The M-Term Pursuit for Image Representation and Progressive Compression

Adel Rahmoune, Pierre Vandergheynst, Pascal Frossard
École Polytechnique Fédérale de Lausanne (EPFL)
Signal Processing Institute
CH- 1015 Lausanne, Switzerland
{adel.rahmoune, pierre.vandergheynst, pascal.frossard}@epfl.ch

Abstract— **This paper introduces a sparse signal representation algorithm in redundant dictionaries, called the M-Term Pursuit (MTP), with an application to image representation and scalable coding. The MTP algorithm belongs to the framework of the matching pursuit (MP) [1]; it expands the image into a linear combination of atoms, selected from a large collection of spatial atoms. The MTP relies on the concept of dictionary partitioning, i.e., as splitting the dictionary into L disjoint sub-dictionaries, each carrying some specific information. Then, it iteratively finds a K-term approximation, by selecting M atoms at a time, where $M \leq L$, followed by an orthogonal projection. The approximation performances of the MTP algorithm have been shown to yield comparable results with those of the matching pursuit. However, it presents the advantage of a reduced computational complexity. For progressive image compression, an embedded quantization and coding step is applied on the series of obtained atoms based on the subset approach [2]; to generate a flexible bitstream. The performances of the MTP image coder are finally shown to compare favorably against those of the state-of-the-art JPEG-2000 scheme, in terms of rate-distortion characteristics.**

I. INTRODUCTION

Signal representations providing progressive and scalable image coding schemes have gained more interest recently from the research community. They provide interesting solutions to an increasing number of multimedia applications that require adaptive signal representations, like image manipulation and transmission over heterogeneous environments such as Internet. Most of the known scalable image coding schemes are generally based on the 2-D wavelet transform, employing a separable 2-D discrete wavelet transform (DWT) [3]–[5]. However, it was demonstrated that the separable wavelet transform in higher dimensional spaces is not ideally suited for representing higher dimensional singularities. For instance, the 2-D DWT fails to capture regular geometric features (e.g. edges or contours) [6], mainly because it lacks directionality and anisotropic scaling. Two-dimensional adaptive image representation has been proposed recently in [7], as an alternative to wavelet-based methods. This scheme generates sparse signal expansions in spatial dictionaries, well adapted to represent natural image features and components. It employs the Matching Pursuit algorithm [1], to obtain a compact signal decomposition using a small number of atoms, and thus allowing for efficient coding at low and medium bit rate. For instance, it was shown that for low bit rates (e.g. up to 0.4 bpp

for 256x256 image), the compression results are comparable to the state-of-the-art JPEG-2000 coder, in terms of rate distortion performance and visual quality. However, the matching pursuit algorithm is characterized by a higher computational complexity, because at each iteration, it calculates all the inner products between the residual image and the atoms. Motivated by this issue, we proposed a greedy algorithm for sparsity-constrained optimization problems, called the M-Term Pursuit (MTP) [8]. The MTP algorithm is a recursive approach for compact signal representation that provides an approximation to the residual signal in M atoms selected from a large dictionary $\mathcal{D}$, at each iteration. The proposed image representation scheme based on the MTP is characterized by a reduced complexity, while maintaining the compression performances of MP.

This paper is organized as follows: Section II discusses the image representation scheme using the MTP algorithm and dictionary partitioning. The computational complexity of the algorithm is described in Section III. Approximation performances are shown in Section IV and compared to those of MP. Section V presents the scalable image coder and its compression performances when compared against state-of-the-art JPEG-2000. Finally, discussions and conclusions are given in Section VI

II. IMAGE REPRESENTATION

A. Image Dictionary

The spatial dictionary $\mathcal{D}$ is built following the same methodology as proposed in [7]. $\mathcal{D}$ is generated by applying affine transformations on two mother atoms, which are a 2-D Gaussian $g_1(x,y) = \frac{1}{\sqrt{\pi}} e^{-(x^2+y^2)}$, and its 2nd partial derivative (a ridge-like function) $g_2(x,y) = \frac{2}{\sqrt{3\pi}}(4x^2 - 2)e^{-(x^2+y^2)}$. The 2-D Gaussian is used in order to extract the low frequency components. Its 2nd partial derivative is used to capture image singularities like edges and contours. The affine operator is a composition of translation, scaling and rotation of the mother atoms, as follows:

$$ \mathcal{U}_{(x_0,y_0,a_1,a_2,\theta)}g = \frac{1}{\sqrt{a_1 a_2}} g(r_{-\theta}(\frac{x - x_0}{a_1}, \frac{y - y_0}{a_2})), \quad (1) $$

where $r_{-\theta}$ is a rotation matrix of angle θ. The coherence [9] $\mu_1(m)$ of the dictionary $\mathcal{D}$, which is related to its redundancy,

is defined as:

$$\mu_1(m) \overset{def}{=} \max_{|\Lambda|=m} \max_{\omega \notin \Lambda} \sum_{\lambda \in \Lambda} |\langle g_\omega, g_\lambda \rangle|. \qquad (2)$$

B. Dictionary Partitioning

The dictionary $\mathcal{D}$ is partitioned into L disjoint sub-dictionaries $\mathcal{D}_i$. Each sub-dictionary $\mathcal{D}_i$ consists of all functions g_l, $l = 1, 2$, whose center, (x_0, y_0) defined in equation 1, belongs to a region Ω_i. In our method, an image of size (I_X, I_Y) is divided into non-overlapping rectangular blocks of size $(I_{X_\Omega}, I_{Y_\Omega})$, where each block is associated to a region Ω_i and thus to a sub-dictionary $\mathcal{D}_i$. This class of partitioning, also called space-partitioning and denoted by $\mathcal{P}_\Omega(\mathcal{D})$, insures that the coherence between atoms in different regions shall be small, except for atoms lying near the border of the regions Ω_i's or those having a large support. The atoms centered near the border of a given region Ω_i generally have a support extending to other regions, though they are regarded to belong to Ω_i.

C. The MTP Decomposition

In a dictionary partition $\mathcal{P}_\Omega(\mathcal{D})$ composed of L ($L = 64$) sub-dictionaries according to the previous description, the MTP algorithm decomposes the image into a finite number of spatial atoms, iteratively as described by Algorithm 1. At each iteration j, the algorithm performs two steps: (i) the selection step, and (ii) the projection step. During the selection step, all the inner products between each atom in $\mathcal{D}$ and the residual image $I^j(x, y)$ are calculated. Then m_j atoms are selected, at most one is selected from each sub-dictionary $\mathcal{D}_i$, (i) according to the threshold parameter γ, (ii) and by imposing that the coherence in the collection of the selected atoms Φ_Λ does not exceed μ_s (or simply $(\mu_1(m)/m) \leq \mu_s$) following the non-decreasing order for their inner product magnitudes, i.e. $|\langle R^j I, g_\omega \rangle|$ in Algorithm 1. The second step consists in an orthogonal projection $P_V R^j I$ of $I^j(x, y)$ onto the span of the atoms indexed in Λ.

A formal description of the algorithm is given in Algorithm 1. The error $\|I - a_J\|_2$ for the obtained approximation a_J is upper-bounded. This bound is derived by following the steps of the proof in [8], Theorem 1, and by knowing that at iteration j all the m_j atoms in Φ_Λ have inner product x_p satisfying $|x_p| \geq \gamma \|X\|_\infty$, where the vector X is $(\langle R^j I, g_\omega \rangle)_{g_\omega \in \mathcal{D}}$. Now, by letting $X_\Lambda = (x_p)_{p \in \Lambda}$, we have $\|X_\Lambda\|_2^2 \geq (m_j \gamma^2)\|X\|_\infty^2$.

So the upper-bound after J iterations is given by Theorem 1:

Theorem 1: Let $\mathcal{P}(\mathcal{D})$ be a partition of $\mathcal{D}$ and I be any signal. Fix a threshold γ s.t,

$$\gamma \leq \frac{1}{\beta} \sqrt{\frac{1 + \mu_1(m_j - 1)}{m_j}}, \text{ for } j = 1, \cdots, J \qquad (3)$$

Suppose that Φ_Λ exists with $m_j = |\Phi_\Lambda|$, and that the MTP returns an approximation a_J. Then the approximation error is bounded by:

$$\|I - a_J\|_2^2 \leq \|I\|_2^2 \prod_{j=1}^{J} \left(1 - \frac{\beta^2 (m_j \gamma^2)}{1 + \mu_1(m_j - 1)} \right), \qquad (4)$$

Algorithm 1 The MTP Image Decomposition Algorithm.

INPUT:
 The image I, the number of iterations J, the threshold γ and μ_s.
OUTPUT:
 The approximation a_J and residual $R^J I$.
PROCEDURE
1. Initialize the residual signal $R^0 I = I$ and the approximation $a_0 = 0$, the set of atoms $\Phi_\Lambda = \emptyset$, and the iteration number $j = 1$.
2. For $i = 1$ to L
  Find $g_{\omega_i} = \arg \sup_{\omega \in \Omega_i} |\langle R^{j-1} I, g_\omega \rangle|$
  If $|\langle R^{j-1} I, g_{\omega_i} \rangle| \geq \gamma \sup_{\omega \in \Omega} |\langle R^{j-1} I, g_\omega \rangle|$
  Then $\Phi_\Lambda \leftarrow \Phi_\Lambda \cup \{g_{\omega_i}\}$
3. Discard g_ω from Φ_Λ if its coherence $(\mu_1(m)/m) > \mu_s$ in Φ_Λ, following the non-decreasing order of $\langle R^{j-1} I, g_\omega \rangle|$.
4. Determine the orthogonal projection $P_V R^{j-1} I$ onto the span of Φ_Λ, compute the approximation and the residual:

$$\begin{aligned} a_j &\leftarrow a_{j-1} + P_V R^{j-1} I \\ R^j I &\leftarrow R^{j-1} I - P_V R^{j-1} I \end{aligned}$$

5. Increment j, empty Φ_Λ ($\Phi_\Lambda = \emptyset$), and go to step 2 if $j \leq J$.

---

where β is related to the dictionary.

III. COMPUTATIONAL COMPLEXITY

The computational complexity of the MTP during an iteration j in any dictionary $\mathcal{D}$ is $O(Np + m_j q)$. The first term comes from performing N inner products between each atom in $\mathcal{D}$ and the residual $R^j I$, each requiring p operations. Whereas the second term is due to the projection operation $P_V R^j I$. The orthogonal projection, which is $P_V R^j I = \sum_{n=1}^{m_j} c_n g_{\omega_n}$, is achieved by solving a linear system of equations. Since $P_V R^j I$ minimizes the error $\|R^j I - \sum_{n=1}^{m_j} c_n g_{\omega_n}\|_2^2$ over all (c_n) for $n = 1 \cdots m_j$, the calculation of the coefficients (c_n) requires to solve the following linear system. For any $g_{\omega_k} \in \Phi_\Lambda$,

$$\langle R^j I, g_{\omega_k} \rangle = \sum_{n=1}^{m_j} c_n \langle g_{\omega_n}, g_{\omega_k} \rangle. \qquad (5)$$

Denote the vectors $C = (c_n)_{1 \leq n \leq m_j}$ and $X_\Lambda = (\langle R^j I, g_{\omega_k} \rangle)_{1 \leq k \leq m_j}$, and the Gram matrix $G = (\langle g_{\omega_n}, g_{\omega_k} \rangle)_{1 \leq k \leq m_j, 1 \leq n \leq m_j}$. A Conjugate Gradient method [10] is implemented to solve the linear in Eq. 5, or $GC = X_\Lambda$, since G is Hermitian if the atoms are linearly independent, thus requiring $O(qm_j)$ operations where q is a constant. The complexity of the matching pursuit is $O(Nm_j p)$ for the same number of terms.

Figure 1 shows the computational complexity as a function of the residual error measured in PSNR, of both algorithms MTP and MP when applied to the Lena image (256x256). One can see clearly that the gain becomes very significant at higher PSNR values. This is due to the fact that at this stage of

the approximation, in a single iteration the number of selected atoms by the MTP tends to the maximum L whereas MP keeps selecting only one atom per iteration. For instance, at a PSNR of 28.6, the MP requires almost 400 units in contrast to the 41 units required for the MTP, thus accounting for a speed-up of 9.68. It is noteworthy that only the complexity of the approximation is compared since the time required for entropy encoding is a small fraction of the total coding time.
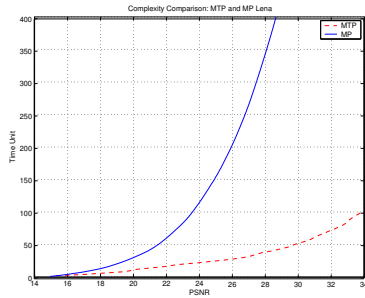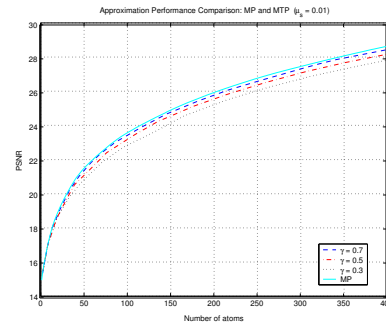

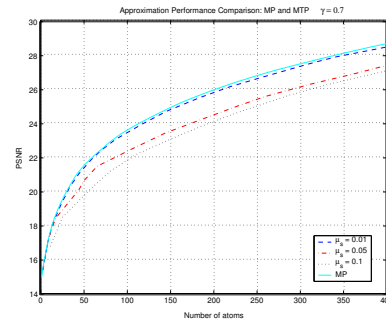Fig. 1.   Complexity comparison between MTP and MP algorithms: Time unit vs PSNR for Lena

IV. MTP APPROXIMATION PERFORMANCE

The MTP algorithm has been applied to the dictionary $\mathcal{D}$ with the partition $\mathcal{P}_\Omega(\mathcal{D})$ as described in the previous section. The threshold γ is varied over $\{0.3, 0.5, 0.7\}$ and the coherence threshold μs is assigned also values from the set $\{0.01, 0.05, 0.1\}$. In all our experiments the dictionary $\mathcal{D}$ is divided into $L = 64$ sub-dictionaries $\mathcal{D}_i$. Figure 2 shows the behavior of the error of MTP, measured in terms of the PSNR, versus the number of atoms compared, against the matching pursuit algorithm, for different γ's and μ_s's. It is clear that the gap between the two algorithms decreases with smaller μ_s and larger γ because in this case the selected atoms are almost uncorrelated and have a large energy contribution (defined by γ) during a single iteration. With $\gamma = 0.7$ and $\mu_s = 0.01$ the MTP performs almost as well as the MP.

An interesting relationship between the cardinality of Φ_Λ at each iteration and the values of the γ and μ_s for the Lena image (256x256) is displayed in Figure 3. The number of selected atoms increases along the pursuit; this is due to the fact that during the first iterations, most of the selected atoms correspond to the smooth regions present in the image, or the low-frequency part of the signal, which tend to have a support expanding over different regions Ω_i. Hence, these atoms with overlapping supports are characterized by somehow a *large* coherence. And by imposing a small coherence threshold μ_s on the atoms of Φ_Λ tends to reduce its cardinality. On the other hand, when iterations run on, the number of selected atoms per iteration stabilizes mainly because these atoms are chosen to represent edges, which are localized, and thus these atoms are almost decorrelated. So their selection restriction comes principally from the fixed threshold γ. A direct consequence of this observation is that the complexity gain of the MTP over MP, which is defined mainly by the cardinality of Φ_Λ, becomes significant after few iterations.

(a) $\mu_s = 0.01$

(b) $\gamma = 0.7$

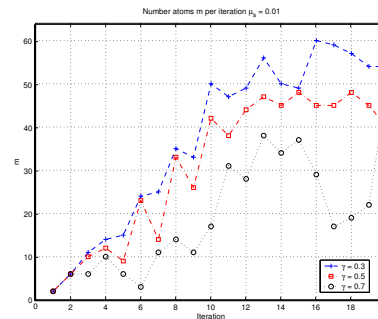Fig. 2.   Approximation performance comparison between MP and MTP for Lena image (256x256)

Fig. 3.   The effect of γ on the number of atoms selected per iteration for Lena image ($\mu_s = 0.01$)

V. PROGRESSIVE IMAGE COMPRESSION

An interesting application for compact image representation is compression and especially progressive compression, which consists in embedded quantization and coding operations.
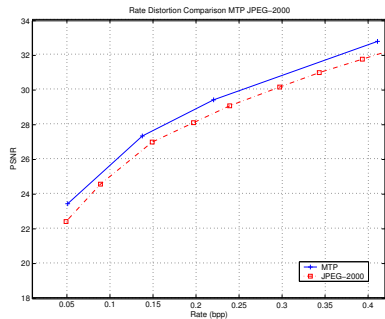
A. Embedded Coding

The embedded quantization and coding is performed by using the subset approach introduced in [2], where the selected atoms (indexes and coefficients) are initially divided into S disjoint subsets s_i. Each subset contains l_i elements. Their number is dictated by scalability requirements (i.e., the number of target decoding rates), and represents a trade-off between
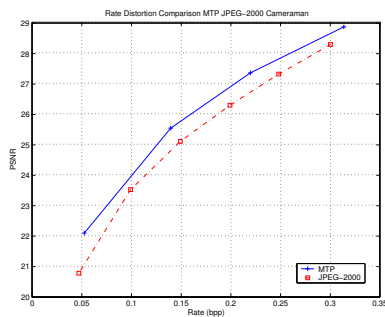
stream flexibility, and coding efficiency, that respectively increases and decreases with S. In each subset, atoms are sorted according to their spatial positions, that are further run-length encoded. Other index parameters and quantized coefficients are encoded with an adaptive arithmetic encoder [11].

B. Rate-Distortion Results

We evaluate the rate-distortion performances of the MTP based image compression scheme and compare it against the state-of-the-art image compression standard JPEG-2000. The standard Lena and Cameraman images are used for comparison in our experiments. It can be seen on Figure 4 that the PSNR of MTP image codec is higher than that of JPEG-2000 by about 0.6 db over the range of low and medium bit rates for both images, i.e. up to 0.4 bpp.

(a) Lena (256x256)

(b) Cameraman (256x256)

Fig. 4. Rate distortion comparison between MTP ($\gamma = 0.7, \mu_s = 0.01$),and JPEG-2000

Figure 5 show the visual quality of the MTP-based codec when applied upon the Lena image. When decoded at a target bit rate of 0.41 bpp, its PSNR is 32.73. And one can see that all the edges, contours and prominent features present in the image have been captured, which gives some smoothness to the reconstructed image.

VI. DISCUSSIONS AND CONCLUSIONS

We proposed an image representation and coding scheme that uses the novel MTP algorithm as a decomposition strategy over redundant visual dictionaries, or a rich collection of visual

(a) Original Lena   (b) MTP encoding

Fig. 5. The Lena image (256x256) decoded using the MTP-based codec at 0.41bpp, the PSNR = 32.73

primitives tuned for image features. The main strength of the MTP algorithm is its reduced complexity by an order of magnitude when compared to the matching pursuit approach. Once the image representation is obtained in terms of a finite sum of visual atoms, image processing and compression tasks can be applied directly on the individual atoms. For instance, to achieve scalable image compression these atoms are further quantized and coded in a progressive fashion based on a subset approach. The performances of this scheme are favorably compared to the state-of-the-art image coders such as JPEG-2000 in terms of rate-distortion characteristics. Moreover, the flexible bit-stream allows for non-octave spatial scaling, due to the structure of the dictionary, without resorting to complex transcoding operations [2]. Future works will be to investigate other types of dictionary partitioning, such as scale-based or phase-based partitioning and to compare the algorithm against other fast matching pursuit-based techniques.

REFERENCES

[1] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, December 1993.

[2] A. Rahmoune, P. Vandergheynst, and P. Frossard, "Flexible motion-adaptive video coding with redundant expansions," *IEEE Transactions on Circuit and Systems for Video Technology*, 2004, submitted.

[3] D. Taubman and M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers, 2001.

[4] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on circuit and systems for video technology*, vol. 6, pp. 234–250, June 1996.

[5] J. Shapiro, "An embedded wavelet hierarchical image coder using zero trees of wavelet coefficients," in *IEEE Data compression conference*, 1993, pp. 214–223.

[6] E. J. Candes and D. L. Donoho, "Curvelets- a surprisingly effective nonadaptive representation for objects with edges," in *Curve and surface fitting*, A. C. C. Rabut and L. L. Schumaker, Eds. Saint-Malo: Vanderbilt University Press, 1999.

[7] P. Frossard, P. Vandergheynst, and R. Figueras i Ventura, "High flexibility scalable image coding," in *Proc. SPIE VCIP*, Lugano (Switzerland), 2003.

[8] A. Rahmoune, P. Vandergheynst, and P. Frossard, "Sparse approximation using m-term pursuits with applications to image and video compression," Signal Processing Institute, Swiss Federal Institute of Technology EPFL, Tech. Rep. ITS-2005.03, 2005, http://itswww.epfl.ch.

[9] J. A. Tropp, "Just relax: Convex programming methods for subset selection and sparse approximation," UT-Austin, ICES 04-04, February 2004.

[10] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed. Johns Hopkins University Press, 1996.

[11] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Comm. ACM*, vol. 30, pp. 520–540, June 1987.