Proceedings of Symposia in Applied Mathematics
Volume **73**, 2016
http://dx.doi.org/10.1090/psapm/073/00633

Compressed sensing and dictionary learning

Guangliang Chen and Deanna Needell

Abstract. Compressed sensing is a new field that arose as a response to inefficient traditional signal acquisition schemes. Under the assumption that the signal of interest is *sparse*, one wishes to take a small number of linear samples and later utilize a reconstruction algorithm to accurately recover the compressed signal. Typically, one assumes the signal is sparse itself or with respect to some fixed orthonormal basis. However, in applications one instead more often encounters signals sparse with respect to a tight frame which may be far from orthonormal. In the first part of these notes, we will introduce the compressed sensing problem as well as recent results extending the theory to the case of sparsity in tight frames.

The second part of the notes focuses on dictionary learning which is also a new field and closely related to compressive sensing. Briefly speaking, a dictionary is a redundant system consisting of prototype signals that are used to express other signals. Due to the redundancy, for any given signal, there are many ways to represent it, but normally the sparsest representation is preferred for simplicity and easy interpretability. A good analog is the English language where the dictionary is the collection of all words (prototype signals) and sentences (signals) are short and concise combinations of words. Here we will introduce the problem of dictionary learning, its applications, and existing solutions.

Contents

1. Introduction

Many signals of interest contain far less information than their ambient dimension suggests, making them amenable to compression. However, traditional

2010 *Mathematics Subject Classification.* 94A12, 41A45, 42A10.

Key words and phrases. ℓ_1-analysis, ℓ_1-synthesis, tight frames, dictionary sparsity, compressed sensing, dictionary learning, image denoising, K-SVD, geometric multiresolution analysis, online dictionary learning.

D. Needell was supported by NSF CAREER #1348721 and the Alfred P. Sloan Fellowship.

signal acquisition schemes sample the entire signal only to discard most of that information during the compression process. This wasteful and costly acquisition methodology leads one to ask whether there is an acquisition scheme in which the compressed samples are obtained directly, without the need for time and resources to observe the entire signal. Surprisingly, the answer is often *yes*.

Compressive signal processing (CSP) or compressed sensing (CS) is a new and fast growing field which seeks to resolve this dilemma [**Can06**, **Don06**, **BS07**, **DSP**]. Work in CSP demonstrates that for a certain class of signals, very few *compressive samples* are necessary to accurately represent the signal. In fact, the number of samples required is proportional to the amount of information one wishes to acquire from the signal, and only weakly dependent on the signal's ambient dimension. These samples can be acquired directly from the signal via a linear mapping, and thus the costly process of observing the entire signal is completely eliminated.

Once a signal is acquired via this CSP technology, one needs an efficient algorithm to recover the signal from the compressed samples. Fortunately, CSP has also provided us with methods for recovery which guarantee tractable and robust signal reconstruction. The CSP methodology continues to impact areas ranging from imaging [**WLD$^+$06**, **LDP07**, **PPM**], analog-to-information conversion [**TWD$^+$06**, **KLW$^+$06**, **ME11**] and radar [**BS07**, **HS09**] to geophysical data analysis [**LH07**, **TSHM09**] and computational biology [**DSMB09**, **MSW$^+$10**].

An important aspect of the application of CSP to real-world scenarios is that the sparsifying basis must be known. Often, they are carefully designed based on a mathematical model of the expected kind of signal, with corresponding requirement that they possess some desired theoretical property, such as the Restricted Isometry Property [**CT05**, **CT06**]. Typical choices are random matrices with subgaussian entries or random sign entries.

The dictionary learning problem is closely related to the CSP but arises in a different context, where the main goal is to find compact and meaningful signal representations and correspondingly use them in signal and image processing tasks, such as compression [**BE08**], denoising [**BCM05**, **EA06**, **MSE**], deblurring [**HX13**], and super-revolution [**PETM09**].

Specifically, given signal data $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^L$, we train a *dictionary* $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_m] \in \mathbb{R}^{m \times L}$, which can be thought of as an overcomplete basis consisting of elementary signals (called *atoms*). We then use the learned dictionary to represent a signal $\mathbf{x} \in \mathbb{R}^L$ by finding the coefficient vector γ that satisfies the equation $\mathbf{x} = \mathbf{D}\gamma$. When the dictionary forms a basis, there is exactly one solution and thus every signal is uniquely represented as a linear combination of the dictionary atoms. While mathematically this is very simple to operate, such a unique representation has very limited expressiveness.

When $\mathbf{D}$ is an overcomplete system, the problem has more than one solution. This gives us greater flexibility in choosing which coefficient to use for the signal, and allows us to seek the most informative representation, often measured by some cost function $C(\gamma)$:

$$\gamma^* = \arg\min C(\gamma) \qquad \text{subject to} \quad \mathbf{x} = \mathbf{D}\gamma.$$

For example, if one chooses $C(\gamma) = \|\gamma\|_0$, which counts the number of nonzero entries, the above program effectively searches for the sparsest solution, a problem commonly referred to as *sparse coding* [**MZ93**, **OF96**, **CDS98**, **BDE09**].

However, similarly to CSP, the choice of the dictionary $\mathbf{D}$ is crucial but it often requires extensive effort to build it. Traditionally, the signal processing community heavily depended on the Fourier and wavelet dictionaries, which perform quite well for 1-dimensional signals. However, these dictionaries are not adequate for representing more complex natural signal data, especially in higher dimensions, so better dictionary structures were sought.

A variety of dictionaries have been developed in response to the rising need. These dictionaries emerge from one of two sources – either a mathematical model of the data [**AR77**, **Mal89**, **Dau92**, **Bas80**, **Jan81**, **CD04**, **CDDY00**], or a set of realizations of the data [**AEB05**, **AEB06**, **ZCP$^+$09**, **MBPS09**, **MBPS10**, **CM10**, **CM11b**, **ACM12**]. Dictionaries of the first type are often referred to as *analytic* dictionaries, because they are characterized by an analytic formulation and often equipped with a fast implicit implementation. In contrast, dictionaries of the second type deliver increased flexibility and possess the ability to adapt to specific signal data, and for this reason they are called *data-dependent* dictionaries. In this manuscript we focus on data-dependent dictionaries.

Organization. The rest of the lecture notes consists of four sections, the first two of which are devoted to compressive sensing and the last two to dictionary learning. In each part, we carefully present the background material, the problem being considered, existing solutions and theory, and its connection to other fields (including frames).

2. Background to Compressed signal processing

2.1. The CSP Model. In the model of CSP, the signal $\boldsymbol{f}$ in general is an element of $\mathbb{C}^d$. Linear measurements are taken of the form

$$y_i = \langle \boldsymbol{\phi}_i, \boldsymbol{f} \rangle \quad \text{for } i = 1, 2, \ldots m,$$

where $m \ll d$. The vectors $\boldsymbol{\phi}_i$ can be viewed as columns from an $m \times d$ matrix $\boldsymbol{\Phi}$, which we call the *sampling operator*, and the measurement vector $\boldsymbol{y}$ is of the form $\boldsymbol{y} = \boldsymbol{\Phi f}$. With $m \ll d$, $\boldsymbol{\Phi}$ clearly has a nontrivial nullspace and thus the problem of reconstructing $\boldsymbol{f}$ from $\boldsymbol{y}$ is ill-posed without further assumptions. The additional assumption in CSP is that the signals of interest contain far less information than the dimension d suggests. A means for quantifying this notion is called *sparsity*. We say that a signal $\boldsymbol{f} \in \mathbb{C}^d$ is s- *sparse* when it has at most s non-zero components:

$$\|\boldsymbol{f}\|_0 \overset{\text{def}}{=} |\operatorname{supp}(\boldsymbol{f})| \leq s \ll d, \tag{2.1}$$

where $\|\cdot\|_0$ denotes the ℓ_0 quasi-norm. For $1 \leq p < \infty$, $\|\cdot\|_p$ denotes the usual p-norm,

$$\|\boldsymbol{f}\|_p := \left(\sum_{i=1}^{d} |\boldsymbol{f}_i|^p \right)^{1/p},$$

and $\|\boldsymbol{f}\|_\infty = \max |f_i|$. In practice, signals are often encountered that are not exactly sparse, but whose coefficients decay rapidly. *Compressible* signals are those satisfying a power law decay:

$$|\boldsymbol{f}_k^*| \leq Rk^{(-1/q)}, \tag{2.2}$$

where $\boldsymbol{f}^*$ is a non-increasing rearrangement of $\boldsymbol{f}$, R is some positive constant, and $0 < q < 1$. Note that in particular, sparse signals are compressible and for small values of q compressibility becomes essentially the same as sparsity. In any case, compressible signals are well approximated by sparse signals since the majority of the energy in the signal is captured by a few components. If we denote by $\boldsymbol{f}_s$ the vector consisting of the s largest coefficients in magnitude of $\boldsymbol{f}$, then we see that for compressible signals $\boldsymbol{f}$ and $\boldsymbol{f}_s$ are close,

$$\|\boldsymbol{f} - \boldsymbol{f}_s\|_2 \leq Rs^{1/2-1/q} \quad \text{and} \quad \|\boldsymbol{f} - \boldsymbol{f}_s\|_1 \leq Rs^{1-1/q}.$$

Therefore when working with compressible signals we may capture the majority of the information in the signal by taking advantage of sparsity.

One observes however that this definition (2.1) of sparsity requires that the signal itself contain few non-zeros. This notion can be generalized by asking instead that the signal $\boldsymbol{f}$ of interest be sparse with respect to some *sparsifying basis*. We fix some orthonormal basis, written as the columns of the matrix $\boldsymbol{D}$. Then formally, we will again call a signal $\boldsymbol{f}$ s-sparse when

$$\boldsymbol{f} = \boldsymbol{D}\boldsymbol{x} \quad \text{with} \quad \|\boldsymbol{x}\|_0 \leq s \ll d. \tag{2.3}$$

We call $\boldsymbol{x}$ the *coefficient vector*. We will say that $\boldsymbol{f}$ is compressible when its coefficient vector $\boldsymbol{x}$ satisfies the power law decay as in (2.2). Many signals in practice are compressible in this sense. Natural signals such as images are often compressible with respect to the identity or wavelet sparsifying basis [**CT05**, **CRT06b**, **CW08**]. Likewise, manmade signals such as those in radar, medical imaging, and communications applications are compressible with respect to the Fourier basis and other sparsifying bases [**BS07**, **Rom08**]. Since $\boldsymbol{D}$ is an orthonormal system, we may think of absorbing $\boldsymbol{D}$ into the sampling operator and attempting to estimate $\boldsymbol{f}$ by estimating $\boldsymbol{x}$. From this viewpoint we can assume $\boldsymbol{f}$ is sparse with respect to the coordinate basis and acknowledge that results for this class of signals apply also to the broader class which are sparse with respect to some fixed orthonormal basis.

2.2. Sampling Mechanisms. The sampling operator $\boldsymbol{\Phi}$ is a linear map from $\mathbb{C}^d$ to some lower dimensional space $\mathbb{C}^m$. It is clear that to recover a sparse signal $\boldsymbol{f}$ from its measurements $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{f}$ one at least needs that $\boldsymbol{\Phi}$ is one-to-one on all sparse vectors. Indeed, if this is the case then to recover $\boldsymbol{f}$ from $\boldsymbol{y}$ one simply solves the minimization problem

$$\hat{\boldsymbol{f}} = \underset{\boldsymbol{g} \in \mathbb{C}^d}{\operatorname{argmin}} \|\boldsymbol{g}\|_0 \quad \text{subject to } \boldsymbol{\Phi}\boldsymbol{g} = \boldsymbol{y}. \tag{2.4}$$

Then since $\boldsymbol{\Phi}$ does not map any two sparse vectors to the same image, it must be that we recover our signal, $\hat{\boldsymbol{f}} = \boldsymbol{f}$. This minimization problem however, is intractable and NP-Hard in general [**Mut05**, Sec. 9.2.2]. We thus consider slightly stronger requirements on $\boldsymbol{\Phi}$. The first assumption one can make on the sampling operator is that its columns are *incoherent*. For a matrix $\boldsymbol{\Phi}$ with unit norm columns $\{\boldsymbol{\phi}_i\}$, we define its coherence μ to be the largest correlation among the columns,

$$\mu = \max_{i \neq j} \langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle .$$

A sampling operator is *incoherent* when its coherence μ is sufficiently small. Incoherent operators are thus those which are approximately orthonormal on sparse vectors.

An alternative property which captures this idea was developed by Candès and Tao and is called the Restricted Isometry Property (RIP) [**CT05**, **CT06**]. The RIP implies incoherence and that the operator $\boldsymbol{\Phi}$ approximately preserves the geometry of all sparse vectors. Formally, they define the restricted isometry constant δ_s to be the smallest constant such that

$$(1-\delta_s)\|\boldsymbol{f}\|_2^2 \leq \|\boldsymbol{\Phi f}\|_2^2 \leq (1+\delta_s)\|\boldsymbol{f}\|_2^2 \quad \text{for all } s\text{-sparse vectors } \boldsymbol{f}. \tag{2.5}$$

We say that the sampling operator $\boldsymbol{\Phi}$ has the RIP of order s when δ_s is sufficiently small, say $\delta_s \leq 0.1$. The important question is of course what type of sampling operators have this property, and how large does the number m of samples have to be. Fortunately, the literature in CSP has shown that many classes of matrices possess this property when the number of measurements m is *nearly linear in the sparsity s*. Two of the most important examples are the following.

Subgaussian matrices. A random variable X is subgaussian if $\mathbb{P}(|X| > t) \leq Ce^{-ct^2}$ for all $t > 0$ and some positive constants C, c. Thus subgaussian random variables have tail distributions that are dominated by that of the standard Gaussian random variable. Choosing $C = c = 1$, we trivially have that standard Gaussian matrices (those whose entries are distributed as standard normal random variables) are subgaussian. Choosing $C = \frac{1}{e}$ and $c = 1$, we see that Bernoulli matrices (those whose entries are uniform ±1) are also subgaussian. More generally, any bounded random variable is subgaussian. It has been shown that if $\boldsymbol{\Phi}$ is an $m \times d$ subgaussian matrix then with high probability $\frac{1}{\sqrt{m}}\boldsymbol{\Phi}$ satisfies the RIP of order s when m is on the order of $s \log d$ [**MPTJ08**, **RV06**].

Partial bounded orthogonal matrices. Let $\boldsymbol{\Psi}$ be an orthogonal $d \times d$ matrix whose entries are bounded by $C/\sqrt{d}$ for some constant C. A $m \times d$ partial bounded orthogonal matrix is a matrix $\boldsymbol{\Phi}$ formed by choosing m rows of such a matrix $\boldsymbol{\Psi}$ uniformly at random. Since the $d \times d$ discrete Fourier transform matrix is orthogonal with entries bounded by $1/\sqrt{d}$, the $m \times d$ random partial Fourier matrix is a partial bounded orthogonal matrix. Rudelson and Vershynin showed that such matrices satisfy the RIP with high probability when the number of measurements m is on the order of $s \log^4 d$ [**RV08**].

Work continues to be done to demonstrate other types of random matrices which satisfy the RIP so that this assumption is quite viable in many practical applications (see e.g. [**HN07**, **KW11**, **PRT11**]). Matrices with structure such as the partial Fourier are particularly important in applications since they can utilize a fast-multiply.

2.3. Current Approaches to CSP.

Since the problem (2.4) is computationally infeasible, alternative methods are needed. In addition, the signal $\boldsymbol{f}$ is often compressible rather than sparse, and the samples are often corrupted by noise so that the measurement vector is actually $\boldsymbol{y} = \boldsymbol{\Phi f} + \boldsymbol{e}$ for some error vector $\boldsymbol{e}$. An ideal recovery method would thus possess the following properties.

Nonadaptive samples: The method should utilize sampling operators $\boldsymbol{\Phi}$ which do not depend on the signal $\boldsymbol{f}$. Note that the operators satisfying the RIP above possess the nonadaptivity property.

Optimal number of samples: The number m of samples required for reconstruction should be minimal.

Uniform Guarantees: One sampling operator should suffice for recovery of *all* signals.

Robust Recovery: The method should be stable and robust to noise and provide optimal error guarantees.

Computational Complexity: The algorithm should be computationally efficient.

There are currently two main approaches in CSP which provide methods for sparse reconstruction with these ideal properties in mind. The first solves an optimization problem and the second utilizes greedy algorithms to recover the signal.

2.3.1. *Optimization based methods.* Initial work in CSP [**DH01**, **CT05**, **Don06**, **CRT06b**, **Tro06**] considered the convex relaxation of the NP-Hard problem (2.4). The closest convex norm to the ℓ_0 quasi-norm is the ℓ_1-norm, and the geometry of the ℓ_1-ball promotes sparsity. We therefore estimate a compressible signal $\boldsymbol{f}$ by the minimizer $\hat{\boldsymbol{f}}$ to the following problem

$$\hat{\boldsymbol{f}} = \operatorname*{argmin}_{\boldsymbol{g}\in\mathbb{C}^d} \|\boldsymbol{g}\|_1 \quad \text{subject to } \|\boldsymbol{\Phi g} - \boldsymbol{y}\|_2 \le \varepsilon, \tag{2.6}$$

where ε bounds the norm of the noise: $\|\boldsymbol{e}\|_2 \le \varepsilon$. This problem can be formulated as a linear program and so standard methods in Linear Programming can be used to solve it. Candès, Romberg and Tao showed that this ℓ_1-minimization problem provides the following error guarantee.

Theorem 2.1 (Candès-Romberg-Tao [**CRT06b**]). *Let $\boldsymbol{\Phi}$ be a sampling operator which satisfies the RIP. Then for any signal $\boldsymbol{f}$ and noisy measurements $\boldsymbol{y} = \boldsymbol{\Phi f} + \boldsymbol{e}$ with $\|\boldsymbol{e}\|_2 \le \varepsilon$, the solution $\hat{\boldsymbol{f}}$ to* (2.6) *satisfies*

$$\|\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2 \le C\left[\varepsilon + \frac{\|\boldsymbol{f} - \boldsymbol{f}_s\|_1}{\sqrt{s}}\right],$$

where $\boldsymbol{f}_s$ again denotes the vector of the s largest coefficients in magnitude of $\boldsymbol{f}$.

This result says that the recovery error is at most proportional to the norm of the noise in the samples and the tail of the signal. The error bound is optimal up to the precise value of the constant C [**CDD09**]. Note that when the signal $\boldsymbol{f}$ is exactly sparse and there is no noise in the samples that this result confirms that the signal $\boldsymbol{f}$ is reconstructed exactly [**CT05**]. For a compressible signal as in (2.2), this bound guarantees that

$$\|\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2 \le C[\varepsilon + Rs^{1/2-1/q}].$$

Therefore, using Gaussian or Fourier samples, by solving a linear program we can achieve an optimal error bound of this form with number of samples m approximately $s\log d$. This result thus provides uniform guarantees with optimal error bounds using few nonadaptive samples. Although linear programming methods are becoming more and more efficient, for some applications the computational cost of this approach may still be burdensome. For that reason, greedy algorithms have been proposed and may provide some advantages.

2.3.2. *Greedy methods.* Orthogonal Matching Pursuit (OMP) is an early greedy algorithm for sparse reconstruction analyzed by Gilbert and Tropp [**TG07**]. Given an exactly s-sparse signal $\boldsymbol{f}$ with noiseless samples $\boldsymbol{y} = \boldsymbol{\Phi f}$, OMP iteratively identifies elements of the support of $\boldsymbol{f}$. Once the support T is located, the signal is reconstructed by $\hat{\boldsymbol{f}} = \boldsymbol{\Phi}_T^{\dagger}\boldsymbol{y}$ where $\boldsymbol{\Phi}_T$ denotes the restriction of $\boldsymbol{\Phi}$ to the columns indexed by T and $\boldsymbol{\Phi}_T^{\dagger}$ denotes its pseudo-inverse. The critical observation which allows OMP to succeed is that when $\boldsymbol{\Phi}$ is Gaussian or more generally is incoherent, $\boldsymbol{\Phi}^*\boldsymbol{\Phi}$ is close to the identity. Thus $\boldsymbol{u} := \boldsymbol{\Phi}^*\boldsymbol{y} = \boldsymbol{\Phi}^*\boldsymbol{\Phi f}$ is in a loose sense close to $\boldsymbol{f}$, and so OMP estimates that the largest coefficient of $\boldsymbol{u}$ is in the true support of

$\boldsymbol{f}$. The contribution from that column is subtracted from the samples, and OMP repeats this process. Gilbert and Tropp showed that OMP correctly recovers a *fixed* sparse signal with high probability. Indeed, in [**TG07**] they proved the following.

Theorem 2.2 (OMP Signal Recovery [**TG07**]). *Let $\boldsymbol{\Phi}$ be an $m \times d$ (sub)Gaussian measurement matrix with $m \geq Cs \log d$ and let $\boldsymbol{f}$ be an s-sparse signal in $\mathcal{R}^d$. Then with high probability, OMP correctly reconstructs the signal $\boldsymbol{f}$ from its measurements $\boldsymbol{y} = \boldsymbol{\Phi f}$.*

Without modifications, OMP is not known to be robust to noise. Also, the result provides non-uniform guarantees; for a given sampling operator and fixed signal, OMP recovers the signal with high probability. In fact, a uniform guarantee from OMP has been proved to be impossible [**Rau08**]. However, the strong advantage of OMP over previous methods is its extremely low computational cost. Using an efficient implementation reduces the overall cost of OMP to $\mathrm{O}(smd)$ in general, and even faster when the sampling operator has a fast-multiply.

CoSaMP. Motivated by a breakthrough greedy algorithm analyzed with the RIP [**NV07b**, **NV07a**], Needell and Tropp developed the greedy method Compressive Sampling Matching Pursuit (CoSaMP) [**NT08b**, **NT08a**], which is similar in spirit to OMP. This algorithm again has a similar structure to OMP. In each iteration, multiple components are selected to be in the support of the estimation. Then the signal is estimated using this support and then pruned to maintain sparsity. A critical difference between CoSaMP and the other two matching pursuits is that in CoSaMP elements of the support which are incorrectly identified may be removed from the estimation in future iterations. Formally, the CoSaMP template is described by the pseudocode below. Note that parameters within the algorithm can of course be tuned for optimal performance.

Compressive Sampling Matching Pursuit (CoSaMP) [**NT08b**]

```
Input: Sampling operator $\boldsymbol{\Phi}$, sample vector $\boldsymbol{y} = \boldsymbol{\Phi f}$, sparsity level $s$
Output: $s$-sparse reconstructed vector $\hat{\boldsymbol{f}} = \boldsymbol{a}$
Procedure:
        Initialize: Set $\boldsymbol{a}^0 = \boldsymbol{0}$, $\boldsymbol{v} = \boldsymbol{y}$, $k = 0$. Repeat the following steps
            and increment $k$ until the halting criterion is true.
        Signal Proxy: Set $\boldsymbol{u} = \boldsymbol{\Phi}^* \boldsymbol{v}$, $\Omega = \operatorname{supp} \boldsymbol{u}_{2s}$ and merge the
            supports: $T = \Omega \cup \operatorname{supp} \boldsymbol{a}^{k-1}$.
        Signal Estimation: Using least-squares, set $\boldsymbol{b}|_T = \boldsymbol{\Phi}_T^\dagger \boldsymbol{y}$ and
            $\boldsymbol{b}|_{T^c} = \boldsymbol{0}$.
        Prune: To obtain the next approximation, set $\boldsymbol{a}^k = \boldsymbol{b}_s$.
        Sample Update: Update the current samples: $\boldsymbol{v} = \boldsymbol{y} - \Phi \boldsymbol{a}^k$.
```

Several halting criteria are offered in [**NT08b**], the simplest of which is to halt after $6s$ iterations. The authors prove the following guarantee for the CoSaMP algorithm.

Theorem 2.3 (CoSaMP [**NT08b**]). *Suppose that $\boldsymbol{\Phi}$ is an $m \times d$ sampling matrix satisfying the RIP. Let $\boldsymbol{y} = \boldsymbol{\Phi f} + \boldsymbol{e}$ be a vector of samples of an arbitrary* *signal, contaminated with noise. The algorithm CoSaMP produces an s-sparse approximation $\hat{\boldsymbol{f}}$ that satisfies*

$$\|\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2 \leq C\left[\varepsilon + \frac{\|\boldsymbol{f} - \boldsymbol{f}_s\|_1}{\sqrt{s}}\right],$$

where $\boldsymbol{f}_s$ is a best (s)-sparse approximation to $\boldsymbol{f}$. The running time is $\mathrm{O}(\mathcal{L} \cdot \log(\|\boldsymbol{f}\|_2))$, where $\mathcal{L}$ bounds the cost of a matrix–vector multiply with $\boldsymbol{\Phi}$ or $\boldsymbol{\Phi}^*$. Working storage is $\mathrm{O}(d)$.

CoSaMP utilizes minimal nonadaptive samples and provides uniform guarantees with optimal error bounds. The computational cost is proportional to the cost of applying the sampling operator and CoSaMP is therefore the first algorithm to provide optimality at every critical aspect. In addition, under a Gaussian noise model, CoSaMP and other greedy methods have guaranteed recovery error similar to the best possible obtained when the support of the signal is known [**GE12**].

Other greedy methods like the iterative hard thresholding algorithm (IHT) can also provide analogous guarantees [**BD09**]. IHT can be described by the simple recursive iteration

$$\boldsymbol{x}_k = H_s(\boldsymbol{x}_{k-1} + \boldsymbol{\Phi}(\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}_{k-1})),$$

where H_s is the thresholding operator which sets all but the largest (in magnitude) s entries to zero, and $\boldsymbol{x}_0$ can be chosen as an arbitrary starting estimate. We focus mainly on the CoSaMP greedy method and its adaptation to tight frames in these notes, but see e.g. [**BD09**, **Blu11**, **GNE$^+$12**] for similar adaptations of methods like IHT.

2.3.3. *Total variation methods.* In numerous CSP applications, the signals of interest are images. Natural images tend to be compressible with respect to some orthonormal basis such as the wavelet basis. With this notion of sparsity, one can use ℓ_1-minimization or a greedy method to recover the image from a small number of measurements. The standard results in CSP then guarantee the reconstruction error will be small, relative to the noise level and the compressibility of the signal. However, the errors using this approach arise as artifacts from high frequency oscillations, and their structure often appears displeasing to the eye, and makes image analysis challenging. An alternative is to consider the sparsity of the signal with respect to the image *gradient*, rather than some orthonormal basis. Minimizing the ℓ_1-norm of the gradient leads to the well-known *total variation* program.

The key to the total variation problem is that because of the structure of natural images, their gradients tend to be sparse. In other words, the matrix whose entries are the distances between neighboring pixels of a natural image is a compressible matrix. Concretely, we define the *total variation* (TV) of an image $\boldsymbol{X}$ as

$$\|\boldsymbol{X}\|_{TV} \overset{\text{def}}{=} \sum_{j,k}\sqrt{(\boldsymbol{X}_{j+1,k} - \boldsymbol{X}_{j,k})^2 + (\boldsymbol{X}_{j,k+1} - \boldsymbol{X}_{j,k})^2} = \sum_{j,k}|(\nabla\boldsymbol{X})_{j,k}|,$$

where $\nabla\boldsymbol{X}$ denotes the (discrete) gradient of the image. The gradient can then be defined by writing

$$\boldsymbol{X}_x : \mathbb{C}^{N\times N} \to \mathbb{C}^{(N-1)\times N}, \qquad (\boldsymbol{X}_x)_{j,k} = \boldsymbol{X}_{j+1,k} - \boldsymbol{X}_{j,k} \tag{2.7}$$

$$\boldsymbol{X}_y : \mathbb{C}^{N\times N} \to \mathbb{C}^{N\times(N-1)}, \qquad (\boldsymbol{X}_y)_{j,k} = \boldsymbol{X}_{j,k+1} - \boldsymbol{X}_{j,k}, \tag{2.8}$$

and then setting

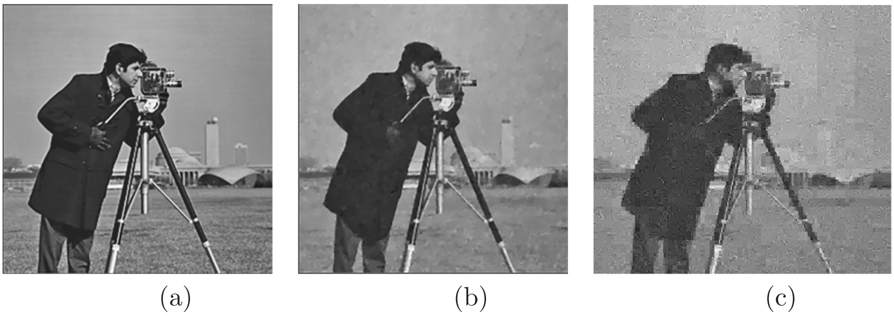
$$[\boldsymbol{\nabla X}]_{j,k} \overset{\text{def}}{=} \begin{cases} \big((\boldsymbol{X}_x)_{j,k}, (\boldsymbol{X}_y)_{j,k}\big), & 1 \le j \le N-1, \quad 1 \le k \le N-1 \\ \big(0, (\boldsymbol{X}_y)_{j,k}\big), & j = N, \quad 1 \le k \le N-1 \\ \big((\boldsymbol{X}_x)_{j,k}, 0\big), & k = N, \quad 1 \le j \le N-1 \\ \big(0, 0\big), & j = k = N. \end{cases} \tag{2.9}$$

Since images have a compressible gradient, we may consider minimizing with respect to the TV-norm:

$$\hat{\boldsymbol{X}} = \operatorname*{argmin}_{\boldsymbol{M}} \|\boldsymbol{M}\|_{TV} \quad \text{subject to} \quad \|\boldsymbol{y} - \mathcal{A}(\boldsymbol{M})\|_2 \le \varepsilon,$$

where $\boldsymbol{y} = \mathcal{A}(\boldsymbol{X}) + \boldsymbol{e}$ are noisy measurements with bounded noise $\|\boldsymbol{e}\|_2 \le \varepsilon$.

Instead of searching for a sparse image in the wavelet basis, the total variation problem searches for an image with a sparse gradient. This reduces the high frequency oscillatory artifacts from the recovered image, as seen in Figure 1.



(a) (b) (c)

Figure 1 *Images from* [**NW13**]. *(a) Original cameraman image, (b) its reconstruction from* 20% *random Fourier coefficients using total-variation minimization and (c)* ℓ_1*-minimization of its Haar wavelet coefficients.*

The benefits of using total variation norm minimization have been observed extensively and the method is widely used in practice (see e.g. [**CRT06b**, **CRT06a**, **CR05**, **OSV03**, **CSZ06**, **LDP07**, **LDSP08**, **LW11**, **NTLC08**, **MYZC08**], [**KTMJ08**, **Kee03**]). Despite this, theoretical results showing robust recovery via TV have only been obtained very recently. In [**NW12**, **NW13**], Needell and Ward prove the first robust theoretical result for total variation minimization:

Theorem 2.4 (Needell and Ward [**NW12**, **NW13**]). *From* $\mathrm{O}(s\log(N))$ *linear RIP measurements with noise level* ε*, for any* $\boldsymbol{X} \in \mathbb{C}^{N\times N}$*, the solution* $\hat{\boldsymbol{f}}$ *to the TV minimization problem satisfies*

$$\|\boldsymbol{X} - \hat{\boldsymbol{X}}\|_2 \lesssim \log(N) \cdot \left[\frac{\|\nabla[\boldsymbol{X}] - \nabla[\boldsymbol{X}]_s\|_1}{\sqrt{s}} + \varepsilon\right]$$

Analogous to the bounds of ℓ_1-optimization, this result guarantees that the recovery error is at most proportional to the noise in the samples and the "tail" of the compressible gradient. The proof technique relies on the development of an improved Sobolev inequality, and the error guarantees obtained are optimal up to the logarithmic factor. The linear measurements can be obtained from a RIP sampling operator, and have also been extended to higher dimensional arrays, see [**NW12**, **NW13**] for details.

2.4. Matrix recovery by CSP. In many applications, the signals of interest are better represented by matrices than by vectors. Such a signal may still be sparse in the sense described previously (2.1), in which case the theory above extends naturally. Alternatively, a data matrix may possess some low-rank structure. Then the question becomes, given measurements of such a low-rank data matrix, can one recover the matrix? This problem gained popularity by the now famous NetFlix problem in collaborative filtering [**RS05**, **Sre04**]. In this problem, the data matrix consists of user ratings for movies. Since not every user rates every movie and not every movie is rated by every user, only partial information about the true rating data matrix is known. From these partial measurements one wishes to obtain the true matrix containing the missing entries so that preferences can be inferred and movie recommendations can be made to the users.

A similar problem arises in the triangulation from partial data. Here, one is given some information about distances between objects in a network and wishes to recover the (low-dimensional) geometry of the network [**LLR95**, **SY07**, **Sch86**, **Sin08**]. This type of problem of course appears in many applications including remote sensing, wireless communications, and global positioning.

Formally speaking, in all of these problems we are given measurements $\boldsymbol{y} = \mathcal{A}(\boldsymbol{X})$ and wish to recover the low-rank data matrix $\boldsymbol{X}$. In general the measurement operator is of the form $\mathcal{A} : \mathbb{R}^{n\times n} \to \mathbb{R}^m$ and acts on a matrix $\boldsymbol{X}$ by

$$(\mathcal{A}(\boldsymbol{X}))_i = \langle \boldsymbol{A}_i, \boldsymbol{X}\rangle \tag{2.10}$$

where $\boldsymbol{A}_i$ are $n \times n$ matrices and $\langle\cdot,\cdot\rangle$ denotes the usual matrix inner product:

$$\langle \boldsymbol{A}, \boldsymbol{B}\rangle \overset{\text{def}}{=} \text{trace}(\boldsymbol{A}^*\boldsymbol{B}).$$

Analogous to the program (2.4), one considers solving

$$\hat{\boldsymbol{X}} = \underset{\boldsymbol{M}}{\text{argmin}}\, \text{rank}(\boldsymbol{M}) \quad \text{such that} \quad \mathcal{A}(\boldsymbol{M}) = \boldsymbol{y}.$$

However, as in the case of (L_0), the problem (2.4) is not computationally feasible in general. We thus consider instead its relaxation, which minimizes the ℓ_1-norm of its singular values.

$$\hat{\boldsymbol{X}} = \underset{\boldsymbol{M}}{\text{argmin}}\, \|\boldsymbol{M}\|_* \quad \text{such that} \quad \mathcal{A}(\boldsymbol{M}) = \boldsymbol{y}. \tag{2.11}$$

Here $\|\cdot\|_*$ denotes the *nuclear norm* which is defined by

$$\|\boldsymbol{X}\|_* = \text{trace}(\sqrt{\boldsymbol{X}^*\boldsymbol{X}}) = \|\sigma(\boldsymbol{X})\|_1,$$

where $\sigma(\boldsymbol{X})$ is the vector of singular values of $\boldsymbol{X}$.

This program (2.11) can be cast as a semidefinite program and is thus numerically feasible. Work in CSP has shown [**NRWY10**, **RFP07**, **OH10**, **CP09**] that $m \geq Cnr$ measurements suffice to recover any $n \times n$ rank-r matrix via (2.11).

2.4.1. *Matrix Decomposition.* In addition to the recovery of a low-rank structure, one may also simultaneously wish to recover a sparse component of a data matrix. That is, given a data matrix $\boldsymbol{X}$, one seeks to identify a low-rank component $\boldsymbol{L}$ and a sparse component $\boldsymbol{S}$ such that $\boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}$. For example, in a surveillance video one may wish to decompose the foreground from the background to detect moving targets. This problem has received much attention recently and can be interpreted as a robust version of principal component analysis [**CSPW11**, **EJCW09**]. The applications of this problem are numerous and include video surveillance, facial recognition, collaborative filtering and many more.

A particular challenge of this application is in the well-posedness of the decomposition problem. For example, if the sparse component also has some low-rank structure or the low-rank component is sparse, the problem does not have a unique solution. Thus some assumptions are placed on the structure of both components. Current results assume that the low-rank component $\boldsymbol{L}$ satisfies an incoherence condition (see Section 1.3 of [**EJCW09**]) which guarantees that its singular vectors are sufficiently spread and that the sparsity pattern in the sparse component $\boldsymbol{S}$ is selected uniformly at random.

The proposed method for solving this decomposition problem is *Principal Component Pursuit* [**EJCW09**, **ZLW$^+$10**] which solves the following convex optimization problem

$$(\hat{\boldsymbol{L}}, \hat{\boldsymbol{S}}) = \operatorname*{argmin}_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{S}\|_1 \quad \text{subject to} \quad \boldsymbol{L}+\boldsymbol{S}=\boldsymbol{X}. \tag{2.12}$$

Under the assumption that the low-rank component $\boldsymbol{L}$ has spread singular vectors and that the sparsity pattern of $\boldsymbol{S}$ is uniformly random, Candès et.al. show that with high probability the $n \times n$ decomposition $\boldsymbol{L}+\boldsymbol{S}$ can be exactly recovered when the rank r of $\boldsymbol{L}$ is proportional to $n/\log(n)$ and the sparsity s of $\boldsymbol{S}$ is a constant fraction of the entries, $s \leq cn^2$. This astonishing result demonstrates that the low-rank component of a data matrix can be identified even when a fixed fraction of the entries in the matrix are corrupted – and that these errors can have arbitrarily large magnitudes!

It is clear that some assumptions must be made on the individual components in the decomposition for the problem to even be well-posed. However, in many applications it may not be practical to impose such randomness in the sparsity pattern of the sparse component. We discuss this further below.

3. Compressed sensing with tight frames

In the usual CSP framework, the signal $\boldsymbol{f}$ is assumed to be sparse as in (2.3) or compressible with respect to some *orthonormal basis*. As mentioned, there are numerous applications in which the signal of interest falls into this class of signals. However, more often than not, sparsity is expressed not in terms of an orthonormal basis but in terms of an *overcomplete* dictionary. In this setting, the signal $\boldsymbol{f} = \boldsymbol{D}\boldsymbol{x}$ where $\boldsymbol{x}$ is sparse or compressible and $\boldsymbol{D}$ is an arbitrary set of column vectors which we refer to as a *dictionary* or *frame*. The dictionary need not be orthonormal or even incoherent and often it will be overcomplete, meaning it has far more columns than rows. There are numerous applications that use signals sparse in this sense, many of which are of importance to ONR. Some examples of dictionaries we encounter in practice in this setting are the following.

Oversampled DFT: Signals which are sparse with respect to the discrete Fourier matrix (DFT) are precisely those which are superpositions of sinusoids with frequencies in the lattice of those in the DFT. In practice, it is of course rare to encounter such signals. Therefore one often considers the oversampled DFT in which the sampled frequencies are taken over even smaller fixed intervals, small intervals of varying lengths, or even randomly selected intervals. This creates an overcomplete frame that may have high coherence.

Gabor frames: Radar, sonar and other imaging systems aim to recover pulse trains whose atoms have a time-frequency structure [**FS98**]. Because of this structure, Gabor frames are widely used [**Mal99**]. Gabor frames are not incoherent and often very overcomplete.

Curvelet frames: Curvelets provide a multiscale decomposition of images, and have geometric features that distinguish them from other bases like wavelets. The curvelet transform can be viewed as a multiscale pyramid with many directions at each length scale, and needle-shaped elements at fine scales [**CD04**, **CDDY00**]. Although the transform has many properties of an orthonormal basis, it is overcomplete, and neighboring columns have high coherence.

Wavelet Frames: The undecimated wavelet transform (UWT) is a wavelet transform with a translation invariance property that the discrete wavelet transform (DWT) does not possess [**Dut89**]. The UWT is missing the downsamplers and upsamplers in the DWT but upsamples the filter coefficients by a factor of 2^k at the $(k-1)$st level. This of course makes it very overcomplete. The Unitary Extension Principle of Ron and Shen [**RS97**] enables constructions of tight wavelet frames for $L^2(\mathbb{R}^d)$ which may also be very overcomplete. The overcompleteness has been found to be helpful in image processing [**SED04**].

Concatenations: In many applications a signal may not be sparse with respect to a single orthonormal basis, but may be a composition of sparse signals from multiple orthonormal bases. For example, a linear combination of spikes and sines is sparse with respect to the concatenation of the identity and the Fourier basis. In imaging applications one may wish to take advantage of the geometry of multiple sparsifying bases such as a combination of curvelets, wavelets, and brushlets. The concatenation of these bases is overcomplete and may be highly coherent.

Such redundant dictionaries are now used widespread in signal processing and data analysis. Often, there may simply be no good sparsifying orthonormal basis such as in the applications utilizing Gabor and Curvelet frames. In addition, researchers acknowledge and take advantage of the flexibility provided by overcomplete frames. In general linear inverse problems such as deconvolution, tomography, and signal denoising, it has been observed that using overcomplete dictionaries significantly reduces artifacts and mean squared error [**SED04**, **SFM07**]. Since CSP problems are special types of inverse problems it is not surprising that redundant frames are equally helpful in this setting.

3.1. The ℓ_1-analysis approach. Since in this generalized setting the sparsity is in the coefficient vector $\boldsymbol{x}$ rather than the signal $\boldsymbol{f}$, it no longer makes sense to minimize the ℓ_1-norm of the signal itself. The intuition behind the ℓ_1-analysis method is that for many dictionaries $\boldsymbol{D}$, $\boldsymbol{D}^*\boldsymbol{f}$ will have rapidly decaying coefficients and thus it becomes natural to minimize the ℓ_1-norm of this vector. Therefore for a signal $\boldsymbol{f} = \boldsymbol{D}\boldsymbol{x}$ and noisy samples $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{f} + \boldsymbol{e}$, the ℓ_1-analysis problem constructs an estimate $\hat{\boldsymbol{f}}$ to $\boldsymbol{f}$ as the solution to the following minimization problem:

$$\hat{\boldsymbol{f}} = \operatorname*{argmin}_{\boldsymbol{g}\in\mathbb{C}^d} \|\boldsymbol{D}^*\boldsymbol{g}\|_1 \quad \text{subject to } \|\boldsymbol{\Phi}\boldsymbol{g} - \boldsymbol{y}\|_2 \leq \varepsilon,$$

where as before $\varepsilon \geq \|\boldsymbol{e}\|_2$ is a bound on the noise level.

Recently, Candès et.al. provide error bounds for ℓ_1-analysis [**CENR10**]. This result holds when the dictionary $\boldsymbol{D}$ is a *tight frame*, meaning $\boldsymbol{D}\boldsymbol{D}^*$ equals the identity. All the dictionaries mentioned above are examples of tight frames. In developing theory in this setting, an important issue that had to be addressed was the assumption on the sampling operator $\boldsymbol{\Phi}$. Since sparsity in this setting is captured in the coefficient vector rather than the signal, the following natural extension of the RIP was developed. For a given dictionary $\boldsymbol{D}$, the sampling operator $\boldsymbol{\Phi}$ satisfies the D-RIP of order s when

$$(1-\delta_s)\|\boldsymbol{D}\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{\Phi}\boldsymbol{D}\boldsymbol{x}\|_2^2 \leq (1+\delta_s)\|\boldsymbol{D}\boldsymbol{x}\|_2^2 \quad \text{for all } s\text{-sparse vectors } \boldsymbol{x}$$

for some small δ_s, say $\delta_s \leq 0.08$. Here sparsity in $\boldsymbol{x}$ is with respect to the coordinate basis. D-RIP, therefore, asks that the sampling operator $\boldsymbol{\Phi}$ be approximately orthonormal on all signals $\boldsymbol{f}$ which are sparse with respect to $\boldsymbol{D}$. Using a standard covering argument it is straightforward to show that for a dictionary $\boldsymbol{D}$ with d columns that subgaussian sampling operators satisfy the D-RIP with high probability when the number m of samples is again on the order of $s \log d$ [**CENR10**]. Moreover, if $\boldsymbol{\Phi}$ satisfies the standard RIP, then multiplying the columns by random signs yields a matrix which satisfies the D-RIP [**CENR10, KW11**]. Often, however, it may not be possible to apply random column signs to the sampling matrix. In MRI for example, one is forced to take Fourier measurements and cannot preprocess the data. Recent work by Krahmer et.al. [**KNW15**] shows that one can instead use *variable density sampling* to remove the need for these column signs. In this case, one constructs for example a randomly sub-sampled Fourier matrix by selecting the rows from the standard DFT according to some specified distribution. This shows that the same class of operators used in standard CSP can also be used in CSP with overcomplete dictionaries.

Under this assumption, the error in the estimation provided by ℓ_1-analysis is bounded by the noise level and the energy in the tail of $\boldsymbol{D}^*\boldsymbol{f}$:

Theorem 3.1 (ℓ_1-analysis Recovery [**CENR10**]). *Let $\boldsymbol{D}$ be an arbitrary tight frame and suppose the sampling operator $\boldsymbol{\Phi}$ satisfies the D-RIP of order s. Then the solution $\hat{f}$ to the ℓ_1-analysis problem satisfies*

$$\|\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2 \leq C\left[\varepsilon + \frac{\|\boldsymbol{D}^*\boldsymbol{f} - (\boldsymbol{D}^*\boldsymbol{f})_s\|_1}{\sqrt{s}}\right],$$

where $(\boldsymbol{D}^*\boldsymbol{f})_s$ denotes the largest s entries in magnitude of $\boldsymbol{D}^*\boldsymbol{f}$.

This result states that ℓ_1-analysis provides robust recovery for signals $\boldsymbol{f}$ whose coefficients $\boldsymbol{D}^*\boldsymbol{f}$ decay rapidly. Observe that when the dictionary $\boldsymbol{D}$ is the identity, this recovers precisely the error bound for standard ℓ_1-minimization. Without further assumptions or modifications, this result is optimal. The bound is the natural bound one expects since the program minimizes a sparsity promoting norm over the image of $\boldsymbol{D}^*$; if $\boldsymbol{D}^*\boldsymbol{f}$ does not have decaying coefficients, there is no reason $\boldsymbol{f}$ should be close to the minimizer.

Another recent result by Gribonval et.al. analyzes this problem using a model of *cosparsity*, which captures the sparsity in $\boldsymbol{D}^*\boldsymbol{f}$ [**NDEG11**]. Their results currently only hold in the noiseless setting, and it is not known what classes of matrices satisfy the requirements they impose on the sampling operator. This alternative model deserves further analysis and future work in this direction may provide further insights.

In most of the applications discussed, namely those using curvelets, Gabor frames, and the UWT, the coefficients of $\boldsymbol{D}^*\boldsymbol{f}$ decay rapidly. Thus for these applications, ℓ_1-analysis provides strong recovery guarantees. When the dictionary is a concatenation of bases, $\boldsymbol{D}^*\boldsymbol{f}$ will not necessarily have decaying coefficients. For example, when $\boldsymbol{D}$ consists of the identity and the Fourier bases, $\boldsymbol{D}^*\boldsymbol{f}$ can be a very flat signal even when $\boldsymbol{f}$ has a sparse representation in $\boldsymbol{D}$.

Although the D-RIP is a natural extension of the standard RIP, recent work suggests that a more generalized theory may be advantageous [**CP10**]. This framework considers sampling operators whose columns are independent random vectors from an arbitrary probability distribution. Candès and Plan show that when the distribution satisfies a simple incoherence and isotropic property that ℓ_1-minimization robustly recovers signals sparse in the standard sense. A particularly useful consequence of this approach is a logarithmic reduction in the number of random Fourier samples required for reconstruction. We propose an extension of this analysis to the setting of overcomplete dictionaries which will reduce the number of samples needed and provide a framework for new sampling strategies as well.

3.2. Greedy methods. Current analysis of CSP with overcomplete dictionaries is quite limited, and what little analysis there is has focused mainly on optimization based algorithms for recovery. Recently however, Davenport et.al. analyzed a variant of the CoSaMP method [**DW11, DNW13**] summarized by the following algorithm. We use the notation $\mathcal{S}_{\boldsymbol{D}}(\boldsymbol{u}, s)$ to denote the support of the best s-sparse representation of $\boldsymbol{u}$ with respect to the dictionary $\boldsymbol{D}$, $\mathcal{R}(\boldsymbol{D_T})$ to denote the range of the subdictionary $\boldsymbol{D}_T$, and $\mathcal{P}_{\boldsymbol{D}}(\boldsymbol{b}, s)$ to denote the signal closest to $\boldsymbol{b}$ which has an s-sparse representation in $\boldsymbol{D}$.

CoSaMP with arbitrary dictionaries

```
Input: Sampling operator $\boldsymbol{\Phi}$, dictionary $\boldsymbol{D}$, sample vector $\boldsymbol{y} = \boldsymbol{\Phi f}$, sparsity level $s$
Procedure:
    Initialize: Set $\hat{\boldsymbol{f}} = \boldsymbol{0}$, $\boldsymbol{v} = \boldsymbol{y}$. Repeat the following:
    Signal Proxy: Set $\boldsymbol{u} = \boldsymbol{\Phi}^*\boldsymbol{v}$, $\Omega = \mathcal{S}_{\boldsymbol{D}}(\boldsymbol{u}, 2s)$ and merge supports:
        $T = \Omega \cup \mathcal{S}_{\boldsymbol{D}}(\hat{\boldsymbol{f}}, 2s)$
    Signal Estimation: Set $\boldsymbol{b} = \operatorname{argmin}_{\boldsymbol{z}} \|\boldsymbol{y} - \boldsymbol{\Phi z}\|_2$ s.t. $\boldsymbol{z} \in \mathcal{R}(\boldsymbol{D_T})$
    Prune: To obtain the next approximation, set $\hat{\boldsymbol{f}} = \mathcal{P}_{\boldsymbol{D}}(\boldsymbol{b}, s)$.
    Sample Update: Update the current samples: $\boldsymbol{v} = \boldsymbol{y} - \Phi\hat{\boldsymbol{f}}$.
Output: $s$-sparse reconstructed vector $\hat{\boldsymbol{f}}$
```

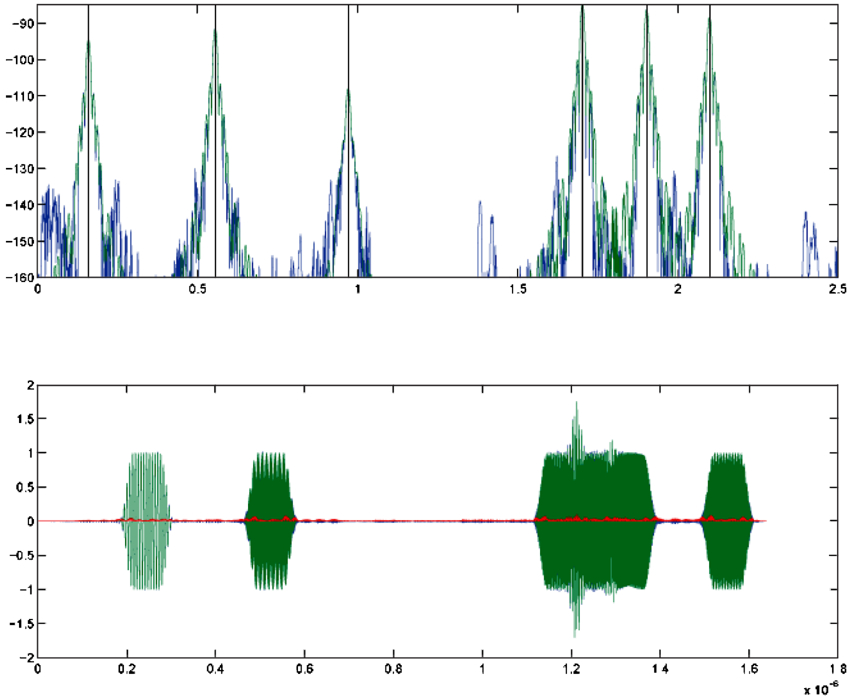


Figure 2 From [**CENR10**]. Recovery in both the time (below) and frequency (above) domains by ℓ_1-analysis after one reweighted iteration. Blue denotes the recovered signal, green the actual signal, and red the difference between the two. The RMSE is less than a third of that in Figure 2

It was recently proved that this version of CoSaMP provides robust recovery of sparse signals with respect to $\boldsymbol{D}$ when the sampling operator satisfies the $\boldsymbol{D}$-RIP [**DNW13**]. Similar results have been obtained using the co-sparse model [**GNE$^+$12**] and Iterative Hard Thresholding (IHT) [**Blu11**].

The major drawback to these results is that the projection operators $\mathcal{P}_{\boldsymbol{D}}$ and $\mathcal{S}_{\boldsymbol{D}}$ cannot in general be implemented efficiently. Indeed, Giryes and Needell [**GN13**] relax the assumptions of these operators from Davenport et al. but still require the following.

Definition 3.2. *A pair of procedures $\mathcal{S}_{\zeta k}$ and $\tilde{\mathcal{S}}_{\tilde{\zeta} k}$ implies a pair of near-optimal projections $\mathbb{P}_{\mathcal{S}_{\zeta k}(\cdot)}$ and $\mathbb{P}_{\tilde{\mathcal{S}}_{\tilde{\zeta} k}(\cdot)}$ with constants C_k and $\tilde{C}_k$ if for any $\mathbf{z} \in \mathbb{R}^d$, $|\mathcal{S}_{\zeta k}(\mathbf{z})| \leq \zeta k$, with $\zeta \geq 1$, $\left|\tilde{\mathcal{S}}_{\tilde{\zeta} k}(\mathbf{z})\right| \leq \tilde{\zeta} k$, with $\tilde{\zeta} \geq 1$, and*

$$(3.1)\quad \|\mathbb{P}_{\mathcal{S}_{\zeta k}(\mathbf{z})}\mathbf{z}\|_2^2 \leq C_k\|\mathbf{z} - \mathbb{P}_{\mathcal{S}_k^*(\mathbf{z})}\mathbf{z}\|_2^2 \quad \textit{as well as} \quad \|\mathbb{P}_{\tilde{\mathcal{S}}_{\tilde{\zeta} k}(\mathbf{z})}\mathbf{z}\|_2^2 \geq \tilde{C}_k\|\mathbb{P}_{\mathcal{S}_k^*(\mathbf{z})}\mathbf{z}\|_2^2,$$

where $\mathbb{P}_{\mathcal{S}_k^*}$ denotes the optimal projection:

$$\mathcal{S}_k^*(\mathbf{z}) = \underset{|T| \leq k}{\operatorname{argmin}}\|\mathbf{z} - \mathbb{P}_T\mathbf{z}\|_2^2.$$

Under the assumption that one has access to such near-optimal projections, signal recovery can be obtained by the following result.

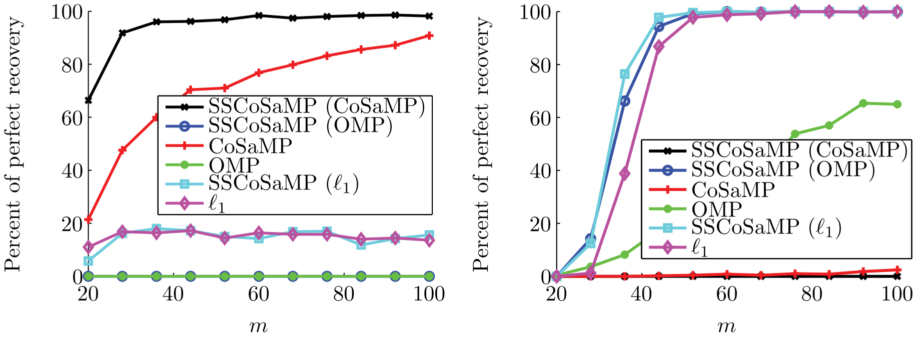


Figure 3 From [**DNW13**]. Percentage of signal recovery for CoSaMP variants and standard methods. Left: Sparse coefficients are clustered together. Right: Sparse coefficients are well separated.

Theorem 3.3 (Signal Space CoSaMP [**GN13**]). *Let* $\mathbf{M}$ *satisfy the* $\mathbf{D}$*-RIP* (3.1) *with a constant* $\delta_{(3\zeta+1)k}$ $(\zeta \geq 1)$*. Suppose that* $\mathcal{S}_{\zeta k}$ *and* $\tilde{\mathcal{S}}_{2\zeta k}$ *are a pair of near optimal projections (as in Definition* 3.2*) with constants* C_k *and* $\tilde{C}_{2k}$*. Apply SS-CoSaMP (with* $a = 2$*) and let* $\mathbf{x}^t$ *denote the approximation after* t *iterations. If* $\delta_{(3\zeta+1)k} < \epsilon^2_{C_k, \tilde{C}_{2k}, \gamma}$ *and*

$$(1 + C_k)\left(1 - \frac{\tilde{C}_{2k}}{(1+\gamma)^2}\right) < 1, \tag{3.2}$$

then after a constant number of iterations t^* *it holds that*

$$\|\mathbf{x}^{t^*} - \mathbf{x}\|_2 \leq \eta_0 \|\mathbf{e}\|_2, \tag{3.3}$$

where γ *is an arbitrary constant, and* η_0 *is a constant depending on* $\delta_{(3\zeta+1)k}$*,* C_k*,* $\tilde{C}_{2k}$ *and* γ*. The constant* $\epsilon_{C_k, \tilde{C}_{2k}, \gamma}$ *is greater than zero if and only if* (3.2) *holds.*

Unfortunately, it is unknown whether there exist efficient approximate-projections for general redundant frames that satisfy Definition 3.2. Empirically, however, traditional CSP methods like OMP, CoSaMP, or ℓ_1-minimization often provide accurate recovery [**DW11**, **DNW13**, **GNE$^+$12**]. We also find that the method used to solve the projection may have a significant impact. For example, Figure 3 (left) shows the percentage of correctly recovered signals (as a function of the number of measurements m) with a $256 \times 4(256)$ oversampled Fourier dictionary in which CoSaMP with projection approximated by CoSaMP clearly outperforms the other CoSaMP methods as well as the standard methods.

On the other hand, if one employs the co-sparse or "analysis-sparse" model, a need for such projections can be eliminated. Rather than assuming the signal is sparse in the overcomplete frame $\mathbf{D}$ (that is, $\mathbf{f} = \mathbf{D}\mathbf{x}$ for sparse $\mathbf{x}$, the analysis-sparse model assumes that the analysis coefficients $\mathbf{D}^*\mathbf{f}$ are sparse (or approximately sparse). Foucart, for example, shows that under the analysis-sparse model hard thresholding algorithms provide the same guarantees as ℓ_1-minimization without the need for approximate projections [**F15**]. Of course, these two models of sparsity can be quite different for frames of interst, and the practicality of either may vary from application to application.

4. Dictionary Learning: An introduction

In the CS framework, the dictionary $\mathbf{D}$ is often chosen as a random matrix satisfying the Restricted Isometry Property (e.g., Subgaussian or partial bounded orthogonal matrices), or designed based on intuitive expectations of the signal of interest (such as the oversampled DFT, Gabor frames, wavelets, and curvelets). The resulting dictionary is thus not directly related to the observed signals. However, in reality, the observed data often do not obey those model assumptions, so that pre-designed dictionaries would not work well. Accordingly, it is important to consider dictionaries which adapt to the observed data, often called *data-dependent dictionaries*. Starting this section, we talk about how to learn such dictionaries directly from data and apply them to image processing tasks.

Notation. We often use boldface lowercase Roman letters to represent vectors (for example $\mathbf{x}, \mathbf{y}, \mathbf{e}$) and boldface uppercase Roman letters for matrices (e.g., $\mathbf{A}, \mathbf{D}$). For any such vector (e.g. $\mathbf{x}$), we use the plain version of the letter plus a subscript (i.e., x_i) to refer to the specific entry of the vector. Meanwhile, for some vectors and matrices that are interpreted as coefficients, we will use Greek letters to denote them. In these cases, we use lowercase letters with subscripts to represent vectors (e.g., γ_i) and uppercase letters for matrices (e.g. Γ). For a matrix $\mathbf{A}$, we write A_{ij} to denote its (i, j) entry; we use $\mathbf{A}(:, j)$ to denote the jth column of $\mathbf{A}$ and $\mathbf{A}(i, :)$ its ith row. For any $0 < p < \infty$, the p-norm of a vector $\mathbf{x} \in \mathbb{R}^L$ is defined as

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{L} |x_i|^p \right)^{1/p}.$$

If $p = 0$, then $\|\mathbf{x}\|_0$ counts the number of its nonzero entries:

$$\|\mathbf{x}\|_0 = \#\{i \mid x_i \neq 0\}.$$

The Frobenius norm of a matrix $\mathbf{A}$ is

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} A_{ij}^2},$$

and its $\ell_{1,1}$ norm is

$$\|\mathbf{A}\|_{1,1} = \sum_j \|\mathbf{A}(:, j)\|_1$$

If $\mathbf{A}$ is a square matrix, then its trace is defined as

$$\operatorname{trace}(\mathbf{A}) = \sum_i A_{ii}.$$

4.1. The Dictionary Learning Problem. Suppose we are given a finite set of training signals in $\mathbb{R}^L$, for example, $\sqrt{L} \times \sqrt{L}$ pixel images or $\sqrt{L} \times \sqrt{L}$ patches taken from a large digital image. We want to to learn a collection of atomic signals called *atoms*, directly from the given signals so that they can be represented as, or closely approximated by, linear combinations of few atoms. A good analog of this problem is the construction of the English dictionary from many sentences or the recovery of the periodic table of chemical elements from a large variety of materials.

Specifically, given the training data $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^L$, and positive integers m, s, we wish to find an $L \times m$ matrix $\mathbf{D}$ and s-sparse vectors $\gamma_1, \ldots, \gamma_n \in \mathbb{R}^m$ such that

$\mathbf{D}\gamma_i$ is "close" to $\mathbf{x}_i$ for all i. Using the ℓ_2 norm to quantify the error, we formulate the following dictionary learning problem:

$$\min_{\mathbf{D},\gamma_1,\ldots,\gamma_n} \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2^2 \qquad \text{such that} \quad \|\gamma_i\|_0 \le s, \ \text{for all } i. \tag{4.1}$$

Here, $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_m] \in \mathbb{R}^{L\times m}$ is called the *dictionary*, and its columns represent atoms. The vector $\gamma_i \in \mathbb{R}^m$, with at most s nonzero entries, contains the coefficients needed by the columns of $\mathbf{D}$ to linearly represent $\mathbf{x}_i$. To make the choices of $\mathbf{D}$ and γ_i unique, we constrain the columns of $\mathbf{D}$ to be on the unit sphere in $\mathbb{R}^L$, i.e., $\|\mathbf{d}_i\|_2 = 1$. The dictionary size m is allowed to exceed the ambient dimension L in order to exploit redundancy. In contrast, the sparsity parameter often satisfies $s \ll L$.

In the special case where each γ_i is enforced to be 1-sparse (i.e., $s = 1$) with the only nonzero entry being 1, the problem in (4.1) aims to use the most similar atom to represent each signal. This corresponds to the Kmeans clustering problem [**Mac67**], where the training data are divided into n disjoint subsets, each surrounding a unique atom as its center, such that points in each subset are closer to the corresponding center than to other centers. Here, we mention a recent paper by Awasthi et al. [**ABC$^+$15**] which provides global recovery guarantees for an SDP relaxation of the Kmeans optimization problem.

Let us look at an example. Suppose we extract all 8×8 patches from a 512×512 digital image and consider them as our signal data. Here, the signal dimension $L = 64$, but the number of signals is very large ($n \approx 512^2$). A typical choice of the dictionary size is $m = 256$, which is four times as large as the signal dimension L so that $\mathbf{D}$ is overcomplete. Lastly, s is often set to some positive integer not more than 10. Performing dictionary learning in this setting is thus equivalent to finding 256 elementary image patches so that each original patch can be most closely approximated by a linear combination of at most 10 elementary patches.

Note that in (4.1) we used the square loss function to quantify the representation error

$$\ell(\mathbf{x}_i, \mathbf{D}) = \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2^2,$$

but this can be replaced by any other loss function, for example ℓ_1. The dictionary is considered "good" at representing the signals if the total loss is "small". Furthermore, the fewer columns $\mathbf{D}$ has, the more efficient it is.

If we let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ and $\Gamma = [\gamma_1, \ldots, \gamma_n]$ be two matrices representing respectively the signals and the coefficients in columns, then the dictionary learning problem in (4.1) can be readily rewritten as a matrix factorization problem

$$\min_{\mathbf{D},\Gamma} \|\mathbf{X} - \mathbf{D}\Gamma\|_F^2 \qquad \text{such that} \quad \|\gamma_i\|_0 \le k, \ \text{for all } i. \tag{4.2}$$

Here, the matrix $\mathbf{D}$ is required to have unit-norm columns while Γ must be column-sparse.

In some cases we are not given the signal sparsity s but a precision requirement ϵ on the approximation error for individual signals. We then reformulate the above problem as follows:

$$\min_{\mathbf{D},\gamma_1,\ldots,\gamma_n} \sum_{i=1}^{n} \|\gamma_i\|_0 \qquad \text{such that} \quad \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2 \le \varepsilon, \ \text{for all } i \tag{4.3}$$

Here, the objective function can be thought of the total cost for representing the signals with respect to a dictionary.

The two formulations of the dictionary learning problem in (4.1) and (4.3) can be unified into a single problem without mentioning s or ϵ:

$$\min_{\mathbf{D},\Gamma} \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2^2 + \lambda\|\gamma_i\|_0. \tag{4.4}$$

Here, λ is a regularization parameter whose role is to balance between representation error and cost (i.e., sparsity). That is, large values of λ force the ℓ_0 penalty term to be small, leading to very sparse representations. On the other hand, smaller values of λ place a smaller weight on sparsity and correspondingly enforce the program to significantly reduce the total error.

Unfortunately, the combinatorial nature of the ℓ_0 penalty requires an exhaustive search for the support set of each coefficient vector γ_i, making none of the problems (4.1)-(4.4) practically tractable (in fact, they are all NP-hard). One often replaces it by the ℓ_1 penalty (which is the closest convex norm) and considers instead

$$\min_{\mathbf{D},\gamma_1,\ldots,\gamma_n} \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2^2 + \lambda\|\gamma_i\|_1, \tag{4.5}$$

or its matrix version

$$\min_{\mathbf{D},\Gamma} \|\mathbf{X} - \mathbf{D}\Gamma\|_F^2 + \lambda\|\Gamma\|_{1,1}, \tag{4.6}$$

hoping that the new problem still preserves, at least approximately, the solution of the original problem. The problem in (4.6) is now convex in each of the variables $\mathbf{D}, \Gamma$, but not jointly convex. It is thus often solved by fixing one of $\mathbf{D}$ and Γ and updating the other in an alternating fashion. From now on, we will focus on (4.5) and its matrix version (4.6) due to its tractability and unifying nature.

4.2. Connections to several other fields. Dictionary learning (DL) is closely related to the following fields.

4.2.1. *Compressive sensing (CS).* In DL both the dictionary and sparse coefficients are simultaneously learned from the training data. When the dictionary $\mathbf{D}$ is fixed, the optimization problem in (4.5) is over the coefficients $\gamma_1, \ldots, \gamma_n$, in which case the n terms in the sum of (4.5) can be decoupled, leading to n similar problems:

$$\min_{\gamma} \|\mathbf{x}_i - \mathbf{D}\gamma\|_2^2 + \lambda\|\gamma\|_1. \tag{4.7}$$

This is exactly the *sparse coding* problem, studied extensively in the CS framework [**BDE09**]. Indeed, the CS research has shown that the relaxation to the $\|\gamma\|_1$ penalty (from $\|\gamma\|_0$) preserves the sparse solution, at least when $\mathbf{D}$ satisfies the RIP condition [**CRT06b**]. Additionally, there are efficient pursuit algorithms for solving this problem, such as the OMP [**TG07**], Basis Pursuit [**CRT06b**], and CoSamp [**NT08b, NT08a**]. Thus, one may solve this problem by using any of these pursuit algorithms, which are described in the first part of the lecture notes.

Although both CS and DL contain the same coding problem (4.7), the interpretations of the variables in (4.7) are markedly different. In the CS setting the matrix $\mathbf{D}$ serves as the sensing matrix whose rows are carefully picked to linearly interact with the unknown signal γ, and $\mathbf{x}$ represents the vector of compressed measurements. The main goal of solving (4.7) is to recover both the support set and entries of the sparse signal γ. In contrast, in the DL framework the emphasis is placed on the columns of $\mathbf{D}$ which are regarded as prototype signals and used to linearly represent the training signals of the same dimension. Accordingly, $\mathbf{x}$ should not be regarded as the measurement vector, but just a training signal. Moreover, the vector γ no longer represents the sparse signal to be recovered, but indicates the sparse linear representation of the training signal $\mathbf{x}$ with respect to the dictionary $\mathbf{D}$.

4.2.2. *Frame theory.* Frame design has been a very active research field for decades, and it lies at the intersection of many subjects, theoretical or applied, such as pure mathematics, harmonic analysis, compressive sensing, dictionary learning, and signal processing. Specifically, a frame for a finite dimensional Hilbert space (i.e., $\mathbb{R}^L$) is a spanning set $\{\mathbf{e}_k\}$ for the space, without requiring linear independence among them, that satisfies the following frame condition [**CK**]: There exist two fixed constants $B \geq A > 0$ such that for every $\mathbf{x} \in \mathbb{R}^L$,

$$A\|\mathbf{x}\|_2^2 \leq \sum_k |\langle \mathbf{x}, \mathbf{e}_k \rangle|^2 \leq B\|\mathbf{x}\|_2^2.$$

The central problem in frame theory is signal representation and reconstruction by using the frame $\{\mathbf{e}_k\}$ and its dual $\{\tilde{\mathbf{e}}_k\}$:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{e}}_k \rangle \mathbf{e}_k = \sum_k \langle \mathbf{x}, \mathbf{e}_k \rangle \tilde{\mathbf{e}}_k.$$

The concept of frames occurred much earlier than that of dictionaries, representing an important intermediate step from orthogonal bases modeling to sparse and redundant modeling. Like dictionaries, frames are overcomplete systems of signals that can represent other signals. Because of the redundancy every vector $\mathbf{x} \in \mathbb{R}^L$ has infinitely many representations, and this great flexibility is what makes both of them (frames and dictionaries) so useful in many applications: By representing a signal in many different ways, we are better able to sustain losses and noise while still having accurate and robust reconstructions.

Though both frames and dictionaries depend on the same notion of redundancy, they use it in different ways. The vectors in a frame must satisfy a frame condition which enables rigorous analysis of the system and guarantees many attractive theoretical properties. For example, though no longer an orthogonal basis, the linear coefficients can still be obtained through dot products between the (dual) frame and the signal. In contrast, dictionaries, especially data-dependent ones, do not require such a condition for its elements, but introduce a new notion of sparsity for the representation coefficients. That is, it places a small upper bound on the number of its elements that can be used for representing any given signal, so as to promote simple and interpretable representations. While the sparsity concept is quite easy to understand, its discrete nature makes it extremely difficult to analyze and often one can only consider a convex relaxation of it. Accordingly, comparing with frames, there is much less theory for dictionaries. Despite the theoretical challenge, dictionaries have proven to improve over frames in many applications, because of its greater flexibility and better ability to adapt to real data. Furthermore, the elements of a dictionary represent prototype signals and thus have a more clear physical interpretation.

4.2.3. *Subspace clustering.* Subspace clustering extends the classic Principle Component Analysis (PCA) to deal with hybrid linear data. The PCA is a linear transform which adapts to signals sampled from a Gaussian distribution, by fitting a low-dimensional subspace to the data with the lowest L_2 approximation error [**Jol02**]. Subspace clustering is a natural extension of PCA by using more than one subspace. Specifically, given a set of signals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in $\mathbb{R}^L$ which are sampled from a mixture of unknown number of subspaces with unknown dimensions, the goal of subspace clustering is to estimate the parameters of the model planes and their bases, and then cluster data according to the identified planes. It has been a very hot topic since the beginning of this century. We refer the reader to [**Vid11**] for a tutorial on this field and for an introduction to state-of-the art algorithms such as SCC [**CL09a**, **CL09b**].

From a DL perspective, the overall collection of the subspace bases forms a dictionary, and every given signal is expressed as a linear combination of several basis vectors, depending on the subspace it belongs to. In other words, the dictionary here consists of a few subdictionaries, each one reserved for a particular group of signals, and the different subdictionaries are not allowed to be mixed to form other kinds of linear representations. Clearly, such a dictionary is a lot more restrictive. Though sparsity is still enforced here, redundancy is not exploited because the dictionary is small for low dimensional subspaces. Finally, the subspace bases, which must be linearly independent, may not have the interpretation of atomic signals.

In (4.2)-(4.4), the signals all have at most s non-zeros in its representation. In other words, the size of the support set of each coefficient is no bigger than s. More importantly, there is no restriction on which combination of atoms can be used for representing a signal. Thus, there are a total of $\binom{m}{s} + \binom{m}{s-1} + \cdots + \binom{m}{1} = \binom{m+1}{s}$ possibilities for a support, where m is the dictionary size. Each such support defines a unique subspace of dimension (at most) s in $\mathbb{R}^L$, and the overall signal model is therefore a union of a large number of subspaces, to one of which each signal is believed to belong. Consequently, this model is a relaxation of the mixture of subspaces model mentioned above, and the representations here are thus more flexible and efficient.

4.3. Applications to image processing tasks.

Sparse and redundant dictionaries offer a new way for modeling complex image content, by representing images as linear combinations of few atomic images chosen from a large redundant collection (i.e., dictionary). Because of the many advantages associated with dictionaries, dictionary-based methods tend to improve over traditional image processing algorithms, leading to state-of-the-art results in practice. Below we briefly survey some of the most common imaging applications and their solution by dictionary learning. To gain a deeper and more thorough understanding of the field, we refer the reader to [**EFM10**], which is also our main reference for writing this part.

4.3.1. *Introduction.* Consider a clean image or a patch taken from it, I, of size $\sqrt{L} \times \sqrt{L}$, where $\sqrt{L}$ is a positive integer. Typically, $\sqrt{L} = 512$ (for full digital images), or $\sqrt{L} = 8$ (when operating on patches taken from a full image). We vectorize the image I to obtain $\mathbf{t} \in \mathbb{R}^L$, by following some fixed order (e.g., lexicographical order). Normally we do not observe the clean image $\mathbf{t}$, but rather a noisy measurement of it (see Fig. 4):

$$\mathbf{x} = \mathbf{t} + \mathbf{e}.$$

Here, $\mathbf{e}$ represents an (unknown) additive noise contaminating the image. Naturally, given $\mathbf{x}$, we would like to recover the true image $\mathbf{t}$, at least as closely as possible. This is the *image denoising* problem [**BCM05**, **EA06**, **MSE**].

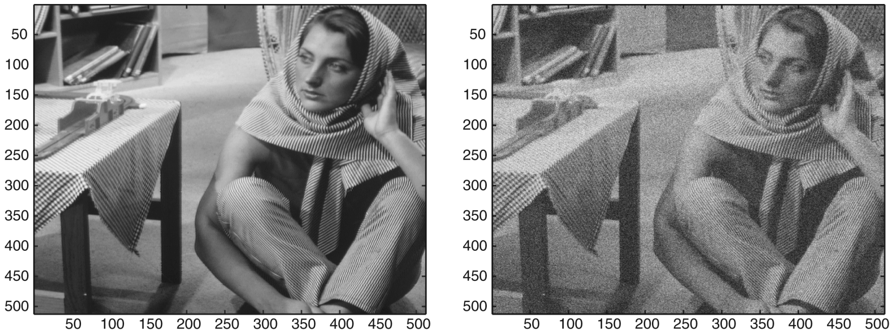


Figure 4 A clean image and its noisy version (by adding zero-mean Gaussian noise with standard deviation $\sigma = 25$). We assume that only the noisy image is given to us, and we wish to recover the clean image.

Assuming that the noise $\mathbf{e}$ has bounded norm ($\|\mathbf{e}\|_2 \le \delta$), the true image $\mathbf{t}$ and its noisy realization $\mathbf{x}$ are within a distance of δ from each other. In theory, if we know the value of δ, then we may search the δ-ball centered at $\mathbf{x}$ for the clean image $\mathbf{t}$. However, we cannot use the concept of cleanness directly. In addition, this space is prohibitively large for performing any practical task. So we need to choose a model for the clean image $\mathbf{t}$ and correspondingly focus on a smaller class of images. The "best" image in that class is then used as an estimator for the clean image $\mathbf{t}$:

$$\hat{\mathbf{t}} = \operatorname*{argmin}_{\mathbf{y}} C(\mathbf{y}) \qquad \text{subject to} \qquad \|\mathbf{x} - \mathbf{y}\|_2 \le \delta.$$

In the above, $C(\cdot)$ represents a cost function, often naturally associated with the selected model, such that smaller cost means better estimation.

For example, if we let $C(\mathbf{y}) = \|\mathbf{L}\mathbf{y}\|_2^2$, where $\mathbf{L}$ is a Laplacian matrix representing the operation of applying the Laplacian filter to the image $\mathbf{y}$, then the cost is the deviation of $\mathbf{t}$ from spatial smoothness. In other words, the class of spatially smooth images that lie in the δ-ball around $\mathbf{t}$ is considered and the most spatially smooth image is selected to estimate $\mathbf{t}$. A second example is $C(\mathbf{y}) = \|\mathbf{W}\mathbf{y}\|_1$, where $\mathbf{W}$ is a matrix representing the orthogonal wavelet transform, and the ℓ_1 norm measures the sparsity of the wavelet coefficients. This corresponds to wavelet denoising, which combines spatial smoothness (of a lower order) and a robust measure in the cost function. There are also many other choices of $C(\mathbf{y})$, e.g., the total variation measure [**ROF92**].

Recently, inspired by sparse and redundant modeling, the sparsity of the coefficient of $\mathbf{y}$ with respect to an overcomplete dictionary $\mathbf{D}$ is adopted as the cost function:

$$\min \|\gamma\|_1 \qquad \text{subject to} \qquad \|\mathbf{x} - \mathbf{D}\gamma\|_2 \le \delta.$$

Here, $\mathbf{D}$ represents a global dictionary, learned in advance from many image examples of the same size. In this case, the dictionary learning and sparse coding parts are actually decoupled from each other, and one thus solves them separately. The minimizer $\hat{\gamma}$ of the sparse coding problem corresponds to the "simplest" image with respect to the global dictionary $\mathbf{D}$, and the clean image estimate is $\hat{\mathbf{t}} = \mathbf{D}\hat{\gamma}$.

In practice, instead of working on the full image and using many similar examples, one often extracts the patches $\mathbf{p}_i$, at a fixed size, of the noisy image $\mathbf{x}$ as training signals. We then learn a (patch) dictionary $\mathbf{D}$, along with coefficients $\hat{\gamma}_i$, directly from those patches:

$$\min_{\mathbf{D},\{\gamma_i\}} \sum_i \|\gamma_i\|_1 \qquad \text{subject to} \quad \|\mathbf{p}_i - \mathbf{D}\gamma_i\|_2 \leq \delta \quad \text{for all } i.$$

The denoised patches are given by $\mathbf{D}\hat{\gamma}_i$, from which we reconstruct the clean image. Such a patch-based approach has two immediate advantages: First, the signal dimension becomes much smaller, which greatly mitigates the computational burden. Secondly, since the dictionary is self-learned, there is no need to use other exemplary images. Fig. 5 displays both a dictionary trained on patches of size 8×8 taken from the noisy image in Fig. 4 and the corresponding denoised image.

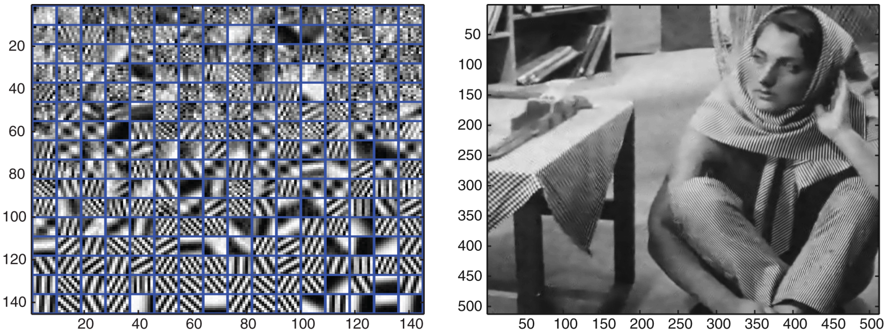


Figure 5 Trained dictionary (left) and corresponding denoised result (right), using the K-SVD algorithm [**AEB05**, **AEB06**].

More generally, we assume that we observe a noisy degraded version of $\mathbf{x}$:

$$\mathbf{x} = \mathbf{H}\mathbf{t} + \mathbf{e}$$

where $\mathbf{H}$ is a linear operator representing some kind of degradation of the signal, such as

- a blur,
- the masking of some pixels,
- the downsampling, and
- a random set of projections.

Our goal is still to recover the true image $\mathbf{t}$ from its noisy observation $\mathbf{x}$. The corresponding problems are respectively referred to as

- image deblurring [**HX13**],
- image inpainting [**MSE**],
- image super-resolution [**PETM09**], and
- compressive sensing.

When $\mathbf{H}$ is taken to be the identity operator, then the problem reduces to image denoising. These problems are all special types of inverse problems in image processing.

Similarly, if we adopt the ℓ_1 cost function and learn a dictionary $\mathbf{D}$ elsewhere from many image examples, we may then consider the following problem:

$$\min \|\gamma\|_1 \qquad \text{subject to} \qquad \|\mathbf{x} - \mathbf{H}\mathbf{D}\gamma\|_2 \leq \delta.$$

Here, we assume $\mathbf{H}$ is known to us. We solve it by regarding $\mathbf{HD}$ as a whole. The minimizer $\hat{\gamma}$ of the above problem then gives the clean image estimate $\hat{\mathbf{t}} = \mathbf{D}\hat{\gamma}$. We refer the reader to the above references (corresponding to the specific applications) for more details as well as experimental results.

5. Dictionary Learning: Algorithms

Since the beginning of this century, many algorithms have been proposed for solving the dictionary learning problem, most of which use the formulation (4.6) and have an iterative fashion. That is, by fixing one of the matrices $\mathbf{D}$ and Γ, they consider the optimization over the other variable and strive to find the best update for it; such an alternating procedure is repeated until convergence. In the following, we review three state-of-the-art dictionary learning algorithms, K-SVD [**AEB05**, **AEB06**], Geometric Multi-Resolution Analysis (GRMA) [**CM10**, **CM11b**, **ACM12**], and Online Dictionary Learning (ODL) [**MBPS09**, **MBPS10**], which have very different flavors and adequately represent their own categories. The review will also enable the reader to learn the different rationals and ideas used in the data-dependent dictionary learning research. For a more complete survey on dictionary learning approaches, we refer the reader to [**RBE10**].

5.1. K-SVD. The K-SVD algorithm is an iterative algorithm, developed by Elad et al. [**AEB05**, **AEB06**], that minimizes the expression in (4.1), or its matrix form (4.2). It consists of two stages, similarly to the Kmeans algorithm [**Mac67**].

First, at any iteration, the dictionary $\mathbf{D}$ is held fixed and the best coefficient matrix Γ is seeked. In this case, the n terms in the sum of (4.1) can be decoupled, leading to n similar problems:

$$\min_{\gamma_i} \; \|\mathbf{x}_i - \mathbf{D}\gamma_i\|_2^2 \quad \text{subject to} \quad \|\gamma_i\|_0 \leq s, \tag{5.1}$$

where i is taken to be from 1 to n. This is exactly the sparse coding problem, being solved n times. Therefore, any of the pursuit algorithms (such as OMP and CoSamp) that are mentioned in the first half of this paper may be used at this stage.

At the second stage of the same iteration, K-SVD then fixes the new coefficient matrix Γ and searches for a better dictionary $\mathbf{D}$ relative to the coefficients. However, unlike some of the approaches described in [**RBE10**] which update the whole matrix $\mathbf{D}$ by treating it as a single variable, the K-SVD algorithm updates one column at a time, fixing the other columns of $\mathbf{D}$. Meanwhile, as a byproduct, new coefficient corresponding to the updated column is also obtained. Such adoptions have at least two important advantages. First, as we shall see, the process of updating only one column of $\mathbf{D}$ at a time is a simple problem with a straightforward solution based on the singular value decomposition (SVD). Second, allowing a change in the coefficient values while updating the dictionary columns accelerates convergence, since the subsequent column updates will be based on the more relevant coefficients.

5.1.1. *Detailed description of the KSVD algorithm.* Let us present such ideas more carefully. Assume that at some iteration both Γ and all columns of $\mathbf{D}$ except one $\mathbf{d}_k$ are fixed. The goal is to update $\mathbf{d}_k$ and $\gamma^{(k)}$ simultaneously so as to reduce the overall representation error. Denote by $\gamma^{(i)}$ the ith row of Γ for all i (note the the ith column of Γ is denoted by γ_i). Then, by writing out the individual rank-1 matrices in the product $\mathbf{D}\Gamma$ and regrouping terms, we obtain from the objective function in (4.2) the following

$$\|\mathbf{X}-\mathbf{D}\Gamma\|_{\mathrm{F}}^2 = \left\|\mathbf{X}-\sum_j \mathbf{d}_j\gamma^{(j)}\right\|_{\mathrm{F}}^2 = \left\|\left(\mathbf{X}-\sum_{j\neq k}\mathbf{d}_j\gamma^{(j)}\right)-\mathbf{d}_k\gamma^{(k)}\right\|_{\mathrm{F}}^2. \tag{5.2}$$

Denoting

$$\mathbf{E}_k = \mathbf{X}-\sum_{j\neq k}\mathbf{d}_j\gamma^{(j)}, \tag{5.3}$$

which stores the errors for all the training data when the kth atom is omitted, the optimization problem in (4.2) becomes

$$\min_{\mathbf{d}_k,\gamma^{(k)}} \left\|\mathbf{E}_k-\mathbf{d}_k\gamma^{(k)}\right\|_{\mathrm{F}}^2. \tag{5.4}$$

Note that in the above equation the matrix $\mathbf{E}_k$ is considered fixed. The problem thus tries to find the closest rank-1 approximation to $\mathbf{E}_k$, expressing each of its columns as a constant multiple of $\mathbf{d}_k$. A seemingly natural solution would be to perform a rank-1 SVD of $\mathbf{E}_k$ to update both $\mathbf{d}_k$ and $\gamma^{(k)}$. However, this disregards any sparsity structure that $\gamma^{(k)}$ presents[1] and the SVD will very likely fill all its entries to minimize the objective function. Collectively, when all atoms along with their coefficients are sequentially updated, such a method would destroy the overall sparsity pattern of the coefficient matrix Γ. As a result, the convergence of the algorithm will be significantly impaired.

It is thus important to preserve the support of $\gamma^{(k)}$, when solving the above problem, to ensure convergence. The K-SVD algorithm introduces the following simple solution to address the issue. Let the support set of $\gamma^{(k)}$ be denoted by

$$\Omega_k = \{i \mid \gamma^{(k)}(i) \neq 0\},$$

and its reduced version by

$$\gamma_\Omega^{(k)} = \gamma^{(k)}(\Omega_k) = \left(\gamma^{(k)}(i)\right)_{i\in\Omega_k}.$$

We also restrict our attention to the same subset of columns of $\mathbf{E}_k$:

$$\mathbf{E}_k^\Omega = \mathbf{E}_k(:,\Omega_k).$$

Using such notation, we may rewrite the above problem as

$$\min_{\mathbf{d}_k,\gamma_\Omega^{(k)}} \left\|\mathbf{E}_k^\Omega-\mathbf{d}_k\gamma_\Omega^{(k)}\right\|_{\mathrm{F}}^2, \tag{5.5}$$

in which $\gamma_\Omega^{(k)}$ now has a full support. Since the sparsity constraint has been removed, this problem bears a simple and straightforward solution, computable from rank-1 SVD. Specifically, if the SVD of $\mathbf{E}_k^\Omega$ is given by

$$\mathbf{E}_k^\Omega = U\Sigma V^T$$

where U, V are orthonormal and Σ is diagonal, then the solution of the above problem is

$$\widetilde{\mathbf{d}}_k = U(:,1), \quad \gamma_\Omega^{(k)} = \Sigma_{11}V(:,1).$$

[1] Recall that $\gamma^{(k)}$ is the kth row of the coefficient matrix Γ which has sparse columns. So it is very likely that each row of Γ also contains many zeros and thus has a (nearly) sparse pattern.

One immediate benefit of such a solution is that the new atom $\widetilde{\mathbf{d}}_k$ remains normalized.

We now summarize the steps of K-SVD in Algorithm 1.

Algorithm 1 Pseudocode for the K-SVD Algorithm

```
Input: Training data $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, sparsity parameter $s$, initial dictionary $\mathbf{D}^{(0)}$
Output: Dictionary $\mathbf{D}$, sparse coefficients $\Gamma$
Steps:
Initialization: $J \leftarrow 1$ (iteration index)
WHILE stopping criterion not met
    • Sparse coding stage: For each data point $\mathbf{x}_i, i = 1, \ldots, n$, solve
          $\min_{\gamma_i} \|\mathbf{x}_i - \mathbf{D}^{(J-1)}\gamma_i\|_2^2$ subject to $\|\gamma_i\|_0 \le s$,
      using any pursuit algorithm (e.g. OMP). Denote the resultant coefficient matrix by $\Gamma$.
    • Dictionary update stage: For each dictionary atom $\mathbf{d}_i$ of $D^{(J-1)}, i = 1, \ldots, n$,
        – Identify the support set $\Omega_i$ of $\gamma^{(i)}$, the $i$th row of the current matrix $\Gamma$.
        – Compute
              $\mathbf{E}_i = \mathbf{X} - \sum_{j \neq i} \mathbf{d}_j \gamma^{(j)}$,
          and restrict it to the subset $\Omega_i$ of columns of $\mathbf{E}_i$ to form $\mathbf{E}_i^{\Omega}$.
        – Apply rank-1 SVD to $\mathbf{E}_i^{\Omega}$ to update $\mathbf{d}_i$ and $\gamma_{\Omega}^{(i)}$ simultaneously
    • $J \leftarrow J + 1$
ENDWHILE
Return $\mathbf{D}, \Gamma$
```

5.1.2. *A few remarks about K-SVD.* We make the following comments on K-SVD.

- The K-SVD algorithm has many advantages. For example, it is simple to implement, fast to run, and converges (assuming the pursuit algorithm used for sparse coding is accurate). It has been successfully applied to many imaging applications (e.g., [**EA06**]).
- However, the success of K-SVD depends on the choice of the initial dictionary. In other words, though it converges, it might be trapped in a suboptimal solution. Its performance also depends on the pursuit algorithm used. For example, convergence of K-SVD is guaranteed only if the pursuit algorithm solves the sparse coding problems accurately.
- The K-SVD algorithm closely resembles the Kmeans algorithm, and can be viewed as a natural extension of it. This explains why K-SVD shares the same advantages and drawbacks with Kmeans, like those mentioned above.

- The dictionary built by K-SVD is completely unstructured, making sparse representation of any new signal (relative to the trained dictionary) a nontrivial task, which requires to use one of the pursuit algorithms.

5.2. Geometric Multi-Resolution Analysis (GMRA). The GMRA [**CM10**, **CM11b**, **ACM12**] is a wavelet-like algorithm based on a geometric multiresolution analysis of the data. It builds data-dependent dictionaries that are structured and multiscale. When the data is sampled from a manifold, there are theoretical guarantees for both the size of the dictionary and the sparsity of the coefficients.

Specifically, let $(\mathcal{M}, \rho, \mu)$ be a metric measure space with μ a Borel probability measure, ρ a metric function, and $\mathcal{M} \subseteq \mathbb{R}^D$ a set. For example, $(\mathcal{M}, \rho, \mu)$ can be a smooth compact Riemannian manifold of dimension d isometrically embedded in $\mathbb{R}^D$, endowed with the natural volume measure. The GMRA construction consists of three steps. First, it performs a nested geometric decomposition of the set $\mathcal{M}$ into dyadic cubes at a total of J scales, arranged in a tree. Second, it obtains an affine approximation in each cube, generating a sequence of piecewise linear sets $\{\mathcal{M}_j\}_{1\le j\le J}$ approximating $\mathcal{M}$. Lastly, it constructs low-dimensional affine difference operators that efficiently encode the differences between $\mathcal{M}_j$ and $\mathcal{M}_{j+1}$, producing a hierarchically organized dictionary that is adapted to the data. Associated to this dictionary, there exist efficient geometric wavelet transforms, an advantage not commonly seen in the current dictionary learning algorithms.

5.2.1. *Multiscale Geometric Decomposition.* For any $\mathbf{x} \in \mathcal{M}$ and $r > 0$, we use $B_r(\mathbf{x})$ to denote the ball in the set $\mathcal{M}$ of radius r centered at $\mathbf{x}$. We start by a spatial multiscale decomposition of $\mathcal{M}$ into *dyadic cubes*, $\{\mathbb{C}_{j,k}\}_{k\in\Gamma_j, j\in\mathbb{Z}}$, which are open sets in $\mathcal{M}$ such that

(i) for every $j \in \mathbb{Z}$, $\mu(\mathcal{M} \setminus \cup_{k\in\Gamma_j}\mathbb{C}_{j,k}) = 0$;

(ii) for $j' \ge j$ either $\mathbb{C}_{j',k'} \subseteq \mathbb{C}_{j,k}$ or $\mu(\mathbb{C}_{j,k} \cap \mathbb{C}_{j',k'}) = 0$;

(iii) for any $j < j'$ and $k' \in \Gamma_{j'}$, there exists a unique $k \in \Gamma_j$ such that $\mathbb{C}_{j',k'} \subseteq \mathbb{C}_{j,k}$;

(iv) each $\mathbb{C}_{j,k}$ contains a point $\mathbf{c}_{j,k}$, called center of $\mathbb{C}_{j,k}$, such that

$$B_{c_1\cdot 2^{-j}}(\mathbf{c}_{j,k}) \subseteq \mathbb{C}_{j,k} \subseteq B_{\min\{c_2\cdot 2^{-j},\mathrm{diam}(\mathcal{M})\}}(\mathbf{c}_{j,k}),$$

for fixed constants c_1, c_2 depending on intrinsic geometric properties of $\mathcal{M}$. In particular, we have $\mu(\mathbb{C}_{j,k}) \sim 2^{-dj}$;

(v) the boundary of each $\mathbb{C}_{j,k}$ is piecewise smooth.

The properties above imply that there is a natural tree structure $\mathcal{T}$ associated to the family of dyadic cubes: for any (j,k), we let

$$\mathrm{children}(j,k) = \{k' \in \Gamma_{j+1} : \mathbb{C}_{j+1,k'} \subseteq \mathbb{C}_{j,k}\}.$$

Note that $\mathbb{C}_{j,k}$ is the disjoint union of its children. For every $\mathbf{x} \in \mathcal{M}$, with abuse of notation we let $(j,\mathbf{x})$ be the unique $k \in \Gamma_j$ such that $\mathbf{x} \in \mathbb{C}_{j,k}$.

5.2.2. *Multiscale Singular Value Decomposition (MSVD).* We start with some geometric objects that are associated to the dyadic cubes. For each $\mathbb{C}_{j,k}$ we define the mean

$$\mathbf{c}_{j,k} := \mathbb{E}_\mu[\mathbf{x}|\mathbf{x} \in \mathbb{C}_{j,\mathbf{x}}] = \frac{1}{\mu(\mathbb{C}_{j,k})}\int_{\mathbb{C}_{j,k}} \mathbf{x}\, d\mu(\mathbf{x}), \tag{5.6}$$

and the covariance operator restricted to $C_{j,k}$,

$$\mathrm{cov}_{j,k} = \mathbb{E}_\mu[(\mathbf{x} - \mathbf{c}_{j,k})(\mathbf{x} - \mathbf{c}_{j,k})^*|\mathbf{x} \in \mathbb{C}_{j,k}]. \tag{5.7}$$

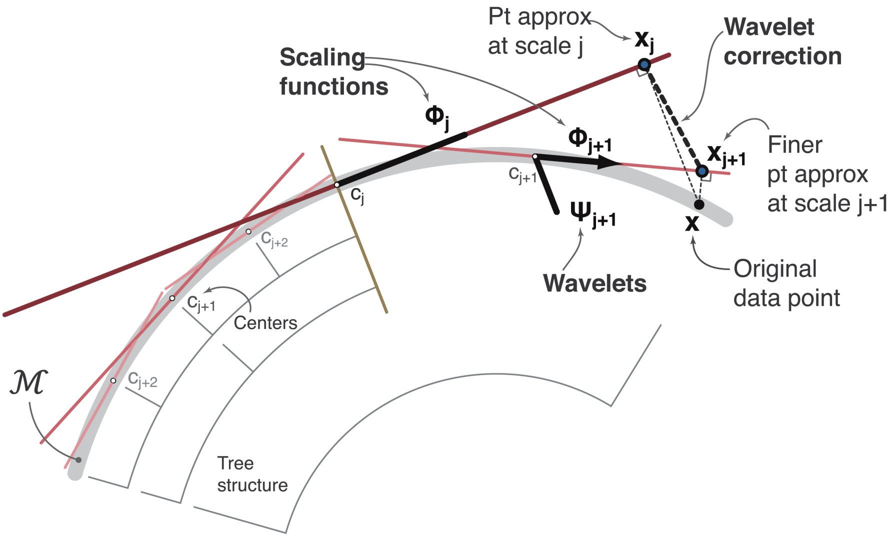


Figure 6 Construction of GMRA.

Let τ_0 be some method that chooses local dimensions $d_{j,k}$ at the dyadic cubes $\mathbb{C}_{j,k}$. For example, when the data is sampled from a manifold of dimension d, τ_0 assigns $d_{j,k} = d$ for all (j,k). In the setting of nonmanifold data, τ_0 can instead picks $d_{j,k}$ so that certain (absolute/relative) error criterion is met. We then compute the rank-$d_{j,k}$ Singular Value Decomposition (SVD) of the above covariance matrix

$$\mathrm{cov}_{j,k} \approx \Phi_{j,k}\Sigma_{j,k}\Phi_{j,k}^*, \tag{5.8}$$

and define the approximate local tangent space

$$\mathbb{V}_{j,k} := \mathcal{V}_{j,k} + \mathbf{c}_{j,k}, \quad \mathcal{V}_{j,k} = \langle \Phi_{j,k} \rangle\,, \tag{5.9}$$

where $\langle \Phi_{j,k} \rangle$ denotes the span of the columns of $\Phi_{j,k}$. Let $\mathbb{P}_{j,k}$ be the associated affine projection onto $\mathbb{V}_{j,k}$: for any $\mathbf{x} \in \mathbb{C}_{j,k}$,

$$\mathbb{P}_{j,k}(\mathbf{x}) := \mathbf{P}_{j,k} \cdot (\mathbf{x} - \mathbf{c}_{j,k}) + \mathbf{c}_{j,k}, \quad \mathbf{P}_{j,k} = \Phi_{j,k}\Phi_{j,k}^*, \tag{5.10}$$

and define a coarse approximation of $\mathcal{M}$ at scale j,

$$\mathcal{M}_j := \cup_{k\in\Gamma_j}\mathbb{P}_{j,k}(\mathbb{C}_{j,k}). \tag{5.11}$$

When $\mathcal{M}$ is a manifold and $d_{j,k} = d$, $\mathcal{M}_j \to \mathcal{M}$ in the Hausdorff distance, as $J \to +\infty$.

5.2.3. *Construction of Geometric Wavelets.* We introduce our wavelet encoding of the difference between $\mathcal{M}_j$ and $\mathcal{M}_{j+1}$, for $j \geq 0$. Fix a point $\mathbf{x} \in \mathbb{C}_{j+1,k'} \subset \mathbb{C}_{j,k}$. There are two ways to define its approximation at scale j, denoted by $\mathbf{x}_j$ or equivalently by $P_{\mathcal{M}_j}(\mathbf{x})$:

$$\mathbf{x}_j := \mathbb{P}_{j,k}(\mathbf{x}); \tag{5.12}$$

and

$$\mathbf{x}_j := \mathbb{P}_{j,k}(\mathbf{x}_{j+1}), \quad \text{for } j < J; \quad \text{and } \mathbf{x}_J = \mathbb{P}_{J,\mathbf{x}}(\mathbf{x}). \tag{5.13}$$

Clearly, the first definition is the direct projection of the point $\mathbf{x}$ onto the approximate local tangent subspace $\mathbb{V}_{j,k}$ (thus it is the closest approximation to x from

$\mathbb{V}_{j,k}$ in the least-squares sense). In contrast, the second definition is the successive projection of $\mathbf{x}$ onto the sequence of tangent spaces $\mathbb{V}_{J,\mathbf{x}}, \ldots, \mathbb{V}_{j,\mathbf{x}}$.

Regardless of which definition, we will see that the difference $\mathbf{x}_{j+1} - \mathbf{x}_j$ is a high-dimensional vector in $\mathcal{R}^D$, however it may be decomposed into a sum of vectors in certain well-chosen low-dimensional spaces, shared across multiple points, in a multiscale fashion.

For reasons that will become obvious later, we define the geometric wavelet subspaces and translations as

$$W_{j+1,k'} := (\mathbf{I} - \mathbf{P}_{j,k})\,\mathcal{V}_{j+1,k'}; \tag{5.14}$$

$$\mathbf{w}_{j+1,k'} := (\mathbf{I} - \mathbf{P}_{j,k})(\mathbf{c}_{j+1,k'} - \mathbf{c}_{j,k}), \tag{5.15}$$

and let $\Psi_{j+1,k'}$ be an orthonormal basis for $W_{j+1,k'}$ which we call a *geometric wavelet basis* (see Fig. 6).

We proceed using the two definitions of $\mathbf{x}_j$ separately. First, with (5.12), we have for $j \leq J-1$

$$\begin{aligned} Q_{\mathcal{M}_{j+1}}(\mathbf{x}) &:= \mathbf{x}_{j+1} - \mathbf{x}_j \\ &= \mathbf{x}_{j+1} - \mathbb{P}_{j,k}(\mathbf{x}_{j+1}) + \mathbb{P}_{j,k}(\mathbf{x}_{j+1}) - \mathbb{P}_{j,k}(\mathbf{x}) \\ &= (\mathbf{I} - \mathbf{P}_{j,k})(\mathbf{x}_{j+1} - \mathbf{c}_{j,k}) + \mathbf{P}_{j,k}(\mathbf{x}_{j+1} - \mathbf{x}). \end{aligned} \tag{5.16}$$

Since $\mathbf{x}_{j+1} - \mathbf{c}_{j,k} = \mathbf{x}_{j+1} - \mathbf{c}_{j+1,k'} + \mathbf{c}_{j+1,k'} - \mathbf{c}_{j,k}$ and $\mathbf{x}_{j+1} - \mathbf{c}_{j+1,k'} \in \mathcal{V}_{j+1,k'}$, we obtain from (5.16), (5.14), (5.15)

$$\begin{aligned} Q_{\mathcal{M}_{j+1}}(\mathbf{x}) = \Psi_{j+1,k'}\Psi^*_{j+1,k'}(\mathbf{x}_{j+1} - \mathbf{c}_{j+1,k'}) + \mathbf{w}_{j+1,k'} \\ - \Phi_{j,k}\Phi^*_{j,k}(\mathbf{x} - \mathbf{x}_{j+1}). \end{aligned} \tag{5.17}$$

Note that the last term $\mathbf{x} - \mathbf{x}_{j+1}$ can be closely approximated by $\mathbf{x}_J - \mathbf{x}_{j+1} = \sum_{l=j+1}^{J-1} Q_{\mathcal{M}_{l+1}}(\mathbf{x})$ as the finest scale $J \to +\infty$, under general conditions. This equation splits the difference $\mathbf{x}_{j+1} - \mathbf{x}_j$ into a component in $W_{j+1,k'}$, a translation term that only depends on the cube (j,k) (and not on individual points), and a projection onto $V_{j,k}$ of a sum of differences $\mathbf{x}_{l+1} - \mathbf{x}_l$ at finer scales.

Second, with (5.13), we may obtain a simpler representation of the difference

$$\begin{aligned} Q_{\mathcal{M}_{j+1}}(\mathbf{x}) &= \mathbf{x}_{j+1} - (\mathbf{P}_{j,k}(\mathbf{x}_{j+1} - \mathbf{c}_{j,k}) + \mathbf{c}_{j,k}) \\ &= (\mathbf{I} - \mathbf{P}_{j,k})(\mathbf{x}_{j+1} - \mathbf{c}_{j+1,k'} + \mathbf{c}_{j+1,k'} - \mathbf{c}_{j,k}) \\ &= \Psi_{j+1,k'}\Psi^*_{j+1,k'}(\mathbf{x}_{j+1} - \mathbf{c}_{j+1,k'}) + \mathbf{w}_{j+1,k'}. \end{aligned} \tag{5.18}$$

The term $\mathbf{x} - \mathbf{x}_{j+1}$ no longer appears in this equation and the difference depends only on a component in $W_{j+1,k'}$ and a translation term.

Comparing (5.17) and (5.18) we see that the main advantage of the construction in (5.17) is that the approximations $\mathbf{x}_j$ have clear-cut interpretations as the best least-squares approximations. However, it is at the expense of the size of the dictionary which must contain the scaling functions $\Phi_{j,k}$. The construction in (5.18), leading to a smaller dictionary, is particularly useful when one does not care about the intermediate approximations, for example, in data compression tasks.

It is also worth mentioning that the definition of wavelet subspaces and translations (see (5.14), (5.15)) is independent of that of the x_j. We present their construction in Alg. 2. Moreover, regardless of the definition of the approximations, we have the following two-scale relationship (by definition of $Q_{\mathcal{M}_{j+1}}$)

$$P_{\mathcal{M}_{j+1}}(\mathbf{x}) = P_{\mathcal{M}_j}(\mathbf{x}) + Q_{\mathcal{M}_{j+1}}(\mathbf{x}), \tag{5.19}$$

Algorithm 2 Pseudocode for the construction of geometric wavelets

```
Input: $\mathbb{X}$: a set of $n$ samples from $\mathcal{M} \subset \mathbb{R}^D$;
    $\tau_0$: some method for choosing local dimensions;
    $\epsilon$: a precision parameter
Output: A tree $\mathcal{T}$ of dyadic cubes $\{\mathbb{C}_{j,k}\}$, with local means $\{\mathbf{c}_{j,k}\}$ and SVD bases
    $\{\Phi_{j,k}\}$, as well as a family of geometric wavelets $\{\Psi_{j,k}\}, \{\mathbf{w}_{j,k}\}$
Steps:
1: Construct a tree $\mathcal{T}$ of dyadic cubes $\{\mathbb{C}_{j,k}\}$ with centers $\{\mathbf{c}_{j,k}\}$.
2: $J \leftarrow$ finest scale with the $\epsilon$-approximation property.
3: Let $\mathrm{cov}_{J,k} = |C_{J,k}|^{-1} \sum_{\mathbf{x} \in C_{J,k}} (\mathbf{x} - \mathbf{c}_{J,k})(\mathbf{x} - \mathbf{c}_{J,k})^*$, for all $k \in \Gamma_J$, and compute
   $\mathrm{SVD}(\mathrm{cov}_{J,k}) \approx \Phi_{J,k} \Sigma_{J,k} \Phi^*_{J,k}$ (where the rank of $\Phi_{J,k}$ is determined by $\tau_0$).
4: FOR $j = J - 1$ down to $0$
         FOR $k \in \Gamma_j$
             • Compute $\mathrm{cov}_{j,k}$ and $\Phi_{j,k}$ as above
             • For each $k' \in \mathrm{children}(j,k)$, construct the wavelet basis
               $\Psi_{j+1,k'}$ and translation $\mathbf{w}_{j+1,k'}$ using (5.14) and (5.15)
         ENDFOR
   ENDFOR
5: Return $\Psi_{0,k} := \Phi_{0,k}$ and $\mathbf{w}_{0,k} := \mathbf{c}_{0,k}$ for $k \in \Gamma_0$.
```

and it may be iterated across scales:

$$\mathbf{x} = P_{\mathcal{M}_j}(\mathbf{x}) + \sum_{l=j}^{J-1} Q_{\mathcal{M}_{l+1}}(\mathbf{x}) + (\mathbf{x} - P_{\mathcal{M}_J}(\mathbf{x})). \tag{5.20}$$

The above equations allow to efficiently decompose each step along low dimensional subspaces, leading to efficient encoding of the data. We have therefore constructed a multiscale family of projection operators $P_{\mathcal{M}_j}$ (one for each node $\mathbb{C}_{j,k}$) onto approximate local tangent planes and detail projection operators $Q_{\mathcal{M}_{j+1}}$ (one for each edge) encoding the differences, collectively referred to as a GMRA structure. The cost of encoding the GMRA structure is at most $\mathrm{O}(dD\epsilon^{-\frac{d}{2}})$ (when also encoding the scaling functions $\{\Phi_{j,k}\}$), and the time complexity of the algorithm is $\mathrm{O}(Dn\log(n))$ [**ACM12**].

Finally, we mention that various other variations, optimizations, and generalizations of the construction, such as orthogonalization, splitting, pruning, out-of-sample extension, etc., can be found in [**ACM12**]. Due to space considerations, we omit their details here.

5.2.4. *Associated Geometric Wavelet Transforms (GWT).* Given a GMRA structure, we may compute a Discrete Forward GWT for a point $\mathbf{x} \in \mathcal{M}$ that maps it to a sequence of wavelet coefficient vectors:

$$\mathbf{q}_{\mathbf{x}} = (\mathbf{q}_{J,\mathbf{x}}, \mathbf{q}_{J-1,\mathbf{x}}, \ldots, \mathbf{q}_{1,\mathbf{x}}, \mathbf{q}_{0,\mathbf{x}}) \in \mathcal{R}^{d+\sum_{j=1}^{J} d^w_{j,\mathbf{x}}} \tag{5.21}$$

where $\mathbf{q}_{j,\mathbf{x}} := \Psi^*_{j,\mathbf{x}}(\mathbf{x}_j - \mathbf{c}_{j,\mathbf{x}})$, and $d^w_{j,\mathbf{x}} := \mathrm{rank}(\Psi_{j,\mathbf{x}})$. Note that, in the case of a d-dimensional manifold and for a fixed precision $\epsilon > 0$, $\mathbf{q}_{\mathbf{x}}$ has a maximum possible length $(1 + \frac{1}{2}\log_2 \frac{1}{\epsilon})d$, which is independent of D and nearly optimal in d [**CM10**]. On the other hand, we may easily reconstruct the point $\mathbf{x}$ using the

```
$\{\mathbf{q}_{j,\mathbf{x}}\}$ =FGWT(GMRA, $\mathbf{x}$)

// Input: GMRA structure, $\mathbf{x} \in \mathcal{M}$
// Output: Wavelet coefficients $\{q_{j,\mathbf{x}}\}$
for $j = J$ down to $0$
        $\mathbf{x}_j = \Phi_{j,\mathbf{x}}\Phi^*_{j,\mathbf{x}}(x - \mathbf{c}_{j,\mathbf{x}}) + \mathbf{c}_{j,\mathbf{x}}$
        $\mathbf{q}_{j,\mathbf{x}} = \Psi^*_{j,\mathbf{x}}(\mathbf{x}_j - \mathbf{c}_{j,\mathbf{x}})$
end
```

```
$\{\mathbf{q}_{j,\mathbf{x}}\}$ =FGWT(GMRA, $\mathbf{x}$)

// Input: GMRA structure, $\mathbf{x} \in \mathcal{M}$
// Output: Wavelet coefficients $\{\mathbf{q}_{j,\mathbf{x}}\}$
for $j = J$ down to $0$
        $\mathbf{q}_{j,\mathbf{x}} = \Psi^*_{j,\mathbf{x}}(\mathbf{x} - \mathbf{c}_{j,\mathbf{x}})$
        $\mathbf{x} = \mathbf{x} - (\Psi_{j,\mathbf{x}}\mathbf{q}_{j,\mathbf{x}} + \mathbf{w}_{j,\mathbf{x}})$
end
```

```
$\mathbf{x}$ =IGWT(GMRA,$\{\mathbf{q}_{j,\mathbf{x}}\}$)

// Input: GMRA structure, wavelet coefficients $\{\mathbf{q}_{j,\mathbf{x}}\}$
// Output: Reconstruction $\mathbf{x}$

$Q_{\mathcal{M}_J}(\mathbf{x}) = \Psi_{J,\mathbf{x}}q_{J,\mathbf{x}} + \mathbf{w}_{J,\mathbf{x}}$
for $j = J - 1$ down to $1$
        $Q_{\mathcal{M}_j}(\mathbf{x}) = \Psi_{j,\mathbf{x}}\mathbf{q}_{j,\mathbf{x}} + \mathbf{w}_{j,\mathbf{x}} - \mathbf{P}_{j-1,\mathbf{x}} \sum_{\ell>j} Q_{\mathcal{M}_\ell}(\mathbf{x})$
end
$\mathbf{x} = \Psi_{0,\mathbf{x}}\mathbf{q}_{0,\mathbf{x}} + \mathbf{w}_{0,\mathbf{x}} + \sum_{j>0} Q_{\mathcal{M}_j}(\mathbf{x})$
```

```
$\mathbf{x}$ =IGWT(GMRA,$\{\mathbf{q}_{j,\mathbf{x}}\}$)

// Input: GMRA structure, wavelet coefficients $\{\mathbf{q}_{j,\mathbf{x}}\}$
// Output: Reconstruction $\mathbf{x}$

for $j = J$ down to $0$
        $Q_{\mathcal{M}_j}(\mathbf{x}) = \Psi_{j,\mathbf{x}}\mathbf{q}_{j,\mathbf{x}} + \mathbf{w}_{j,\mathbf{x}}$
end
$\mathbf{x} = \sum_{0 \le j \le J} Q_{\mathcal{M}_j}(\mathbf{x})$
```

Figure 7 Pseudocodes for the Forward (top row) and Inverse (bottom row) GWTs corresponding to different wavelet constructions (5.17) (left column) and (5.18) (right column)

GMRA structure and the wavelet coefficients, by a Discrete Inverse GWT. See Fig. 7 for the pseudocodes of both transforms.

5.2.5. *A toy example.* We consider a 2-dimensional *SwissRoll* manifold in $\mathbb{R}^{50}$ and sample 2000 points from it without adding any noise. We apply the GMRA to this synthetic data set to illustrate how the GMRA works in general. The corresponding results are shown in Fig. 8.

5.2.6. *A few remarks about GMRA.* We make the following comments.

- The GMRA algorithm presents an appealing framework for constructing data-dependent dictionaries using a geometric multiresolution analysis. Unlike the K-SVD dictionaries, the GMRA outputs dictionaries that are structured and hierarchically organized. Moreover, the different subgroups of such a dictionary correspond to different scales and have clear interpretations as detail operators.
- It has many other advantages, for example the construction is based on many local SVD and thus is fast to execute. In addition, there are theoretical guarantees on the size of the dictionary and the sparsity of the representation, at least when the data follows a manifold model. It is also associated with fast transforms, making the sparse coding component extremely simple and fast, which is typically unavailable for other algorithms.
- The GMRA algorithm naturally extends the wavelet transform for 1-dimensional signals to efficient multiscale transforms for higher dimensional data. The nonlinear space $\mathcal{M}$ replaces the classical function spaces,

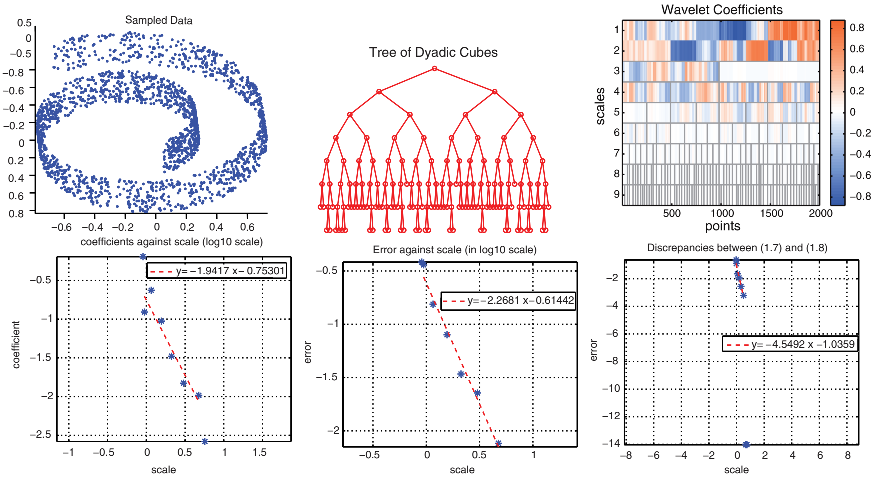


Figure 8 Illustration of the GMRA (with the projection defined in (5.13)) on a data set of 2000 points sampled from a 2-dimensional SwissRoll manifold in $\mathbb{R}^{50}$. Top left: sampled data; top middle: the tree of dyadic cubes obtained by the METIS algorithm [**KK99**]; top right: matrix of wavelet coefficients. The x-axis indexes the points (arranged according to the tree), and the y axis indexes the scales from coarse (top) to fine (bottom). Note that each block corresponds to a different dyadic cube. Bottom left: (average) magnitude of wavelet coefficients versus scale; bottom middle: approximation error of the projection (5.13) to the data as a function of scale; bottom right: deviation of the projection (5.13) from the best possible one (5.12) at each scale (also in $\log_{10}$ scale). The last plot shows that the projection (5.13) deviates from (5.12) at a rate of 4 and in particular, the two almost coincide with each other at fine scales.

the piecewise affine approximation at each scale substitutes the linear projection on scaling function spaces, and the difference operators play the role of the linear wavelet projections. But it is also quite different in many crucial aspects. It is nonlinear, as its adapts to the nonlinear manifolds modeling the data space, but every scale-to-scale step is linear. Translations or dilations do not play any role here, while they are often considered crucial in classical wavelet constructions.

5.3. Online dictionary learning (ODL). The ODL algorithm, developed by Mairal et al. [**MBPS09**, **MBPS10**], is an online algorithm that is designed to handle extremely large data sets. It starts by assuming a generative model $\mathbf{x}_t \sim p(\mathbf{x})$, where $p(\mathbf{x})$ represents a probability density function governing the data distribution.[2] At each time $t = 1, 2, \ldots$, it draws a new sample $\mathbf{x}_t$ from the distribution and uses it to refine the dictionary $\mathbf{D}_{t-1}$ obtained at time $t-1$. This procedure is repeated until a stopping criterion has been met (for example, t has reached an upper bound T, or the dictionary $\mathbf{D}_t$ no longer changes noticeably).

[2] For real data sets one does not know the underlying probability distribution; in this case, the uniform discrete measure can be used. This is equivalent to first randomly permuting the data points and then sequentially processing them, one at a time.

5.3.1. *Detailed description of the ODL algorithm.* To present the specific ideas, we consider iteration t when a new sample $\mathbf{x}_t$ arrives. By this time, the first $t-1$ samples $\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}$ have already been drawn from the distribution and used to train a dictionary $\mathbf{D}_{t-1}$. Of course, if $t = 1$, then $\mathbf{D}_0$ represents an initial guess of the dictionary provided by the user to start with. We now would like to use $\mathbf{x}_t$ to update the dictionary $\mathbf{D}_{t-1}$ to $\mathbf{D}_t$. This is achieved in two steps: First, we find the sparse coefficient of $\mathbf{x}_t$ relative to $\mathbf{D}_{t-1}$ by solving the sparse coding problem

$$\gamma_t = \operatorname*{argmin}_{\gamma} \ \frac{1}{2}\|\mathbf{x}_t - \mathbf{D}_{t-1}\gamma\|^2 + \lambda\|\gamma\|_1$$

where λ is the user-specified tuning parameter, fixed throughout all iterations. We have repeatedly encountered this problem and recall that it can be easily solved by any of those pursuit algorithms mentioned in the CS framework. Next, we fix the coefficient γ_t obtained above, together with the previous ones $\gamma_1, \ldots, \gamma_{t-1}$, and minimize the same objective function, this time with respect to the dictionary $\mathbf{D}$ (constrained to have at most unit-norm columns), hoping to find a new dictionary $\mathbf{D}_t$ that is better suited to all coefficients $\gamma_i, 1 \le i \le t$:

$$\mathbf{D}_t = \operatorname*{argmin}_{\mathbf{D}} \frac{1}{2t}\sum_{i=1}^{t} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|^2 + \lambda\|\gamma_i\|_1.$$

In this step, since the γ_i are fixed, we may remove the second term from the objective function and consider instead

$$\mathbf{D}_t = \operatorname*{argmin}_{\mathbf{D}} \sum_{i=1}^{t} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|^2.$$

To see how the square-loss objective is minimized, we rewrite it using matrix notation:

$$\begin{aligned}
\sum_{i=1}^{t} \|\mathbf{x}_i - \mathbf{D}\gamma_i\|^2 &= \sum_{i=1}^{t} \operatorname{trace}\left((\mathbf{x}_i - \mathbf{D}\gamma_i)(\mathbf{x}_i - \mathbf{D}\gamma_i)^T\right) \\
&= \sum_{i=1}^{t} \operatorname{trace}\left(\mathbf{x}_i\mathbf{x}_i^T - \mathbf{x}_i\gamma_i^T\mathbf{D}^T - \mathbf{D}\gamma_i\mathbf{x}_i^T + \mathbf{D}\gamma_i\gamma_i^T\mathbf{D}^T\right) \\
&= \sum_{i=1}^{t} \left(\operatorname{trace}\left(\mathbf{x}_i\mathbf{x}_i^T\right) - 2\operatorname{trace}\left(\mathbf{x}_i\gamma_i^T\mathbf{D}^T\right) + \operatorname{trace}\left(\mathbf{D}^T\mathbf{D}\gamma_i\gamma_i^T\right)\right) \\
&= \operatorname{trace}\left(\sum_{i=1}^{t} \mathbf{x}_i\mathbf{x}_i^T\right) - 2\operatorname{trace}\left(\mathbf{D}^T\sum_{i=1}^{t} \mathbf{x}_i\gamma_i^T\right) \\
&\quad + \operatorname{trace}\left(\mathbf{D}^T\mathbf{D}\sum_{i=1}^{t} \gamma_i\gamma_i^T\right)
\end{aligned}$$

Letting

$$\mathbf{A}_t = \sum_{i=1}^{t} \gamma_i\gamma_i^T \in \mathbb{R}^{m\times m}, \quad \mathbf{B}_t = \sum_{i=1}^{t} \mathbf{x}_i\gamma_i^T \in \mathbb{R}^{L\times m}$$

and discarding the first term that does not depend on $\mathbf{D}$, we arrive at the final form

$$\mathbf{D}_t = \operatorname*{argmin}_{\mathbf{D}} \ \operatorname{trace}(\mathbf{D}^T\mathbf{D}\mathbf{A}_t) - 2\operatorname{trace}(\mathbf{D}^T\mathbf{B}_t)$$

This problem can be solved by block coordinate descent, using $\mathbf{D}_{t-1}$ as an initial dictionary. Specifically, we hold all except the kth column $\mathbf{d}_k$ of $\mathbf{D} = \mathbf{D}_{t-1}$ fixed and consider the resulting optimization problem, which is only over $\mathbf{d}_k$. Using straightforward multivariable calculus, we obtain the gradient of the objective function as follows:

$$2(\mathbf{D}\,\mathbf{A}_t(:,k) - \mathbf{B}_t(:,k)).$$

Setting it equal to zero and solving the corresponding equation yields a unique critical point

$$\frac{1}{(\mathbf{A}_t)_{kk}}(\mathbf{B}_t(:,k) - \sum_{j\neq k} \mathbf{d}_j\,(\mathbf{A}_t)_{jk}) = \mathbf{d}_k - \frac{1}{(\mathbf{A}_t)_{kk}}\left(\mathbf{D}\,\mathbf{A}_t(:,k) - \mathbf{B}_t(:,k)\right).$$

The Hessian matrix of the objective function, $(\mathbf{A}_t)_{kk}\,\mathbf{I}$, is strictly positive definite (because $\mathbf{A}_t$ also is), implying that the critical point is a global minimizer. Thus, we update $\mathbf{d}_k$ by setting

$$\mathbf{d}_k \leftarrow \mathbf{d}_k - \frac{1}{(\mathbf{A}_t)_{kk}}\left(\mathbf{D}\,\mathbf{A}_t(:,k) - \mathbf{B}_t(:,k)\right),$$

in order to reduce the objective function as much as possible (if $\|\mathbf{d}_k\|_2 > 1$, we then need to normalize it to unit norm). We sequentially update all the columns of $\mathbf{D}$ by varying k, always including the updated columns in $\mathbf{D}$ for updating the remaining columns, in order to accelerate convergence. We repeat this procedure until the objective function no longer decreases and denote the corresponding dictionary by $\mathbf{D}_t$.

We now summarize the steps of ODL in Algorithm 3. We refer the reader to [**MBPS09**, **MBPS10**] for convergence analysis and performance evaluations.

5.3.2. *A few remarks about the ODL algorithm.*

- ODL and K-SVD are similar in several ways: (1) Both involve solving the sparse coding problem (4.7) many times (once per signal, per iteration). (2) The columns of $\mathbf{D}$ are updated sequentially in both algorithms, which speeds up convergence. (3) Both require an initial guess of the dictionary, which affects convergence.
- Both ODL and GMRA assume a generative model for the data, so they both aim to build dictionaries for some probability distribution, not just for a particular data set. In contrast, K-SVD only aims to achieve the minimal empirical cost on the training data.
- The ODL is also quite different from K-SVD and GMRA, in the sense that the latter two are batch methods which must access the whole training set in order to learn the dictionary. In contrast, the ODL uses one sample at a time, without needing to store or access the entire data set. This advantage makes ODL particularly suitable for working with very large data sets, which might have millions of samples, and with dynamic training data changing over time, such as video sequences. Additionally, since it does not need to store the entire data set, it has a low memory consumption and lower computational cost than classical batch algorithms.

5.4. Future directions in dictionary learning. Despite all the impressive achievements by sparse and redundant modeling, many questions remain to be answered and there are a large number of future research directions. We list only

Algorithm 3 Pseudocode for the Online Dictionary Learning (ODL) Algorithm

```
Input: Density $p(\mathbf{x})$, tuning parameter $\lambda$, initial dictionary $\mathbf{D}_0$, and number of
    iterations $T$
Output: Final dictionary $\mathbf{D}_T$
Steps:
Initialization: Set $\mathbf{A}_0 \leftarrow 0$ and $\mathbf{B}_0 \leftarrow 0$
FOR $t = 1 : T$
        • Sampling: Draw a new sample $\mathbf{x}_t$ from $p(\mathbf{x})$
        • Sparse coding: Find the sparse coefficient of $\mathbf{x}_t$ relative to $\mathbf{D}_{t-1}$ by
          solving
              $\gamma_t = \underset{\gamma}{\operatorname{argmin}} \ \|\mathbf{x}_t - \mathbf{D}_{t-1}\gamma\|_2^2 + \lambda\|\gamma\|_1.$
          using any pursuit algorithm (e.g., OMP).
        • Recording: Update $\mathbf{A}_t$ and $\mathbf{B}_t$ to include $\mathbf{x}_t$ and $\gamma_t$:
              $\mathbf{A}_t \leftarrow \mathbf{A}_{t-1} + \gamma_t\gamma_t^T, \quad \mathbf{B}_t \leftarrow \mathbf{B}_{t-1} + \mathbf{x}_t\gamma_t^T$
        • Dictionary update: Update the columns $\mathbf{d}_k$ of $\mathbf{D}_{t-1}$ sequentially,
          using
              $\mathbf{d}_k \leftarrow \mathbf{d}_k - \frac{1}{(\mathbf{A}_t)_{kk}} \left(\mathbf{D}\mathbf{A}_t(:,k) - \mathbf{B}_t(:,k)\right).$
          If $\|\mathbf{d}_k\|_2 > 1$, then normalize it to have unit norm. Repeat this procedure until convergence.
    ENDFOR
Return $\mathbf{D}_t$
```

a few below, while referring the reader to [**Ela12**] for a detailed discussion on this topic.

- **Theoretical justification of dictionary learning algorithms**. So far most of the dictionary learning algorithms are essentially empirical methods (e.g., K-SVD [**AEB05**, **AEB06**]). Little is known about their stability, especially in the presence of noise. Furthermore, it is unclear which conditions would guarantee the algorithm to succeed. Finally, new measures of dictionary quality (not just worst-case conditions) need to be developed in order to study the goodness of the learned dictionary.
- **Introducing structure to dictionaries**. Currently, most learned dictionaries (such as the K-SVD dictionary) are completely unstructured. Finding the sparse representation of a signal relative to such a dictionary is a nontrivial task. This causes a great computational burden when dealing with large data sets. It is thus desirable to impose structures to the dictionary atoms to simplify the sparse coding task. It is noteworthy to mention that the GMRA dictionary represents an advancement in this direction, as it organizes its atoms into a tree which makes the coding part extremely simple and fast. It will be interesting to explore other ways to construct structured dictionaries.
- **Developing next generation models**. Sparse and redundant modeling represents the current state of the art, having evolved from transforms and

union-of-subspaces models. Inevitably such a model will be replaced by newer, more powerful models, just like its predecessors. It will be exciting to see any new research devoted in this direction.

Acknowledgment

The images in Fig. 5 are generated by the K-SVD software available at `http://www.cs.technion.ac.il/~elad/software/`.

References

[ABC$^+$15] P. Awasthi, and A. S. Bandeira, and M. Charikar, and R. Krishnaswamy, and S. Villar, and R. Ward. Relax, No Need to Round: Integrality of Clustering Formulations. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, pages 191–20, January 2015.

[ACM12] William K. Allard, Guangliang Chen, and Mauro Maggioni, *Multi-scale geometric methods for data sets II: Geometric multi-resolution analysis*, Appl. Comput. Harmon. Anal. **32** (2012), no. 3, 435–462, DOI 10.1016/j.acha.2011.08.001. MR2892743

[AEB05] M. Aharon, M. Elad, and A. M. Bruckstein. The k-svd algorithm. In *SPARSE*, 2005.

[AEB06] M. Aharon, M. Elad, and A. M. Bruckstein. K-svd: An algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE T. Signal Proces.*, 54(11):4311–4322, November 2006.

[AR77] J. B. Allen and L. R. Rabiner. A unified approach to short-time fourier analysis and synthesis. In *Proceedings of the IEEE*, volume 65, pages 1558–1564, 1977.

[Bas80] M. J. Bastiaans. Gabor's expansion of a signal into gaussian elementary signals. In *Proceedings of the IEEE*, volume 68, pages 538–539, 1980.

[BCM05] A. Buades, B. Coll, and J. M. Morel, *A review of image denoising algorithms, with a new one*, Multiscale Model. Simul. **4** (2005), no. 2, 490–530, DOI 10.1137/040616024. MR2162865

[BD09] Thomas Blumensath and Mike E. Davies, *Iterative hard thresholding for compressed sensing*, Appl. Comput. Harmon. Anal. **27** (2009), no. 3, 265–274, DOI 10.1016/j.acha.2009.04.002. MR2559726 (2010i:94048)

[BDE09] Alfred M. Bruckstein, David L. Donoho, and Michael Elad, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM Rev. **51** (2009), no. 1, 34–81, DOI 10.1137/060657704. MR2481111 (2010d:94012)

[BE08] O. Bryt and M. Elad. Compression of facial images using the k-svd algorithm. *Journal of Visual Communication and Image Representation*, 19:270–283, 2008.

[Blu11] Thomas Blumensath, *Sampling and reconstructing signals from a union of linear subspaces*, IEEE Trans. Inform. Theory **57** (2011), no. 7, 4660–4671, DOI 10.1109/TIT.2011.2146550. MR2840482 (2012i:94102)

[BS07] R. Baraniuk and P. Steeghs. Compressive radar imaging. In *Proc. IEEE Radar Conf*, pages 128–133. Ieee, 2007.

[Can06] Emmanuel J. Candès, *Compressive sampling*, International Congress of Mathematicians. Vol. III, Eur. Math. Soc., Zürich, 2006, pp. 1433–1452. MR2275736 (2008e:62033)

[CD04] Emmanuel J. Candès and David L. Donoho, *New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities*, Comm. Pure Appl. Math. **57** (2004), no. 2, 219–266, DOI 10.1002/cpa.10116. MR2012649 (2004k:42052)

[CDD09] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore, *Compressed sensing and best k-term approximation*, J. Amer. Math. Soc. **22** (2009), no. 1, 211–231, DOI 10.1090/S0894-0347-08-00610-3. MR2449058 (2010d:94024)

[CDDY00] E. J. Candès, L. Demanet, D. L. Donoho, and L. Ying. Fast discrete curvelet transforms. *Multiscale Model. Simul.*, 5:861–899, 2000.

[CSPW11] Venkat Chandrasekaran, Sujay Sanghavi, Pablo A. Parrilo, and Alan S. Willsky, *Rank-sparsity incoherence for matrix decomposition*, SIAM J. Optim. **21** (2011), no. 2, 572–596, DOI 10.1137/090761793. MR2817479 (2012m:90128)

[CDS98] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput. **20** (1998), no. 1, 33–61, DOI 10.1137/S1064827596304010. MR1639094 (99h:94013)

[CENR10] Emmanuel J. Candès, Yonina C. Eldar, Deanna Needell, and Paige Randall, *Compressed sensing with coherent and redundant dictionaries*, Appl. Comput. Harmon. Anal. **31** (2011), no. 1, 59–73, DOI 10.1016/j.acha.2010.10.002. MR2795875 (2012d:94014)

[CK] *Finite frames*, Applied and Numerical Harmonic Analysis, Birkhäuser/Springer, New York, 2013. Theory and applications; Edited by Peter G. Casazza and Gitta Kutyniok. MR2964005

[CL09a] Guangliang Chen and Gilad Lerman, *Foundations of a multi-way spectral clustering framework for hybrid linear modeling*, Found. Comput. Math. **9** (2009), no. 5, 517–558, DOI 10.1007/s10208-009-9043-7. MR2534403 (2010k:62299)

[CL09b] G. Chen and G. Lerman. Spectral curvature clustering (SCC). *Int. J. Comput. Vision*, 81(3):317–330, 2009.

[CM10] G. Chen and M. Maggioni. Multiscale geometric wavelets for the analysis of point clouds. In *Proc. of the 44th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, March 2010.

[CM11a] G. Chen and M. Maggioni. Multiscale geometric and spectral analysis of plane arrangements. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[CM11b] G. Chen and M. Maggioni. Multiscale geometric dictionaries for point-cloud data. In *Proc. of the 9th International Conference on Sampling Theory and Applications (SampTA)*, Singapore, May 2011.

[CMW92] Ronald R. Coifman, Yves Meyer, and Victor Wickerhauser, *Wavelet analysis and signal processing*, Wavelets and their applications, Jones and Bartlett, Boston, MA, 1992, pp. 153–178. MR1187341

[CP09] Emmanuel J. Candès and Yaniv Plan, *Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements*, IEEE Trans. Inform. Theory **57** (2011), no. 4, 2342–2359, DOI 10.1109/TIT.2011.2111771. MR2809094

[CP10] Emmanuel J. Candès and Yaniv Plan, *A probabilistic and RIPless theory of compressed sensing*, IEEE Trans. Inform. Theory **57** (2011), no. 11, 7235–7254, DOI 10.1109/TIT.2011.2161794. MR2883653

[CR05] E. Candès and J. Romberg. Signal recovery from random projections. In *Proc. SPIE Conference on Computational Imaging III*, volume 5674, pages 76–86. SPIE, 2005.

[CRT06a] Emmanuel J. Candès, Justin Romberg, and Terence Tao, *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory **52** (2006), no. 2, 489–509, DOI 10.1109/TIT.2005.862083. MR2236170 (2007e:94020)

[CRT06b] Emmanuel J. Candès, Justin K. Romberg, and Terence Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math. **59** (2006), no. 8, 1207–1223, DOI 10.1002/cpa.20124. MR2230846 (2007f:94007)

[CSZ06] Tony F. Chan, Jianhong Shen, and Hao-Min Zhou, *Total variation wavelet inpainting*, J. Math. Imaging Vision **25** (2006), no. 1, 107–125, DOI 10.1007/s10851-006-5257-3. MR2254441 (2007g:94006)

[CT05] Emmanuel J. Candes and Terence Tao, *Decoding by linear programming*, IEEE Trans. Inform. Theory **51** (2005), no. 12, 4203–4215, DOI 10.1109/TIT.2005.858979. MR2243152 (2007b:94313)

[CT06] Emmanuel J. Candes and Terence Tao, *Near-optimal signal recovery from random projections: universal encoding strategies?*, IEEE Trans. Inform. Theory **52** (2006), no. 12, 5406–5425, DOI 10.1109/TIT.2006.885507. MR2300700 (2008c:94009)

[CW08] E. J. Candès and M. Wakin. An introduction to compressive sampling. *IEEE Signal Proc. Mag.*, 25(2):21–30, 2008.

[Dau92] Ingrid Daubechies, *Ten lectures on wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 61, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992. MR1162107 (93e:42045)

[DH01] David L. Donoho and Xiaoming Huo, *Uncertainty principles and ideal atomic decomposition*, IEEE Trans. Inform. Theory **47** (2001), no. 7, 2845–2862, DOI 10.1109/18.959265. MR1872845 (2002k:94012)

[DNW13] Mark A. Davenport, Deanna Needell, and Michael B. Wakin, *Signal space CoSaMP for sparse recovery with redundant dictionaries*, IEEE Trans. Inform. Theory **59** (2013), no. 10, 6820–6829, DOI 10.1109/TIT.2013.2273491. MR3106865

[Don06] David L. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory **52** (2006), no. 4, 1289–1306, DOI 10.1109/TIT.2006.871582. MR2241189 (2007e:94013)

[DSMB09] W. Dai, M. Sheikh, O. Milenkovic, and R. Baraniuk. Compressive sensing DNA microarrays. *EURASIP journal on bioinformatics and systems biology*, pages 1–12, 2009.

[DSP] Compressive sampling webpage. http://dsp.rice.edu/cs.

[Dut89] *Wavelets*, Inverse Problems and Theoretical Imaging, Springer-Verlag, Berlin, 1989. Time-frequency methods and phase space; Edited by J. M. Combes, A. Grossmann and Ph. Tchamitchian. MR1010895 (90g:42062)

[DW11] Mark A. Davenport and Michael B. Wakin, *Compressive sensing of analog signals using discrete prolate spheroidal sequences*, Appl. Comput. Harmon. Anal. **33** (2012), no. 3, 438–472, DOI 10.1016/j.acha.2012.02.005. MR2950138

[EA06] Michael Elad and Michal Aharon, *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Trans. Image Process. **15** (2006), no. 12, 3736–3745, DOI 10.1109/TIP.2006.881969. MR2498043

[EFM10] M. Elad, M. Figueiredo, and Y. Ma. On the role of sparse and redundant representations in image processing. *Proceedings of the IEEE - Special Issue on Applications of Sparse Representation and Compressive Sensing*, 98(6):972–982, 2010.

[EJCW09] Y. M. E. J. Candès, X. Li and J. Wright. Robust Principal Component Analysis? *Journal of ACM*, 58(1):1–37, 2009.

[Ela12] M. Elad. Sparse and redundant representation modeling–what next? *IEEE Signal Processing Letters*, 19(12):922–928, 2012.

[FS98] *Gabor analysis and algorithms*, Applied and Numerical Harmonic Analysis, Birkhäuser Boston, Inc., Boston, MA, 1998. Theory and applications; Edited by Hans G. Feichtinger and Thomas Strohmer. MR1601119 (98h:42001)

[F15] S. Foucart. Dictionary-sparse recovery via thresholding-based algorithms. Submitted.

[Gab46] D. Gabor. Theory of communication. *J. Inst. Electr. Eng.*, 93(26):429–457, 1946.

[GE12] Raja Giryes and Michael Elad, *RIP-based near-oracle performance guarantees for SP, CoSaMP, and IHT*, IEEE Trans. Signal Process. **60** (2012), no. 3, 1465–1468, DOI 10.1109/TSP.2011.2174985. MR2859009 (2012m:94114)

[GN13] Raja Giryes and Deanna Needell, *Greedy signal space methods for incoherence and beyond*, Appl. Comput. Harmon. Anal. **39** (2015), no. 1, 1–20, DOI 10.1016/j.acha.2014.07.004. MR3343799

[GNE$^+$12] R. Giryes, S. Nam, M. Elad, R. Gribonval, and M. E. Davies, *Greedy-like algorithms for the cosparse analysis model*, Linear Algebra Appl. **441** (2014), 22–60, DOI 10.1016/j.laa.2013.03.004. MR3134336

[HN07] J. Haupt and R. Nowak. A generalized restricted isometry property. *Univ. of Wisconsin-Madison, Tech. Rep. ECE-07-1*, 2007.

[HS09] Matthew A. Herman and Thomas Strohmer, *High-resolution radar via compressed sensing*, IEEE Trans. Signal Process. **57** (2009), no. 6, 2275–2284, DOI 10.1109/TSP.2009.2014277. MR2641823 (2011a:94028)

[HX13] H. Huang and N. Xiao. Image deblurring based on sparse model with dictionary learning. *Journal of Information & Computational Science*, 10(1):129–137, 2013.

[Jan81] A. Janssen. Gabor representation of generalized functions. *J. Math. Anal. and Applic.*, 83(2):377–394, 1981.

[Jol02] I. T. Jolliffe, *Principal component analysis*, 2nd ed., Springer Series in Statistics, Springer-Verlag, New York, 2002. MR2036084 (2004k:62010)

[Kee03] Stephen L. Keeling, *Total variation based convex filters for medical imaging*, Appl. Math. Comput. **139** (2003), no. 1, 101–119, DOI 10.1016/S0096-3003(02)00171-6. MR1949379 (2003k:92013)

[KK99] George Karypis and Vipin Kumar, *A fast and high quality multilevel scheme for partitioning irregular graphs*, SIAM J. Sci. Comput. **20** (1998), no. 1, 359–392 (electronic), DOI 10.1137/S1064827595287997. MR1639073 (99f:68158)

[KLW$^+$06] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. Baraniuk. Analog-to-information conversion via random demodulation. In *Proc. IEEE Dallas Circuits and Systems Workshop (DCAS)*, pages 71–74. IEEE, 2006.

[KTMJ08] B. Kai Tobias, U. Martin, and F. Jens. Suppression of MRI truncation artifacts using total variation constrained data extrapolation. *Int. J. Biomedical Imaging*, 2008.

[KNW15] Author = F. Krahmer, and D. Needell and R. Ward, Compressive Sensing with Redundant Dictionaries and Structured Measurements. Submitted.

[KW11] Felix Krahmer and Rachel Ward, *New and improved Johnson-Lindenstrauss embeddings via the restricted isometry property*, SIAM J. Math. Anal. **43** (2011), no. 3, 1269–1281, DOI 10.1137/100810447. MR2821584 (2012g:15052)

[LDP07] M. Lustig, D. Donoho, and J. Pauly. Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007.

[LDSP08] M. Lustig, D. Donoho, J. Santos, and J. Pauly. Compressed sensing MRI. *IEEE Sig. Proc. Mag.*, 25(2):72–82, 2008.

[LH07] T. Lin and F. Herrmann. Compressed wavefield extrapolation. *Geophysics*, 72(5):SM77, 2007.

[LLR95] Nathan Linial, Eran London, and Yuri Rabinovich, *The geometry of graphs and some of its algorithmic applications*, Combinatorica **15** (1995), no. 2, 215–245, DOI 10.1007/BF01200757. MR1337355 (96e:05158)

[LW11] Y. Liu and Q. Wan. Total variation minimization based compressive wideband spectrum sensing for cognitive radios. Preprint, 2011.

[Mac67] J. MacQueen, *Some methods for classification and analysis of multivariate observations*, Proc. Fifth Berkeley Sympos. Math. Statist. and Probability (Berkeley, Calif., 1965/66), Univ. California Press, Berkeley, Calif., 1967, pp. Vol. I: Statistics, pp. 281–297. MR0214227 (35 #5078)

[Mal89] S. Mallat. A theory of multiresolution signal decomposition: the wavelet representation. *IEEE T. Pattern Anal.*, 11:674–693, 1989.

[Mal99] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, London, 2nd edition, 1999.

[MBPS09] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, volume 11, pages 689–696, 2009.

[MBPS10] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro, *Online learning for matrix factorization and sparse coding*, J. Mach. Learn. Res. **11** (2010), 19–60. MR2591620 (2011i:62121)

[ME11] Moshe Mishali and Yonina C. Eldar, *Xampling: compressed sensing of analog signals*, Compressed sensing, Cambridge Univ. Press, Cambridge, 2012, pp. 88–147, DOI 10.1017/CBO9780511794308.004. MR2963168

[MPTJ08] Shahar Mendelson, Alain Pajor, and Nicole Tomczak-Jaegermann, *Uniform uncertainty principle for Bernoulli and subgaussian ensembles*, Constr. Approx. **28** (2008), no. 3, 277–289, DOI 10.1007/s00365-007-9005-8. MR2453368 (2009k:46020)

[MSE] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration.

[MSW$^+$10] M. Mohtashemi, H. Smith, D. Walburger, F. Sutton, and J. Diggans. Sparse sensing DNA microarray-based biosensor: Is it feasible? In *IEEE Sensors Applications Symposium (SAS)*, pages 127–130. IEEE, 2010.

[Mut05] S. Muthukrishnan. *Data Streams: Algorithms and Applications.* Now Publishers, Hanover, MA, 2005.

[MYZC08] S. Ma, W. Yin, Y. Zhang, and A. Chakraborty. An efficient algorithm for compressed mr imaging using total variation and wavelets. In *IEEE Conf. Comp. Vision Pattern Recog.*, 2008.

[MZ93] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE T. Signal Proces.*, 41(12):3397–3415, 1993.

[NDEG11] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, *The cosparse analysis model and algorithms*, Appl. Comput. Harmon. Anal. **34** (2013), no. 1, 30–56, DOI 10.1016/j.acha.2012.03.006. MR2981332

[NJW01] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856, 2001.

[NRWY10] S. Negahban, P. Ravikumar, M. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. *Adv. Neur. In.*, 1348–1356, 2009.

[NT08a] D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. ACM Technical Report 2008-01, California Institute of Technology, Pasadena, July 2008.

[NT08b] D. Needell and J. A. Tropp, *CoSaMP: iterative signal recovery from incomplete and inaccurate samples*, Appl. Comput. Harmon. Anal. **26** (2009), no. 3, 301–321, DOI 10.1016/j.acha.2008.07.002. MR2502366 (2010c:94018)

[NTLC08] B. Nett, J. Tang, S. Leng, and G. Chen. Tomosynthesis via total variation minimization reconstruction and prior image constrained compressed sensing (PICCS) on a C-arm system. In *Proc. Soc. Photo-Optical Instr. Eng.*, volume 6913. NIH Public Access, 2008.

[NV07a] D. Needell and R. Vershynin. Signal recovery from incomplete and inaccurate measurements via Regularized Orthogonal Matching Pursuit. *IEEE J. Sel. Top. Signa.*, 4:310–316, 2010.

[NV07b] D. Needell and R. Vershynin. Uniform uncertainty principle and signal recovery via Regularized Orthogonal Matching Pursuit. *Found. Comput. Math.*, 9(3):317–334, 2007.

[NW12] Deanna Needell and Rachel Ward, *Near-optimal compressed sensing guarantees for total variation minimization*, IEEE Trans. Image Process. **22** (2013), no. 10, 3941–3949, DOI 10.1109/TIP.2013.2264681. MR3105153

[NW13] Deanna Needell and Rachel Ward, *Stable image reconstruction using total variation minimization*, SIAM J. Imaging Sci. **6** (2013), no. 2, 1035–1058, DOI 10.1137/120868281. MR3062581

[OF96] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[OH10] S. Oymak and B. Hassibi. New Null Space Results and Recovery Thresholds for Matrix Rank Minimization. Preprint, 2010.

[OSV03] Stanley Osher, Andrés Solé, and Luminita Vese, *Image decomposition and restoration using total variation minimization and the H^{-1} norm*, Multiscale Model. Simul. **1** (2003), no. 3, 349–370 (electronic), DOI 10.1137/S1540345902416247. MR2030155 (2004k:49004)

[PETM09] M. Protter, M. Elad, H. Takeda, and P. Milanfar. Generalizing the non-local-means to super-resolution reconstruction. *IEEE T. Image Process.*, 16(1):36–51, January 2009.

[PPM] S. Pudlewski, A. Prasanna, and T. Melodia. Compressed-sensing-enabled video streaming for wireless multimedia sensor networks. *IEEE T. Mobile Comput.*, (99):1–1.

[PRT11] Götz E. Pfander, Holger Rauhut, and Joel A. Tropp, *The restricted isometry property for time-frequency structured random matrices*, Probab. Theory Related Fields **156** (2013), no. 3-4, 707–737, DOI 10.1007/s00440-012-0441-4. MR3078284

[Rau08] Holger Rauhut, *On the impossibility of uniform sparse reconstruction using greedy methods*, Sampl. Theory Signal Image Process. **7** (2008), no. 2, 197–215. MR2451767 (2010m:94051)

[RBE10] R. Rubinstein, A. M. Bruckstein, and M. Elad. Dictionaries for sparse representation modeling. In *Proceedings of the IEEE*, volume 98, pages 1045–1057, 2010.

[RFP07] Benjamin Recht, Maryam Fazel, and Pablo A. Parrilo, *Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization*, SIAM Rev. **52** (2010), no. 3, 471–501, DOI 10.1137/070697835. MR2680543 (2012a:90137)

[ROF92] Leonid I. Rudin, Stanley Osher, and Emad Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D **60** (1992), no. 1-4, 259–268. Experimental mathematics: computational issues in nonlinear science (Los Alamos, NM, 1991). MR3363401

[Rom08] J. Romberg. Imaging via compressive sampling. *Signal Processing Magazine, IEEE*, 25(2):14–20, 2008.

[RS97] Amos Ron and Zuowei Shen, *Affine systems in $L_2(\mathbf{R}^d)$: the analysis of the analysis operator*, J. Funct. Anal. **148** (1997), no. 2, 408–447, DOI 10.1006/jfan.1996.3079. MR1469348 (99g:42043)

[RS05] J. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proc. 22nd int. conf. on Machine learning*, pages 713–719. ACM, 2005.

[RV06] M. Rudelson and R. Vershynin. Sparse reconstruction by convex relaxation: Fourier and Gaussian measurements. In *Proc. 40th Annual Conf. on Info. Sciences and Systems*, Princeton, Mar. 2006.

[RV08] Mark Rudelson and Roman Vershynin, *On sparse reconstruction from Fourier and Gaussian measurements*, Comm. Pure Appl. Math. **61** (2008), no. 8, 1025–1045, DOI 10.1002/cpa.20227. MR2417886 (2009e:94034)

[Sch86] R. Schmidt. Multiple emitter location and signal parameter estimation. *Antennas and Propagation, IEEE Transactions on*, 34(3):276–280, 1986.

[SED04] J.-L. Starck, M. Elad, and D. Donoho. Redundant multiscale transforms and their application for morphological component analysis. *Adv. Imag. Elect. Phys.*, 132, 2004.

[SFM07] Jean-Luc Starck, Jalal Fadili, and Fionn Murtagh, *The undecimated wavelet decomposition and its reconstruction*, IEEE Trans. Image Process. **16** (2007), no. 2, 297–309, DOI 10.1109/TIP.2006.887733. MR2462723 (2009j:94052)

[Sin08] Amit Singer, *A remark on global positioning from local distances*, Proc. Natl. Acad. Sci. USA **105** (2008), no. 28, 9507–9511, DOI 10.1073/pnas.0709842104. MR2430205 (2009h:62064)

[SM00] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[Sre04] Nathan Srebro, *Learning with matrix factorizations*, ProQuest LLC, Ann Arbor, MI, 2004. Thesis (Ph.D.)–Massachusetts Institute of Technology. MR2717223

[SY07] Anthony Man-Cho So and Yinyu Ye, *Theory of semidefinite programming for sensor network localization*, Math. Program. **109** (2007), no. 2-3, Ser. B, 367–384, DOI 10.1007/s10107-006-0040-1. MR2295148 (2007j:90097)

[TG07] Joel A. Tropp and Anna C. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, IEEE Trans. Inform. Theory **53** (2007), no. 12, 4655–4666, DOI 10.1109/TIT.2007.909108. MR2446929 (2009h:94042)

[Tro06] Joel A. Tropp, *Just relax: convex programming methods for identifying sparse signals in noise*, IEEE Trans. Inform. Theory **52** (2006), no. 3, 1030–1051, DOI 10.1109/TIT.2005.864420. MR2238069 (2007a:94064)

[TSHM09] G. Tang, R. Shahidi, F. Herrmann, and J. Ma. Higher dimensional blue-noise sampling schemes for curvelet-based seismic data recovery. *Soc. Explor. Geophys.*, 2009.

[TWD$^+$06] J. Tropp, M. Wakin, M. Duarte, D. Baron, and R. Baraniuk. Random filters for compressive sampling and reconstruction. In *Proc. 2006 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, volume 3. IEEE, 2006.

[Vid11] R. Vidal. Subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, 2011.

[WLD$^+$06] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R. Baraniuk. An architecture for compressive imaging. In *Image Processing, 2006 IEEE International Conference on*, pages 1273–1276. IEEE, 2006.

[ZCP$^+$09] M. Zhou, H. Chen, J. Paisley, L. Ren, G. Sapiro, and L. Carin. Non-parametric bayesian dictionary learning for sparse image representations. In *Advances in Neural and Information Processing Systems (NIPS)*, 2009.

[ZLW$^+$10] Z. Zhou, X. Li, J. Wright, E. Candes, and Y. Ma. Stable principal component pursuit. In *IEEE International Symposium on Information Theory Proceedings (ISIT)*, pages 1518–1522. IEEE, 2010.

Dept. of Mathematics and Statistics
Current address: San Jose State University
E-mail address: `guangliang.chen@sjsu.edu`

Dept. of Mathematical Sciences
Current address: Claremont McKenna College
E-mail address: `dneedell@cmc.edu`