

CLASS-ADAPTED IMAGE COMPRESSION USING INDEPENDENT COMPONENT ANALYSIS

Artur J. Ferreira

Institute of Telecommunications, and
Dept. of Electronics, Telecom., and Comp. Eng.
Instituto Superior de Engenharia de Lisboa
1949-014 Lisboa, Portugal
arturj@cc.isel.ipl.pt

Mário A. T. Figueiredo

Institute of Telecommunications, and
Dept. of Electrical and Computer Eng.
Instituto Superior Técnico
1049-001 Lisboa, Portugal
mtf@lx.it.pt

ABSTRACT

This paper exploits *independent component analysis* (ICA) to obtain transform-based compression schemes adapted to specific image classes. This adaptation results from the data-dependent nature of the ICA bases, learnt from training images. Several coder architectures are evaluated and compared, according to both standard (SNR) and perceptual (*picture quality scale* – PQS) criteria, on two classes of images: faces and fingerprints. For fingerprint images, our coders perform close to the well-known special-purpose wavelet-based coder developed by the FBI. For face images, our ICA-based coders clearly outperform JPEG at the low bit-rates herein considered.

1. INTRODUCTION

Independent component analysis (ICA) considers a class of probabilistic generative models in which a random vector \mathbf{X} is obtained according to $\mathbf{X} = \mathbf{AS}$, where \mathbf{A} is an $N \times M$ unknown mixing matrix and \mathbf{S} is a vector of independent sources [2, 6, 9]. The standard goal of ICA is to infer (learn) \mathbf{A} from a set of samples of \mathbf{X} . To apply ICA to images, each sample of \mathbf{X} usually contains the pixels in an image block.

It is known that natural images are well modelled when the columns of \mathbf{A} are wavelet-like (or Gabor-like) and the independent sources (elements of \mathbf{S}) have super-Gaussian (also called sparse) distributions [1, 6, 9]. Specially in the case of over-complete ICA ($M > N$), this sparse nature of the distribution of \mathbf{S} means that only a small number of its components have significant values; this fact underlies the potential usefulness of ICA to compression and denoising of natural images [6, 9]. Despite this, few attempts have been made at using ICA for image compression [12].

In this paper, we exploit the data-dependent nature of the ICA decomposition (onto the basis defined by the columns

of \mathbf{A}) to develop low bit-rate compression schemes adapted to specific image classes. We use Hyvärinen's FastICA algorithm (see [6, 7] for details) to learn complete and over-complete bases from training images. Since these bases are non-orthogonal, we apply variants of the *matching pursuit* (MP) algorithm [10] to perform image decomposition.

The paper is organized as follows. Section 2 describes how the basis vectors are extracted from faces and fingerprint images using ICA. In Section 3, we review orthogonal and non-orthogonal matching pursuit algorithms, and present their energy compaction ability on the ICA bases. The coder architecture is described in Section 4, while Section 5 shows experimental results. Finally, Section 6 presents some concluding remarks.

2. BASIS ESTIMATION

We apply the FastICA algorithm [6, 7] to estimate both complete ($N = M$) and over-complete ($M > N$) bases for randomly selected training sets of (8×8) image blocks (400 per image), after mean removal and sphering by principal component analysis [6]. In this paper, we consider a set of fingerprint images¹ and a set of face images². Fig. 1

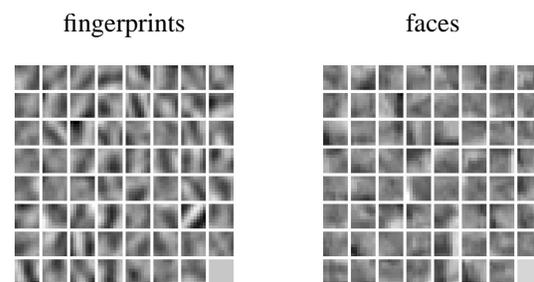


Fig. 1. Complete bases obtained by FastICA.

This work was partially supported by the (Portuguese) Foundation for Science and Technology (FCT), grant POSI/33143/SRI/2000.

¹bias.csr.unibo.it/fvc2000/databases.asp

²www.uk.research.att.com/facedatabase.html

shows the vectors of complete ($M = N = 64$) bases obtained from one fingerprint and one face image; notice the more pronounced edge-like nature of the fingerprint basis. Observation of histograms of relative angles between all the ICA basis vectors reveals that most angles are above 40° . Since these bases are non-orthogonal, representations can not be obtained by the standard orthogonal projection procedure. Alternatively, we consider the use of matching pursuit algorithms, described in the next section.

3. MATCHING PURSUIT

Matching pursuit (MP) is a greedy iterative algorithm that approximates a signal by successive projections on the vectors of a (possibly over-complete) basis [10]. Formally, let $D = \{f_1, \dots, f_M\}$ be a dictionary with M unit vectors ($\|f_i\| = 1$) on an N -dimensional Hilbert space \mathcal{H} with inner product $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \rightarrow \mathcal{R}$. Given some function $g \in \mathcal{H}$, MP obtains a sequence of linear representations

$$\hat{g}^n = \sum_{i=1}^n \alpha_i f_i, \quad n = 1, 2, \dots$$

by applying the following steps.

Step 0: Let $n = 0$ and $\hat{g}^0 = 0$.

Step 1: Compute $R_g^n = g - \hat{g}^n$, the residue of the representation with n terms.

Step 2: Choose the index of the next basis vector to include in the representation according to

$$\delta_n = \arg \max_{\delta \in \{1, \dots, M\}} |\langle R_g^n, f_\delta \rangle|.$$

Step 3: Update the representation,

$$\hat{g}^{n+1} = \hat{g}^n + \alpha_{n+1} f_{\delta_n},$$

where $\alpha_{n+1} = \langle R_g^n, f_{\delta_n} \rangle$.

Step 4: Check stopping condition; if it is not verified, let $n \leftarrow n + 1$ and go back to Step 1.

The stopping condition depends on the particular application; usually, it is of the type $\|R_g^n\| \leq d$, where d is a threshold. The residue energy $\|R_g^n\|_2^2$ converges exponentially to zero, if the dictionary is at least complete [3, 10].

Orthogonal matching pursuit (OMP) [3] is a variant of MP, in which Step 3 is redefined to be

Step 3: Update the representation,

$$\hat{g}^{n+1} = \hat{g}^n + \alpha_{n+1} u_{n+1},$$

where

$$u_{n+1} = \frac{f_{\delta_n} - \sum_{p=1}^n \langle f_{\delta_n}, u_p \rangle u_p}{\|f_{\delta_n} - \sum_{p=1}^n \langle f_{\delta_n}, u_p \rangle u_p\|},$$

and $\alpha_{n+1} = \langle R_g^n, u_{n+1} \rangle$.

In words, each selected basis vector, before being included in the representation, is orthogonalised with respect to all previously selected basis vectors, and then normalised. All the other steps of the algorithm remain unchanged. The orthogonalisation procedure guarantees convergence in a maximum of N iterations [3].

3.1. Energy Compaction

In this section, we compare the MP and OMP algorithms in terms of energy compaction, that is, in terms of their ability to extract good representations with as few as possible terms. To quantify the goodness of the representations, we consider the standard SNR measure

$$SNR_n = 10 \log_{10} (\sigma^2 / MSE_n) \quad [dB],$$

where σ^2 is the original image variance, and MSE_n is the mean squared error between the original image and its block by block n -term representation.

Fig. 2 shows SNR_n , for MP and OMP with complete and over-complete bases extracted from a set of three images (not including the one being compressed). The results of the *Karhunen-Loève transform* (KLT) [8] are also displayed for comparison, since, among all orthogonal transforms, the KLT has the highest energy compaction ability.

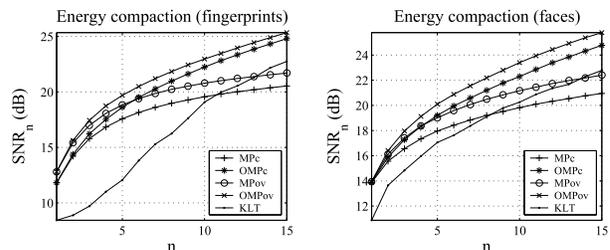


Fig. 2. Energy compaction for image representations obtained with MP and OMP on complete (suffix “c”) and over-complete (“ov”) ICA bases.

These results show that, for both image classes, an over-complete basis yields up to 2 dB more energy compaction than the complete basis. For the same basis, in the first few iterations, OMP and MP produce roughly the same results; that is, MP chooses nearly orthogonal basis vectors without the need for explicit orthogonalisation (see also [3]). Due to its larger coding and decoding complexity (compared to MP), OMP will thus not be further considered. The comparison with KLT suggests that MP expansions will be advantageous in the case of a small number of coefficients.

3.2. Coding/Decoding Complexity: Incomplete Bases

Consider a set of M vectors from an N -dimensional space. To represent (*i.e.*, code) a vector (*i.e.* an image block) with

respect to this set, each iteration of the MP algorithm involves MN scalar products (M inner products of N -dimensional vectors). If each block is coded with L coefficients, the total number of scalar multiplications is LMN . To re-synthesize (*i.e.*, decode) the block, LN scalar multiplications are performed. Thus, reducing the cardinality M of the basis, reduces linearly the coding complexity. The same applies to L with respect to both coding and decoding. These facts suggest that it would be advantageous to use an incomplete ($M < N$) basis; notice also that we typically have $L \ll M$.

To obtain an incomplete basis, with $M < N$, we have devised the following procedure. We start by obtaining an over-complete basis with $M' > N$ vectors. Each of these vectors receives one vote for each time it is chosen by the MP algorithm to represent one of the training blocks. The M most voted vectors constitute the incomplete basis.

4. CODER ARCHITECTURE

The proposed image coder is transform-based, as shown in Fig. 3. The transform coefficients are obtained by MP over the (complete, over-complete, or incomplete) ICA basis. Operation modes with fixed and variable block sizes are considered. Two methods to code the coefficients are studied: sending only the non-zero coefficients and the corresponding indexes (this option is represented by the dashed lines in Fig. 3); sending all M coefficients, regardless of their being zero or not. The mean value of each block is separately quantised (Lloyd I) and entropy-coded.

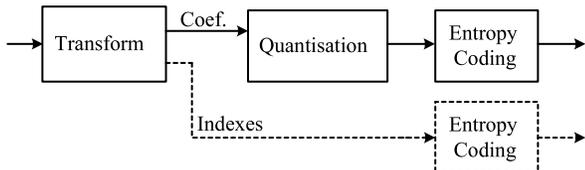


Fig. 3. Non-orthogonal transform-based coder architecture.

Coefficient quantisation is performed using a Lloyd I quantiser, learnt off-line from the MP coefficients. Entropy coding of the quantiser output and of the indexes is carried out by adaptive arithmetic coders, using source models (histograms obtained off-line from several test images of the specific class being considered). In the coding method with indexes, the first coefficient of each block is quantised with a larger number of bits and entropy-coded separately using an arithmetic coder.

4.1. Fixed and Variable Size Blocks

We first consider blocks with fixed size (8×8). Each block is encoded with a variable number of coefficients; this number is selected as the smallest that guarantees that the maximum absolute difference between the original and the coded

block does not exceed a predefined threshold Δ . Blocks for which this criterion can not be met, are coded with a predefined maximum number of coefficients L_{\max} . Several tests showed that most image blocks require less than L_{\max} coefficients. Consequently, this method reduces the coding complexity and the bit rate, when compared to the use of a fixed number of coefficients.

In the case of variable size blocks, image analysis is performed using blocks of sizes 16×16 , 8×8 , and 4×4 , organised in a quad-tree structure [8]. Splitting each 16×16 or 8×8 block into its four sub-blocks is done when the “maximum absolute difference” (referred in the preceding paragraph) exceeds a given value. The resulting tree decomposition is encoded using an adaptive arithmetic coder.

5. EXPERIMENTAL RESULTS

5.1. Face Images

In our experiments with face images, we use $L_{\max} = 10$, $\Delta = 24$, and fixed 8×8 block size. We consider incomplete, complete, and over-complete bases, obtained from one or three images, and the two coding methods described: without indexes (with 5 bits/coefficient); with indexes (5 bits for the first coefficient and 4 bits for the remaining ones). In both cases, the block mean value is quantised with 5 bits.

Fig. 4 shows SNR as a function of bit-rate. The plot on the left hand side refers to bases extracted from one single image, while the one on the right corresponds to bases extracted from three images. For comparison, we also include JPEG results, since it is the standard (8×8) block coder.

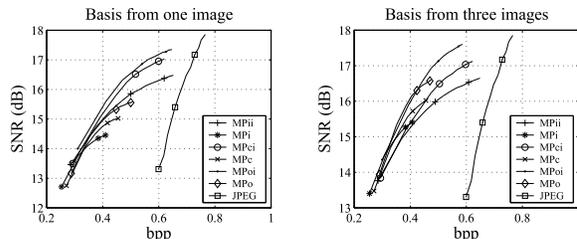


Fig. 4. Face image tests: SNR versus bit-rate. Legend: MP_{ai} , where $a \in \{i, c, o\}$ denotes the type of basis (incomplete, complete, over-complete), and the presence of “i” indicates the use of the coding method with indexes.

There is no meaningful performance difference between the ICA bases obtained from just one or from three images. The proposed coder clearly outperforms JPEG for the low bit-rates considered. Fig. 5 shows a face image coded with JPEG and with the proposed coder (using an incomplete basis of 32 vectors and index-free coding), and the corresponding values of two distortion measures: SNR and *picture quality scale* (PQS), which is based on a model of the

SNR=13.7 dB PQS= -3.7



JPEG

SNR=16.0 dB PQS= -2.3



ICA

Fig. 5. Coded images with JPEG, and ICA at 0.62 bpp.

SNR=13.2 dB PQS= 1.9



WSQ

SNR=11.7 dB PQS= 1.4



ICA

Fig. 6. Coded images with WSQ, and ICA at 0.31 bpp.

human visual system [11]. In addition to better SNR and PQS, the ICA-coded image is clearly less blocky.

5.2. Fingerprint Images

WSQ (*wavelet scalar quantisation* [4]) is a wavelet-based special-purpose coder for fingerprint images. In tests similar to those reported in Fig. 4, WSQ outperforms the ICA-based coders by $2 \sim 4$ dB. The variable block-size coder (Sec. 4.1), with a complete basis learnt from 3 images, yields ~ 2 dB lower distortion than the fixed block-size coder. Fig. 6 shows images compressed with WSQ and the variable block-size ICA coder. Although the ICA-coded image has slightly worse SNR and PQS values than the WSQ-coded image, they are visually indistinguishable. Thus, our approach was able to *learn* a coder which is competitive with a method specially tailored to a specific image class.

6. CONCLUDING REMARKS

We have shown how to exploit the data-dependent nature of ICA to obtain low bit-rate transform-based compression schemes adapted to specific image classes (concretely, face and fingerprint images). Several image representation bases and coder architectures, supported on *matching pursuit* [10], were evaluated and compared against standard coders (JPEG for face images, and WSQ [4] for fingerprints).

Incomplete, complete, and over-complete bases, learnt from one or several images, were shown to produce roughly the same results. This suggests that, for specific classes, it is not worth to use over-complete bases. For face images, at low bit-rates, the 8×8 fixed block-size coder, using an incomplete basis, yields better SNR and PQS than JPEG, and also less blocky images. For fingerprint images, the variable block-size coder, using a complete basis, has performance close to that of the special-purpose WSQ coder. Elsewhere [5], we show that orthogonalised ICA bases perform similarly to WSQ over a wide range of bit-rate values.

7. REFERENCES

- [1] A. Bell and T. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37:3327–3338, 1997.
- [2] P. Comon. Independent component analysis: A new concept? *Signal Proc. Elsevier*, 36(3):287–314, 1994.
- [3] G. Davis. *Adaptive Nonlinear Approximations*. PhD thesis, Courant Institute of Mathematical Sciences, New York University, 1994.
- [4] Federal Bureau of Investigation. *WSQ Gray-Scale Fingerprint Image Compression Specification*. IAFIS-IC-0110v2 (rev. 2.0), 1993.
- [5] A. Ferreira. *Image Compression Using Independent Component Analysis*. MSc thesis, Instituto Superior Técnico, Technical University of Lisbon, 2002.
- [6] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley Interscience, 2001.
- [7] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.
- [8] A. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 3rd edition, 1989.
- [9] T.-W. Lee. *Independent Component Analysis - Theory and Applications*. Kluwer Academic Publishers, 1998.
- [10] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. on Signal Processing*, 41(12):3397–3415, 1993.
- [11] M. Miyahara, K. Kotani, and V. Algazi. Objective picture quality scale (PQS) for image coding. *IEEE Trans. on Comm.*, 46(9):1215–1226, 1998.
- [12] A. Puga and A. Alves. An experiment on comparing PCA and ICA in classical transform image coding. In *ICA99*, pages 105–108, Aussois, 1999.