Random time-frequency Subdictionary design for sparse representation with greedy algorithms

Manuel Moussallam, Laurent Daudet, Gaël Richard

RANDOM TIME-FREQUENCY SUBDICTIONARY DESIGN FOR SPARSE REPRESENTATIONS WITH GREEDY ALGORITHMS

Manuel Moussallam[1,2] *, Laurent Daudet*[2]*, Gaël Richard*[1]

[1]Institut Telecom - Telecom ParisTech - CNRS/LTCI
37/39, rue Dareau 75014 Paris, France

[2]Institut Langevin - ESPCI ParisTech - UMR7587
10 rue Vauquelin, 75005 Paris, France

ABSTRACT

Sparse signal approximation can be used to design efficient low bit-rate coding schemes. It heavily relies on the ability to design appropriate dictionaries and corresponding decomposition algorithms. The size of the dictionary, and therefore its resolution, is a key parameter that handles the tradeoff between sparsity and tractability. This work proposes the use of a non adaptive random sequence of subdictionaries in a greedy decomposition process, thus browsing a larger dictionary space in a probabilistic fashion with no additional projection cost nor parameter estimation. This technique leads to very sparse decompositions, at a controlled computational complexity. Experimental evaluation is provided as proof of concept for low bit rate compression of audio signals.

Index Terms— Matching Pursuits, Random Subdictionaries, Sparse Audio Coding

1. INTRODUCTION

Randomness has proven surprisingly useful in a wide variety of computational and statistical fields. In communications, spread spectrum techniques, where a signal is modulated by a random binomial sequence before transmission allows for better bandwidth management. Quantization has long taken advantage of the dithering technique that uses randomness to avoid perceptually disturbing artefacts linked to the quantization noise. More generally, stochastic resonance theory has shown how a moderate amount of added noise can increase the average behavior of many non-linear systems. More recently, tremendous work has been achieved on the compressive sampling scheme, making use of random measurement matrices. Behind all these examples lies a common intuition: controlling the random part of a system is better than having to deal with colored measurement noise or transmission errors, or signal-dependent deviations. The key point is to spread the information where it can be efficiently found. Having to guess where discriminant low-level features are hidden in huge-dimensional spaces is too costly or simply not feasible. In such cases, randomness can be used as a powerful sieve by information miners.

At the opposite of uniform random distributions is the concept of sparsity. Sparse coding of digital signals has been the subject of many works in the past few years for audio [9], images [4] or video streams. Low bitrate coders have been designed and proved to be competitive with state of the art industrial solutions. The core idea is to decompose an original signal f as a combination of (a few) objects from a dictionary Φ of indexed elementary waveforms. In a coding framework, the coder has to transmit the set of indexes of the non-zero coefficients, together with their quantized values. Here, a crucial yet often underestimated issue is the choice of the size of Φ, that always resorts to a tradeoff. If the dictionary is small (slight or no overcompleteness), computations are fast, the index coding cost per coefficient is low, but many coefficients may be needed. If it is large (i.e., dense in the parameter space), we have a fast decay of the approximation error as a function of the approximation order, but the cost of encoding the indexes is higher, and computations get cumbersome. Indeed, the computational complexity associated with these sparse techniques, as opposed to suboptimal but much simpler transform-based coders, is probably the main limitation to their widespread use in practical applications. Strategies have been proposed to lower the computational cost of using large dictionaries, based on local adaptation of the selected atoms [5] or probabilistic approaches [3] where successive runs with random sub-optimal atom selection are performed, then averaging yields a robust sparse approximation. Yet, these approaches are still associated with high index coding costs, if the atoms parameter space is large.

In this work, we propose a different paradigm that mitigates the drawbacks of using a large dictionary while keeping most of the benefits. Based on the algorithm described in [8], a single run is performed using varying subdictionaries. These subdictionaries have limited size, but are designed so as to evenly span a much larger dictionary space. In this work, we use a simplistic audio coding example as a proof of concept to demonstrate the usefulness of randomization for sparse representation problems. The key point in our technique is that the choice of subdictionary is *not* adaptive, but is parameterized by a fixed pseudo-random sequence, also known by the decoder. In other words, we have the (theoretical) complexity of working with a small dictionary, and the small coding costs, but the whole large dictionary is spanned. It should be emphasized that, unlike a compressive sensing framework, we our goal is to design a 'standard' coding scheme with maximal efficiency at the cost of computational complexity at the encoder, and minimal decoding complexity at the decoder.

The rest of this paper is organized as follows. Section 2 recalls the Matching Pursuit framework for compression. In Section 3, the novel approach is presented along with considerations on the random sequence design, and in Section 4 trivial audio compression serves as a proof of concept for the suitability of the proposed method.

2. SIGNAL COMPRESSION WITH GREEDY ALGORITHMS

Let f be a discrete signal living in $\mathbb{R}^N$. Greedy algorithms iteratively decompose f using m elements from a dictionary $\Phi = \{\phi_\gamma\}$ of elementary objects called *atoms* by alternating two steps 1: Select an atom in the dictionary and 2: Update approximant and residual.

LD is on a joint position between Univ. Paris Diderot and Institut Universitaire de France

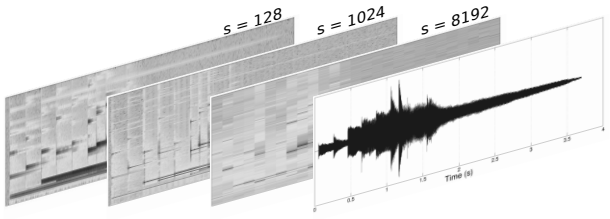

Fig. 1. *Projection of a glockenspiel signal quantized with 3 different time frequency lattices* $\sharp_s$ *for* $s \in [128, 1024, 8192]$.

The resulting representation $f_m = \alpha.\Phi_{\Gamma^m}$ is a (usually suboptimal) solution to the NP-hard problem (1), using the subset Γ^m of columns of Φ

$$\min m \text{ w.r.t. } \delta(f, f_m) \leq \epsilon \tag{1}$$

where $\delta(f, f_m)$ is a distorsion measure, usually in the form of a quadratic reconstruction error. Ideally, the dictionary is chosen to be related to the signal's characteristics. Speech can be efficiently compressed using carefully designed codebooks, 2D wavelets bases are useful for image compression. Music representation is better achieved using dictionaries of windowed cosines (MDCT) or gabor functions. The algorithm selects at iteration i the atom that maximizes a correlation function, usually an energy criterion:

$$\phi^{\star}_{\gamma^i} = \arg\max_{\phi_\gamma \in \Phi} |\left\langle R^i f, \phi_\gamma \right\rangle| \tag{2}$$

where $R^i f = f - f_i$ is the residual signal. The approximant update depends on the nature of the algorithm but can generally be seen as a descent in a direction defined by the newly selected atom (plain Matching Pursuit (MP) [7]), the subspace spanned by all previously selected atoms (Orthogonal MP) or a gradient estimate ([1]). This criterion can also be modified to take perceptual models into account or dealing with pre-echo artefacts. However, there is always, as mentioned in the introduction, the central problem of choosing the size of Φ. From a continuous parameter space, choosing a dictionary Φ for practical use amounts to discretizing the parameters. In the compressive sensing framework, this leads to the so-called basis mismatch problem [2]: the chosen representation space is not exactly the one where f has the sparsest representation.

2.1. Pursuits in time-frequency dictionaries

Time frequency dictionaries such as Gabor Dictionaries and MDCT-based dictionaries are well suited for audio signals. Time and frequency resolutions are constrained by the scale of the chosen transform that defines the analysis window size and the overlap intervals between consecutive analysis windows. The finer this analysis grid, the better the chances of choosing well-localized atoms that remove a lot of perceptually relevant energy. However, the complexity gets higher. Both for tractability and compression purposes, the time-frequency grid that serves for inner product computations need to be quantized. Let $\Phi = \{\phi_{s,u,\xi}\}$ be a dictionary of localized waveforms of scale $s \in S$, time position $u \in U$ and frequency index $\xi \in \Xi$. The set of available atom indexes is denoted $\Gamma = S \times U \times \Xi$.

2.2. Quantization of the time frequency plane

A convenient way to model Φ that coincides with practical implementations is to see it as a union of monoscale dictionaries: $\Phi = \bigcup_{s \in S} \Phi_{\sharp_s}$ each of which defines a quantization of the time frequency plane with resolution defined by the set of indexes $\sharp_s = U_s \times \Xi_s$. If the quantization is uniform, the size of $\Phi_{\sharp_s}$ is

$$T_s = \frac{N}{\Delta_s u} \times \frac{F_s}{2\Delta_s \xi} \tag{3}$$

where $\Delta_s u$ is the time interval between frames for the scale s, F_s is the sampling frequency and $\Delta_s \xi$ is the frequency resolution. The subdictionary $\Phi_{\sharp_s}$ can be an orthonormal base (e.g an MDCT with 50 % overlap [9]), in this case $\Delta_s u = s/2$ and $\Delta_s \xi = F_s/s$ and then $T_s = N$. It can also be overcomplete (Gabor dictionaries with more than 50% overlap: $\Delta_s u < s/2$, $T_s > N$) or span only a limited subspace ($\Delta_s u > s/2$, $T_s < N$). $\sharp_s$ defines a two dimensional lattice in the time-frequency plane that can be seen as a quantization of the underlying continuous time-frequency parameters. Indeed, the choice of a suboptimal atom can be understood as a quantization error artefact. Figure 1 shows how different lattices can fit different components of a signals, here transients and harmonics of a glockenspiel signal. The quantization error is greatly lowered by the concatenation of all the bases in the dictionary [9]. Nonetheless, traditional Matching Pursuit-based strategies are using the same dictionary during the whole decomposition process. By doing so, the a priori choice of lattices introduces a bias in the decomposition as explained in Durka's work [3]. While their solution would be to run multiple decompositions with different lattices and averaging the results in a Monte Carlo fashion, we propose an novel approach inspired by the *dithered quantization* technique.

3. PURSUITS WITH A RANDOM SEQUENCE OF SUBDICTIONARIES

Instead of choosing an analysis grid once and for all, the lattices are chosen so as to span the largest possible space in an ergodic fashion during the decomposition. For each scale, a sequence of lattices $\Delta_s\sharp = [\sharp_s^0, \sharp_s^1, ..\sharp_s^k]$ is used and at iteration i an atom is selected in $\Phi_i = \bigcup_{s \in S} \Phi_{\sharp_s^i}$. By doing so, we expect the equivalent of the quantization error to be evenly spread among the selected atoms, thus removing the bias. This technique is conceptually equivalent to adding a uniform noise in the time-frequency domain before quantizing it. In the overall, we hope to promote the selection of more salient features than with a fixed lattice. This technique is completely non-adaptive. The sequences $\Delta_s\sharp$ are known in advance and independant from the signal. They are also known by the decoder and therefore there is no need to encode them. The virtual cost of the index of atom i is then down to $\mathcal{O}(\log_2(\sum_{s \in S} T_s^i))$.

3.1. Learning the sequence

Since we want the subset sequence to be independant of the signal, we can try to estimate its desirable properties. When a signal model is available (e.g sinusoidal+transient modelling of audio, edges and textures modelling of images), one can try to manually design a sequence that will minimize the quantization error (i.e the suboptimality factor) under a dictionary size constraint. In this work however, we have no signal model and we are interested in designing a universal sequence for audio signals. Figure 2 shows a decomposition of a short glockenspiel signal with MP over a full temporal-resolution multiscale discrete Gabor dictionary. Frequency resolution is constrained in each scale, the full temporal resolution is achieved by performing Short Time Fourier Transforms with high overlapping between consecutive analysis frames (i.e $\Delta_s u = 1$ sample). Atoms are clearly not uniformly distributed in the time-frequency plane, which would be the case for white noise. Most real life signals are

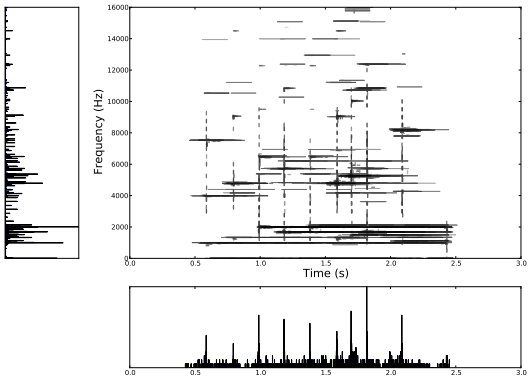

Fig. 2. *Time-frequency joint and marginal distribution of atoms in the decomposition of a glockenspiel signal with MP on a multiscale Gabor dictionary with full time resolution.*

structured, a complete randomness in the choice of an analysis lattice would not give optimal performance.

3.2. Time resolution subsampling

In particular, for audio signals, the frequency resolution problem seems to be efficiently adressed by the use of multiple scales. The time resolution, however, presents a more interesting challenge. Let us consider a reference coarse lattice $\sharp_s$ with overlap of $\Delta_s u = s/2$ between consecutive frames and define the local time shift τ of an atom relative to this coarse grid. Then the distribution of τ in the interval $[-\frac{s}{4}, \frac{s}{4}]$ ressembles a uniform distribution. To verify this statement, we decomposed short audio signals from the MPEG SQAM test database up to the first 1000 atoms in a full resolution multiscale Gabor dictionary and calculated the Gini index of τ. This index quantifies how far from the uniform distribution a candidate distribution stands and has been demonstrated [6] to be a suitable sparsity measure.

Figure 3 shows that distribution for the decomposition of the orchestra signal ressembles the one for the decomposition of white noise: atoms become more and more uniformly spread. The glockenspiel signal's atoms are less uniformly distributed, but their distribution can not be considered sparse after a few hundreds iterations. From this observation, we state that an efficient and simple way to simulate a pursuit in a large dictionary is to use orthonormal basis randomly shifted at each iteration by a random variable τ.

4. SCALABLE SPARSE CODING OF AUDIO SIGNALS

4.1. Performances with a simple encoding model

In order to demonstrate the potential benefits of this technique, we compared traditional approaches to the new one in a simplified audio coding task. We considered 3 cases:

MP with a union of 8 MDCT: $\forall s, \Delta_s u = s/2, T_s = N$ and $T = SN$.

MP with a union of 8 MDCT with full temporal resolution (for computational reasons, an approximate based on local optimization after a coarse grid search is used as in [8]) $T_s = Ns/2$, $T = \sum_{s \in S} Ns/2$.

MP with Random Sequence of Subdictionaries that are 8 MDCT randomly shifted in time. The subdictionary size is constant $T_s^i = N$.

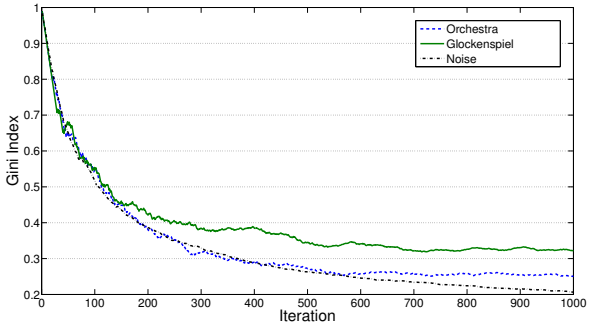

Fig. 3. *Gini index for the time localization of the first 1000 atoms in a full resolution multiscale MDCT dictionary for glockenspiel, orchestra and white noise signals.*

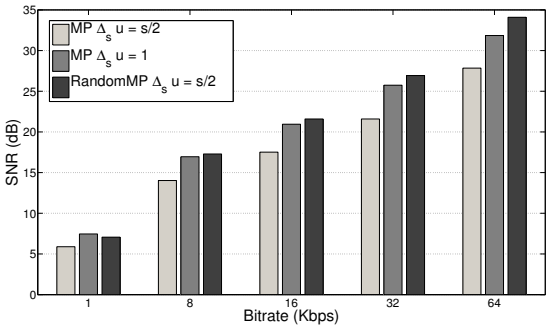

Fig. 5. *SNR achieved at different rates for 5 signals (6 seconds) from MPEG - SQAM database. Results averaged over 20 runs.*

The distorsion measure that we used was the Signal to Noise Ratio (SNR), and bitrates are estimated using the upper bound $\mathcal{C}(f_m) = m\,(\log_2(T) + Q)$, where we assume a simple uniform mid-tread quantizer with Q bits per coefficient. Note that using an entropy coder instead has also been tested, with similar results. Figure 4 summarizes the coding scheme. Figure 5 shows that the novel algorithm gives better performances in all cases but at very low bit-rates. This can be explained by the fact that, as exposed in 3.2, in the first iterations, the full resolution dictionary succesfully locates the most prominent features and it actually compensates the additional costs. At bigger rates, however, atoms are more evenly spread and the lower index cost favors the randomized method. To summerize our results, this randomized greedy pursuit allows to have the costs of the small dictionary with a decay rate close to the one on the large dictionary.

4.2. Controlling computational complexity

Although the theoretical complexity of the novel algorithm is equivalent to the one of the fixed small dictionary case, in practice it can get much higher. Inner products computation, actually, are usually performed using Fast transforms, and in the fixed dictionary case, most of these products remains unchanged from one iteration to the next, thus greatly reducing the cost of all but the first iteration. In our case, however, this trick cannot be applied anymore since the central point is to change all the projections at every iteration. There

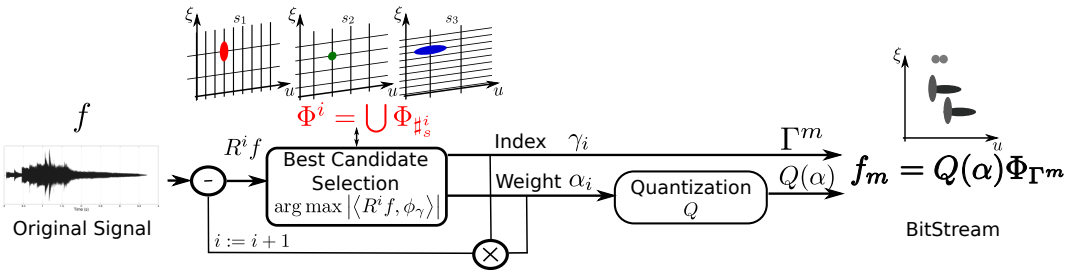

Fig. 4. *Simplistic encoding scheme with greedy decompositions: bitstream includes atom indexes from the sequence of subdictionaries Φ^i and the quantized weights.*

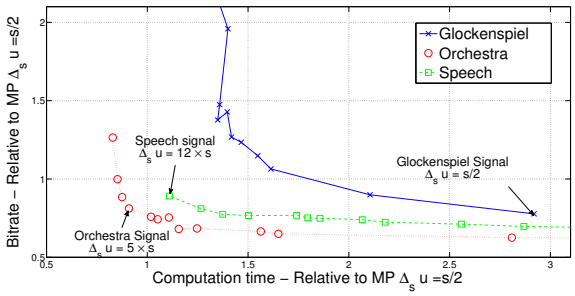

Fig. 6. *Compromise between coding efficiency and complexity with Randomized MP of 3 short audio signals. Results averaged over 40 runs.*

is nonetheless one parameter that allows us to control the complexity of the algorithm and it is the size of the subdictionaries. So far, we have considered Φ_i to be overcomplete, by decreasing the time resolution of each lattices $\sharp_s^i$, less inner products need to be computed at each iteration and the algorithm is much faster. By doing so, however, we can severely damage the convergence rate of the algorithm.

To evaluate the performances, we used different time resolution parameters $\forall s, \Delta_s u \in [s/2, s, 2\times s, .., 12\times s]$. We used the same simple encoding scheme and compared execution times and bitrates between the randomized method and a reference MP with fixed dictionary of 8 MDCT with the same optimizations than in [9]. All algorithms were given the same approximation quality target of 5 dB of SNR.

Figure 6 shows that the time resolution parameter can serve as a control parameter for limiting complexity while still improving the rate. A time resolution of $\Delta_s u = 5\times s$ which is equivalent to subdictionaries of size $T_s^i = N/10$ yields a bitstream 20% smaller than MP with the fixed grid, with a slightly faster computation time. With a speech signal, all cases took slightly longer than the reference but the bitrate was always smaller. Finally, the glockenspiel signal gave the less competitive results: its components are so well localized that the subsampling quickly penalizes the achieved bitrate. However in this case, the randomized MP with overcomplete subdictionaries took less than 3 times longer than the reference MP. This can be explained by the fact that our algorithm selects better atoms than the reference and thus, much less iterations are needed to achieve the same approximation level.

5. CONCLUSION

The proposed algorithm appears to be suitable for sparse approximation of complex signals. The potential benefits are in low bitrate compression, and we exhibit several sound examples were these advantages show off. The unsupervised nature of the algorithm and the randomness introduced in the atom selection makes it very easy to design worst-case scenarii for which the algorithm would converge slower than a pursuit over a fixed dictionary. However, on average, and with a small empirical variance, the proposed scheme appears to have the coding costs of the small dictionary with a decay rate close to the one on the large dictionary.

In conclusion, adding randomness to the parameter space within a greedy sparse decomposition process can be highly beneficial. Although on different paradigms, this is reminiscent of both dithered quantization techniques and compressive sampling strategies. Whether such random sub-dictionary techniques can be combined with other sparse decomposition schemes (e.g. iterated thresholding) is still an open issue.

6. REFERENCES

[1] T. Blumensath and M.E. Davies. Gradient pursuits. *IEEE Trans. Sig. Proc*, 2008.

[2] Y. Chi, L.L. Scharf, A. Pezeshki, and A.R. Calderbank. Sensitivity to basis mismatch in compressed sensing. *Signal Processing, IEEE Transactions on*, 59(5):2182 –2195, may 2011.

[3] P.J. Durka, D.Ircha, and K.J. Blinowska. Stochastic time-frequency dictionaries for matching pursuit. *IEEE Trans. Sig. Proc*, 2001.

[4] R.M. Figueras i Ventura, P. Vandergheynst, and P. Frossard. Low-rate and flexible image coding with redundant representations. *IEEE Trans. Image Proc.*, 2006.

[5] R. Gribonval. Fast matching pursuit with a multiscale dictionary of gaussian chirps. *IEEE Trans. Sig. Proc*, 2001.

[6] N. Hurley and S. Rickard. Comparing measures of sparsity. *IEEE Trans, Inf. Theo.*, 2009.

[7] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Sig. Proc*, 1993.

[8] M. Moussallam, L. Daudet, and G. Richard. Matching pursuits with random sequential subdictionaries. *preprint available at http://arxiv.org/abs/1107.2509*, 2011.

[9] E. Ravelli, G. Richard, and L. Daudet. Union of MDCT bases for audio coding. *IEEE Trans. Audio Speech Lang. Proc.*, 2008.