References

1 EAKINS, J. *et al.*: 'Content-based image retrieval'. JISC Technology Application Program Report: 39, 1999

2 SCHMID, C., and MOHR, ROGER: 'Local grayvalue invariants for image retrieval', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1997, **19**, (5), pp. 530–535

3 HARRIS, C., and STEPHENS, M.: 'A combined corner and edge detector'. Proc. 4th Alvey Vision Conf., Manchester, United Kingdom, 1998, pp. 147–151

4 KHOTANZAD, A., and HONG, Y.H: 'Invariants image recognition by Zernike moments', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1990, **12**, (5), pp. 489–497

Video coding using greedy decompositions on generalised bit-planes

R. Caetano, E.A.B. da Silva and A.G. Ciancio

A novel method for performing greedy decompositions using generalised bit-planes is proposed. It provides an elegant solution to the trade-off between quantisation of coefficients and number of passes in the matching pursuits algorithm. In addition, when replacing the matching pursuits algorithm in video coding, it provides a significant performance improvement.

Introduction: The classical algorithms used in video coding are based on the block discrete cosine transform (DCT). An effective alternative for such methods is given by greedy decompositions over redundant dictionaries using the matching pursuits (MP) algorithm [1]. An efficient video encoder using the MP algorithm has been presented by Neff and Zakhor [2]. It provides good coding efficiency and is free from blocking artefacts. Its success has encouraged research on this topic.

In the MP algorithm we usually decompose a signal $\mathbf{x}$ of dimension N on a redundant dictionary $\mathcal{D}=\{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_M\}$, $\|\mathbf{g}_i\|=1$, $\forall i$. The $\mathbf{g}_i$ are in general referred to as atoms. The dictionary is said to be redundant because, in general, $M>N$. The signal $\mathbf{x}$ is then approximated in P passes as [1]

$$\mathbf{x} \simeq \sum_{n=1}^{P} p_n \mathbf{g}_{\gamma_n} \tag{1}$$

The index γ_1 corresponds to the atom in the dictionary over which the projection of $\mathbf{x}$ is largest, with p_1 being the value of this projection. We then compute the residual $\mathbf{r}_1=\mathbf{x}-p_1\mathbf{g}_{\gamma_1}$ and find the pair (p_2, γ_2) corresponding to the largest projection of $\mathbf{r}_1$ on every atom of $\mathcal{D}$. From these we compute $\mathbf{r}_2=\mathbf{r}_1-p_2\mathbf{g}_{\gamma_2}$. This process is repeated recursively until we reach the number of passes P that meets a predefined rate and/or distortion criterion.

From this we see that the MP algorithm performs a kind of successive approximation of a signal $\mathbf{x}$, since, for each atom added, the error in the approximation decreases [1]. Therefore, in principle, the approximation error can be controlled by the number of atoms used. However, when the coefficients p_n are quantised, the approximation error also depends on how the quantisation is performed. Several strategies have been proposed for dealing with this problem, as, for example, the adaptive modulus quantiser proposed in [3], and the one in [4], based on R-D optimisation.

In this Letter we propose a novel algorithm to perform an MP-like greedy decomposition in which a signal is decomposed in generalised bit-planes, each generalised bit-plane being composed of a set of atoms. In it, unlike the classical MP algorithm, there are no coefficients to be quantised, since only the atoms corresponding to each generalised bit-plane need to be transmitted.

Signal decomposition in generalised bit-planes: In [5] a convergent greedy algorithm was proposed for finding a generalised bit-plane decomposition of a signal based on the same philosophy as the MP algorithm. There, a signal $\mathbf{x}$ was decomposed as

$$\mathbf{x}=\sum_{j=1}^{\infty}\alpha^j\sum_{l=1}^{L_j}\bar{\mathbf{g}}_{i_{j,l}} \tag{2}$$

where $\bar{\mathbf{g}}_n \in \bar{\mathcal{D}}=\{\pm\mathbf{g}_1, \pm\mathbf{g}_2, \ldots, \pm\mathbf{g}_M\}$ and $0<\alpha<1$.

The generalised bit-plane j is composed of the functions $\bar{\mathbf{g}}_{i_{j,l}}$ for $l=1, \ldots, L_j$. The only other condition imposed for the convergence was $\Theta(\bar{\mathcal{D}})\leq\pi/3$ where $\Theta(\bar{\mathcal{D}})$ is the largest angle between any signal $\mathbf{x}\in\mathbb{R}^N$ and the closest atom in dictionary $\bar{\mathcal{D}}$. However, even for signals of moderate dimension (e.g. $N\geq 64$), the dictionaries that could provide $\Theta(\bar{\mathcal{D}})\leq\pi/3$ would have very large cardinality. This would lead to inefficient decompositions from an R-D perspective, since a large number of bits would be needed to encode each index $i_{j,l}$.

In this Letter we propose a novel greedy algorithm for performing the decomposition in (2) that does not have the limitation of $\Theta(\bar{\mathcal{D}})\leq\pi/3$. The algorithm is as follows.

Algorithm 1:

1. Start with $\mathbf{w}=\mathbf{x}$, $m=1$.
2. Repeat until a stop criterion is met
(a) Choose $r_m\in\{1, \ldots, q\}$ such that

$$\mathbf{w}\cdot\mathbf{v}_{r_m}=\max_{1\leq j\leq q}\{\mathbf{w}\cdot\mathbf{v}_j\}.$$

(b) Choose

$$k_m=\left\lfloor\frac{\ln(\mathbf{w}\cdot\mathbf{v}_{r_m})}{\ln(\alpha)}\right\rfloor$$

where $\lceil\mathbf{y}\rceil$ is the smallest integer larger than or equal to $\mathbf{y}$.
(c) Replace $\mathbf{w}$ by $\mathbf{w}-\alpha^{k_m}\mathbf{v}_{r_m}$.
(d) Increment m.
3. Stop.

Note that Algorithm 1 approximates $\mathbf{x}$ in P passes as

$$\mathbf{x}^{(P)}=\sum_{m=1}^{P}\alpha^{k_m}\mathbf{v}_{r_m} \tag{3}$$

If we define L_j as the number of values m such that $k_m=j$, we can rename the corresponding indexes r_m as $i_{j,l}$ for $l=1, \ldots, L_j$. Therefore, if we make the dictionary $\mathcal{C}$ in Algorithm 1 equal to $\bar{\mathcal{D}}$, then (3) is equivalent to (2) for $P\rightarrow\infty$.

We can say that Algorithm 1 is convergent if $\lim_{P\rightarrow\infty}\mathbf{x}^{(P)}=\mathbf{x}$. In this sense, its convergence is guaranteed by Theorem 1.

Theorem 1: Be $\mathbf{x}\in\mathbb{R}^N$, $\|\mathbf{x}\|\leq 1$, such that it is approximated by Algorithm 1 using a dictionary $\mathcal{C}$ with P steps, generating $\mathbf{x}^{(P)}$ as in (3), and $\Theta(\mathcal{C})$ is the largest angle between any signal $\mathbf{y}\in\mathbb{R}^N$ and the closest atom in dictionary $\mathcal{C}$. We have that $\|\mathbf{r}^{(M)}\|=\|\mathbf{x}-\mathbf{x}^{(M)}\|\leq\beta^M$, where $\beta=\sqrt{[1-(2\alpha-\alpha^2)\cos^2(\Theta(\mathcal{C}))]}<1$ for every $0<\alpha<1$ and $0\leq\Theta(\mathcal{C})<\pi/2$.

Note that convergence is guaranteed because, since $\beta<1$, then $\lim_{M\rightarrow\infty}\mathbf{r}^{(M)}=0$. The representation output by Algorithm 1 is given by just a sequence of pairs of indexes (k_m, r_m), $m=1, 2, \ldots, P$. This implies that there is no need for quantisation of coefficients as in the classical MP algorithm. In other words, the decomposition and quantisation operations are merged, and cannot be carried out separately. Thus, it presents an elegant solution to the coefficient quantisation problem inherent in the classical MP algorithm.

Implementation of video encoder: The effectiveness of Algorithm 1 was evaluated by employing it in the framework of Neff and Zakhor's MP video encoder [2]. Essentially, Algorithm 1 replaced the decomposition and quantisation strategy employed in [2], using exactly the same dictionary $\mathcal{D}$, as well as the same atom encoding procedure. In our case, instead of encoding the value of the quantised inner product p_n, we encoded the index k_m of the bit-plane corresponding to the atom of index r_m. An adaptive arithmetic coder [6] was used for this purpose. Besides, we needed to send, for each video frame, the largest norm of the macroblocks, $X_{\max}$. Since the atoms can be centred anywhere inside a macroblock, there is a great deal of overlap with its neighbouring macroblocks. Then, $X_{\max}$ was computed considering 50×50 windows centred in every macroblock. We also needed to send the approximation scaling factor (α) at the header of the video sequence. The strategy used for bit-rate allocation was to divide the bit-budget of the sequence equally among all its frames.

Experimental results: We have coded the sequences Container, Hall-monitor, Mother-and-daughter, Silent-voice and Foreman with 300

QCIF frames at 30 frame/s, sub-sampled in time by factors of 4 (rates under 20 kbit/s) and 3 (other rates) to generate 7.5 frame/s and 10 frame/s, respectively. Coding was performed only on luminance component in bit-rates that vary in the range 10–100 kbit/s.

Note that in (2) and (3), there is a dependency on the parameter α. We have verified experimentally that, for α in the range [0.5, 0.85], the PSNR performance is quite insensitive to α. In our experiments, we have used an $\alpha = 0.56$ for all cases.

Table 1 compares the average peak signal-to-noise ratio (PSNR) of the original matching pursuits video encoder (MP) [2] with our adaptation using generalised bit-planes (MPGBP) for some rates.

Table 1: Comparison, in terms of PSNR, between two matching pursuits implementations

Seq + Rate	MPGBP	MP [2]	MPGBP-MP
Container10	32.54	32.45	0.06
Mother10	33.42	33.35	0.07
Hall10	33.43	33.30	0.13
Container24	34.70	34.47	0.23
Mother24	36.39	36.18	0.21
Hall24	36.59	36.13	0.46
Silent24	32.77	32.73	0.04
Container48	36.95	36.43	0.52
Mother48	39.21	38.45	0.76
Hall48	39.14	38.00	1.14
Silent48	36.35	35.90	0.45
Mother96	42.27	41.02	1.25
Foreman96	35.54	35.35	0.19

Fig. 1 shows the variation of the average PSNR with rate for both implementations of the matching pursuits encoders. We can see from this Figure that the use of the generalised bit-planes scheme consistently improves the performance of the matching pursuits encoder from [2] for all rates. In addition, this improvement increases with the bit rate. Indeed, our results are compatible with the best ones in the literature, that have been obtained using sophisticated adaptive strategies [3]. Note that the knee on the curves around 20 kbit/s is due to the increase of the frame rate from 7.5 to 10 frame/s.

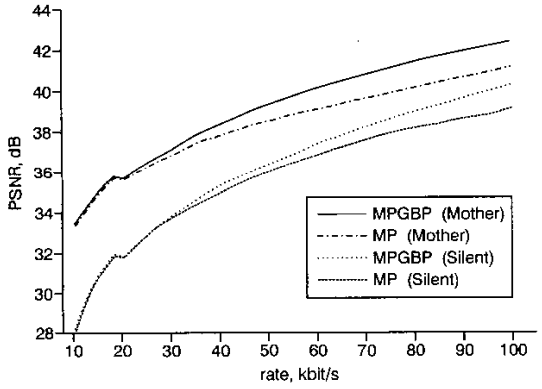

Fig. 1 *Variation of average PSNR with rate for Mother and Silent sequences*

Conclusions: We have proposed a novel algorithm for performing greedy decompositions on redundant dictionaries. Instead of generating at its output a sequence of pairs comprising indexes of atoms and corresponding coefficients, as in the classical MP algorithm, it only generates a sequence of indexes of atoms. The results obtained are very promising, yielding a significant improvement over the classical MP-based video compression algorithm [2].

31 January 2002
Electronics Letters Online No: 20020357
DOI: 10.1049/el:20020357

R. Caetano, E.A.B. da Silva and A.G. Ciancio (*PEE/COPPE/DEL/EE/Universidade Federal do Rio de Janeiro, Cx. P. 68504, Rio de Janeiro, RJ, 21945-970, Brazil*)

E-mail: eduardo@lps.ufrj.br

References

1 MALLAT, S.G., and ZHANG, Z.: 'Matching pursuits with time-frequency dictionaries', *IEEE Trans. Signal Process.*, 1993, **41**, pp. 3397–3415
2 NEFF, R., and ZAKHOR, A.: 'Very low bit rate video coding based in matching pursuits', 1997, *IEEE Trans. Circuits Syst.*, **7**, pp. 158–171
3 NEFF, R., and ZAKHOR, A.: 'Modulus quantization for matching pursuits video coding', *IEEE Trans. Circuits Syst. Video Technol.*, 2000, **10**, pp. 895–912
4 VANDERGHEYNST, P., and FROSSARD, P.: 'Adaptive entropy-constrained matching pursuits quantization'. IEEE Int. Conf. on Image Processing, 2001, pp. 423–426
5 CRAIZER, M., DA SILVA, E.A.B., and RAMOS, E.C.: 'Convergent algorithms for successive approximation vector quantisation with applications to wavelet image compression', *IEE Proc., Vis. Image Signal Process.*, 1999, **146**, pp. 159–164
6 BELL, T.C., CLEARY, J.G., and WITTEN, I.H.: 'Text compression' (Prentice Hall, Englewood Cliffs, NJ, 1990)

Classes of impossible differentials of advanced encryption standard

R.C.W. Phan

Recently, a class of generalised four-round impossible differentials of the advanced encryption standard (AES) was presented. The previous work is extended and applies more flexibility to construct two new classes of impossible differentials of the AES

Introduction: The advanced encryption standard (AES) is a 128-bit block cipher with a 128-, 192- or 256-bit secret key [1]. A data block of the AES is expressed as an array of 4×4 bytes with row and column indices, i, j {0, 1, 2, 3}. The input block is passed through a round function which is iterated 10 times. At the same time, the 128-bit secret key is input to a key schedule to obtain round keys for use in each round. Each round function consists of SubBytes, a nonlinear 8×8 S-box byte substitution; ShiftRows, a cyclic shift of each row by different byte offsets; MixColumns, a linear combination of all 4 bytes in the same column; and KeyAddition, an exclusive-OR (XOR) of the data block with the round key. Each round is identical except that an extra KeyAddition is added before the first round and MixColumns is excluded from the last round.

XOR patterns and truncated differentials: Let a pair of AES data blocks P and P* differ in certain (active) byte positions and are equal in other (passive) bytes.

Definition 1: An XOR pattern is a 4×4 array that specifies the active and passive byte positions of a pair of AES data blocks P and P*

Consider the influence of the round function components on the distribution of the active bytes in the XOR pattern. SubBytes operates on each byte independently hence it does not affect the XOR pattern. KeyAddition does not affect the XOR pattern either because XORing twice with the round key cancels out its effect. ShiftRows only shifts an active byte to another position in the same row but does not diffuse it over to other byte positions. MixColumns causes an active byte to spread to all four byte positions in the same column. The input XOR pattern and its corresponding output XOR pattern after i rounds of the AES are collectively known as an i-round truncated differential.

It is obvious that MixColumns greatly influences the behaviour of truncated differentials as it causes the sole diffusion of active bytes. Since MixColumns operates on each column of the data block independently, it is sufficient to consider the XOR patterns of these individual columns, which are called the column XORs. The distribution of the individual input and output column XORs of MixColumns is given in Table 1 [2]. The '1' in the column XOR denotes an active byte while a '0' denotes a passive byte.

Phan and Siddiqu's impossible differentials: In [2], Phan and Siddiqi constructed a class of generalised 4-round impossible differentials of the AES by concatenating two probability-one truncated differentials such that they form a contradiction [3] in the middle, hence causing