Multi-Scale Dictionary Learning using Wavelets

Boaz Ophir, Michael Lustig and Michael Elad

Abstract

In this paper we present a multi-scale dictionary learning paradigm for sparse and redundant signal representations. The appeal of such a dictionary is obvious - in many cases data naturally comes at different scales. A multi-scale dictionary should be able to combine the advantages of generic multi-scale representations (such as Wavelets), with the power of learnt dictionaries, in capturing the intrinsic characteristics of a family of signals. Using such a dictionary would allow representing the data in a more efficient, i.e. sparse, manner, allowing applications to take a more global look at the signal. In this work we aim to achieve this goal without incurring the costs of an explicit dictionary with large atoms. The K-SVD using Wavelets approach presented here applies dictionary learning in the analysis domain of a fixed multi-scale operator. This way, sub-dictionaries at different data scales, consisting of small atoms, are trained. These dictionaries can then be efficiently used in sparse coding for various image processing applications, potentially outperforming both single-scale trained dictionaries and multi-scale analytic ones. In this paper we demonstrate this construction and discuss its potential through several experiments performed on fingerprint and coastal scenery images.

Index Terms

Multi-scale, Dictionary Learning, K-SVD, Sparse, Redundant

B. Ophir and M. Elad are with the Department of Computer Science, Technion – Israel Institute of Technology, Haifa 32000, Israel. e-mail: {boazo,elad}@cs.technion.ac.il.

M. Lustig is with the Department of Electrical Engineering and Computer Sciences, U.C. Berkeley, Berkeley, CA, 94720 . email: mlustig@eecs.berkeley.edu.

I. Introduction

A. General

Sparse representations of signals over redundant dictionaries is an evolving field with state of the art results in many signal and image processing tasks. The basic assumption of this model is that natural signals can be expressed as a *sparse* combination of atom signals. Formally, for a signal $\mathbf{y} \in \mathbb{R}^{n\times 1}$, this can be described by $\mathbf{y} = \mathbf{D}\mathbf{x}$, where $\mathbf{D} \in \mathbb{R}^{n\times K}$ is a dictionary that contains the atoms as its columns, and $\mathbf{x} \in \mathbb{R}^{K\times 1}$ is the representation vector.

Given the signal, finding its representation is done using the following sparse approximation problem:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0^0 \quad \text{subject to} \quad \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2 \leq \epsilon, \tag{1}$$

where ϵ is a permitted deviation in the representation accuracy, and the expression $\|\mathbf{x}\|_0^0$ is a count of the number of non-zeroes in the vector $\mathbf{x}$. The process of solving the above optimization problem is commonly referred to as "sparse-coding".

While the sparse-coding problem in itself is generally NP-hard, approximate solutions can be found by a wide variety of algorithms. Such methods include "pursuit" algorithms such as matching pursuit (MP) [1] and orthogonal matching pursuit (OMP) [2]. Another popular alternative is to substitute this problem with a simpler one by replacing the ℓ^0-norm with an ℓ^p-norm with $p = 1$ or $p \leq 1$ as is done in the basis pursuit [3] and FOCUSS [4] algorithms.

A fundamental question in practicing the above model is the choice of dictionary to be used [5]. Most approaches to this problem can be divided into one of two categories: the analytic approach and the learning based approach. In the analytic approach a mathematical model of the data is formulated, leading to an implicit dictionary described by it's algorithm rather than by an explicit matrix. These dictionaries include the Fourier transform, the DCT, Hadamard, Wavelets, Curvelets and Countourlets among others. The dictionaries generated by these approaches are highly structured and have fast implementation. A common theme among many of these methods is a multi-scale approach to signal representation.

In contrast, the second approach infers the dictionary from a set of training examples. The

dictionaries learnt are typically represented as explicit matrices. Dictionary learning algorithms range from the well-known and simple PCA, through the seminal work by Olshausen and Field [6], the MOD [7] and K-SVD [8] follow-up methods, and all the way to generalizations of the PCA (e.g. GPCA [9]). All these methods target the same goal – finding a direct sparsifying transform [5]. This approach yields dictionaries more finely fitted to the data, thus producing better performance in many applications. However, this comes at a price of unstructured dictionaries, which are more costly to apply. Complexity constraints limit such learnt dictionaries and specifically the atom size that can be learnt. This constraint is the reason why low-dimensional (a typical dimension is of the order of 100) patch-based processing is so often practiced when using such dictionaries.

In this paper we present an attempt to merge the two approaches described above to create a truly multi-scale learnt dictionary, hopefully gaining the advantages of both methods. In a nut-shell, we propose training the dictionary, in parts, over the analysis range of an analytic multi-scale transform, applied to the training set.

B. Related Work

The idea of learning multi-scale dictionaries is not new. In [10], [11], [12] the Wavelet pyramid structure is maintained (thus achieving multi-scale learning). The Wavelet parameters are trained so that the Wavelet coefficients conform with a sparsity inducing prior distribution. The results are Wavelet-like filters that give a slightly sparser representation for the training images.

In [13], [14] the first steps are taken towards more general multi-scale learnt dictionaries. Using a Quadtree structure, different sized blocks are used to make up one joint global dictionary. This dictionary is learnt and used in a similar manner as in the K-SVD algorithm. Although different sized atoms are used, the computational constraint limits the maximal atom size used, just as in the single-scale approach.

The work reported in [15] could be interpreted as yet another way to train a multi-scale dictionary. This work suggests to form the effective learnt dictionary as a multiplication of a

fixed and fast transform (we will refer to it as the core dictionary) by a sparse matrix[1]. The meaning of this structure is that every atom in the effective dictionary is a linear combination of *few* and arbitrary atoms from the core dictionary. The learning procedure is a variant of the K-SVD (termed Sparse-K-SVD), where the dictionary update stage amounts also to a series of sparse coding problems.

While the work in [15] used this construction for working with somewhat larger patches using the redundant DCT as the core dictionary, one could envision applying the same algorithm with a Wavelet (or any other multi-scale transform) core dictionary, thereby leading to a highly structured and learnt multi-scale dictionary. We note that the work in [15] did not address such an option, and in particular did not consider the numerical complexities that such large matrices give rise to.

The approach presented in our paper bares some similarity to the work in [15], as we too create atoms by combining core Wavelet atoms. However, our construction forces a close spatial proximity between the combined core-atoms, leading to a more constrained structure that directly targets the spatial redundancy that Wavelet coefficients tend to exhibit when handling images. More on the relation between these two algorithms will be given after we introduce our algorithm.

Another line of similar work is found in [17], [18]. The work in [17] suggests combining Bandlets with a multi-scale Wavelet transform. This idea is used in [18] for compression, using either fixed or learnt (PCA) union of ortho-bases dictionaries. We take a somewhat similar approach, combining Wavelets with learnt dictionaries.

C. This Paper

All the above-mentioned recent attempts to tie multi-scale representations to dictionary learning are innovative, but unfortunately they all lead to marginal improvements over existing non-multi-scale methods. This may be explained by the heavily constrained structures these methods force.

[1]Interestingly, a similar dictionary construction is used in [16], but for a different purpose, of approximating a desired frame by combinations over another core-dictionary.

In this paper we present a way to construct multi-scale learnt dictionaries and apply them in an efficient and effective manner. We train the dictionary by learning patch-based dictionaries (using K-SVD) in the analysis domain of the Wavelet transform. When using the overall dictionary, the atoms are effectively interpolated by the inverse Wavelet transform, creating a truly multi-scale learnt dictionary that has the potential to outperform both multi-scale analytic transforms and single-scale learnt dictionaries. Due to its specific structure, sparse coding can be done very easily and with local operations only, while handling arbitrarily high-dimensional signals. In this work we present the core algorithm proposed, and demonstrate it through several examples that show its potential.

The paper is organized as follows. In Section II we describe the background to our work, discussing both multi-scale representations and single-scale dictionary learning. In Section III we motivate and present the construction of multi-scale learnt dictionaries in the analysis domain of a multi-scale transform. Section IV includes experiments comparing our methodology to standard Wavelets and single-scale learnt dictionaries. We test these representation options by evaluating the M-term approximation performance, core-denoising of images, and compressed sensing. In all these tests we demonstrate the potential of our approach to better represent image-content. Section V concludes this paper.

II. Background

A. Multi Scale Image Representations

Multi-scale analysis for images took a center stage in image processing since the 1980's, starting with the Gaussian and Laplacian pyramids (first suggested by Burt and Adelson [19]), and the Gabor Transform. In the 1990's Wavelets [20] became the premier multi-scale analysis tool in signal processing. Being however better suited for single dimensional signals, their usefulness in image processing was limited. To overcome the Wavelet shortcomings in handling two dimensional signals, more advanced "Wavelet like" decompositions were developed starting in the late 1990's and into the 2000's. Several different families of multi-scale decompositions are available for us to consider:

1) Laplacian Pyramid [19], [21],
2) Steerable Pyramid [22],
3) Standard Wavelet decompositions [20],
4) Wavelet Packet decompositions [23],
5) Advanced multi-scale decompositions - Contourlets [24], [25], Curvelets [26], [27], Ridgelets [28], Shearlets [29], Bandlets [30], [17], [18] and more.

While all these decompositions create multi-scale representations of an image, they all suffer from the curse of generality. Designed to handle any and all images, these decompositions can not, and do not, handle any subset of images optimally.

Of the above decompositions, the Curvelet, Contourlet, Shearlets and Bandlets decompositions feature some sort of optimality for two dimensional signals. This optimality is usually measured by the decay of the the M-term approximation error, the distance $\|\mathbf{y}-\mathbf{y}_M\|^2$ between the signal $\mathbf{y}$ and it's approximation $\mathbf{y}_M$, using the M strongest representation coefficients. A low M-term approximation error would suggest that these representations are well suited for many image processing applications. However, even in these cases, optimality is shown only for specific classes of images, such as piece-wise smooth images, which do not necessarily reflect true image content.

That said, many useful applications have been found for these decompositions. A key feature that makes these transforms appealing is their tendency to sparsify specific image content. We shall build on this important property and target it directly by merging a learning procedure on top of an existing multi-scale transform. In that respect, our work follows the intuition of the Bandlets, as advocated by Stephane Mallat and his co-authors: Rather than seek the direct transform to get the ultimate sparsification of our signals, start by using an existing transform that does reasonably well in that respect, and then add another layer of processing that squeezes more over the already simplified signals.

Parameters: K (number of atoms), n (size of signals)
Initialization: Set the dictionary matrix $\hat{\mathbf{D}} \in \mathbb{R}^{n \times K}$ (using examples, or a pre-chosen matrix)
Loop: Repeat until convergence (or according to a stopping rule)

- *Sparse Coding*: Fix $\hat{\mathbf{D}}$ and use OMP to compute the representation vector $\mathbf{x}_i$ for each example $\mathbf{y}_i$ $(i = 1, 2, \ldots, N)$ by solving:

$$\min_{\mathbf{x}_i} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}_i\|_0^0 \leq T \tag{3}$$

- *Dictionary Update*: update each atom $\mathbf{d}_k$ $(k = 1, 2, \ldots, K)$ in turn by:
  - Select ω_k, the group of examples that use atom $\mathbf{d}_k$.
  - For each example j in ω_k, compute its residual $\mathbf{e}_{j,k}$ without the contribution of $\mathbf{d}_k$.
  - Create residual matrix $\mathbf{E}_k$ as the matrix whose columns are $\mathbf{e}_{j,k}$.
  - Update the atom $\mathbf{d}_k$ and weights x_j^k by minimizing:

$$(\mathbf{d}_k, x_j^k) = \underset{\mathbf{x},\mathbf{d}}{\operatorname{argmin}} \|\mathbf{E}_k - \mathbf{d}\mathbf{x}^T\|_F^2 \quad \text{subject to} \quad \|\mathbf{d}\|_2 = 1 \tag{4}$$

    This one-rank approximation is performed by truncated SVD of $\mathbf{E}_k$.

Fig. 1. The Single-scale K-SVD algorithm. The description assumes a fixed number of non-zeroes T in every representation, which could be replaced with a fixed representation error.

B. Learning Signal/Image Dictionaries

Learnt image dictionaries have been of much interest in recent years. Two prominent examples are the MOD algorithm [7] and the K-SVD algorithm [8], [31], [32]. Both these algorithms try to minimize the representation error:

$$\underset{\mathbf{D},\mathbf{X}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \quad \text{subject to} \quad \|\mathbf{x}_i\|_0^0 \leq T \quad \forall i, \tag{2}$$

where $\mathbf{Y} = [\mathbf{y}_1\ \mathbf{y}_2\ \ldots\ \mathbf{y}_N] \in \mathbb{R}^{n \times N}$ denote the set of training examples, $\mathbf{D}$ is the dictionary, and $\mathbf{X} \in \mathbb{R}^{K \times N}$ the sparse representation matrix ($\mathbf{x}_i$ are the columns of $\mathbf{X}$).

For a given (and necessarily low) signal dimension, such a dictionary can be trained using for example the K-SVD algorithm (Figure 1). The problem with this algorithm (and others similar to it) is that the atoms (columns of the dictionary $\mathbf{D}$) are of the same size as the signal they represent. While in theory this gives the learning algorithm maximal freedom to shape the atoms to describe different scales of the data, in practice, due to limited computational resources, this severely limits the size of the signals represented.

Any attempt to increase the signal dimension n beyond few hundreds immediately implies an intolerable amount of computations for the training phase, and an unrealistic size of training set to use. Even if the training has been done somehow, using such a dictionary in applications is prohibitive, as any multiplication by this dictionary (as an explicit matrix) leads to high-complexity algorithms. As said above, this is the prime reason for the popular patch-based processing that is commonly practiced with learned dictionaries in recent years.

Using these algorithms for larger images, in practice, means breaking the input image into blocks and treating each block independently. When these blocks have large overlap the end result is a very redundant representation of the original image. While this representation may be very useful for purposes of denoising [33], it is counter intuitive to our goal of representing the *complete* signal with a sparse combination of atoms. On the other hand, small, or no, overlap of the blocks leads to boundary issues when reconstructing the image.

Dictionary learning algorithms that attempt to learn small, shift invariant, atoms from large signals also exist [32], [34], [35]. One such algorithm is the MoTIF algorithm [34], [35]. In this algorithm small atoms are learnt from larger training signals, however the main driving force is not sparsity of the representation. Instead, the algorithm forces the learnt atoms to be as different from one another as possible by penalizing correlation between them. This approach works well when the atoms are indeed very dissimilar, however this is not the case in most natural signals/images, where we expect many atoms to have a relatively high correlation with each other. Most MoTIF atoms will thus be "difference" atoms that are only relevant when used in conjunction with a "base" atom during coding.

III. Dictionary Learning in the Wavelet Domain

A. The Core Approach

The Wavelet transform gives a sparse representation of the original signal to some degree. What we would like to do is squeeze out some of the redundancy left by the Wavelet decomposition, specifically the spatial correlation between Wavelet coefficients in the same band, or between bands, thus producing sparser image representations than plain Wavelet decompositions.

Following the work reported in [15], we begin our derivation by looking at the learning problem expressed by the following modification to Equation (2):

$$\underset{\mathbf{D},\mathbf{X}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{W}_S\mathbf{D}\mathbf{X}\|_F^2 \quad \text{subject to} \quad \|\mathbf{x}_i\|_0^0 \leq T \quad \forall i. \tag{5}$$

Here $\mathbf{D}$ denotes the learnt dictionary, $\mathbf{X}$ the (sparse) representation vectors, and $\mathbf{Y}$ are the training set images. The matrix $\mathbf{W}_S$ denotes the Wavelet synthesis operator (inverse Wavelet), or equivalently the Wavelet atom dictionary. This model suggests that the data can be expressed by a sparse combination of atoms, which are themselves combinations of atoms from a fixed multi-scale core dictionary, e.g Wavelet. This problem is however intractable, in general, for reasonably sized data, without additional constraints or assumptions on the unknown $\mathbf{D}$. In [15], the assumption chosen is that $\mathbf{D}$ has very sparse columns. This implies that the overall dictionary atoms are linear combination of few (and arbitrary) Wavelet atoms.

Assuming that $\mathbf{W}_S$ is square and unitary (i.e orthogonal Wavelet with periodic extension), we can equivalently write:

$$\underset{\mathbf{D},\mathbf{X}}{\operatorname{argmin}} \|\mathbf{W}_A\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \quad \text{subject to} \quad \|\mathbf{x}_i\|_0^0 \leq T \quad \forall i, \tag{6}$$

where $\mathbf{W}_A$ denotes the Wavelet analysis operator. This formulation suggests that we can train our dictionary not in the image domain but in the analysis domain of the multi-scale decomposition operator, specifically the Wavelet transform.

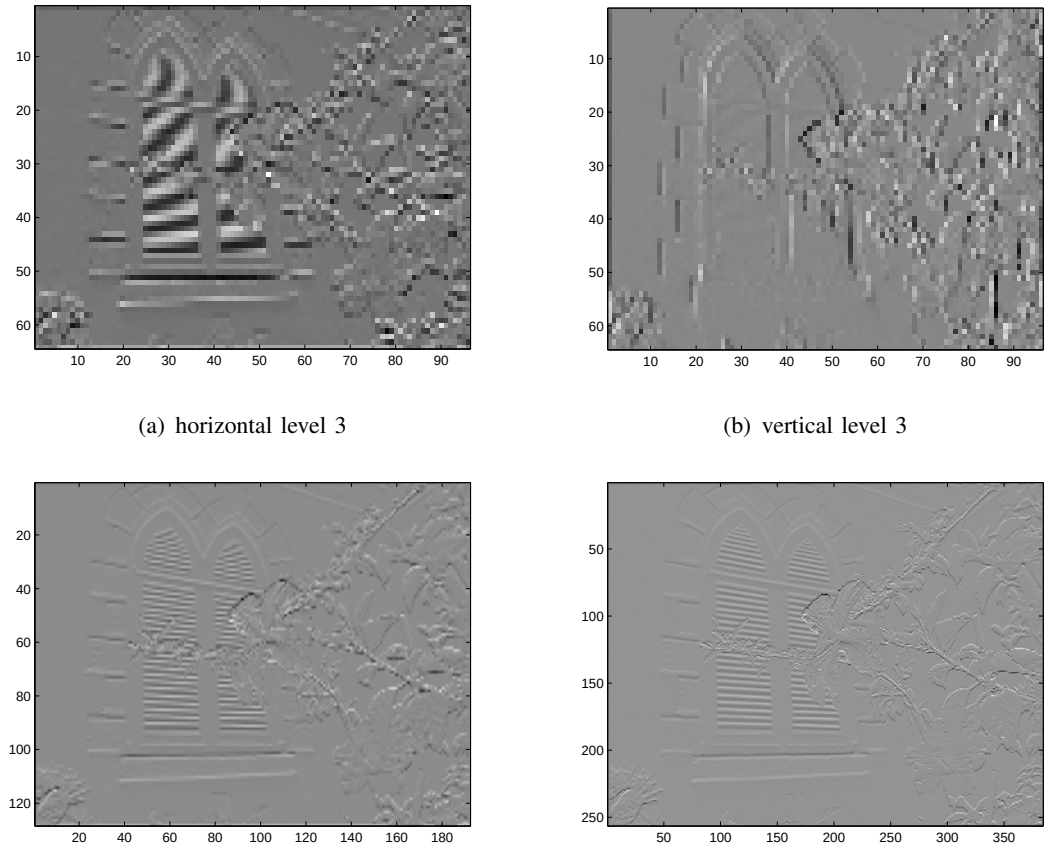
A natural way to view the Wavelet analysis domain is not as a single vector of coefficients, but rather as a collection of coefficient "images" or bands. The different Wavelet coefficient images contain data at different scales and orientations (horizontal,vertical and diagonal). As such it makes sense that separate dictionaries be used to represent these images. We achieve this by training our dictionary in parts, training separate sub-dictionaries $\mathbf{D}_b$ for each Wavelet band (or group of bands):

$$\forall b \quad \underset{\mathbf{D_b},\mathbf{X_b}}{\operatorname{argmin}} \|(\mathbf{W}_A\mathbf{Y})_b - \mathbf{D}_b\mathbf{X}_b\|_F^2 \quad \text{subject to} \quad \|\mathbf{x}_{i,b}\|_0^0 \leq T \quad \forall i, \tag{7}$$

where subscript b denotes the different Wavelet coefficient bands.

While the learning process can be applied as is on small images, this is computationally impossible with larger images (the coefficient band images for the first decomposition level are for instance one forth the size of the original image). We solve this problem by returning to the patch-based approach, prevalent in many image processing methods.

This approach makes sense from another perspective as well – the coefficient images themselves have local structure - adjacent Wavelet coefficients tend to be correlated. Figure 2 demonstrates this. Capturing this structure is the essence of the learning process.



(a) horizontal level 3

(b) vertical level 3

(c) horizontal level 2

(d) horizontal level 1

Fig. 2. Several bands taken from the Wavelet transform of an arbitrary image. As can be seen, there is a large amount of correlation between the Wavelet coefficients, suggesting that more can be done to sparsely represent the image content.

In contrast to patch-based approaches in the image domain that emphasize only the local correlation between pixels, in this approach even a small patch in the deeper decomposition

Parameters: The following parameters should be chosen:
- Wavelet type to use,
- S – number of decomposition levels (scales),
- K – number of atoms per dictionary, and
- n – size of the dictionaries' atoms.

Initialization: Set the dictionary matrices for all bands, $\hat{\mathbf{D}}_\mathbf{b} \in \mathbb{R}^{n\times K},\ b = 1, 2, \ldots, 3S+1$.
Wavelet Decomposition: Decompose each of the training-set images using the chosen 2D-Wavelet transform, each into $3S+1$ bands.
For each band:
- **Extract Patches:** Extract maximally overlapping patches of size $\sqrt{n} \times \sqrt{n}$ from the same band of all training set decompositions.
- **K-SVD:** Apply K-SVD separately for each decomposition band to train the sub-dictionary $\hat{\mathbf{D}}_\mathbf{b}$. This process should be repeated $3S+1$ times, once per each band.

Algorithm Output: The set $\hat{\mathbf{D}}_\mathbf{b} \in \mathbb{R}^{n\times K},\ b = 1, 2, \ldots, 3S+1$, combined with the Wavelet transform used, define the effective multiscale dictionary learned.

Fig. 3. The proposed multi-scaled dictionary learning – K-SVD applied to each band in the Wavelet domain

levels affects a large area in the image domain. This allows our approach to have a more global, as well as local, outlook. The complete learning algorithm is described in Figure 3.

In effect, the effective dictionary we created is $\mathbf{W_s D}$, replacing the standard Wavelet dictionary $\mathbf{W_s}$. The "effective" atoms are thus interpolated versions of the atoms in the learnt dictionary $\mathbf{D}$, interpolated by the Wavelet synthesis process. This dictionary enjoys the multi-scale capabilities of the Wavelet transform while adding to it information specific to the training domain. Note that in the training we use maximally overlapping patches. This creates a "richness" in the training data that generates a level of shift-invariance in the resulting dictionary. Some effective atoms from different scales and bands, obtained by training on a corpus of fingerprint images, are presented in Figure 4. In order to visualize a single effective atom, all the coefficients, except one, are set to zero. The coefficients are then multiplied by the learnt dictionary and passed through a Wavelet synthesis process.

While the idea presented here seems quite simple – applying dictionary learning in the transform domain – it leads to an elegant way of creating a truly multiscale dictionary, while still retaining reasonable computational cost for both learning the dictionary and using it in practice.

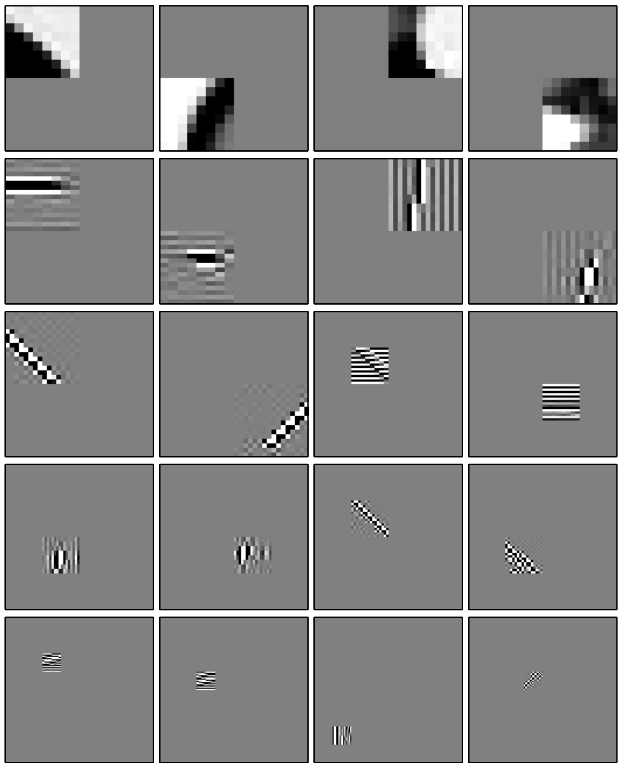


Fig. 4. Some effective atoms from different levels/bands trained on fingerprint images and using a 3 level Haar Wavelet transform. A separate sub-dictionary was trained for each band. Top row - approximation band. 2^{nd} row - coarsest level horizontal and vertical bands and so on.

One of the most appealing aspects of this structure is the ease with which sparse coding is performed. Before we turn to describe this, we first discuss several flexibilities and options that could be incorporated into the above scheme.

B. Options and Flexibilities in the Proposed Scheme

In the above algorithm, the degrees of freedom are the choice of the Wavelet filters, the depth of the Wavelet decomposition and the size of the sub-dictionaries (number of atoms and their size). An obvious extension would be to allow the sub-dictionaries to have different sizes at different scales and orientations. This may be warranted for specific data types where the variability needed to be expressed by the dictionary is high in some bands (requiring a higher level of redundancy in the dictionary to allow for sparse coding) while it is low in others.

Other options arise from the fact that in the above algorithm, each Wavelet decomposition level and each directional band is treated separately and has it's own sub-dictionary. This, of

course, in not mandatory. Two additional options may have merit:

1) Grouping by decomposition level - a single sub-dictionary is trained for all bands at the same Wavelet decomposition level. The rational is that each such sub-dictionary will express all the data at the same scale.
2) Grouping by directional band - a *single* sub-dictionary could be trained for all bands of the same *orientation* in all different scales. The rational for this approach is that the directional features are self similar at different scales, and this should be exploited in our construction. In a way, this directly extends what analytic multi-scale transform are naturally doing.

An advantage of both these approaches is that they increase the amount of data available for training of each sub-dictionary. In some scenarios, there is very little data to train on, especially at the lower decomposition levels. While these options affect which data is used to train each sub-dictionary, and which sub-dictionary is associated with each band, the atoms themselves still represent two dimensional patches in the Wavelet coefficient domain.

A different direction we can take is departing from the two dimensional patch approach, and create multi-band atoms. These atoms are created by concatenating two dimensional patches from different decomposition bands in such a way that all the 2D patches are mapped to the same location in the image. We offer two options for this construction:

1) Grouping by decomposition level - group same-sized patches from all three bands at each decomposition level. Thus, the sub-dictionary could be trained on 3D patches of size $\sqrt{n} \times \sqrt{n} \times 3$, containing three $\sqrt{n} \times \sqrt{n}$ matched patches merged together. The rational is that a correlation exists, not only spatially within each band, but also between "brother"-bands at the same scale, and this should be utilized.
2) Grouping by directionality - group patches from all bands with the same direction. For orthogonal Wavelets we have three directions - horizontal, vertical and diagonal, but for redundant Wavelets (or other transforms such as Contourlets there may be more). Each sub-dictionary could be trained on pyramidal patches, containing matched patches of sizes $\sqrt{n} \times \sqrt{n}$ (for the course level), $2\sqrt{n} \times 2\sqrt{n}$, $4\sqrt{n} \times 4\sqrt{n}$ ... The rational for this approach

is that sharp image features (edges), are composed of many frequency components. The multi-scale transform actually partitions the same edge into multiple bands. The pyramidal atom merges the partitions back together.

Another option available to us is to replace the standard orthogonal Wavelet transform (with periodic extension) by non-unitary transforms such as bi-orthogonal Wavelet and more advanced transforms such as Contourlets or Curvelets. While these decompositions generally give a sparser representation of the data, especially for images, there is still redundancy in the representation that can be reduced by using our scheme. However, losing the unitary property means that Equation (5) and Equation (6) are no longer equivalent, and this will affect the way the sparse-coding should be performed. In this work we will restrict our study to unitary Wavelet transforms, leaving the more general transforms for a future work.

C. Relation to Sparse K-SVD Algorithm

As already mentioned, a work closely related to ours is the one introducing the Sparse K-SVD algorithm [15]. Both works share the same starting point – Equation (5). However, in Sparse K-SVD the effective dictionary is given by the multiplication $\mathbf{BA}$, where $\mathbf{B}$ is a core dictionary and $\mathbf{A}$ is a sparse matrix. Thus, the Sparse-K-SVD seeks effective atoms that are sparse combinations of atoms from the core dictionary. Put formally, the Sparse-K-SVD defines the following optimization task

$$\underset{\mathbf{B},\mathbf{X}}{\operatorname{argmin}} \|\mathbf{Y}-\mathbf{BAX}\|_F^2 \quad \text{subject to} \quad \begin{array}{ll} \|x_i\|_0^0 \le t & \forall i \\ \|a_j\|_0^0 \le p & \forall j \end{array}, \tag{8}$$

In comparison, our approach also creates effective atoms by combining core atoms from a fixed dictionary. However, in our work we combine atoms that are in close spatial (or scale) proximity. Adding such a constraint to Sparse K-SVD would be difficult and cumbersome, and our paradigm bypassed these difficulties while leading to simple and elegant learning procedure.

More specifically, the computational complexity of Sparse K-SVD algorithm is much higher than in our approach. This limits the Sparse K-SVD algorithm to relatively small fixed dictio-

naries which translates to working again on relatively small patches. Our work in the analysis domain, decoupling the multi-scale structure from the learning procedure, enables a much simpler construction and allows working with a much larger core dictionary, e.g. the Wavelet dictionary.

D. Sparse Coding

Assuming that we have been able to train a multi-scale dictionary somehow, one of the main issues we face is using it in applications. Every sparse coding algorithm requires a multiplication by the dictionary and its adjoint, as part of the numerical process of computing the representation. A major problem with explicit multi-scale dictionaries, limiting their usefulness, is the prohibitively high cost of applying them for sparse coding. Atoms with large support simply require too many operations to be effective. This is where the proposed multi-scale dictionary approach shows it's advantage.

The sparse representation of an image $\mathbf{y}$ with respect to a dictionary $\mathbf{D}$ is the solution $\mathbf{x}$ of the problem

$$(P_{0,\epsilon}) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0^0 \ \text{ subject to } \ \|\mathbf{y}-\mathbf{D}\mathbf{x}\|_2 \leq \epsilon, \tag{9}$$

where we aim at getting the sparsest representation that explains $\mathbf{y}$ as $\mathbf{D}\mathbf{x}$ with an error that is at most ϵ. In our approach we replace this problem by

$$(P_{0,\epsilon}^{W})_{synth} \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0^0 \ \text{ subject to } \ \|\mathbf{y}-\mathbf{W}_S\mathbf{D}\mathbf{x}\|_2 \leq \epsilon \tag{10}$$

in the synthesis domain, or equivalently,

$$(P_{0,\epsilon}^{W})_{analysis} \quad \min_{\mathbf{x}} \|\mathbf{x}\|_0^0 \ \text{ subject to } \ \|\mathbf{W}_A\mathbf{y}-\mathbf{D}\mathbf{x}\|_2 \leq \epsilon, \tag{11}$$

in the analysis domain.

All these problems are NP hard in general, and approximate solutions can be found using various methods. The two most prevalent approaches in sparse coding are greedy methods (such as OMP) and relaxation based methods. The proposed dictionary that combines Wavelets and K-SVD can easily be used in both.

In the greedy methods, signals are coded by adding coefficients one at a time until a stopping criterion is met. This is typically done in the signal domain. We propose instead to first apply the multi-scale decomposition $\mathbf{W}_A$ to the image. Then, patches (atom sized) in the analysis domain can be coded using the appropriate sub-dictionary, by using the sparse coding greedy algorithm of choice. These operations are done using the small atoms in the sub-dictionary, at the same cost as when using a single-scale dictionary.

The stopping criterion can be calculated locally per patch, independently of the other patches, or using a more global look. In such a local approach, the k-th patch from the band b, denoted by $[\mathbf{W}_A\mathbf{y}]_b^k$, will be allocated a fixed number of non-zero coefficients ℓ (or a per-patch noise threshold), based on

$$(P_0^w)_b^k \quad \min_{\mathbf{x}} \left\| [\mathbf{W}_A\mathbf{y}]_b^k - \mathbf{D}\mathbf{x}_b^k \right\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}_b^k\|_0^0 \leq \ell, \tag{12}$$

and this should be repeated for all patches k from all bands b.

A global counterpart approach can also be proposed, where patches are coded in conjunction with the coding of the other patches. In this case, we seek the sparse representation vector $\mathbf{x}$ that solves

$$(P_0^w)^{global} \quad \min_{\mathbf{x}} \sum_b \sum_k \left\| [\mathbf{W}_a\mathbf{y}]_b^k - \mathbf{D}\mathbf{x}_b^k \right\|_2^2 \quad \text{subject to} \quad \sum_b \sum_k \|\mathbf{x}_b^k\|_0^0 \leq \ell, \tag{13}$$

Patches will essentially "compete" for additional coefficients, until a global criteria is met (either a fixed number of non-zero coefficients coding the *whole* image or a global noise threshold).

The process can be viewed as if we have competing local pursuits running on all the patches, together acting as a sort of global pursuit. At each step we will compare the gain (in terms of the residual energy) obtained by "activating" an additional coefficient for each patch, thus allowing the local pursuit to add another coefficient to the patch's representation. In other words, we let each local pursuit show us what can be gained by letting it code with one more non-zero coefficient, and then choose the best one. We are thus considering a set of size $|\sum_b \sum_k 1|$ of possibilities (i.e., the number of overall patches in the complete Wavelet domain). The patch

that gives maximum benefit will then have an additional coefficient allocated to it. Since the multi-scale transform is energy preserving all these operations and comparisons can be done purely in the analysis domain, without need to apply the synthesis operator.

Once an atom has been chosen to be added to a certain patch, a least-squares step is required to update the patch's representation (in OMP), followed by updating the residual. Since the patches in the same band are non-overlapping and the Wavelet transform is orthogonal (so one band's representation does not affect the others), these steps are all local. In addition, when moving to the next iteration of the algorithm for choosing the next atom, one need not reevaluate all the $|\sum_b \sum_k 1|$ inner-products, but only update the last patch's gain (how that patch will reduce the residual by having another coefficient allocated to it).

While the algorithm described above is an extension to pursuit algorithms that add coefficients one at a time, such as OMP, our scheme can also be similarly adapted to pursuit methods such as CoSaMP [36] and Subspace Pursuit [37] that activate coefficients in groups.

As for the relaxation based approach, the obtained optimization task can be solved numerically by algorithms such as iterative shrinkage. There are two operations that need to be computed efficiently in order for this process to be feasible – the multiplication by the dictionary $\mathbf{D}_{eq}$ and multiplication by it's transpose $\mathbf{D}_{eq}^T$. Multiplication by the complete equivalent dictionary $\mathbf{D}_{eq}$ is simply done by multiplying each explicit sub-dictionary by the appropriate (sub)-coefficient vector, aggregating the results for all patches k and bands b, and applying the synthesis multi-scale operator (i.e. the inverse Wavelet transform)

$$\widehat{\mathbf{y}} = \mathbf{D}_{eq}\mathbf{x} = \mathbf{W}_s \left(\bigcup_b \bigcup_k \mathbf{D}_b \mathbf{x}_b^k \right). \tag{14}$$

Multiplication by the transpose $\mathbf{D}_{eq}^T$ is also a simple procedure. For unitary transforms this operation is the inverse transform. For orthogonal Wavelets this is simply the forward (analysis) Wavelet transform. We note that for non-orthogonal Wavelets the transpose of the synthesis operation can still be computed efficiently by using the Wavelet analysis operator. In this case the synthesis and analysis filters are no longer equal, and thus the analysis operator is applied

with the synthesis filters instead of the standard analysis ones. Multiplication by $\mathbf{D}_{eq}^T$ is thus computed by applying the (forward) Wavelet transform to the signal followed by breaking each Wavelet coefficient image into atom sized blocks and multiplying each block by the appropriate (explicit) sub-dictionary transposed.

In the experiments that follow we shall use both the greedy and the relaxation methods, and demonstrate their effectiveness.

IV. Experiments

In the following experiments we aim to demonstrate the advantages that our multi-scale learnt dictionary has, compared to standard multi-scale representations and compared to single-scale patch processing in the image domain. We show that by replacing standard Wavelet dictionaries with learnt ones, based on the same multi-scale structure described above, we get an improved representation. We also show that compared to the single-scale patch processing in the image domain, we obtain a more compact representation and better denoising.

A. M-term Approximation

To demonstrate the potential of our scheme to various image processing applications we start by looking at the M-term approximation error, as a measure of how well our dictionary describes the particular features of a set of images.

An input image representation, using a trained dictionary, can be created by the following steps:

- Apply the Wavelet transform to the image.
- Each coefficient band (at each level) should be broken into non-overlapping blocks.
- Using the global approach and local OMP coding, a sparse representation of the transformed image is found with L non-zeros.
- Each band's representation is thus an $M \times N$ sparse matrix, M being the number of atoms in the appropriate sub-dictionary and N being the number of non-overlapping blocks in the band.

- The total image representation is the collection of the representations of all the bands.

From this representation the image can be then reconstructed by:

- Multiplying each block's representation vector by the appropriate sub-dictionary.
- Reconstructing the Wavelet coefficient images for each band and level by tiling the non-overlapping blocks.
- Applying the inverse Wavelet transform.

We trained dictionaries on two sets of images with different characteristics, and using different Wavelets commonly used in various image processing tasks. We trained one dictionary on 50 fingerprint images (Figure 5 presents three of them), and using a 2D separable 16-tap Symlet Wavelet transform (Matlab 'sym8') three layers deep. A second dictionary was trained on 20 coastal scenery images [38] (Figure 6 presents three of them), using Daubachies 8-tap Wavelets (Matlab 'db4') also three layers deep. These transforms produce ten coefficient bands from each input image: horizontal, vertical, and diagonal coefficient images for each level, plus an approximation image at the last level. A separate sub-dictionary was trained for each band, 10 sub-dictionaries in total. Each sub-dictionary is a 64×64 matrix, i.e 64 atoms, each of size 8×8. The training samples used were maximally overlapped blocks extracted from the training set Wavelet coefficient images.

Fig. 5. Training Images - fingerprint data set.

We compare our reconstruction to a Wavelet clipping scheme, where small Wavelet coefficients are nullified, and to single-scale K-SVD, where non-overlapping image patches are coded using a dictionary trained on the same input images.

For both data sets, our reconstruction gives, for the same number of active coefficients, sig-

Fig. 6. Training Images - coastal scenery data set.

nificantly higher image quality at low bit-rates (see Figure 7 for an example from the fingerprint data set, and 8 for an example from the coastal scenery data set). In Figure 9 we show PSNR as a function of the number of active (non-zero) coefficients for all three schemes. The results shown are the average PSNR over 15 test images for the fingerprints and 10 images for the coastal scenery images.

Fig. 7. M-Term approximation of fingerprint image using 3 level sym8 Wavelet. From left to right - input image, Wavelet reconstruction (PSNR=20.68dB), Single-scale K-SVD reconstruction (PSNR=20.44dB), K-SVD on Wavelet reconstruction (PSNR=26.20dB). All reconstructions have 3300 active coefficients for an 448×448 image.

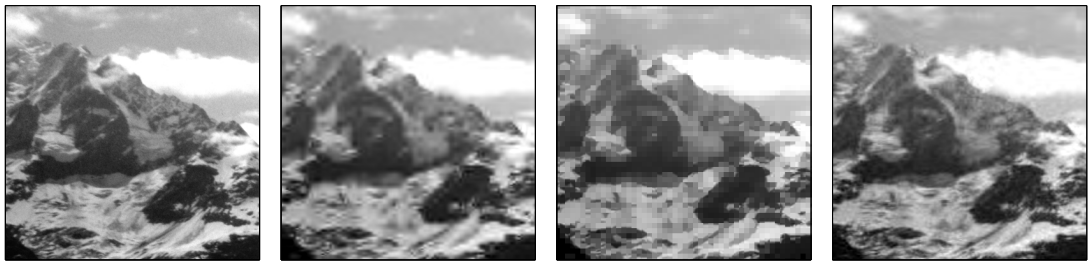


Fig. 8. M-Term approximation of coastal scenery image using 3 level db4 Wavelet. From left to right - input image, Wavelet reconstruction (PSNR=28.36dB), Single-scale K-SVD reconstruction (PSNR=27.82dB), K-SVD on Wavelet reconstruction (PSNR=30.43dB). All reconstructions have 32000 active coefficients for an 1152×1728 image (only a 400×400 segment of the images is shown).

At the low end of the graph, we can see that our reconstruction continues to give good quality images even with a very low number of coefficients. At these levels, both Wavelet clipping and single-scale K-SVD distort the image beyond recognition. The effect is much more prominent in

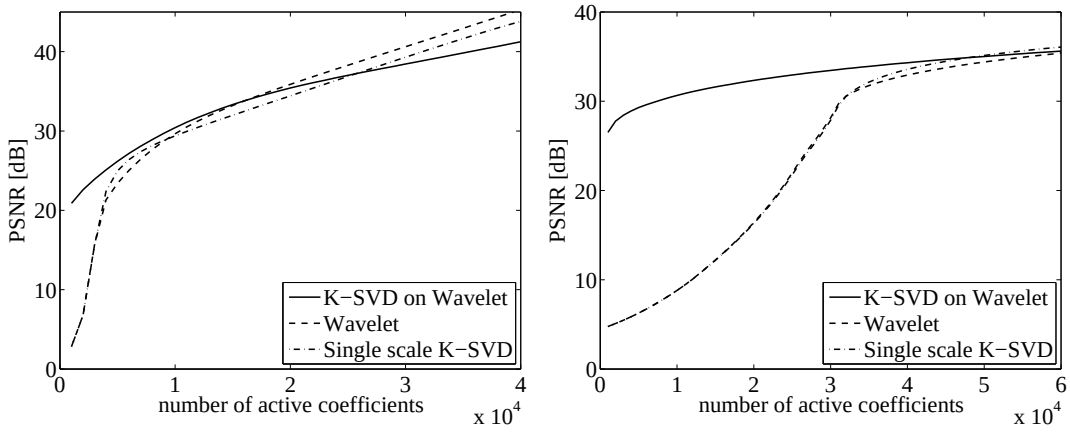


Fig. 9. M-Term approximation - PSNR as function of number of active coefficients. Left - fingerprint images (averaged over 15 test images), Right - coastal scenery images (averaged over 10 test images). For fingerprint images a 3-level 'sym8' Wavelet is used, and for coastal scenery images a 3-level 'db4' Wavelet. All dictionaries are 64×64.

the scenery images which exhibit true multi-scale, compared to the fingerprint images which are mostly single scale. At the high end of the graph the Wavelet clipping gives better quality. Both these results stem from the fact that we train the dictionary specifically to represent the signal well using few active coefficients. Thus for a very sparse representation our scheme performs better. When we test for a much denser representation, our scheme seems inferior. However, our construction is still valid in this case, except that the dictionary needs to be trained for a denser representation to begin with.

We note that these comparisons are fair one since both our representation, the Wavelet representation and the single-scale K-SVD representation are all non-redundant. Needless to say, these tests were all done on images outside the training set.

B. M-term Approximation of Noisy Images

We can use the M-term approximation, described above, as a form of rudimentary denoising. We can seek (by exhaustive search) the best threshold to denoise the images by performing hard thresholding. In the following experiment we add noise ($\sigma = 20$) to fingerprint and scenery images, and plot the denoising achieved by the M-term approximation in our scheme, as a function of the number of non-zero coefficients. We compare our result to a simplified version on K-SVD denoising in the image domain. In this scheme a dictionary is trained on image patches

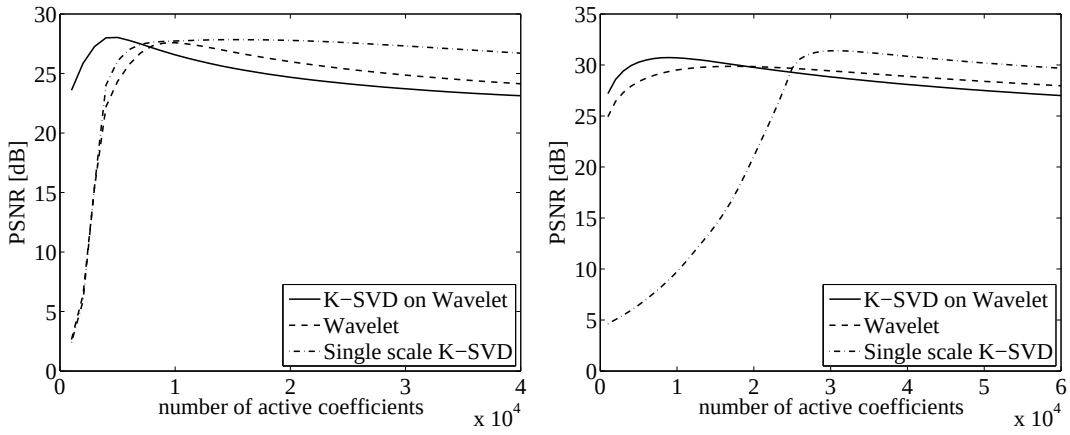


Fig. 10. M-Term denoising - PSNR as function of number of active coefficients. Additive Gaussian white noise ($\sigma = 20$) is cleaned by sparse coding of non-overlapping patches in the Wavelet and image domains. All dictionaries are 64×256. Left - fingerprint images (averaged over 15 test images), Right - coastal scenery images (averaged over 10 test images). For fingerprint images a 3-level 'dmey' Wavelet is used, and for coastal scenery images a 6-level 'dmey' Wavelet. All dictionaries are 64×256.

(from the same training data set). The noisy image is broken into *non-overlapping* patches and global coding is applied in a similar manner to our construction.

We used the discrete Meyer (Matlab 'dmey') Wavelet transform with three levels of decomposition for the fingerprint images and six levels deep for the scenery images. To make the comparison fair, all the dictionaries used are 64×256 (with only one dictionary trained for all Wavelet bands together).

The results (Fig. 10) show that our scheme reaches about the same PSNR level ($\pm 0.3dB$) but this is achieved at a fraction ($\frac{1}{3} - \frac{1}{2}$) of the number of non-zero coefficients compared to the single-scale approach. The effect is much more dramatic for the coastal scenery images which exhibit true multi-scale, but is also seen for the fingerprint images. We note that this configuration (one dictionary trained to represent all bands) is biased against our scheme. The dictionary training procedure, as is, will tend to focus more on the approximation band, and will therefore not represent the directional bands particularly well. Adding a higher redundancy to our dictionary (by training per band sub-dictionaries) improves the result, while the coding cost of the coefficients remains the same. The graph also shows the denoising results for Wavelet coefficient hard-thresholding. These results are significantly inferior, but the comparison is not really fair as the Wavelet dictionary on its own is non-redundant.

This process, coding non-overlapping patches, is a simplified or naive denoising scheme. A complete state-of-the-art denoising solution would be considerably more complex and may include among other things:

- coding and averaging overlapping patches,
- circular shifting of the input image (taking advantage of the fact that the wavelet transform is not shift-invariant),
- per-band parameter optimization.

The fact that our scheme achieves the same level of denoising using significantly fewer coefficients leads us to believe that a full denoising solution based on our scheme will be able to consistently outperform single-scale approaches. This is currently part of ongoing work.

C. Compressed Sensing

In the compressed sensing scenario [39] a signal $\mathbf{f}$ is sampled by a linear measurement process that computes inner products between the signal and a collection of random vectors (whose number is significantly smaller than the signal dimension),

$$\mathbf{y} = \mathbf{\Phi f}. \tag{15}$$

Assuming the signal $\mathbf{f}$ can be expressed as a sparse combination $\mathbf{x}$ of atoms from a sparsifying dictionary $\mathbf{D}$,

$$\mathbf{y} = \mathbf{\Phi D x}, \tag{16}$$

the signal can be reconstructed by minimum l_0 or l_1 norm reconstruction. In our test we used l_1 norm minimization,

$$\widehat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{y} - \mathbf{\Phi D x}\|_2^2 + \lambda\|\mathbf{x}\|_1. \tag{17}$$

We compare three options for the sparisfying basis $\mathbf{D}$,

- A standard Wavelet base,
- Wavelet + Overcomplete DCT (ODCT) Dictionary,

- Wavelet + Trained Dictionary.

We used the separable surrogate functionals (SSF) method [40], [41] to minimize (17).

Our test and training images were again fingerprint images (a different database). We used 16-tap orthogonal Symlet Wavelet (Matlab 'sym8') as the Wavelet base in all tests. The input image was 4096 pixels, and 1024 measurements were taken. The SSF algorithm was allowed to run for 10000 iterations. The results are shown in Figure 11. The "Wavelt+ODCT" shows a clear advantage over plain Wavelet, and is in turn outperformed by the "Wavelet+Trained Dictionary".

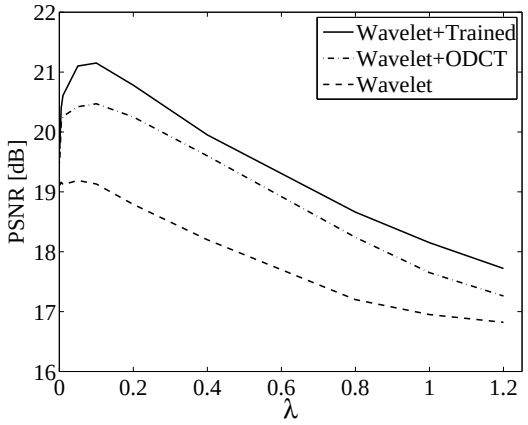


Fig. 11. Compressed Sensing test - PSNR vs. λ.

V. Conclusions

In this paper we presented a novel construction of a multi-scale learnt dictionary. This dictionary combines the multi-scale properties of the Wavelet transform with the data matching capabilities of learnt dictionaries. The learnt dictionary is trained, in parts, in the analysis domain of the Wavelet transform. This allows for a simple and efficient learning process. The multi-scale dictionary can then be seamlessly incorporated in a variety of sparse coding schemes. The benefits of the multi-scale dictionary are demonstrated both in M-term representation and for sample applications, denoising and compressed sensing. Our work on the subject is far from done. We believe this approach has potential for many other image processing tasks such as

inpainting and compression. Another topic of ongoing research is replacing the unitary Wavelet transform with more advanced redundant multi-scale representations.

ACKNOWLEDGMENTS

This research was partly supported by the European Community's FP7-FET program, SMALL project, under grant agreement no. 225913, and by the ISF grant number 599/08.

REFERENCES

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

[2] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of the 27th Annual Asilomar Conference on Signals, Systems, and Computers*, vol. 1, 1993, pp. 40–44.

[3] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM REview*, vol. 43, no. 1, pp. 129–159, 2001.

[4] I. Gorodnitsky and B. Rao, "Sparse signal reconstruction from limited data using focuss: A re-weighted norm minimization algorithm," *IEEE Transaction on Signal Processing*, vol. 45, pp. 600–616, 1997.

[5] R. Rubinstein, A. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *IEEE Proceedings - Special Issue on Applications of Sparse Representation & Compressive Sensing*, vol. 98, no. 6, pp. 1045–1057, June 2010.

[6] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[7] K. Engan, S. Aase, and J. Hakon-Husoy, "Method of optimal directions for frame design," in *IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 5, 1999, pp. 2443–2446.

[8] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, November 2006.

[9] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (gpca)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1945–1959, 2005.

[10] B. Olshausen, P. Sallee, and M. Lewicki, "Learning sparse image codes using a wavelet pyramid architecture," in *Proc. Conf. Advances in Neural Information Processing Systems*, vol. 13, 2000, pp. 887–893.

[11] P. Sallee and B. Olshausen, "Learning sparse multiscale image representations," in *Proc. Conf. Advances Neural Information Processing Systems*, vol. 15, 2002, pp. 1327–1334.

[12] P. Sallee, "Statistical methods for image and signal processing," Ph.D. dissertation, University of California, Davis, 2004.

[13] J. Mairal, G. Sapiro, and M. Elad, "Multiscale sparse image representation with learned dictionaries," in *Proceedings of the IEEE International Conference on Image Processing*, 2007, pp. 105–108.

[14] ——, "Learning multiscale sparse representations for image and video restoration," *SIAM Multiscale Modeling and Simulation*, vol. 7, no. 1, pp. 214–241, April 2008.

[15] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning sparse dictionaries for sparse signal representation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1553–1564, March 2010.

[16] R. Neff and A. Zakhor, "Matching pursuit video coding - part 1: Dictionary approximation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 1, pp. 13–26, January 2002.

[17] S. Mallat and G. Peyré, "Orthogonal bandlet bases for geometric images approximation," *Communications on Pure and Applied Mathematics*, vol. 61, no. 9, pp. 1173–1212, February 2008.

[18] Delaunay, M. X. Chabert, V. Charvillat, and G. Morin, "Satellite image compression by post-transforms in the wavelet domain," *Signal Processing*, vol. 90, no. 2, pp. 599–610, February 2010.

[19] P. Burt and E. Adelson, "The lapalacian pyramid as a compact image code," *IEEE Transactions on Communication*, vol. COM-31, pp. 532–540, April 1983.

[20] I. Daubechies, *Ten Lectures on Wavelets*, ser. (CBMS-NSF Regional Conference Series in Applied Mathematics). Soc for Industrial & Applied Math, December 1992.

[21] M. Do and M. Vetterli, "Framing pyramids," *IEEE Transactions on Signal Processing*, vol. 51, no. 9, pp. 2329–2342, September 2003.

[22] E. Simoncelli and W. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *IEEE International Conference on Image Processing*, vol. 3, Oct 1995, pp. 444–447.

[23] R. Coifman, Y. Meyer, and M. Wickerhauser, "Wavelet analysis and signal processing," in *Wavelets and Their Applications*. Jones and Bartlett, 1992.

[24] M. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Transactions Image on Processing*, vol. 14, no. 12, pp. 2091–2106, December 2005.

[25] M. Do, "Directional multiresolution image representations," Ph.D. dissertation, Swiss Federal Institute of Technology Lausanne (EPFL), November 2001.

[26] E. Candès and D. Donoho, *Curves and Surfaces*. Nashville, TN: Vanderbilt University Press, 1999, ch. Curvelets - a surprisingly effective nonadaptive representation for objects with edges.

[27] E. Candès, L. Demanet, D. Donoho, and L. Ying, "Fast discrete curvelet transforms," *Multiscale Modeling and Simulation*, vol. 5, no. 3, pp. 861–899, 2006.

[28] E. Candès, "Ridgelets: theory and applications," Ph.D. dissertation, Stanford University, August 1998.

[29] D. Labate, W. Lim, G. Kutyniok, and G. Weiss, "Sparse multidimensional representation using shearlets," *Proceedings of SPIE: Wavelets XI*, vol. 5914, pp. 254–262, September 2005.

[30] E. LePennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Transactions on Image Processing*, vol. 14, no. 4, pp. 423–438, April 2005.

[31] M. Aharon, M. Elad, and A. Bruckstein, "On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them," *Journal of Linear Algebra and Applications*, vol. 416, pp. 48–67, July 2006.

[32] M. Aharon, "Overcomplete dictionaries for sparse representation of signals," Ph.D. dissertation, Technion, Israel Institute of Technology, November 2006.

[33] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, December 2006.

[34] P. Jost, S. Lesage, P. Vandergheynst, and R. Gribonval, "MoTIF: An efficient algorithm for learning translation invariant dictionaries," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, May 2006.

[35] P. Jost, "Algorithmic aspects of sparse approximations," Ph.D. dissertation, Swiss Federal Institute of Technology Lausanne (EPFL), 2007.

[36] D. Needell and J. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2008.

[37] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

[38] National Oceanic and Atmospheric Administration / U.S. Department of Commerce, "NOAA photo library," www.photolib.noaa.gov.

[39] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[40] I. Daubechies, M. Defrise, and C. De-Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, August 2004.

[41] M. Zibulevsky and M. Elad, "L1-l2 optimization in signal and image processing," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 76–88, 2010.