better results due to the motion correlation. For zero motion vectors, the result is the same in both cases. The figure also shows that for more than 60% cases we can find the current best pattern within the first-fourth best patterns for the motion-translated co-located MB. Thus, we can successfully predict the actual pattern for current MB more frequently by comparing this kind of side information and syndrome coding (i.e., coset index). For conventional video coding, the decoder is embedded within the encoder. Thus, when we are coding an MB by pattern mode, we know whether the prediction is successful. If it is not successful we may not consider the MB as a RMB and encode it using other H.264 modes. The final mode, including pattern mode, is selected by the Lagrangian optimization function.

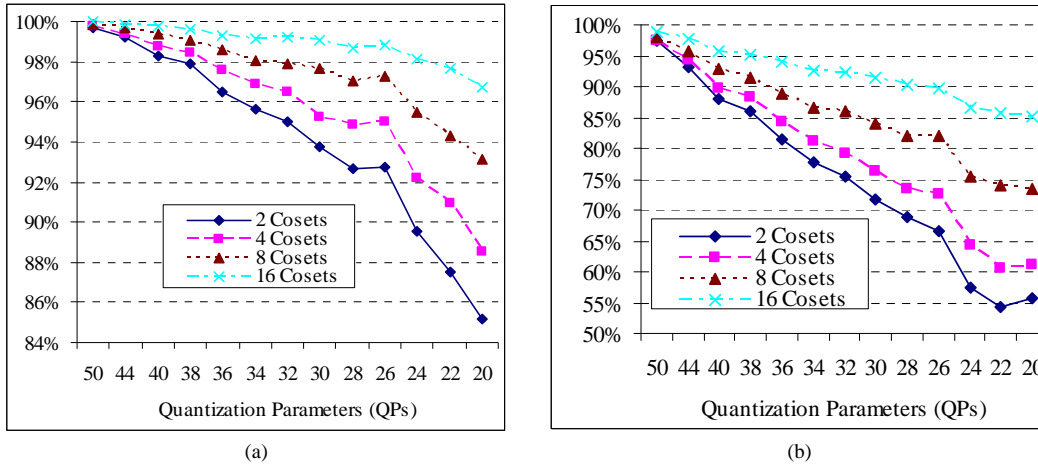


Figure 6: Percentage of accurate pattern prediction by first four best patterns (a) and by the best pattern (b) using different cosets.

4. Pattern Index Retrieval

Sometimes after motion estimation the residual error is zero or is less significant due to the smooth translational motion and coarse quantization. Moreover, a zero motion vector is observed for a significant number of cases in typical video sequences. These two features can be exploited in our pattern prediction scheme to increase the pattern retrieval accuracy. In the existing PVC scheme, only the pattern covered MR is considered for the ME&MC. The rest of the region of an MB is copied from the co-located/motion-translated co-located MB as a static region from the decoded frame. When a pattern could not accurately approximate the MR of an MB, the pattern mode does not provide the best rate-distortion performance due to the pattern mismatch error. Thus, our proposal is to consider the whole MB instead of pattern covered MR for ME&MC. By doing this, we not only eliminate pattern miss-approximation error but reduce the extra ME computational cost as we already have the motion vector for the entire MB from the 16×16 mode. Thus, after quantization when residual error is zero, retrieving a wrong pattern through our prediction scheme could not affect in rate-distortion performance.

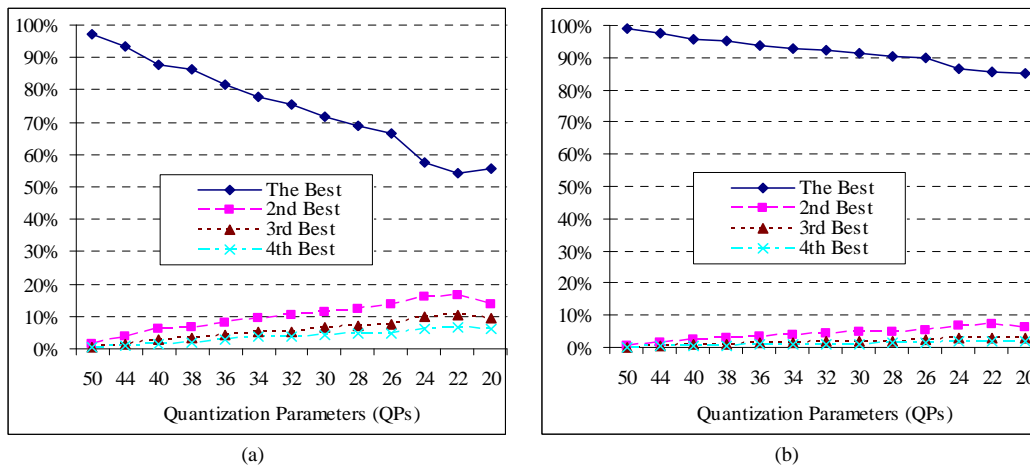


Figure 7: Pattern prediction by the first four best patterns for different number of cosets with different QPs, (a) 2 cosets and (b) 16 cosets.

In our proposed scheme, we determine the best pattern for the current MB by (3) and (4). Then we find the coset index of this pattern. We also find the best pattern (i.e., side information) from the motion-translated co-located MB in the decoded frame. From the coset index and side information, we apply a nearest neighborhood technique to find the predicted pattern. If the predicted pattern and the actual pattern (i.e., the best pattern for the current MB) are the same pattern, we will encode this MB by the best pattern. Otherwise we will try with second/third/fourth best pattern at the same way. If none are successful, we will encode this MB by other H.264 modes instead of pattern mode.

5. Simulation Results

The existing PVC, proposed scheme (termed PVCcosetX {2, 4, 8, 16}), and the H.264 algorithms are implemented based on the *Baseline profile* of H.264/AVC [24] with full search fractional pel motion estimation for the number of standard QCIF [24] video sequences at 15 frames per second (fps) with 16 group of picture (GOP) size. The experimental results reveal that 85~100% correct decisions can be achieved up to 20 *quantization parameter* (QP) [24] (see **Figure 6** and **Figure 7**) using all four best patterns. The prediction is improving with the number of cosets (see **Figure 6**) because the probability of getting actual patterns increases due to the less number of patterns in a coset. **Figure 6** (a) shows that around 97% accurate patterns can be achieved using 16 cosets whereas 85% accurate patterns can be achieved using 2 cosets at QP = 20 (see). **Figure 6** (b) also shows that only the best pattern can predict from 55% (2 cosets) to 85% (16 cosets) patterns accurately when QP = 20.

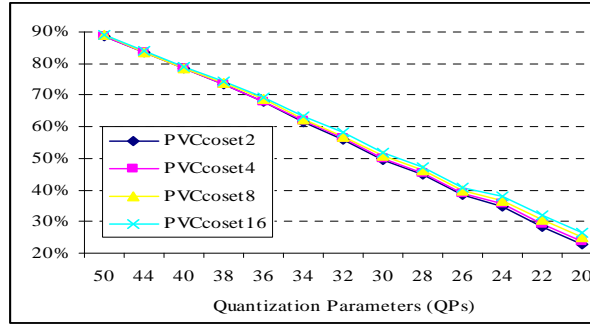


Figure 8: Average number of RMBs with different QPs using eight standard video sequences.

Figure 7 shows relative contributions in pattern retrieval using first four best patterns for different cosets. Undoubtedly, the best pattern can retrieve the actual pattern in the most cases. The rest of the patterns (other three best) have relatively more contribution when the number of cosets is small. For example, when the number of cosets is only two, the rest of the patterns have contributed around 45% of cases whereas when the number of cosets is 16, the rest of the patterns have contributed 15% of cases.

Table 2: Coding performance using PVCcoset, PVC, and H.264 algorithms

Video	PSNR (dB)	Bit Rate (kbps)				PVC	H.264
		PVCcoset					
		2	4	8	16		
Miss America	36	14.5	<u>15.5</u>	16.5	17.0	19	33
	40	26.5	<u>27.5</u>	27.7	28.5	31	43
	43	55.5	<u>56.2</u>	56.5	57.2	59	67
	44	69.8	<u>70.2</u>	70.6	71.2	73	79
	45	90.5	91.5	91.0	<u>90.8</u>	94.3	99
Foreman	27	30.5	<u>32.0</u>	33.0	34.0	39	58
	30	44.0	<u>45.0</u>	46.0	47.0	51	67

	32	64.0	<u>65.0</u>	66.0	67.0	71	84
	34	80.5	<u>81.0</u>	81.2	82.0	86	98
	35	124.5	<u>124.6</u>	124.8	124.8	128	138
Claire	35	15.2	<u>16.2</u>	17.2	18.2	21	35
	39	23.8	<u>24.5</u>	25.0	25.5	29	40
	41	37.0	<u>37.5</u>	37.8	38.0	42	51
	43	54.5	<u>54.5</u>	54.8	55.0	59	67
	44	66.0	<u>66.2</u>	66.6	66.6	70	76
Salesman	31	22.0	23.0	24.0	26.0	29	47
	35	36.5	37.5	38.5	39.5	42	61
	37	53.4	53.5	54.5	55.0	58	75
	39	77.0	78.0	78.0	78.5	81	100
	40	90.5	<u>91.5</u>	91.8	91.8	94	115

Figure 8 confirms that the number of RBMs is decreasing with bit rates. This rate is slightly less when the number of cosets is higher. Thus, rate-distortion gain using the proposed and the PVC schemes have diminishing return with the bit rates compared to the H.264.

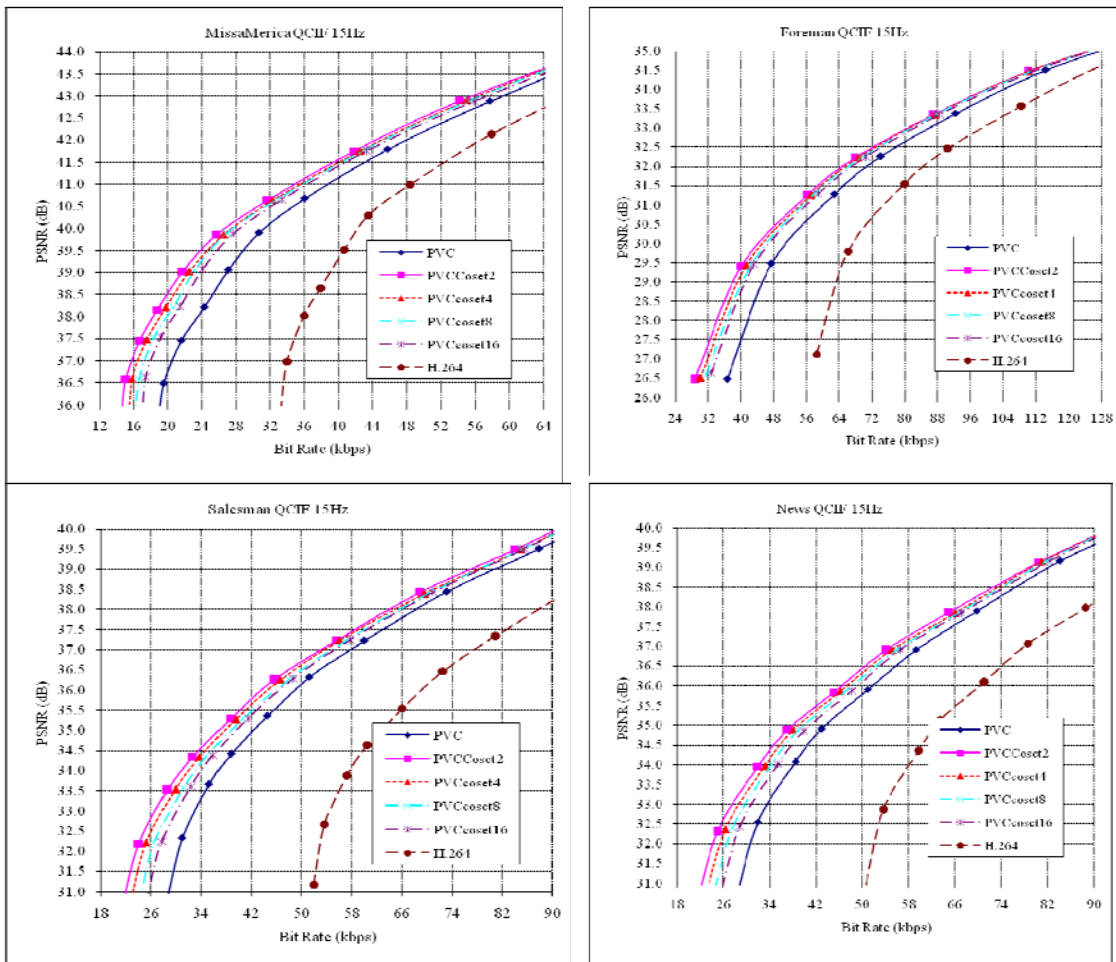


Figure 9: Rate-distortion performance using the H.264, existing PVC, and proposed PVCcosetX {2, 4, 8, 16} for four video sequences.

The new PVCcosetX scheme reduces the bitstream by 26% to 29% and 7% to 12% on average (see Table 2) compared to the H.264 and the existing PVC which is significant especially at very low bit rate. Note that the PVC using more cosets performs better with bit rates as it accommodates more RMBs but the PVC with 4 cosets will provide reasonable performance for all bit rates. The PVC with two cosets will outperform in all cases if we can use better side information. Note that the bold and underline bit rates are the lowest and second lowest bit rates for a given quality. Rate-distortion performance is also shown for four different video sequences using different cosets in **Figure 9**. The figure confirms that more than 0.2 and 1.5 dB PSNR is improved at 64 kbps using two cosets compared to the existing PVC and the H.264 algorithms respectively.

Undoubtedly, using four patterns instead of one will increase the computational time in ME&MC, however, this can be reduced using already available motion vector generated by the 16×16 mode. From the experimental results we also observed that motion estimation using whole block instead of pattern covered moving region provides better results where the mismatch area between the pattern and the moving region of an MB is relatively large. Thus, proposed scheme is comparable with the existing PVC in terms of computational complexity. The computational complexity against the H.264 is investigated and plotted in **Figure 10**. The figure shows that at relatively high QPs (50...32) i.e., at extreme low bit rate, the proposed algorithm took around 3.5% more computational time compared to the H.264. At these bit rate areas, the H.264 mostly used large size block modes for typical video sequences, whereas the proposed scheme used extra pattern mode together with other large block modes. At the relatively moderate QPs (32...24), the H.264 started to encode some blocks with small block size modes, whereas the proposed algorithm still encodes some of those blocks using pattern modes, thus, the proposed scheme used only 1% more computational time. But again the computational time remained at around 3% more at the relatively low QPs(<24), because the both schemes used almost all modes (including pattern modes in the proposed scheme). Thus, we can conclude that the proposed scheme does not require more than 4% computational time compared to the H.264 at any target bit rates.

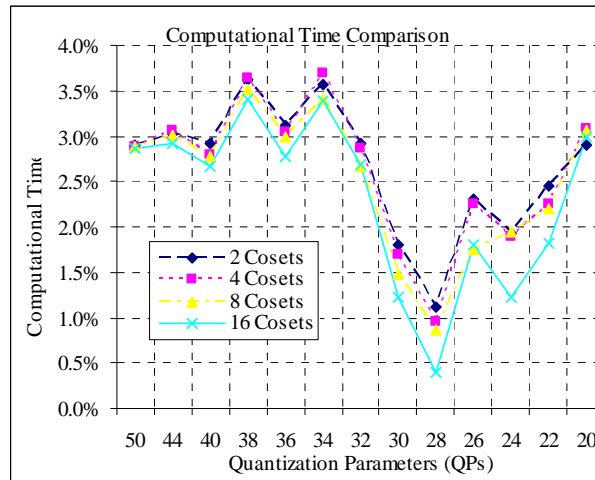


Figure 10: Computational time comparison of the proposed algorithm against the H.264 coding standard.

6. Conclusions

In this paper, we have proposed an efficient index coding technique using distributed video coding concept so that less number of bits is required to encode an index. The main contributions of the research work are:

- (i) We have used two distributed video coding concepts such as syndrome coding (i.e., index of Coset) and side information (i.e., predicted pattern) to predict the pattern index at the decoder so

that we don't need to include actual pattern index codes to transmit. Thus, more compressed video stream can be generated;

- (ii) For the first time we have formed Coset using the angular distance;
- (iii) We have also used angular distance to predict the original data;
- (iv) We have proposed multiple pattern modes to optimize the prediction and rate-distortion performance.

In this approach, the pattern-based video coding can accommodate a greater number of patterns for better approximation of moving region within a macroblock without increasing the index code size. As a result, the rate-distortion performance is improved compared with both the existing methods, and the H.264 video coding standard. The experimental results reveal that 26% to 29% and 7% to 12% bitstream reduction can be achieved on average compared to the H.264 and the existing PVC. This new scheme promises to promote further developments in the field of syndrome coding applications.

7. References

- [1] ITU-T Rec. H.264/ISO/IEC 14496-10 AVC. Joint Video Team (JVT) of ISO MPEG and ITU-T VCEG, JVT-G050, 2003.
- [2] 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.
- [3] M. Paul, M. Frater, and J. Arnold, "An efficient Mode Selection Prior to the Actual Encoding for H.264/AVC Encoder," ,” *IEEE Transactions on Multimedia*, vol. 11, no. 4, pp. 581-588, 2009.
- [4] M. Paul and M. Murshed, "Video Coding focusing on block partitioning and occlusions," *IEEE Transaction on Image Processing*, vol. 19, no. 3, pp. 691-701, March, 2010.
- [5] M. Paul and M. M. Murshed, "Superior VLBR video coding using pattern templates for moving objects instead of variable-bloc size in H.264," *7th IEEE International Conference on Signal Processing (ICSP-04)*, Beijing, China, 717–720, 2004.
- [6] E. M. Hung, L. Ricardo, D. Queiroz, and D. Mukherjee, "On macroblock partition for motion compensation," *IEEE International Conference on Image Processing (ICIP-06)*, pp. 1697–1700, 2006.
- [7] A. A. Muhit, M. R. Pickering, M. R. Frater, and J. F. Arnold, "Video coding using fast geometry-adaptive partitioning and an elastic motion model," *J. Visual Communication and Image Representation* 23(1): 31-41, 2012.
- [8] J. Chen, S. Lee, K.-H. Lee and W.-J. Han, "Object boundary based motion partition for video coding," *Picture Coding Symposium*, 2007.
- [9] J. H. Kim, A. Ortega, P. Yin, P. Pandit, and C. Gomila, "Motion compensation based on implicit block segmentation ,” *IEEE International Conference on Image Processing*, (ICIP-08), 2008.
- [10] S. Chen, Q. Sun, X. Wu, and L. Yu, "L-shaped segmentations in motion-compensated prediction of H.264," *IEEE International Conference on Circuits and Systems*, (ISCAS-08), 2008.
- [11] T. Fukuhara, K. Asai, and T. Murakami, "Very low bit-rate video coding with block partitioning and adaptive selection of two time-differential frame memories," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 7, 212–220, 1997.
- [12] K.-W. Wong, K. -M. Lam, and W. -C. Siu, "An efficient low bit-rate video-coding algorithm focusing on moving regions," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11(10), 1128–1134, 2001.
- [13] M. Paul, M. Murshed, and L. Dooley, "A real-time pattern selection algorithm for very low bit-rate video coding using relevance and similarity metrics," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 6, pp. 753–761, 2005.

- [14] M. Paul and M. Murshed, "Video Coding Using Arbitrarily Shaped Block Partitions in Globally Optimal Perspective," *EURASIP Journal on Advances in Signal Processing*, doi:10.1186/1687-6180-2011-16, 2011.
- [15] Z. Ping and P. Tian, "A Global to Local Motion Estimation for H.264/AVC Video Coding", *International Journal of Advancements in Computing Technology*, Vol. 4, No. 1, pp. 516 ~ 524, 2012.
- [16] C. Duanmu and D. Zhou, "A Fast Inter Mode Selection Algorithm Based on the Statistical Features for the H.264", *Journal of Convergence Information Technology*, Vol. 7, No. 10, pp. 372 ~ 380, 2012.
- [17] A. Aron A. and B. Girod, "Towards practical wyner ziv coding of video," *Proc. of IEEE International Conference on Image Processing*, vol. 3, 869–872, 2003.
- [18] W. J. Chien, L. J. Karam, and G. P. Abousleman, "Distributed video coding with lossy side information," *Proc. of IEEE Int. Conference on Acoustics, Speech, and Signal Proc.*, vol. 2, 69–72, 2006.
- [19] B. Girod, A. Aaron, S. Rane, and D. Robello-Monedero, "Distributed video coding," *Proc. IEEE, Special Issue on Advances in Video Coding and Delivery*, 1–11, 2004.
- [20] F. Xiao, J. Liu, L. Sun, and R. Wang, "A New Side Information Estimate Based Distributed Video Coding Algorithm for Wireless Multimedia Sensor Networks", *International Journal of Advancements in Computing Technology*, Vol. 4, No. 1, pp. 320 ~ 326, 2012.
- [21] R. Puri, and K. Ramchandran, "PRISM: a video coding architecture based on distributed compression principles," *Proc. Allerton Conf. on Compression, Control and Computing*, 2002.
- [22] D. Slepian, and J. k. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. On Information Theory*, vol. 19, 471–480, 1973.
- [23] A. Wyner, and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. On Information Theory*, vol. 22, 1–10, 2003. Maragos, P., "Tutorial on advances in morphological image processing and analysis," *Opt. Eng.*, vol. 26(7), 623–632, 1987.
- [24] I. E. G. Richardson, "H.264 and MPEG-4 video compression," *Wiley Press*, 2003.
- [25] M. Paul and M. Murshed, "A novel pattern identification scheme using distributed video concept," *IEEE International conference on Acoustics, Speech, and Signal processing (IEEE ICASSP-09)*, pp. 729-732, 2009.