Optimal Arbitrary Shaped Pattern-based Video Coding

Manoranjan Paul[1] and Manzur Murshed[2]

Gippsland School of Information Technology, Monash University
Northways Road, Churchill, Victoria-3842, Australia
[1] Manoranjan.Paul@infotech.monash.edu.au
[2] Manzur.Murshed@infotech.monash.edu.au

Abstract— **Very low bit-rate video coding algorithms using content-based generated patterns to segment out moving regions at macroblock level have exhibited good potential for improved coding efficiency when embedded into the H.264 standard as extra mode. This content-based pattern generation (CPG) algorithm provides local optimal result as only one pattern can be optimally generated from a given set of moving regions. But, it failed to provide optimal results for multiple patterns from entire sets. Obviously, a global optimal solution for clustering the set and then generation of multiple patterns enhances the performance farther. But a global optimal solution is not achievable due to the non-polynomial nature of the clustering problem. In this paper, we proposed a near optimal content-based pattern generation (OCPG) algorithm which outperforms the existing approach. Coupling OCPG, generating a set of patterns after clustering the macroblocks into several disjoint sets, with direct pattern selection algorithm by allowing all the macroblocks in multiple pattern modes outperforms the existing pattern-based coding while both embedded into the H.264.**

I. Introduction

Applications such as video conferencing, video telephony, Internet streaming through the public switched telephone network or other bandwidth restricted networks require very low bit rate (VLBR) video coding. Existing standard H.264 [1] introduced seven rectangular variable blocks for motion estimation (ME) and motion compensation (MC) to cover entire range of bit rates by selecting the relatively large blocks at VLBR and small blocks at high bit rates video coding using tradeoff between bit rates and distortion through Lagrangian Multiplier (LM) [2]. A 16×16-pixel block popularly known as macroblock (MB) requires 1 to 16 motion vectors depend on the type of blocks selected for ME &MC. Fukuhara *et al.* [3] also proposed variable block ME and MC using rectangular and non-rectangular blocks. Main limitation of these approaches, if there is static background region in a MB, termed as *intra-block temporal redundancy* (ITR) (see in Figure 1), which has no motion and no residual error compared to the reference MB, still the H.264 or the approach in [3] requires some bits, which are significant especially at VLBR, to signal the decoder for zero motion vector and empty residual error. Thus, there is a scope to improve the coding efficiency of H.264 by efficient handling of the ITR.

Pattern-based video coding (PVC) initially proposed by Wong *et al.* [4] and later extended by Paul *et al.* [5] used pre-defined regular-shaped binary rectangular and non-rectangular pattern templates to segment out the moving regions (MR) in a MB to exploit the ITR. The best-matched MR of an MB with a pattern template (see in Figure 1) through an efficient similarity measure [6] estimates the motion and compensates the residual error using only pattern covered-region and ignores the remaining region of the MB from signalling any bits for motion vector and residual errors. Successful pattern matching can, therefore, theoretically attain maximum compression ratio of 4:1 for an MB as the size of the pattern is 64-pixel. The actual compression however will be lower due to the overheads of identifying this special type of MB as well as the best matched pattern for it and the matching error for approximating the MR using the pattern. An example of pattern approximation using pre-defined thirty two patterns [5] for *Miss America* video sequence is shown in Figure 2.

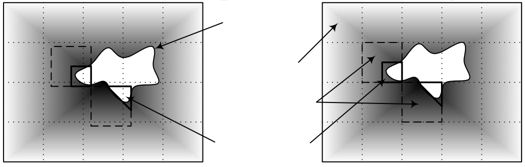

Figure 1: An example on how pattern based coding can exploit the intra-block temporal correlation in improving coding efficiency.

As the objects in video sequences are widely varied, thus, not necessarily the MR is well-matched with any predefined regular shaped pattern template. Intuitively, an efficient coding is possible if the MR is encoded using the pattern templates generated from the content of the video sequences. Very recently, Paul and Murshed [7] proposed a content-based pattern generation (CPG) algorithm to generate eight patterns from the given MRs. The PVC using those generated patterns outperformed the H.264 and the existing PVC by 1.0dB and 0.5dB respectively [7]. They also mathematically proved that this pattern generation technique is optimal if only one pattern would be generated for a given division of MRs. Thus, they get a local optimal solution as they can generate single pattern not multiple patterns. But for efficient coding, multiple patterns are necessary for different shape of MRs.

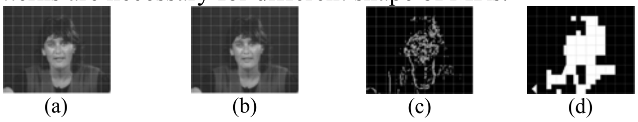

Figure 2: An example of pattern approximation for the *Miss America* standard video sequence, (a) frame number one, (b) frame number two, (c) detected moving regions, and (d) results of pattern approximation.

It is obvious that a global optimal solution improves the pattern generation process for multiple patterns, and hence, eventually the coding efficiency. A global optimal solution can be achieved if we are able to divide entire MRs optimally. But, this problem is a non polynomial (NP-complete) problem, as no clustering techniques provide optimal clusters. In this paper, we propose a heuristic to find the near optimal clusters and apply local optimal CPG algorithm on each cluster to get the near global optimal solution.

Moreover, the existing PVC used a pre-defined threshold to reduce the number of MBs coded using pattern to control the computational complexity as it requires extra motion estimation. It is experimentally observed that any fixed threshold for different video sequences may overlook some potential MBs from the pattern mode. Obviously, eliminating this threshold by allowing all MBs to be motion estimated and compensated using pattern and finally selected by the Lagrangian optimization function will provide better rate-distortion performance by increasing computational time. To reduce the computational complexity we assume the already known motion vector of the H.264 in pattern mode, which may degrade the performance. But we predicted that the net performance gain would outweigh this.

Note that, as the best pattern selection process relies on solely the similarity measures, it is not guaranteed that the best pattern will always result in maximum compression and better quality, which also depends on the residual errors after quantization and LM. This paper also exploited this possibility by introducing additional pattern modes that select the pattern in order of similarity ranking. In addition, a new LM is also developed as the pattern modes provide relatively less bits and slightly higher distortion compared to the other modes of the H.264. The experimental results confirm that this new scheme successfully improves the rate-distortion performance compared to the existing PVC as well as the H.264.

II. Background

The PVC with a set of content-based patterns termed as pattern codebook (PC) performs in two phases. In first phase MRs are collected from the given number of frames and *PC* is generated from those MRs using the CPG algorithm [7]. In the second phase, actual coding is taken place using the this *PC*.

A. Collection of MRs and Generation of PC

Moving region in a current MB is defined based on the number of pixels whose intensities are different from the corresponding pixels of reference MB. The MR *M* of an MB *C* in the current frame is obtained using the co-located MB *R* in the reference frame [4] as follows:

$$M(x,y)=T(|C(x,y)\bullet B-R(x,y)\bullet B|),\quad 0\le x,y\le 15 \tag{1}$$

where B is a 3×3 unit matrix for the morphological closing operation • [8], which is applied to reduce noise, and the thresholding function *T*(*v*) = 1 if *v* > 2 and 0 otherwise. Let $|Q|_\ell$ be the total number of ℓ 's in the matrix *Q*. If $8\le|M|_1<2QP/3+64$ where *QP* is the quantization parameter, the corresponding MB, i.e., *C*, is defined as a candidate active-region MB (CRMB) as it has a reasonable number of moving pixels to be covered by a 64-pixel pattern thus avoiding high matching error.

Once all such CRMBs are collected for a certain number of consecutive frames, decided by the rate-distortion optimizer [2] when the rate-distortion gain outweighs the overhead of encoding the shape of new patterns, these are divided into α sets to generate α patterns. In order to generate patterns with minimal overlapping, a simpler greedy heuristic is employed where these CRMBs are divided into α clusters such that the average distance among the gravitational centers of CRMBs within a cluster is small while the same among the centers of CRMBs taken from different clusters is relatively large. The CPG algorithm generates μ-pixel pattern for a cluster by the μ-most-frequent pixels among all the CRMBs in the cluster.

B. Encoding and Decoding of PVC using Content-based PC

The LM [2] is used to trade off between the quality of the compressed video and the bit rate generated for different modes. In this method, the LM (λ) is first calculated with an empirical formula using the selected QP for every MB:

$$\lambda=0.85\times2^{(QP-12)/3}\,. \tag{2}$$

During the encoding process, all possible modes including pattern mode are first motion estimated and compensated for each MB and the resulting rates and the distortions are determined. The final mode m_n is selected as follows:

$$m_n=\arg\min_{\forall m_i}(D(m_i)+\lambda R(m_i)) \tag{3}$$

where $R(m_i)$ is the total bits for mode m_i, including mode type, motion vectors, (extra pattern index code for pattern mode) and residual error after quantization, while $D(m_i)$ is measured as the *sum of square difference* between original MB and corresponding reconstructed MB for mode m_i.

III. Proposed Algorithm

As mentioned earlier, the CPG algorithm can generate an optimal pattern from given MRs but there is no guarantee to generate optimal multiple patterns from given entire set of MRs. For simplicity it uses a clustering technique which divides the MRs into α clusters to generate α patterns. Thus, it is obvious that the performance of CPG also depends on the efficiency of clustering technique. Clustering problem is an NP-complete problem and thus, global optimization algorithm would be computationally unworkable. We propose a heuristic which can solve this problem near optimally.

A. Optimal Content-based Pattern Generation algorithm

Without losing any generality, we can assume that an optimal clustering technique with the CPG algorithm can provide optimal *PC*. We can define an optimal *PC*, if each MR is best-matched by the pattern which is generated from the cluster of that MR. This can be defined as:

$$\exists M\exists P_i[M\in C_i\wedge P_i=G(C_i)] \tag{4}$$

where $P_i=G(C_i)$ means pattern P_i is generated by the CPG algorithm if and only if one pattern is allowed to generate from C_i cluster of MRs.

In actual coding phase, a CRMB of a cluster will be approximated by the pattern generated from its corresponding cluster or by the best matched-pattern from the PC irrespective of its clusters. The first approach is termed as direct pattern selection (DPS) and later approach is exhaustive pattern selection (EPS). Right classification, τ can be defined as the fraction of the number of CRMBs matched by the pattern using DPS against entire CRMBs. Due to the overlapped regions of the patterns, there is a probability to better approximate a CRMB with a pattern generated other than its cluster. Obviously the probability of τ will increase with the number of patterns in a *PC* due to the better similarity between MR and the corresponding pattern. Moreover, a small number of patterns cannot better approximate the CRMBs [1], as a result there is always a possibility of ignoring a CRMB using pattern mode, if only the extracted pattern from a cluster is used to match against the CRMBs of the same cluster. Thus, this system requires reasonable number of patterns. On the other hand, we can called CPG algorithm as globally optimal algorithm if it produces a pattern set in such a way that each CRMB is best-similarity-matched by the pattern which is generated from its own cluster, i.e., the value of τ is 100%. When we get τ = 100% we can easily claim that we get optimal solution using clustering and the CPG algorithm. To do this we need to modify CPG algorithm where a generic clustering technique using pattern similarity [6] metric as a part of this algorithm. The dissimilarity of the MR of a MB against a pattern, P_n is defined [6] as:

$$\psi = |M|_1 - |M \wedge P_n|_1. \tag{5}$$

where, M and P_n are the CRMB and n-th pattern respectively. The best-matched pattern, P_i is selected as

$$P_i = \arg \min_{\forall P_n \in \mathrm{PC}} \left(|M|_1 - |M \wedge P_n|_1 \right) \tag{6}$$

Unlike the CPG, optimal CPG (OCPG) (detailed in Figure 3) continues to clustering and pattern formation until τ is 100% in each iteration. For a *seed* PC, it ensures that each CRMB will be best-matched by the pattern generated from its own cluster i.e., clustering process is optimum. However, it does not guarantee the global optimality of clustering because of trapping in local optima. To ensure the global optimality we need to determine ψ^t_{avg} using *PC* generated in iteration. The final *PC* is selected based on the minimum ψ_{avg} for a given number of iterations with random starts as ψ_{avg} indicates the optimal *PC*. For one random start, we will get a candidate global solution for a seed *PC*. There would be multiple solutions for given MRs. When the search space is really large and there is no suitable algorithm to find the optimum solution, *k*-change neighbourhood may be considered as a *k*-Opt solution [13]. Lin [14] empirically found that a 3-Opt for the *travelling salesman problem* (TSP) has a probability of ≈ 0.05 of being optimal, and hence for 100 random starts yields the optimum with a probability of 0.99. Lin also importantly demonstrated that a 3-Opt solution is much better than 2-Opt solution, however a 4-Opt solution is not sufficiently superior to the 3-Opt solution to justify the additional computational cost. In our approach we also use 100 random starts and replace 3-pixel in each pattern to get the optimal solution. We terminate each iteration of a random start when either the average dissimilarity is not reduced in successive iteration or τ = 100%. Thus, OCPG ensures convergence by providing near optimal solutions.

```
Algorithm PC = OCPG(α, μ, K, C)
Precondition: Given a set of CRMBs, C, Given iterations K;
Post condition: A pattern codebook $PC = \{P_1,\ldots,P_\alpha\}$ of μ-pixel content-based patterns;
k = 0; τ = 0; $\psi_{avg} = \infty$; Replace = 0;
WHILE (k<K)
    Randomly generate α number of patterns, $P^t_1,\ldots,P^t_\alpha$ of μ-pixel
    Divide C into α clusters based on the equation (6) using PC or any clustering algorithm.
    WHILE (Replace<=1)
        WHILE ((τ<100) && ( $\psi^{t-1}_{avg}$ - $\psi^t_{avg}$ ))
            For i=1,…,α
                For x = 0,…,15
                    For y = 0,…, 15
                        $P^t{}_i(x,y) = 0;$
                        $T_i(16x+y) = f_i(x,y) = \sum_{j=1}^{|C_i|} M_{i,j}(x,y)$ where
                        $M_{i,j}$ is the MR of the $j^{th}$ CRMB in $C_i$;
                        $\{l_0,\ldots,l_{255}\}$ =ranked indices of $T_i$ such that
                        $T_i(l_j) \geq T_i(l_{j+1})$ for 0≤j<255;
                For j=0,…,μ-1
                    $P^t{}_i(\lfloor l_j/16 \rfloor, l_j \bmod 16) = 1;$
            Divide C into α clusters based on the equation (6) using
            new PC and calculate τ and $\psi^t_{avg}$ for all M.
        Replace 3-pixel for each pattern; Replace=Replace+1; $\tau = 0$;
    IF $\psi_{avg} > \psi^t_{avg}$
        $\psi_{avg} = \psi^t_{avg}$;  $PC = \{P^t_1,\ldots,P^t_\alpha\}$
    $k = k + 1;$
```

Figure 3: The optimal content-based pattern generation (OCPG) algorithm for near optimal multiple pattern sequence generation.

The main advantage of this OCPG approach over previous version is that it takes whole MR information to cluster the CRMB against pattern instead of a gravitation centre of a CRMB [7]. Moreover, multiple of iterations ensure the quality of *PC* to represent the CRMBs and this approach does not require EPS pattern matching so that it reduces the computational time needed to select the best-match pattern from a *PC* against each CRMB.

B. *Impact of OCPG algorithm on Right Classification τ, Dissimilarity ψ, and the number of Iterations*

Figure 4 shows average number of iterations needed for each random start to provide τ = 100% using ten standard QCIF video sequences. The whole average is 9.73 per random start.

Figure 5 shows that how the OCPG algorithm generates the optimal *PC*. For each random start the OCPG algorithm reduces the dissimilarity, ψ_{avg} (see Figure 5 (right)) by classifying the CRMBs (line 15 of the OCPG algorithm) using the best-matched pattern. Thus, it increases the right classification τ (see Figure 5 (left)) gradually and also ensures the convergence of the OCPG algorithm.

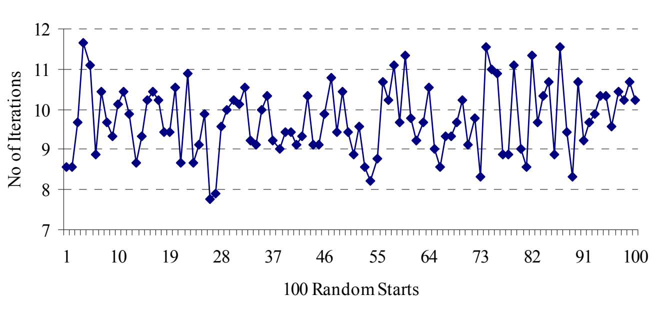

Figure 4: Average number of iterations is needed to get τ = 100% with 100 random starts using 10 standard QCIF video sequences where the total average is 9.73.

It is clear that the coding performance will decrease with the group of frames (GOF) participated in pattern formation process of the OCPG algorithm as the generated PC is gradually losing its quality to approximate the shape of the CRMBs. This restricts the GOF size. Obviously, GOF size is less than the size of a video sequence. Thus, we need to refresh pattern codebook with regular interval. To find the various options of interval, we experimentally find that a group of picture (GOP) size would be good candidate to test whether we refresh the *PC*. The detailed procedure of PC transmission will be described in the sub section *D*.

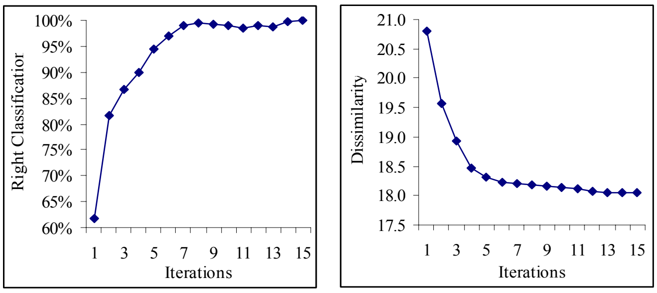

Figure 5: Improvement of clustering process using the OCPG algorithm for the best random start using the first group of picture (GOP) of *Miss America* sequence where right classification, τ increases (left) and dissimilarity, ψ_{avg} decreases with the iterations (left).

C. Clustering Techniques

The CPG algorithm uses *K*-means clustering technique [11] where it uses gravitational centre of the CRMBs to cluster them. The average value of τ is at best 70% using the CPG algorithm. It is due to the gravitational centre which represents entire 256 pixels by a point. We also investigate Fuzzy C-means (FCM) [9][10] clustering technique, but the results is almost same. Neural network is not good candidate due to the computational time. It is interesting to note that the performance of OCPG algorithm does not depend on any specific clustering algorithm because whatever the clustering algorithm used initially will defer in generating only seed *PC* but subsequently it converges quickly with our pattern similarity matching system.

D. Pattern Codebook Refreshing

For content base pattern generation, we need to transmit *PC* after a certain interval. To determine whether we transmit the newly generated *PC* or with the current *PC*, we consider two bitstreams together with their corresponding bits and distortions generated independently by using both current and previous *PC*. We have to take decision after certain interval. Group of picture (GOP) [12] may be the best choice as after a GOP we need to send fresh *Intra* picture in the bitstream. Note that this GOP may be different from GOF. To trade off the bitstream size and quality we can use Lagrangian optimization function as it is used to control the rate-distortion performance. Here we consider average distortion and bits per MB in both cases for processed GOF. We will select current *PC* if it provides less Lagrangian cost compared to the previous PC. Obviously we add cost of bits for transmission of PC with encoded bits.

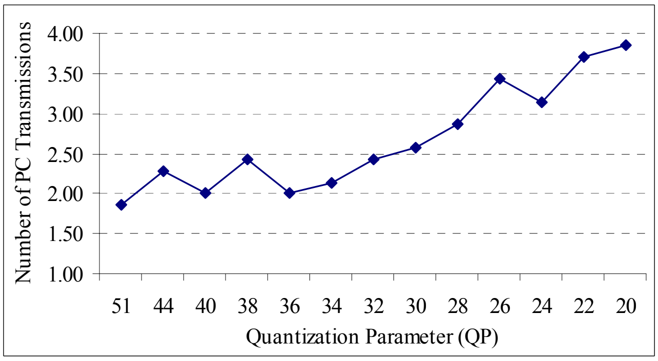

Figure 6: The average number of pattern code transmissions with quantization parameters when we processed first 100 frames using seven standard QCIF video sequences namely *Miss America*, *Suzie*, *Claire*, *Salesman*, *Car phone*, *Foreman*, and *News* of 30 frames per second.

From the experimental results we observe that around 2 to 4 times we need to refresh arbitrary patterns while we use first 100 frames of standard QCIF video sequences (see Figure 6). The figure also shows that the number of transmission increases with the bit rates because almost fixed amount of bits for pattern transmission has significant contribution at the VLBR but insignificant at the high bit rates. Note that 5 times refreshment means we need to refresh the *PC* in all GOP.

E. Multiple Pattern Modes and Allowing all MBs as CRMBs

As we mentioned in the introduction, the best pattern selection process relying on the similarity measures does not guarantee that the best pattern would always result in coding efficiency because of residual errors after quantization and the LM. To eliminate this possibility we also propose multiple pattern modes that select the pattern in order of similarity. Since the similarity measure is a good estimator, we only consider higher ranked patterns. Eliminating the CRMB classification threshold $8 \le |M|_1 < 2QP/3 + 64$, by allowing all

MBs to be motion estimated and compensated using pattern modes and finally selected by the Lagrangian optimization function, will provide better rate-distortion performance. Obviously both will increase the computational complexity which is checked by using already known motion vector determined by the 16×16 mode.

IV. Real-Time OCPG Algorithm

In order to find the computational overhead of the proposed PVC using near-global-optimal arbitrary shaped PVC (ASPVC-Global), let us compare it with the H.264 standard. From now on, previous content-based PVC is termed as ASPVC-Local. The H.264 encodes each MB with motion compensation requiring one motion search for each mode. When ASPVC-Global is embedded into the H.264 as extra mode, additional one-fourth motion compensation is required per MB as pattern size is quarter of an MB. In addition, each MB also takes part in the OCPG algorithm and the best pattern is selected at the end. For detailed analysis of OCPG algorithm is described as follows.

We can divide the entire process into i) Binary matrix calculation, ii) clustering and right classification τ calculation, iii) pixel frequency calculation of each cluster, and iv) sorting the pixels based on the frequency. Let N, α, M^2, k, and I be the total number of MBs, total number of cluster, block size, total number of random starts, and number of iterations to get the right classification $\tau = 100\%$ respectively.

- [i] Binary matrix calculation requires one subtraction, one absolute and one compare operations. Thus, it requires $3NM^2$ operations.
- [ii] Clustering requires one comparison and one addition operations. Thus it requires $2\alpha NM^2$ operations. Right classification calculation requires one comparison operation. Thus, it requires αN operations.
- [iii] Pixel frequency calculation requires one addition operation. Thus it requires NM^2.
- [iv] Sorting the pixel frequency requires $2\alpha M^2 \ln M$ operations.

The pattern generation algorithm, OCPG requires $3NM^2 + kI(2\alpha NM^2 + \alpha N + NM^2 + 2\alpha M^2 \ln M)$ operations. If we simplified the number of operations by assuming $N >> \alpha$ and $N >> M$, the required operations would be $nM^2(4+16K)$ where K is the total number of iterations including the number of random starts and internal iterations. On the other hand, motion search using any mode requires $3(2d+1)^2 NM^2$ operations where d is the total width of motion search. Thus, the ASPVC-Global with 100 random starts and internal iterations (i.e. 9.73 according to Figure 4) until τ = 100% requires no more than 5.4 times operations compared to the full motion search by a mode where search length is 15.

Compared to the fractional as well as multi modes motion search this extra operation does not restrict it from real time operations. The experimental results also show that maximum of dissimilarity is within 7% of minimum dissimilarity of 100 random starts. It means that if we consider only one start, we only lose 7% of clustering accuracy. Thus, according to the hardware availability we can make OCPG real time by reducing the iterations. The experimental results show that with only 5 iterations we can achieve very similar performance of optimal one and better than the existing approach. The OCPG with 5 random starts and the number of iterations (i.e. 9.73 according to Figure 4) until τ = 100% requires no more than 0.3 times operations compared to the full motion search using a mode where search length is 15 pixels.

For multiple pattern modes, the ASPVC-Global needs only bits and distortion calculation without motion estimation. Motion estimation, irrespective of a scene's complexity, typically comprises more than 60% of the processing overhead required to encode an inter picture with a software codec using the DCT [10], when full search is used. Thus, maximum of 10% operations are needed for one pattern mode as each pattern mode will process one-fourth of MB. As a result, the ASPVC-Global algorithm using 5 random starts and up to four pattern modes may requires extra 0.58 mode ME&MC operations compared to the H.264 which would not be a problem in real time processing.

V. Embedding into H.264

To accommodate extra pattern mode in existing H.264 video coding standard, we need to modify its bitstream structure. The H.264 recommendation documents [16] provided binarization for MB and sub-MB in *P* and *SP* slice on Table 9-27 and Table 9-28. Experimental results show that most of the cases 8×8 mode is less frequent compared to the other large modes. Thus we use first part of MB type header for pattern mode using '001' code and then assign variable length code for pattern mode, 8×8, 8×4, 4×8, and 4×4. Using frequency of MB type we assigned pattern mode, 8×8, 8×4, 4×8, and 4×4 as '0', '10', '111', '1100', and '1101' respectively. After the header of MB type we need to send pattern type with the maximum of length codes when fixed length pattern code will be used.

We also investigate LM after embedding new pattern modes in the H.264 standard. It is already mentioned that a pattern mode provides less bits and sometimes higher distortion compared to the other modes. Due to this, it is biased to the bit rate. To keep the fairness to the other modes, it is rational to reduce the value of LM. Our suggested value is $\lambda = 0.4 \times 2^{(QP-12)/3}$. The experimental results of LM and rate-distortion performance also justify the new value. Detailed results are not included due to page restriction.

VI. Simulation Results

In this paper, experimental results in Figure 7 are presented using the first 100 frames of four standard video sequences, comprising *I*- and *P*- type frames of QCIF digital video format [12]. Full-search motion estimation and H.264 recommended 'baseline' profile were employed to obtain the encoding results for standalone H.264, ASPVC-Global, ASPVC-Local, and PVC. ASPVC-Global improved image quality by 1.5~2.0 dB and 0.5~0.75 dB compared to H.264 and ASPVC-Local respectively. The ASPVC-Global exhibits better performance due to the global optimization, allowing all MBs into multiple pattern generation and pattern modes.

VII. Conclusions

In this paper, we proposed an arbitrary shaped pattern-based VLBR video coding scheme which uses optimal content-based pattern generation strategy, multiple pattern modes, and all MBs into the pattern generation process and pattern modes. A new Lagrangian multipler is also derived due to the change of the modes from the original H.264 standard. The experimental results show that this new scheme improves the video quality by 0.5dB and 1.5dB compared to the existing PVC and the H.264 standard respectively.

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] T. Wiegand, H. Schwarz, A. Joch, and F. Kossentini, "Rate-constrained coder control and comparison of video coding standards," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 13(7), pp. 688-702, 2003.

[3] 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 Transaction on Circuits and Systems for Video Technology*, vol. 7, pp. 212–220, 1997.

[4] K. –W. Wong, K.-M. Lam, and W.-C. Siu, "An Efficient Low Bit-Rate Video-Coding Algorithm Focusing on Moving Regions," *IEEE Transaction on Circuits and System for Video Technology*, vol. 11(10), pp. 1128-1134, 2001.

[5] 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 Transaction on Circuits and System for Video Technology*, vol. 15(6), pp. 753-761, 2005.

[6] M. Paul, M. Murshed, and L. Dooley, "A new efficient similarity metric and generic computation strategy for pattern-based very low bit-rate video coding," *Proc. of IEEE Int. Conf. of Acoustics, Speech, and Signal Processing* , vol. 3, pp. 165-168, 2004.

[7] M. Paul and M. Murshed,, "An optimal content-based pattern generation algorithm," *IEEE Signal Processing Letters*, vol. 14, no. 12, pp. 904-907, December, 2007.

[8] P. Maragos, "Tutorial on advances in morphological image processing and analysis," Opt. Eng., vol. 26(7), pp. 623-632, 1987.

[9] Dunn, J. C. "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", *Journal of Cybernetics* vol. 3, pp. 32–57, 1973.

[10] Bezdek, J. C. "Pattern Recognition with Fuzzy Objective Function Algoritms", Plenum Press, New York, 1981.

[11] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," *Proc. of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, vol. 1, pp. 281-297, 1967.

[12] I. E. G. Richardson, *H.264 and MPEG-4 video compression*, JOHN WILLEY & SONS, LTD, 2003.

[13] C. H. Papadimitriou, and K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Prentice-Hall, 1939.

[14] S. Lin, and B. W. Kernighan, "An effective Heuristic Procedure for the Traveling-Salesman Problem," OR, vol. 21, pp. 498-516, 1973.

[15] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to lower spatio-temporal resolutions and different encoding formats," *IEEE Trans. on multimedia*, vol. 2(2), pp. 101–110, 2000.

[16] ITU-T Telecommunication standardization sector of ITU, "Advanced video coding for generic audiovisual services," May, 2003.

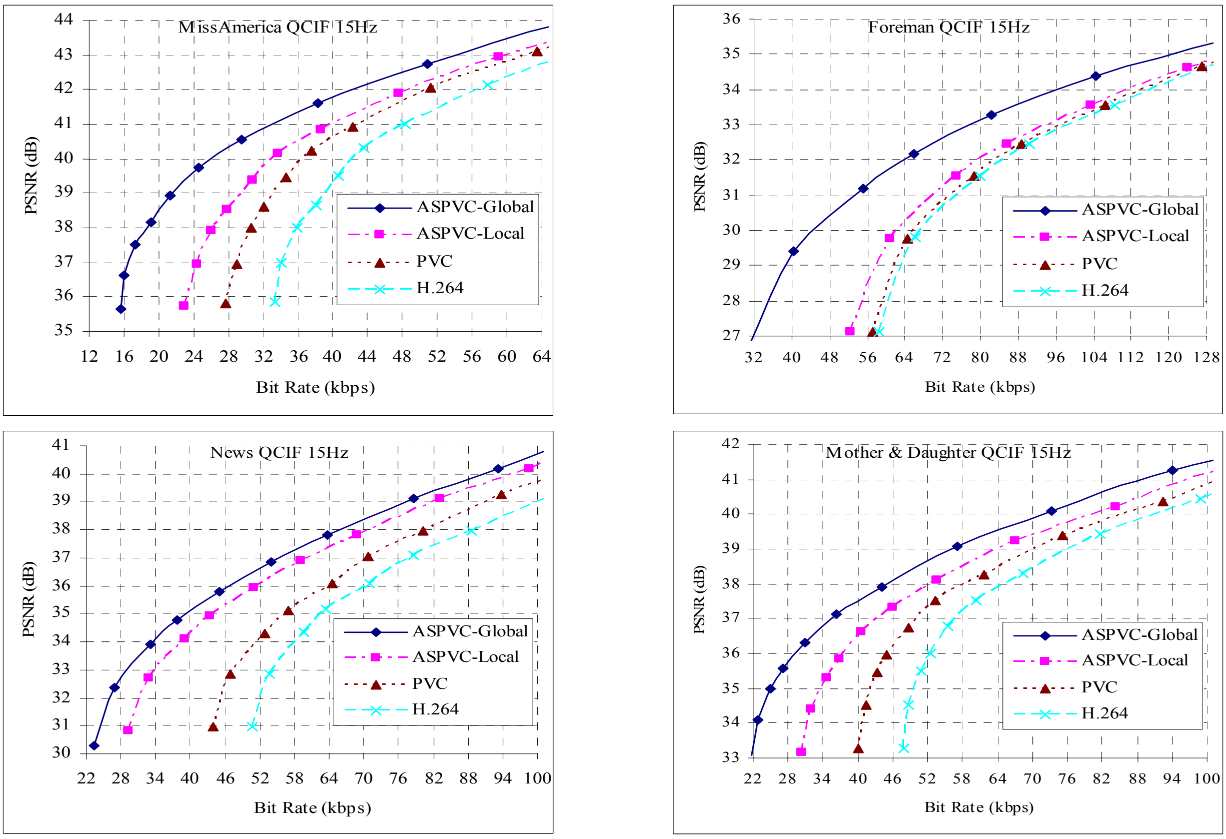

Figure 7: Rate-distortion performance on standard video sequences.