A real time generic variable pattern selection algorithm for very low bit-rate video coding

Conference or Workshop Item

A REAL TIME GENERIC VARIABLE PATTERN SELECTION ALGORITHM FOR VERY LOW BIT-RATE VIDEO CODING

Manoranjan Paul, Manzur Murshed, and Laurence Dooley

Gippsland School of Computing and IT, Monash University, Churchill Vic 3842, Australia
E-mail: {Manoranjan.Paul,Manzur.Murshed,Laurence.Dooley}@infotech.monash.edu.au

ABSTRACT

The selection of an optimal regular-shaped pattern set for very low bit-rate video coding, focusing on moving regions has been the objective of much recent research in order to try and improve bit-rate efficiency. Selecting the optimal pattern set however, is an NP hard problem. This paper presents a Generic Variable Pattern Selection (GVPS) algorithm, which introduces a pattern selection parameter that is able to control the performance in terms of computational complexity as well as bit-rate and picture quality. While using a sub-optimal variable pattern set, GVPS obtains a coding performance comparable to near-optimal algorithms, such as the k-change neighbourhood solution, while being much less computationally intensive, so that it is able to process all types of video sequences in real-time, with minimal pre-processing overheads.

1. INTRODUCTION

Reducing the transmission bit-rate while concomitantly retaining image quality is the most daunting challenge to overcome in the area of very low bit-rate video coding, e.g., H.26X standards [5]–[8]. The MPEG-4 [4] video standard introduced the concept of content-based coding, by dividing video frames into separate segments comprising a background and one or more moving objects. This idea has been exploited in several low bit-rate macroblock-based video coding algorithms [1][14] using a simplified segmentation process which avoids handling arbitrary shaped objects, and therefore can employ popular macroblock-based motion estimation techniques. Such algorithms focus on moving regions through the use of regular pattern templates, from a *pattern codebook* (Figure 1), of non-overlapping rectangular blocks of 16×16 pixels, called *macroblocks* (MB).

The algorithm proposed by Wong et al [14], used eight fixed patterns (*Fixed* 8), with *macroblocks* classified according to the following three mutually exclusive classes: 1) *Static MB* (SMB): Blocks containing little or no motion; 2) *Active MB* (AMB): Blocks that contain moving object(s) with little static background; and 3) *Active-Region MB* (RMB): Blocks that contain both static background and some part(s) of moving object(s). In [1], a pattern codebook of four 128-pixel patterns was used. Further improvements were obtained in the *Fixed 8* algorithm by using a pattern codebook of eight 64-pixel patterns (P_1–P_8 in Figure 1). In [10], Paul *et al.* presented a *Variable Pattern Selection* (VPS) algorithm to select the λ best-matched patterns from a codebook of patterns P_1–P_{24} in Figure 1 using a "greedy" algorithm, where $\lambda \in \{4,8,16,24\}$. The VPS algorithm selected a preset number of the best-matched patterns, from a 24-pattern codebook, by eliminating the least frequent pattern per iteration. This strategy was computationally expensive as $24-\lambda$ iterations were required in order to select λ best-matched patterns. To achieve greater efficiency, an *Extended VPS* (EVPS) [11] algorithm was developed to improve the pattern elimination process. In EVPS, the best-match pattern selection process still exploited the matching frequency of each pattern, however, unlike VPS, the EVPS algorithm allowed more than one pattern per iteration to be eliminated. The assumption being that λ best-matched patterns always lie in the first ρ ($\geq \lambda$) most frequent patterns so their cumulative frequency is greater than or equal to 75%. Empirically, it was proven that only $\lceil 16/\lambda \rceil + 1$ iterations were required to select the λ best-matched patterns, where $\lambda \in \{4, 8, 16, 24\}$. Recently [12], a new parametric approach for MB classification was proposed, which outperformed the previous definition used in the *Fixed 8* algorithm for $\delta = 96$, where δ is the number of moving pixels in an RMB.

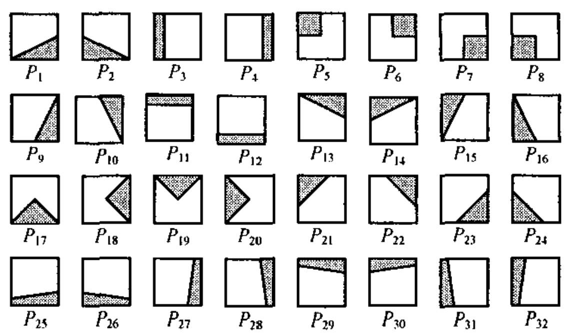

Figure 1: The pattern codebook of 32 regular shaped, 64-pixel patterns, defined in 16×16 blocks, where the shaded region represents 1 (motion) and white region represents 0 (no motion).

It has been proven experimentally that the *Fixed 8*, VPS, and EVPS algorithms do not provide the optimal pattern set for all types of video sequence. To find the optimal pattern set (in terms of the minimum bit-rate and the maximum image quality) from a large number of patterns codebook for a particular video sequence is a non-deterministic polynomial time (NP) hard problem [15]. Sub-optimal pattern sets may be generated; however these can only be used for stored and not real time video

0-7803-7750-8/03/$17.00 ©2003 IEEE

transmission because of the high computational complexity involved.

This paper presents a new *Generic Variable Pattern Selection* (GVPS) algorithm, which incorporates both the VPS and EVPS algorithms with new MB parametric definition [12]. Its performance is compared with the H.263 standard, *Fixed 8* algorithm and 3-change neighborhood (known as 3-Opt). Experimental results will confirm that the GVPS algorithm outperforms both the H.263 standard and *Fixed 8* pattern algorithm and provides comparable results with the 3-Opt algorithm, while crucially achieving a real-time capability.

The paper is organized as follows. The video coding strategy using variable patterns to represent moving regions is described in Section 2. Section 3 presents the new algorithm for obtaining the near-optimal pattern set, while simulation results are discussed in Section 4. Section 5 concludes the paper.

2. LOW BIT-RATE VIDEO CODING USING VARIABLE PATTERNS

The main motivation [10][11] for using a variable pattern selection algorithm is to attempt to derive the λ best-matched patterns from a large pattern codebook. The EVPS algorithm achieved an improved pattern elimination strategy compared with the original VPS algorithm, by using a threshold of 75%. This paper proposes a *Generic Variable Pattern Selection* (GVPS) algorithm by using an arbitrary pattern selection parameter γ to obtain near-optimal performance.

2.1 Moving region detection

Let $C_k(x,y)$ and $R_k(x,y)$, denote the k^{th} macroblock of the current and reference frames respectively, where the frame dimension is W pixels $\times H$ lines. The moving region $M_k(x,y)$ in the k^{th} macroblock of the current frame is obtained as follows:

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

where B, which has size 3×3 is the structuring element of a morphological closing operation $\bullet$ [2][9], $|v|$ is the absolute value of v, $T(v) = 1$ if $v > 2$ or 0 otherwise, $0 \le x, y \le 15$, and $0 \le k < W/16 \times H/16$.

Let the *likelihood* of the k^{th} macroblock with pattern P_n be calculated as

$$D_{k,n} = \frac{1}{256}\sum_{x=0}^{15}\sum_{y=0}^{15} |M_k(x,y) - P_n(x,y)|$$

where $1 \le n \le 32$.

The k^{th} macroblock is then classified as follows, for all k:

1) IF $\sum M_k < 8$ THEN the k^{th} macroblock is classified as an SMB.
2) ELSE IF $\sum M_k < \delta$, where $\delta \in \{64,96,128\}$, AND $D_{k,n} < 0.25$ THEN the k^{th} macroblock is classified as an RMB, whose moving region is represented by the first pattern P_i in the codebook where, $D_{k,i} = \min(D_{k,n})$.
3) ELSE the block is classified as an AMB.

2.2 Encoding and Decoding

Since each SMB and the static regions of RMBs are considered as having no motion, they can be omitted from both coding and transmission as they can be obtained from the reference frame. For each AMB, as well as the moving region of each RMB, motion vectors and residual errors are calculated using conventional block-based methods, with the obvious difference in having the shape of the blocks for the moving regions of RMBs as that of the best-match pattern, rather than being square.

To process the RMB, a motion vector is calculated from only the 64 moving pixels of the best-match pattern. To avoid multiple 8×8 blocks of DCT calculations for 64 residual error values per RMB, these 64 values are rearranged into an 8×8 block. A similar inverse rearrangement is performed during the decoding. The aim is to design an encoder for low motion video sequences, and since these types video sequence have more SMB than any other type, for efficient variable length encoding is used: '0' for SMB, '10' for AMB and '11' for RMB.

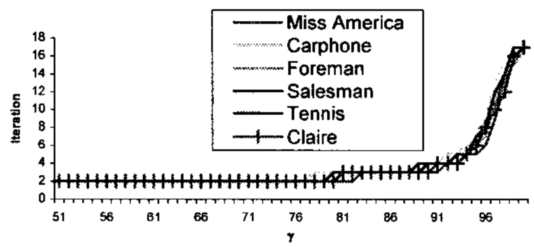

Figure 2: Pattern selection parameter γ, and its corresponding iterations for different standard video sequences in GVPS(16,90) where δ=128.

2.3 GVPS Algorithm

The GVPS algorithm is the generalised form of the variable pattern selection algorithm described in [10][11], and incorporates a pattern selection parameter γ, which controls the computational complexity as well as bit-rate and picture quality (PSNR). For example, when γ = 100%, GVPS is the same as the VPS algorithm, while when γ = 75%, GVPS becomes the EVPS algorithm. Figure 2 plots the effect of adjusting γ on the computational complexity and it is clear that setting the pattern selection parameter γ = 90%, then fewer than 5 iterations are required for standard video sequences. This means that GVPS can be processed in real time, with the only requirement being for some preliminary pre-processing to select the best pattern set. Our experiments also revealed that using GVPS with γ = 90% provided exactly the same λ best patterns as VPS, with a computational cost comparable with that of EVPS. The full GVPS algorithm is now formulated:-

Algorithm GVPS(λ,γ)

Parameter: λ = *Size of the optimal pattern set.*

γ = *Pattern selection window parameter*

Return: Ω = *The optimal pattern set.*

Step 1: $\Omega = \{P_1, P_2, \ldots, P_{32}\}$;

Step 2: Calculate the frequency of each pattern in Ω;

Step 3: IF $|\Omega| = \lambda$ THEN stop;

Step 4: ELSE sort Ω to $(P_{i_1}, P_{i_2}, \ldots, P_{i_{|\Omega|}})$ such that the frequency $\text{Freq}(P_{i_j}) \geq \text{Freq}(P_{i_{j+1}})$ for all $j < |\Omega|$;

Step 5: Find the minimum value of $\rho \geq \lambda$ where $\sum_{j=1}^{\pi} \text{Freq}(P_{i_j}) \geq \gamma$;

Step 6: $\Omega = \Omega - \{\forall j > \rho : P_{i_j}\}$;

Step 7: Go to Step 2;

Figure 3: The GVPS Algorithm.

3. THE OPTIMAL PATTERN SET

Since objects in a video sequence are both arbitrarily shaped and their movements equally varied, a single prescribed set of patterns is not suitable for all types of objects. In [14], the *Fixed 8* pattern algorithm claimed that the first eight patterns are always the most popular. Our observations however, show that no one pattern set is suitable for all types of video. The pattern codebook size may be an arbitrarily large number to capture all types of objects. In order to find a suitable pattern from the codebook set, a heuristic search is necessary. The searching criteria adopted needs to be the minimum bit-rate and the maximum image quality. Assuming a codebook size of α and the optimum pattern set size of λ, then ${}^{\alpha}c_{\lambda}$ combinations need to be searched. For α =32 and λ =8 around 11 million search spaces need to be checked to find the best pattern set which precludes real time applications. So in order to find the optimum pattern set from a relatively larger pattern codebook size is an NP hard problem [15].

When the search space is really large and there is no suitable algorithm to find the optimum solution, k-change neighborhood may be considered as a k-Opt solution [15]. The k-change neighborhood ($k \geq 1$) for the pattern set f is defined by

$N_k(f) = \{g \mid g \in \alpha$ and g is obtained as follows:
remove k patterns from f; then replace them with
ramdomly selected k patterns from set α *that are not in set* $f\}$

To find the k-Opt pattern set, define the function *improve* (t), where $t \in \alpha$, as:-

$$improve\ (t) = \begin{cases} \exists s \mid s \in N_k(t) \text{ AND } RMB(s) > RMB(t) \\ \text{'}no\text{' otherwise} \end{cases}$$

This function searches $N_k(t)$ for a better pattern set s. If one is found, it returns the improved pattern set; otherwise it returns the value 'no'. The algorithm for finding a k-Opt pattern set is:

```
BEGIN
    t:=some initial starting pattern set in α;
    while improve(t)≠'no' do
    t:= improve(t);
    return t;
END
```

Lin [16] found empirically that a 3-Opt for the Traveling-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.

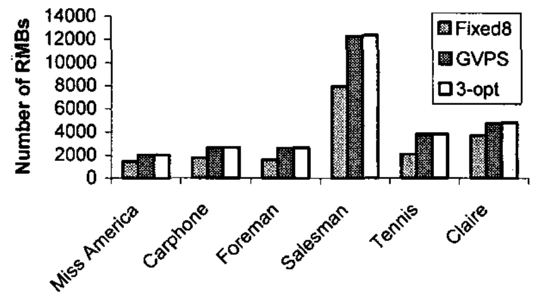

Figure 4: The number of RMBs comparison for some standard video sequences in different methods.

In our experiments, we considered 100 times random start 3-Opt pattern set. The RMBs selected by 3-Opt, GVPS (16,90) where δ=128, and *Fixed 8* algorithm are plotted in Figure 4. The more RMBs selected, the greater the probability of reducing the bit-rate. Figure 5 shows the RMB frequencies of corresponding selected pattern set by GVPS and 3-Opt for the "Miss America" video sequences. The experimental results show that the number of RMBs selected by 3-Opt is more than that of GVPS, however while this means a lower bit-rate, GVPS patterns are selected in real time, which is not possible when using the k-Opt algorithm.

4. SIMULATION RESULTS

Each algorithm along with the H.263 standard have been tested on a large number of standard and non-standard video sequences of CIF and QCIF digital video formats [13] with different degrees of object and camera motion. For the purposes of this paper, experimental results are presented using the first 100 frames of six standard video sequences. Full-search motion estimation and the H.26X recommended variable length coding are employed for obtaining the encoding results using the proposed GVPS approach, as well as the *Fixed 8*, and 3-Opt schemes. The bit-rate is calculated assuming a frame rate of 10fps and half-pel accuracy is used throughout.

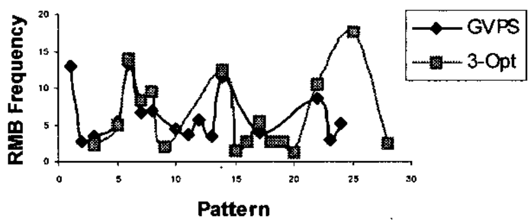

Figure 5: RMBs of corresponding selected patterns by 3-Opt and GVPS(16,90) algorithm for Miss America where δ=128.

Table I confirms that GVPS gives better results compared to both the H.263 standard and *Fixed 8* algorithm for low and high bit rates. It gives almost identical results to the near-optimal 3-Opt algorithm in terms of both bit rate and (PSNR) performance, while being much less computational intensive. GVPS is therefore able to provide real time video transmission with a near-optimal pattern set selection, while requiring minimal pre-processing to select the initial best pattern set.

5. CONCLUSIONS

In this paper, a novel *Generic Variable Pattern Selection* (GVPS) algorithm has been developed using a 32-pattern codebook. As an optimal variable pattern selection algorithm is a NP hard problem, the near-optimal 3-Opt algorithm is used for results comparison, because it yields a very high probability of obtaining an optimal solution. The experimental results proved that GVPS algorithm gave superior results for all video sequence types, compared with the *Fixed 8* algorithm and H.263 standard. It also provided comparable results to the 3-Opt algorithm in terms of bit-rate and picture quality (PSNR). GVPS is however much more computationally efficient enabling real time video transmission with minimal pre-processing requirements.

6. REFERENCES

[1] Fukuhara, T., 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 Syst. Video Techno.*, 7, 212–220, 1997.

[2] Gonzalez, R.C. and R. E. Woods, *Digital Image Processing*, Addison-Wesley, 1992.

[3] ISO/IEC 13818, MPEG-2 International Standard, 1995.

[4] ISO/IEC N4030, MPEG-4 International Standard, 2001.

[5] ITU-T Recommendation H.261, "Video codec for audio-visual services at px64kbits/s," 1993.

[6] Draft ITU-T Recommendation H.263, "Video coding for low bit-rate communication," 1996.

[7] Draft ITU-T Recommendation H.263, "Video coding for low bitrate communication," Version 2, 1998.

[8] ITU-T Q6/SG16 VCEG-L45, "H.26L Test Model Long Term Number 6 (TML-6) draft 0," March 2001.

[9] Maragos, P., "Tutorial on advances in morphological image processing and analysis," *Opt. Eng.*, 26(7), 623–632, 1987.

[10] Paul, M., M. Murshed, and L. Dooley, "A Low Bit-Rate Video-Coding Algorithm Based Upon Variable Pattern Selection," Proc. of 6th Int. Conf. on Signal Processing, Beijing, China, August 26–30, Vol-2, 933–936, 2002.

[11] Paul, M., M. Murshed, and L. Dooley, "A Variable Pattern Selection Algorithm with Improved Pattern Selection Technique for Low Bit-Rate Video-Coding Focusing on Moving Objects," Proc. of Int. Workshop on Knowledge Management Technique, Crema, Italy, September 16–18, Vol-2, 1560-1564, 2002.

[12] Paul, M., M. Murshed, and L. Dooley, "Impact of Macroblock classification on low bit rate video coding focusing on moving region," IEEE Proc. of Conference of Computer and Information Technology, Dhaka, Bangladesh, Dec 2002.

[13] Shi, Y.Q. and H. Sun, *Image and Video Compression for Multimedia Engineering Fundamentals, Algorithms, and Standards*, CRC Press, 1999.

[14] Wong, K.-W., K.-M. Lam, and W.-C. Siu, "An Efficient Low Bit-Rate Video-Coding Algorithm Focusing on Moving regions," *IEEE trans. circuits and systems for video technology*, 11(10), 1128–1134, 2001.

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

[16] Lin, S., and B. W. Kernighan, " An effective Heuristic Procedure for the Traveling-Salesman Problem," *OR*, 21 (1973), 498-516.

Table I: Relative bit-rate and PSNR for six standard video sequences using 3-Opt, H.263 standard, Fixed 8, and the GVPS algorithm

Video sequences	Video format	3-Opt		H.263		Fixed 8		GVPS (λ= 16, γ = 90)					
								$\delta = 64$		$\delta = 96$		$\delta = 128$	
		Bit-rate (kbps)	PSNR (dB)	Bit-rate (kbps)	PSNR (dB)	Bit-rate (kbps)	PSNR (dB)	Bit-rate (kbps)	PSNR (dB)	Bit-rate (kbps)	PSNR (dB)	Bit-rate (kbps)	PSNR (dB)
Miss America	QCIF (176 × 144)	12.12	42.92	13.94	42.36	13.40	42.51	12.77	43.00	12.21	42.92	12.15	42.90
Car phone	QCIF (176 × 144)	29.00	35.32	31.42	35.16	30.87	35.18	30.72	35.48	29.31	35.32	29.06	35.29
Foreman	QCIF (176 × 144)	33.97	33.30	35.46	33.17	35.90	33.18	35.88	33.46	34.38	33.30	34.06	33.25
Salesman	CIF (352 × 288)	119.86	37.16	133.60	36.96	129.21	36.94	123.21	37.24	120.29	37.19	119.95	37.15
Tennis	CIF (352 × 240)	149.58	30.72	139.96	30.38	151.71	30.57	151.94	30.83	150.26	30.76	149.62	30.72
Claire	CIF (352 × 288)	30.95	43.36	35.08	43.07	32.29	43.13	32.05	43.41	31.34	43.40	31.20	43.39