11-27-2018

Degraded Visual Environment Tracker

Mohammed Eid Abu Hussein

DEGRADED VISUAL ENVIRONMENT TRACER

by

Mohammed Eid Abu Hussein

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Major: Electrical Engineering

The University of Memphis

December 2018

ACKNOWLEDGMENTS

First and foremost, I give all thanks to God for His grace on my life to accomplish this degree at this stage in my life.

ABSTRACT

Compressive Sensing (CS) has proven its ability to reduce the number of measurements required to reproduce images with similar quality to those reconstructed by observing the Shannon-Nyquist sampling criteria. By exploiting spatial redundancies, it was shown that CS can be used to denoise and enhance image quality. In this thesis we propose a method that incorporates an efficient use of CS to locate a specific object in zero-visibility environments. This method was developed after multiple implementations of dictionary learning, reconstruction, detection, and tracking algorithms in order to identify the shortcomings of existing techniques and enhance our results. We show that with the use of an over-complete dictionary of the target our technique can perceive the location of the target from hidden information in the scene. This thesis will summarize the previously implemented algorithms, detail the shortcomings evident in their outputs, explain our setups, and present quantified results to support its efficacy in the results section.

TABLE OF CONTENTS

List of Figures

Abbreviations

Abbreviation	Meaning
DFT	Discrete Fourier transform
CoSaMP	CoSaMP Compressive sampling matching pursuit
OMP	Orthogonal Matching Pursuit
RIP	Restricted Isometry Property
SNR	Signal-to-noise ratio
StRIP	Statistical Restricted Isometry Property
HOG	Histogram of Oriented Gradients

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

Lau and Woodward [1] detailed a method for seeing through obscurants in a severe degraded visual environment (DVE). The method allows for the extraction of hidden information from the raw sensor data via computational imaging technologies. These authors begin by presenting an image that to the human eye represents a zero-visibility view of an target captured by a sensor through dense obscurants. The authors then show that it is possible to recover the hidden visual information of the object and display its visual cues to aircraft pilots to aid in landing and maneuvering. Lau and Woodward [1] used an iterative dictionary trained on the current location and view of the object and predicted its position in subsequent frames. This thesis expands that initial investigation by examining the effects of additional information regarding the shape of the object as well as the location in the extraction of target position. In addition to the shape investigation, results showing the effects of shape-based window size selection are also presented.

1.2 Contributions of This Study

The major contributions of this study are as follows:

- Extension and improvements to CS based tracking algorithms
- Quantification of the effects on shape and noise type on CS-based tracking
- Illustration of the effects of pristine vs iterative dictionary construction.

1.3 Notation

We define some basic notation used throughout this thesis to eliminate confusion.

- Bolded lower case letters ($\mathbf{x}$, $\mathbf{y}$, etc) are used exclusively for vectors.

- Double bars around a vector with an subscript p (ex. $\|\mathbf{x}\|_p$) indicate the lp norm of the vector $\mathbf{x}$. $\|\mathbf{x}\|_p = \sqrt[p]{\sum_i |x_i|^p}$

- Matrices are denoted with upper-case, bold letter ($\mathbf{\Phi}$, etc).

- $\mathbf{\Phi}^*$ indicates the Hermitian transpose of $\mathbf{\Phi}$.

- $\mathbf{\Phi}^\dagger$ indicates the pseudo-inverse of $\mathbf{\Phi}$ such that $\mathbf{\Phi}^\dagger = (\mathbf{\Phi}^*\mathbf{\Phi})^{-1}\mathbf{\Phi}^*$.

- T^C indicates the compliment of set T.

- We use the subscript notation $|T$ to show that a vector or matrix is being restricted to only certain elements or columns. For example, $\mathbf{x}_{|T}$ indicates the vector $\mathbf{x}$ is restricted only the elements given in T. $\mathbf{\Phi}_{|T^C}$ indicates that the matrix $\mathbf{\Phi}$ is restricted to the columns contained in T^C.

1.4 Basic Terminologies

(a) **Sampling Theorems:** Sampling is a fundamental way to represent and recover continuous signals (analog domain) in the field of signal processing. The Sampling theorem connects continuous signals and discrete signals by establishing guidelines connecting signal content, sample rates, and reconstruction error.

(b) **Shannon-Nyquist Sampling Theorem:** This theorem is named after Harry Nyquist and Claude Shannon to honor them for their contribution to the field of signal processing. Simply put, the theorem states that in order to obtain all relevant information in a signal, the sampling rate must be at least 2 times the bandwidth of the signal. For instance, if a simple sinusoid has the highest frequency as 50 Hz, then to capture all the relevant information, the Nyquist rate states that it must be sampled at at least 100 Hz.

Mathematically, if a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced

$\frac{1}{2B}$ seconds apart [35]. For a given sample rate f_s, perfect reconstruction is guaranteed possible for a band limit $B < \frac{f_s}{2}$. Hence, any sampling rate that is larger than $2B$ samples/second is said to be sufficient.

Meanwhile, it is noteworthy to remark here that perfect reconstruction may still be possible when the sample-rate criterion defined above is not satisfied using for example, compressed sensing or when other constraints on the signal are known.

(c) **Signal:** The description of signal(s) presented here copies heavily from Baraniuk [43]. Consider a real-valued, finite-length, one-dimensional, discrete-time signal $\boldsymbol{x}$, which we view as an $N \times 1$ column vector in $\mathbb{R}^N$ with elements $x[n], n = 1, 2, \cdots, N$. Any signal in $\mathbb{R}^N$ can be represented in terms of a basis of $N \times 1$ vectors $\{\Psi_i\}_{i=1}^N$. Forming the $N \times N$ basis matrix $\boldsymbol{\Psi} := [\Psi_1|\Psi_2|\cdots|\Psi_N]$ by stacking the vectors $\{\Psi_i\}_{i=1}$ as columns, we can express any signal $\boldsymbol{x}$ as

$$\boldsymbol{x} = \sum_{i=1}^{N} s_i \Psi \quad \text{or} \quad \boldsymbol{x} = \boldsymbol{\Psi x} \tag{1.1}$$

where $\boldsymbol{s}$ is the $N \times 1$ column vector of weighting coefficients $s_i = \langle \boldsymbol{x}, \Psi_i \rangle = \Psi_i^T \boldsymbol{x}$. Clearly, $\boldsymbol{x}$ and $\boldsymbol{s}$ are equivalent representations of the same signal, with $\boldsymbol{x}$ in the time domain and $\boldsymbol{s}$ in the $\boldsymbol{\Psi}$ domain.

(d) **Signal processing:** Signal processing concerns the analysis, synthesis, and modification of signals, which are broadly defined by Priemer [8] as functions conveying "information about the behavior or attributes of some phenomenon", such as sound, images, and biological measurements [10]. For example, signal processing techniques are used to improve signal transmission fidelity, storage efficiency, and subjective quality, and to emphasize or detect components of interest in a measured signal [6].

(e) **Compressible Signals:** If the sorted components of a signal decay rapidly obeying power law, then these signals are called compressible signals. A power law in statistics is a functional relationship between two quantities, where a relative change in one quantity results in a proportional relative change in the other quantity, independent of the initial size of those quantities: one quantity varies as a power of another. For instance, considering the area of a square in terms of the length of its side, if the length is doubled, the area is multiplied by a factor of four [36]. Meanwhile, there are several variants of power law, one variant is a power law with an exponential cutoff. This is simply a power law multiplied by an exponential function [38]:

$$f(x) \propto x^{\alpha} e^{\beta x} \tag{1.2}$$

(f) **Matroid:** A matroid is a mathematical structure that generalizes the notion of linear independence from vector spaces to arbitrary sets. If an optimization problem has the structure of a matroid, then the appropriate greedy algorithm will solve it optimally [28].

(g) **Sparsity:** Sparsity is a common term in compressed sensing. Sparsity is the inherent property of those signals for which, the whole of the information contained in the signal can be represented only with the help of few significant components, as compared to the total length of the signal. A signal can have sparse/compressible representation either in the original domain or in some other transform domain. Representative transforms are the Fourier transform, cosine transform, wavelet transform, etc. In this thesis, we will focus on signals that have a sparse representation, where $\boldsymbol{x}$ is a linear combination of just K basis vectors, with $K \ll N$. That is, only K of the s_i in equation (1.1) are nonzero and $(N - K)$ are zero. A few examples of representative signals with sparse representations in certain domains are as follows:

- Natural images which have sparse representations in the wavelet domain

- Speech signals can be represented by fewer components using Fourier transform
- Better models for medical images can be obtained through use of the Radon transform

. Mathematically, a sparse collection of data has a small number of non-zero values in some domain. If the data is such that only a few non-zero values contain sufficient magnitude to represent meaningful data it can also be classified as compressible. Sparsity is motivated by the fact that many natural and man-made signals are compressible in the sense that there exists a basis $\mathbf{\Psi}$ where the representation (1.1) has just a few large coefficients and many small coefficients.

Compressive/compressed sensing generally refers to techniques/methods designed to obtain the few non-zero coefficients directly in the sparse domain and thus avoid the processing and hardware overhead associated with traditional signal capture and subsequent domain transformation. The inefficiencies associated with the acquisition of sparse signals using traditional methods can be summarized as follows:

- Native domain sampling using Nyquist-criterion results in too many samples when compared to the actual information content of the signal. This is especially true for applications such as tracking where perfect reconstruction of the target is not required and only a few features of the target are necessary to identify it's current target location.
- Compression of the Nyquist sampled signal is completed by computing the necessary transform coefficients for all the samples, retaining only larger coefficients and discarding the smaller ones for storage/transmission purposes.

(h) **DFT Matrix:** A DFT matrix is an expression of a discrete Fourier transform

(DFT) as a transformation matrix, which can be applied to a signal through matrix multiplication. For instance, an N-point DFT is expressed as the multiplication $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{x}$, where $\boldsymbol{x}$ is the original input signal, $\boldsymbol{W}$ is the N-by-N square DFT matrix, and $\boldsymbol{X}$ is the DFT of the signal.

(i) **Kalman Filtering:** The filter is named after Rudolf E. Kalman, one of the primary developers of its theory. Kalman filtering, also known as linear quadratic estimation (LQE), is an algorithm that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone, by estimating a joint probability distribution over the variables for each time frame [30].

(j) **Tracking:** Tracking is the problem of generating an inference about the motion of an object given a sequence of images [48]. Analogies of "tracking" includes a spy "following" you, a missile "targeting" a ship, a detective "spotting" the suspects, and a typical car chase scene in a Hollywood movie [48].

(k) **Feature Descriptor:** A feature descriptor is a representation of an image or an image patch that simplifies the image by extracting useful information and throwing away extraneous information.

1.5 Theses Outline

The rest of the thesis is presented as follows: A summary explanation of traditional and compressed signal acquisition is presented. This is followed by an overview of the algorithmic implementations used to obtain the initial target tracking and reconstruction results. A detailed synopsis of the algorithmic changes and motivation behind their inclusion is presented next. The next section contains the results of the thesis along with a discussion of their relevant interpretations. The thesis is then concluded with some appropriate conclusions and avenues for future research directions.

CHAPTER 2

SIGNAL (DATA) ACQUISITION METHOD

Two types of signal acquisition (traditional and compressive sensing) are described in this chapter.The traditional methods can be classified into the two distinct categories of Shannon/Nyquist Sampling and the more modern Compression Approach. Due to the relevance of this thesis, more detail will be provided on the Compression Approach.

2.1 Traditional Approach

2.1.1 Shannon/Nyquist Sampling

The most common traditional methods of image and signal reconstruction from measured data follow the Shannon/Nyquist sampling theorem (defined above). The procedure for data acquisition, transmission and reception using Nyquist sampling is presented in Figure 1.

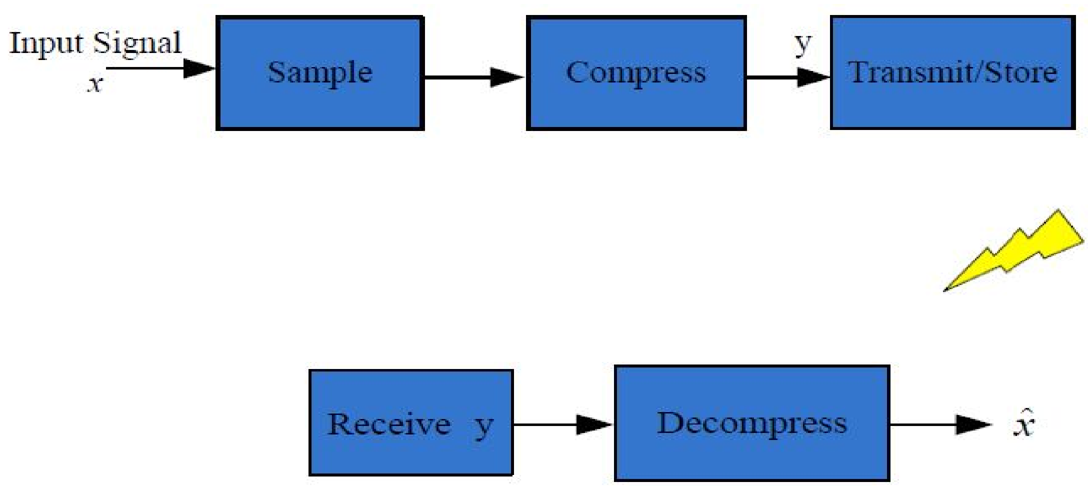

Figure 1: Traditional Data Acquisition. *Copied from* [47]

According to Fornasier and Rauhut [22], this principle is integral to the operation of most devices of current technology. Operations such as analog to digital conversion, medical imaging, or audio and video electronics all rely on the Nyquist criteria to ensure proper signal discretization and reconstruction. However, the major disadvantage of Nyquist sampling criteria is that in many important applications, the prescribed rate is so high that the resultant discretized data actually has more samples than is needed to complete a particular task. Due to cost and physical limitations in emerging areas such as smart cities and distributed or smart sensor networks, the data acquisition and processing of signals in application using Nyquist sampling continues to be a concern.

2.1.2 Compression Approach

To address the difficulties and challenges of using Nyquist sampling especially when dealing with high-dimensional data, scholars have suggested another method of signal acquisition known as compressible sampling. The purpose of which is to obtain the most succinct representation of the original signal while also achieving an acceptable level of distortion in the data. Transform coding, one of the most popular techniques for signal compression, typically dedicates on finding a basis or frame that provides sparse or compressible representations for signals [23]. Sparse representation, in particular, is a common way to sparsify a signal by transforming the signal to the orthonormal basis (e.g. Wavelet basis).

Both sparse and compressible signals can be represented with high fidelity by preserving only the values and locations of the largest coefficients of the signal. Mallat [42] reports that compressible signals are well approximated by K-sparse representations (the basis of transform coding). For instance, natural images tend to be compressible in the discrete cosine transform (DCT) and wavelet bases [42] on which the JPEG [25], and JPEG2000 [26] compression standards are based [43].

Transform coding plays a central role in data acquisition systems like digital cameras where the number of samples is high but the signals are compressible. In this framework, we acquire the full N-sample signal $\boldsymbol{x}$; compute the complete set of transform coefficients $\{s_i\}$ via $\boldsymbol{s} = \boldsymbol{\Psi}^T \boldsymbol{x}$; locate the K largest coefficients and discard the $(N - K)$ smallest coefficients; and encode the K values and locations of the largest coefficients [43]. Unfortunately, the sample-then-compress framework suffers from some inherent inefficiencies [43]: We enumerate the process below:

1. We must start with a potentially large number of samples N even if the ultimate desired K is small.

2. The encoder must compute all of the N transform coefficients $\{s_i\}$, even though it will discard all but K of them.

3. The encoder faces the overhead of encoding the locations of the large coefficients.

Overall, since typical signals have some compressible structure, the process of massive data acquisition followed by compression is extremely wasteful.

2.2 Compressive Sensing

Recall that the Nyquist-Shannon sampling theorem requires a certain minimum number of samples be obtained in order to perfectly capture an arbitrary band-limited signal. However, a great reduction in the required number of stored measurements is possible if the signal is sparse in a known basis. Thus, the Nyquist-Shannon theorem represents a sufficient condition but not a necessary condition for perfect reconstruction since signal sparsity or compresssibility implies signal recovery could be obtained from far fewer saved data points.

To address this, Candès, Romberg, Tao, Donoho in 2006 ([3], [4], [5]) proposed compressive sensing (CS); sometimes referred to as compressed sensing, compressive sampling, or sparse sampling by some authors and researchers. CS as a method of signal reconstruction which uses far fewer samples for signal acquisition. Thus, CS provides a stricter sampling condition when the signal is known to be sparse or compressible. The fundamental idea behind CS is, instead of first sampling the data at a high rate and then compressing the sampled data, directly sense the data in a compressed form at a lower sampling rate.

This sampling scheme relies heavily on the following conditions: (1) the sparsity of the signals (2) incoherence. These two conditions are discussed in section (2.3). Meanwhile, there are three main problems with the use of CS:

- sparse representation,
- measurement matrix construction, and
- reconstruction algorithm.

The pictorial representation of CS data acquisition is presented in Figure 2. Basically,

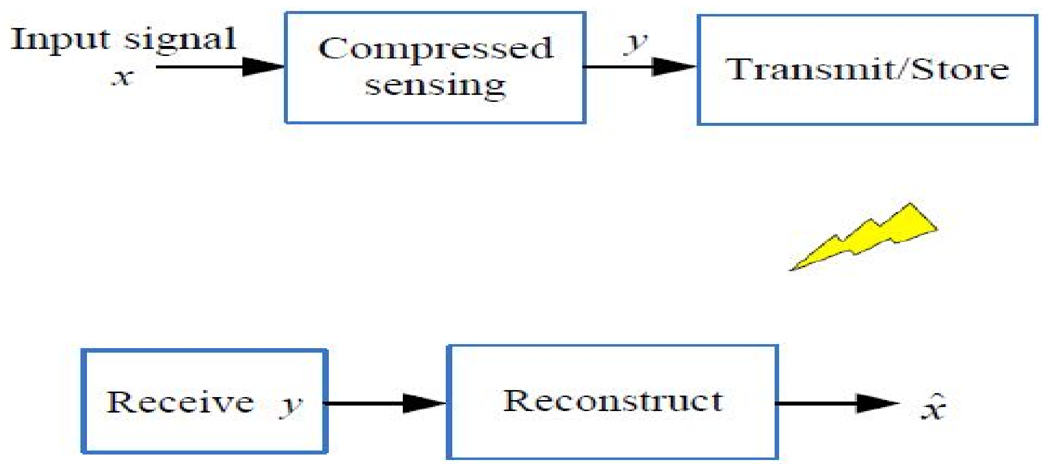

Figure 2: Compressive sensing data acquisition. *Copied from*[47]

CS is used for the acquisition of signals which are either sparse or compressible in either the original domain or in some transform domain (like Fourier, sine). To recover the sample, there are two distinct approaches and each has its own set of drawbacks. One approach is random and the other is deterministic. Examples of the random method of signal and image recovery are those which make use of Gaussian and Bernoulli random matrices; these have been investigated by several authors in the past. In fact, it is the first approach employed for image recovery. Although it enjoys sound theoretical backing [18], it suffers the disadvantage that it is complex and requires large storage during its implementation [19].

Meanwhile, a number of methods have been suggested by several scholars to address the shortcomings of the random method describe above. Some of these methods developed and applied a deterministic matrix for the measurement strategies. For instance, chirp sequences [20], Kerdock and Delsarte-Goethals codes [12], dual BCH codes [13] and second order Reed-Muller codes [14]. Other techniques for deterministic construction, based on finite fields, representation theory, and character sequences, can be found in ([17], [15], [16]). The deterministic matrices can guarantee reconstruction performance that is empirically reliable, with fast processing and low complexity.

2.2.1 Basic Math Behind CS Acquisition Model

In this section, we present the basic math behind CS signal acquisition and note that in our presentation, we draw heavily from Baraniuk [43].

In real a world setting, theoretical calculations require that N measurements of the signal of interest be obtained to satisfy traditional reconstruction requirements. However, in practice (perhaps due to cost, computational consideration, and difficulty in dealing with the population) we can only take M measurements ($M < N$). Basically, for the mathematical setup of CS, the following components are required:

- **Signal:** Let $\boldsymbol{x}$ be $N \times 1$ dimensional signal of interest (vector representing our original signal) with $\boldsymbol{x} = \sum_{i=1}^{N} x_i$ and assuming the sparsity $\boldsymbol{x} = \boldsymbol{\Psi} b$ where b is sparse.

- **Sensing Matrix:** Let $\boldsymbol{\Phi}$ be $M \times N$ matrix with ($M \ll N$), $\boldsymbol{\Phi}$ is usually defined as the sensing matrix. The job of the matrix $\boldsymbol{\Phi}$ is to ensure passage from $\boldsymbol{x}$ to $\boldsymbol{y}$ either through a random measurements or transformations, and sometimes by the combination of the two.

- **Output:** Let $\boldsymbol{y}$ ($M \times 1$) be vector of our output values i.e., a vector denoting the measurements taken. That is

$$\boldsymbol{y} = \boldsymbol{\Phi x} \tag{2.1}$$

As stated earlier, CS is a more general signal acquisition technique that condenses the signal directly into a compressed representation without going through the intermediate stage of taking N samples. Consider the more general linear measurement process that computes $M < N$ inner products between $\boldsymbol{x}$ and a collection of vectors $\{\phi_j\}_{j=1}^{M}$ as in $y_j = \langle \boldsymbol{x}, \phi_j \rangle$. Stacking the measurements y_j into the $M \times 1$ vector $\boldsymbol{y}$ and the measurement vectors ϕ_j^T as rows into an $M \times N$ matrix $\boldsymbol{\Phi}$ and substituting in (1.1), we can write

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{\Phi x} \\ &= \boldsymbol{\Phi \Psi s} \\ &= \boldsymbol{\Theta s} \end{aligned} \tag{2.2}$$

where $\boldsymbol{\Theta} = \boldsymbol{\Phi\Psi}$ is an $M \times N$ matrix. The pictorial description of (2.2) is presented in Figure 3.

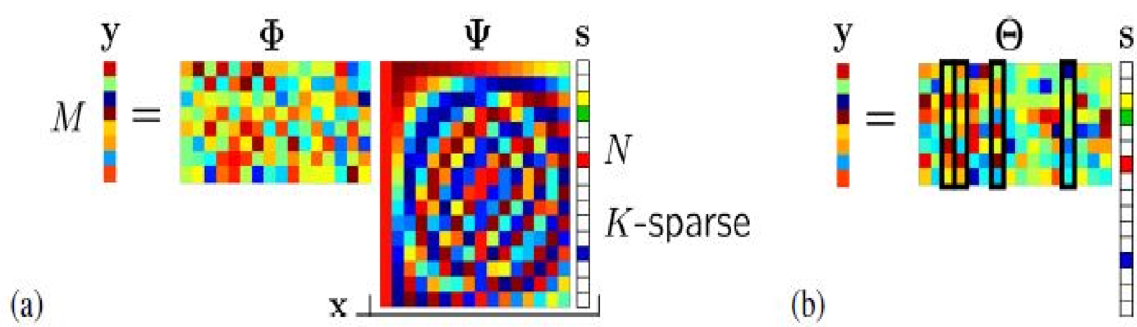

Figure 3: (a) Compressive sensing measurement process with (random Gaussian) measurement matrix $\boldsymbol{\Phi}$ and discrete cosine transform (DCT) matrix $\boldsymbol{\Psi}$. The coefficient vector s is sparse with $K = 4$. (b) Measurement process in terms of the matrix product $\boldsymbol{\Theta} = \boldsymbol{\Phi\Psi}$ with the four columns corresponding to nonzero s_i highlighted. The measurement vector $\boldsymbol{y}$ is a linear combination of these four columns. *Copied from* [43]

Usually in equation (2.1), some known, unknown, and some assumed values are required for operation. For instance, the $M \times N$ sensing matrix $\boldsymbol{\Phi}$ and the output vector $\boldsymbol{y}$ are assumed known, however, $\boldsymbol{x}$ is unknown. The assumption here is that vector $\boldsymbol{x}$ is sparse. The question then is how to determine or figure out what $\boldsymbol{x}$ is? This is where CS recovery methods comes in.

Understanding the orthonormal bases in CS

In this section, we present two orthonormal bases (or orthobases) which are critical for CS.

(a) Representation basis: Re-emphasizing the importance of signal sparsity, we note that the signal must be sparse in order to perform compressed sensing. The representation basis denoted Ψ is the mathematical way of getting the signal into a sparse domain. Assume a native time domain application. In some cases, $\boldsymbol{\Psi}$ can be as simple as an identity matrix with dimension I_N. However, in specific situations, the time-domain signal may be sparse in a different domain (e.g., the frequency domain). Thus, Ψ will be the basis which transforms a time-domain signal into the frequency domain, i.e. the discrete Fourier transform (DFT) matrix. According to Candès & Wakin [2], there are many possibilities for Ψ, and picking the right one is dependent on the application at hand. For frequency-sparse signals, the DFT is applicable and for 2-D images, one of the many Wavelet transform bases may be applicable.

(b) Sensing basis: The second orthornormal basis for CS is known as the sensing basis and it is usually denoted by $\boldsymbol{\phi}$. $\boldsymbol{\phi}$ represents the domain from which we extract values from the signal. For instance, it may be as simple as the matrix of random Gaussian entries and sometimes may be spikes. Usually, spikes is employed to obtain a smaller (compressed) amount of data.

2.2.2 Noisy CS

Sparse signal recovery in the presence of noise has been intensively investigated in many recent literature treatments because real-world devices are subject to at least a small amount of noise [44]. The noisy measurement is represented as

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{y} + \boldsymbol{z} \\ &= \boldsymbol{\Phi x} + \boldsymbol{z} \end{aligned} \tag{2.3}$$

where the measurements are corrupted by $\boldsymbol{z}$, which is the additive white Gaussian noise of zero mean and variance σ^2.

2.3 Necessary and Sufficient Conditions for Perfect Recovery

2.3.1 Restricted Isometry Property

The restricted isometry property (RIP) of a compressed sensing matrix is an important necessary condition to guarantee the sparse signal recovery [40]. This condition is given as

$$(1 - \delta_s) \leq \frac{\|\boldsymbol{\Theta x}\|_2}{\|\boldsymbol{x}\|_2} \leq (1 + \delta_s) \tag{2.4}$$

where k is the sparsity of vector s, $\boldsymbol{x}$ is a vector having the same k-nonzero entries as s and $\delta_s > 0$ is known as RIP constant [43].

A sensing matrix $\boldsymbol{\Theta}$ satisfies the restricted isometry property (RIP) if δ_s is not too close to 1 [40]. The RIP is a very restrictive condition and the currently known measurement matrices obeying the RIP with near-optimal number of measurements fall into two categories according to [41]:

(a) Random matrices such as Gaussian or Bernoulli matrices with the entries of Gaussian or Bernoulli distribution,

(b) Random partial Fourier matrix or Hadamard transform matrix are obtained by choosing K rows uniformly at random from a normalized $N \times N$ Fourier or Hadamard transform matrices.

Due to the storage limitations of random matrices, in some applications, deterministic sensing matrices have been applied in literature. For instance, [19] demonstrated an approach on deterministic sensing matrix on $\boldsymbol{\Theta}$ to ensure that $\boldsymbol{\Theta}$ is nearly-isometric with high probability regarding s-sparse signals on a uniform distribution.

The condition in (2.4) states that matrix $\boldsymbol{\Theta}$ must preserve the distance between two k-sparse vectors. However, a sufficient condition for a robust solution is that matrix $\boldsymbol{\Theta}$ must satisfy the relation given in (2.4) for an arbitrary k-sparse vector $\boldsymbol{x}$. One of the limitations of the above condition is that it is difficult to calculate δ, thus, another simpler condition which guarantees stable solution is proposed. This new condition is known as incoherence (see, [11]).

2.3.2 Incoherent Sampling

This is an alternative approach to ensure that the measurement matrix is stable. Following from (2.1), let $(\boldsymbol{\Phi}, \boldsymbol{\Psi})$ be a pair of orthonormal bases of $\mathbb{R}^N$.

(a) $\boldsymbol{\Phi} = (\phi_j)$ is used for sensing: A is a subset of rows of Φ^*

(b) $\boldsymbol{\Psi} = (\psi_k)$ is used to sparsely represent x: $x = \boldsymbol{\Psi} b$, and b is assumed sparse.

Definition: The coherence between $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ is

$$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) = \sqrt{N} \max_{1 \leq k,j \leq N} |\langle \phi_j, \psi_k \rangle| \tag{2.5}$$

If $\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) = C$ a constant, then $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are called incoherent. A classical example of incoherent sampling is as follows: if $\boldsymbol{\Psi}$ is identity (e,g., signal is sparse in canonical/Kroneker basis) and $\boldsymbol{\Phi}$ is discrete Fourier basis, the Kronecker and Fourier bases are incoherent because $\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) = 1$

2.4 Strategies for CS Acquisition Model

Several strategies and procedures have been proposed in the literature to ensure that the CS reconstruction is properly conducted. The main requirement of CS for proper reconstruction is that the measurements must be taken randomly. In this section, we briefly discuss three operating principles/techniques of signal acquisition that have been proposed in the literature to satisfy this requirement. Meanwhile, we would like to note that this section benefits and draws heavily from Rani et al., [49].

1. **Random Demodulator (RD)**: the random demodulator (RD), also referred to as analog information converter (AIC) was proposed by [53], is a compressive sampler used to sample signals at a rate below the Nyquist. It is an efficient wideband signal sampler. The details of this method as described in a comprehensive review by [49] is as follows:

    - The input signal $x(t)$ is first multiplied with a pseudorandom sequence consisting of $+/-1s$, known as chipping sequence $p_c(t)$. This is equivalent to the convolution in frequency domain and results in spreading the signal frequency to low frequency regions.
    - The next stage is an integrator, serving as a low pass filter (LPF), which is used to obtain a unique frequency signature of signal in lower frequency region.
    - In the next stage, two unique frequency signatures are obtained from the RD for two different frequency signals and the highest frequency of the signal so obtained lies in lower frequency region (and hence can be sampled using a low rate ADC to obtain vector of digital measurements). These fewer compressive measurements can then easily be stored or transmitted.
    - The unique frequency signature is the information about the original signal that is contained in random measurements and helps in reconstructing the original signal back from compressive measurements.

2. **Random Filtering**: This technique, proposed by [46] is applicable for compressible, continuous and streaming signals. The input signal x, is acquired by performing convolution with a random-tap finite impulse response (FIR) filter

h. The first stage is then followed by downsampling the filtered signal by a factor of $\lfloor n/m \rfloor$ to obtain compressive measurements y, this is shown in Fig.7 of of [49]. The filter taps are random and can be obtained from random distributions like Gaussian distribution $\mathcal{N}(0,1)$ with zero mean and variance one, Bernoulli distribution of $+/-1s$.

3. **Random Equivalent Sampling** (RES): This is another technique which is based on random sampling mechanism. This method is used to sample the periodic high frequency analog signals at sub-Nyquist-rate. The use of CS reconstruction for the signals acquired using RES, was proposed by [50]. CS reconstruction for RES achieves higher SNR while requiring fewer RES samples compared to the traditional method. The block diagram of signal acquisition using RES is shown in Fig.9 of [49]. RES samples the signal at random positions by dithering the phase of ADC sampling clock with the help of a variable delay circuit implemented using the control module. A level-triggering circuit is used to provide fixed reference trigger-pulses to the control module to align the samples. The time-to-digital converter (TDC) circuitry is used to measure the relative sample positions, which are required to generate the measurement matrix using Whittaker-Shannon interpolation formula and the measurement matrix so generated is used for applying the CS reconstruction on RES sampled signal [49].

2.5 Reconstruction Approaches for CS

In this section, we discuss some of the reconstruction methods and or algorithms employed in CS schemes. In particular, we shall focus on some convex optimization and greedy methods. These are required to find out the sparse estimation of the original input signal from the compressive samples collected in some suitable frame or dictionary. We note again that some of the descriptions in this section draw directly from [49] with slight or no modifications.

2.5.1 Convex Optimization Methods

(1) **Basis Pursuit**: Basis Pursuit (BP) was proposed by [55]. It is a convex optimization problem, which searches for a solution having minimum ℓ_1-norm, subject to the equality constraint given below

$$\widehat{s} = arg\min_{s} \|s\|_1; \quad \text{subject to} \quad \Theta s \tag{2.6}$$

BP is used in CS to find the sparse approximation $\widehat{s}$ of input signal x, in dictionary or matrix Θ, from compressive measurements y. BP can recover faithfully only if, the measurements are noise-free [49].

(2) **Basis Pursuit Denoising:** If the measurements are corrupted by noise, then to suppress the noise, exact reconstruction is not desired. The denoising can be achieved by relaxing the equality constraint in equation (2.6) to account for measurement noise. The widely used formulations for robust data recovery from noisy measurements are Dantzig selector, basis pursuit denoising (BPDN), total variation (TV) minimization based denoising, etc. However, BPDN is here in what follows:

BPDN is also introduced by [55], in the field of computational harmonics. It is similar to the Least Absolute Shrinkage Selection operator (LASSO) method suggested by Tibshirani [51]. To account for the noise in measurements, BPDN poses the sparse estimation problem as an optimization problem given in equation (2.7). It shows that, BPDN searches for a solution having minimum ℓ_1-norm subject to the relaxed condition on constraint. The quadratic inequality constraint used by BPDN states that for the obtained solution, the squared ℓ_2-norm of the error between y and Θs should be less than or equal to ϵ.

$$\widehat{s} = arg\min_{s} \|s\|_1; \quad \text{subject to} \quad \frac{1}{2}\|(y - \Theta s)\|_2^2 \leq \epsilon \tag{2.7}$$

where, ℓ_2, also known as euclidean norm, represents the length or size of a vector [52]. Some algorithms solve BPDN in its Lagrangian form, which is an unconstrained optimization problem and can be rewritten as

$$\widehat{s} = arg\min_{s} \|s\|_1 + \frac{1}{2}\|(y - \Theta s)\|_2^2 \tag{2.8}$$

Equations (2.7) and (2.8) are equivalent for certain value of λ, which is unknown a priori. The value of λ balances between error and sparsity of solution. Popular algorithms that have been used to solve (2.8) are primal-dual interior-point method, fixed-point continuation, etc.

(3) **Solvers:** Solvers are the techniques or methods required to obtain solutions to the optimization problems described above. The BP problem in equation (2.7) can be solved by linear programming algorithms like the simplex algorithm which is also known as BP-simplex, and the interior-point algorithm with is also known as BP-interior. Here, simplex can be defined as a convex polyhedron formed by the set of all feasible solutions or points [55]. The algorithmic steps of these solvers are described below [49]:

  (a) **Basis pursuit - simplex algorithm:** The basic steps for solving the BP problem using simplex algorithm is as described below:

    (i) **Initial basis selection**: Initial bases are a set of n linearly independent columns selected from a dictionary. Generally, the initial bases are used to find the initial feasible solution. This solution corresponds to one of vertices of the simplex.

    (ii) **Swapping**: Swap one column in current basis with the column external to the basis that gives best improvement in objective function. This is equivalent to jumping on the vertices of simplex or searching the solution, in the direction of improving the objective function.

    (iii) Repeat step (ii) until no further improvement is possible. At last, the optimal solution is achieved.

  (b) **Basis pursuit - interior algorithm:** The basic steps for solving the BP problem using interior-point algorithm are described below [49]:

(i) Initial solution: start from a non-sparse initial solution which is well inside the interior of simplex.

(ii) Apply transformation that sparsifies the solution. This corresponds to moving the solution inside the simplex in the direction of a vertex.

(iii) Repeat step (ii), until a solution having $\leq n$ significant non-zero entries, is reached. The result so obtained is a feasible solution and corresponds to the vertex of simplex.

Apart from simplex and interior-point algorithms, the other popular algorithms for solving convex optimization problems are fixed point continuation, gradient projection for sparse representation, Bregman iteration algorithm, etc.

2.5.2 Greedy Methods

There are no watertight rules applicable to all computation problems as different problems usually require different approaches. This is especially true for problem requiring algorithm design as an integral part of their solution. This also holds true for the "Greedy Methods" employed as techniques to obtain system solutions. Black [29] defines greedy algorithm as an algorithm that always takes the best immediate, or local, solution while finding an answer. Greedy algorithms find the overall, or globally, optimal solution for some optimization problems, but may find less-than-optimal solutions for some instances of other problems [29].

Greedy algorithms have proven to be quite successful in some problems (e.g., Huffman encoding which is used to compress data); However in many problems (e.g., NP-complete cases), a greedy strategy does not produce an optimal solution. While the greedy algorithms are quite efficient in terms of execution time, the choice of how to make the local decisions in the algorithm may affect performance. Using a purely random approach may seem natural; however, it causes slower performance when compared to heuristic methods.

To solve any problem using greedy algorithms, the following properties must be satisfied:

(a) **Greedy choice property:** A global (overall) optimal solution can be reached by choosing the optimal choice at each step.

(b) **Optimal substructure:** A problem has an optimal substructure if an optimal solution to the entire problem contains the optimal solutions to the subproblems.

In other words, greedy algorithms work on problems for which it is true that, at every step, there is a choice that is optimal for the problem up to that step, and after the last step, the algorithm produces the optimal solution of the complete problem [27]. The choice made at each step of the algorithmn must be feasible (must satisfy the problem's constraints), locally optimal (be the best local choice among all feasible choices available on that step), and irrevocable (meaning once made, it cannot be changed on subsequent steps of the algorithm.) Essentially, Greedy Algorithms solve combinatorial problems having the properties of a matroid. In general, greedy algorithms have five components [31]:

(a) A candidate set, from which a solution is created

(b) A selection function, which chooses the best candidate to be added to the solution

(c) A feasibility function, that is used to determine if a candidate can be used to contribute to a solution

(d) An objective function, which assigns a value to a solution, or a partial solution, and

(e) A solution function, which will indicate when we have discovered a complete solution

2.6 Orthogonal Matching Pursuit Algorithm

This section describes an iterative greedy algorithm for signal recovery known as the orthogonal matching pursuit (OMP). The algorithm was proposed/developed in Tropp & Gilbert [34]. It is commonly used as an algorithm for recovery of sparse signals due to its low complexity and simple implementation. The procedure of OMP is defined in [34] as follows:

The input is a $K \times N$ measurement matrix, a K-dimensional measurement vector $\boldsymbol{y}$, and the sparsity level $\boldsymbol{s}$. As for the output, we have an index set Λ containing $\boldsymbol{s}$ elements and a signal estimate $\widehat{\boldsymbol{x}} \in R^n$. The procedure for recovering signal through OMP is as follows [54]:

(a) Initialize a residual vector $\boldsymbol{r}_0 = \boldsymbol{y} = (y_0, \cdots, y_{k-1})^T$ and $\Lambda = \phi$ at iteration $i = 0$.

(b) At iteration i, compute $\boldsymbol{f} = \boldsymbol{A}^H \boldsymbol{r}_i = (f_0, \cdots, f_{N-1})^T$, find the peak of $\boldsymbol{f}$, and record its position as n_i i.e., $n_i = arg \max\limits_{t=0,\cdots,N-1} |ft|$.

(c) Update the index set $\Lambda \leftarrow \Lambda \cup \{n_i\}$ and the submatrix $\boldsymbol{A_{i+1}} = [\boldsymbol{A_i a_{n_i}}]$. Note that $\boldsymbol{A_0}$ is an empty matrix.

(d) Solve a least-square problem to obtain $\boldsymbol{b_i} = arg \min\limits_{b} \|\boldsymbol{y} - \boldsymbol{A_{i+1} b}\|_2$.

(e) Update the residual by $\boldsymbol{r}_{i+1} = \boldsymbol{y} - \boldsymbol{A_{i+1} b}$.

(f) If $i < s - 1$, then $i \leftarrow i + 1$ and repeat (b) - (e). If $i = s - 1$, stop the iteration. The nonzero entry of $\widehat{\boldsymbol{x}}$ is set by $\widehat{\boldsymbol{x}}_{n_j} = \boldsymbol{b}_j$ for $n_j \in \Lambda$, where $\boldsymbol{b}_j$ is the jth element of $\boldsymbol{b}_{s-1}$.

Note that the measurement procedure in compressed sensing is summarized by $y = \boldsymbol{Ax}$, where $\boldsymbol{y}$ is a linear combination of $\boldsymbol{s}$ columns in $\boldsymbol{A}$. In the reconstruction part, we have to determine which columns of $\boldsymbol{A}$ participated in this measurement and the coefficients of these columns contributed in the measurement. The idea behind this algorithm is to choose columns in a greedy fashion [39]. At each iteration, we choose the column of $\boldsymbol{A}$ that is the most strongly correlated with the remaining part of vector $\boldsymbol{y}$. Then the coefficients of the chosen columns are calculated in a least-square manner. Finally, we subtract off these columns' contribution to $\boldsymbol{y}$ and iterate on the residual. After $\boldsymbol{s}$ iterations, the algorithm will have identified the correct set of columns together with their corresponding coefficients [54].

2.7 CoSaMP Based Recovery

This section introduces the Compressive Sampling Matching Pursuit (CoSaMP) algorithm described in Needell & Tropp [45]. The algorithm is useful and general for recovery of sparse signal or image. As input, the CoSaMP algorithm requires four pieces of information:

1. A $M \times N$ measurement matrix $\mathbf{\Phi}$
2. A M-dimensional measurement vector $\boldsymbol{y}$
3. The sparsity level $\boldsymbol{s}$ of the approximation to be produced.
4. A halting criterion.

The algorithm is initialized with a trivial signal approximation, which means that the initial residual equals the unknown target signal. During each iteration, CoSaMP performs five major steps [45]:

- **Identification**. The algorithm forms a proxy of the residual from the current samples and locates the largest components of the proxy.
- **Support Merger**. The new support set is united with the set of components that appear in the current approximation.

- **Estimation**. The algorithm solves a least-squares problem to approximate the target signal on the merged set of components.
- **Pruning**. The algorithm produces a new approximation by retaining only the largest entries in this least-squares signal approximation.
- **Sample Update**. The samples are updated so that they reflect the residual, the part of the signal that has not been approximated

These steps are repeated until the hating criterion is triggered. The following table lists the steps of the CoSaMP algorithm [45]:

---

1) Initialization:

$\mathbf{a}^0 = 0$ ($\mathbf{x}_k$ is the estimate of $\mathbf{a}$) at the k^{th} iteration

$\mathbf{v} = \mathbf{u}$ (the current residual)

---

2) Loop until convergence

i) Form signal proxy or compute the current error:

$\mathbf{y} = \mathbf{\Phi}^* \mathbf{v}$.

ii) Compute the best $2s$ support set of the error (index set):

$\Omega = \mathbf{y}_{2s}$.

iii) Merge the strongest support sets:

$T = \Omega \bigcup \mathrm{supp}(\mathbf{a}^{k-1})$.

iv) Perform a Least-Squares Signal Estimation:

$\mathbf{b}_{|T} = \mathbf{\Phi}^{\dagger}_{|T}\mathbf{u}$, $\mathbf{b}_{|T^c} = 0$.

v) Prune $\mathbf{a}^k$ and compute $\mathbf{v}$ for next round:

$\mathbf{a}^k = \mathbf{b}_s$,

$\mathbf{v} = \mathbf{u} - \mathbf{\Phi}\mathbf{a}^k$.

2.8 Kalman Filter Tracker

Generally Kalman filters refer to the set of algorithms that take advantage of a series of measurements collected over time to produce an estimate of some unknown variable. They are often used for object tracking and find use in a wide range of applications such as the Space Shuttle, the Patriot missile system, and the NY stock exchange. They are especially convenient for objects in which the motion model is known since they incorporate some extra information in order to estimate the next object position more robustly. They can be used for general purpose single object tracking assuming some constraints.

2.8.1 Tracking with Kalman Filter

Tracking problem applications can usually be broken down into two subproblems

(1) Acquisition/Detection: finding the object of interest (the target) for the first time

(2) Tracking/Prediction: guessing where it's going to be in the next frame Kalman Filter

The Kalman filter is the optimal filter (in the least mean squared error sense) for target track prediction.

2.8.2 Kalman Filter Algorithm

The algorithm works in a two-step process. In the prediction step, the Kalman filter produces estimates of the current state variables, along with their uncertainties. Once the outcome of the next measurement (necessarily corrupted with some amount of error, including random noise) is observed, these estimates are updated using a weighted average, with more weight being given to estimates with higher certainty. The algorithm is recursive. It can run in real time, using only the present input measurements and the previously calculated state and its uncertainty matrix; no additional past information is required.

Using a Kalman filter does not assume that the errors are Gaussian Kalman [9]. However, the filter yields the exact conditional probability estimate in the special case that all errors are Gaussian.

Mathematical Model of Tracking

The assumption for the general model of tracking includes the following:

- There are moving objects, which have an underlying state X
- There are measurements Y, some of which are functions of this state
- There is a clock
  - at each tick, the state changes
  - at each tick, we get a new observation

Meanwhile, the mathematical model of tracking involve three main steps, these are;

- **Prediction:** We have seen $\boldsymbol{y}_0, \cdots, \boldsymbol{y}_{i-1}$, what state does this set of measurements predict for the ith frame? To solve this problem, we need to obtain a representation of $Pr\left(\boldsymbol{X}_i|\boldsymbol{Y}_0 = \boldsymbol{y}_0, \cdots, \boldsymbol{Y}_{i-1} = \boldsymbol{y}_{i-1}\right)$.
- **Data Association:** Some of the measurements obtained from the ith frame may tell us about the object's state. Typically, we use $Pr\left(\boldsymbol{X}_i|\boldsymbol{Y}_0 = \boldsymbol{y}_0, \cdots, \boldsymbol{Y}_{i-1} = \boldsymbol{y}_{i-1}\right)$ to identify these measurements.
- **Correction:** Now, after having $\boldsymbol{y}_i$ - the relevant measurements - we need to compute a representation $Pr\left(\boldsymbol{X}_i|\boldsymbol{Y}_0 = \boldsymbol{y}_0, \cdots, \boldsymbol{Y}_i = \boldsymbol{y}_i\right)$

2.9 Histogram of Oriented Gradients (HOG) for Object Detection

One of the most popular, fast and efficient feature descriptor methods that can be widely employed on several domains to characterize objects through their shapes (especially in computer vision and image processing) is the Histograms of Oriented Gradients (HOG). It was introduced by Dalal and Triggs [32] at the Computer Vision and Pattern Recognition (CVPR) conference in 2005. The main purpose of HOG is detection of an object. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform (SIFT) descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

As opposed to SIFT object recognition, HOG detection is a fairly simple to understand dense feature extraction method for images . One of the main reasons for this is that it uses a global features to describe an object rather than a collection of local features. In other words, the entire object is represented by a single feature vector as opposed to many feature vectors representing smaller parts of the object.

2.9.1 Procedure for HOG Feature Descriptor

The description of the procedure for HOG calculation draws heavily from the article by Vocal Technology website [33].

The first step in HOG detection is to divide the source image into blocks (for example 16 × 16 pixels). Each block is divided into small regions, called cells (for example 8×8 pixels). Blocks usually overlap each other so that the same cell may be in several blocks. For each pixel within the cell the vertical and horizontal gradients are obtained. The simplest method to do that is to use 1-D Sobel vertical and horizontal operators:

$$G_x(y,x) = Y(y,x+1)Y(y,x-1) \tag{2.9}$$

$$G_y(y,x) = Y(y+1,x)Y(y-1,x) \tag{2.10}$$

where $Y(y,x)$ is the pixel intensity at coordinates x and y, $G_x(y,x)$ is the horizontal gradient, and $G_y(y,x)$ is the vertical gradient.

The magnitude and phase of the gradient are determined as:

$$G(y,x) = \sqrt{\left[G_x(y,x)\right]^2 + \left[G_y(y,x)\right]^2} \tag{2.11}$$

$$\theta(y,x) = \arctan\left[\frac{G_y(y,x)}{G_x(y,x)}\right] \tag{2.12}$$

Next, the HOG is created for each cell. For the histogram, Q bins for the angle are chosen (for example $Q = 9$). Usually unsigned orientation is used, so angles below $0°$ are increased by $180°$.

Since different images may have different contrasts, contrast normalization can be very useful. Dalal and Triggs [32] explored four different methods for block normalization. Let v be the non-normalized vector containing all histograms in a given block, $\|v\|_k$ be its k-norm for $k = 1, 2$ and s be some small constant (the exact value, hopefully, is unimportant). Then the normalization factor can be one of the following [37]:

(a) L2 Norm: $f = \frac{v}{\sqrt{\|v\|_2^2 + s^2}}$

(b) L1 Norm: $f = \frac{v}{\|v\|_1 + s}$

(c) L1 sqrt-norm: $f = \sqrt{\frac{v}{\|v\|_1 + s}}$

(d) L2-hys: L2-norm followed by clipping (limiting the maximum values of v to 0.2) and renormalizing, as in [7]

A descriptor is assigned to each detector window. It consists of all the cell histograms for each block in the detector window and can be used as information for object recognition, training and testing. The data obtained from the descriptor may then be passed on to the many possible existing methods used to classify objects such as support vector machines, neural networks, etc.

CHAPTER 3

APPROACH DEVELOPMENT AND EVALUATION

In order to reach the improved results of the approach demonstrated in this thesis, multiple reconstruction algorithms and parameters were tested. This chapter will list the tested strategies and report the results from the experiments conducted to study the performance of each presented image reconstruction algorithm.

3.1 Simulation Study and Dataset

In order to normalize the attained results the same data set is used to compare the performance of each approach included in this thesis. The data set consists of a video containing 270 frames of a bouncing ball. The video was captured at a frame rate of 30 frames per second (fps) with each frame size being 670×670 pixels.

The dictionary contains 270 samples of pristine shifted images of the target. Patches of 8 pixels are extracted to train the dictionary. We test each approach by adding high level white noise (Var= 1.2), salt and pepper with density of 0.8 , and speckle noise with a variance of 0.8. Sample images are displayed in figure 4.

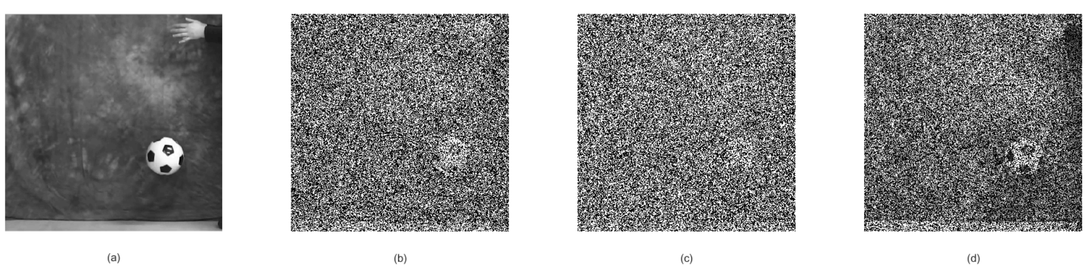

Figure 4: Figure(a) shows the original image. Figure (b) shows the image after white noise is added, with a PSNR of 6.92 dB, Figure (c) shows the noisy image when salt and pepper is added with PSNR of 6.34 dB, Figure (d) shows the image when speckle noise is added with a PSNR of 8.06

The Kalman tracker is included to predict the location of the object in the next frame K_{i+1} from information obtained from the current frame K_i. The filter effectively sets the search area within the next frame. This step will help reduce calculation time per frame since we are only reconstructing the search area instead of the entire scene. The search location as well as the dictionary will be updated on a frame by frame basis.

The simulation performance is evaluated by comparing the PSNR between the original, noisy, and algorithmically reconstructed images. If the reconstruction scores higher than the noisy image then the reconstruction approach is successful. The remaining segments of this section will detail the changes implemented to improve the performance of the tracker. The first section of this chapter will explain the training of the SVM detector and the dataset used. The second and third sections will contain the transitional approach tested to reach our final and optimal setup. The algorithm is demonstrated in section 3.

3.1.1 SVM Detector

The SVM classifier is trained using a 1000 sample dataset consisting of translated poses of the object of interest. The samples are divided into a training set using 80% of the samples, and a test set using 20%. The trained model is tested with the test set. $accuracy = 86.1\%$, run time 1.42 seconds.

3.2 Basis Pursuit

In this approach, the ℓ_1-norm was used to reconstruct our noisy frames. As mentioned in section 2.5.1, convex-optimization techniques do not perform well for noisy images. Our test results confirm this statement. After calculating the ℓ_1 norm of the images in Figure 4, the reconstructions are shown in Figure 5. Even for a large ϵ, the process was not effective in reducing noise effects in the image. Due to the poor performance, the object detector algorithm was not able to detect and objects in the reconstructed scene for SNR of 5 dB or less.

This method was evaluated by measuring the error in the object location with respect to the correct location (in pixels), and then subsequently comparing it to the

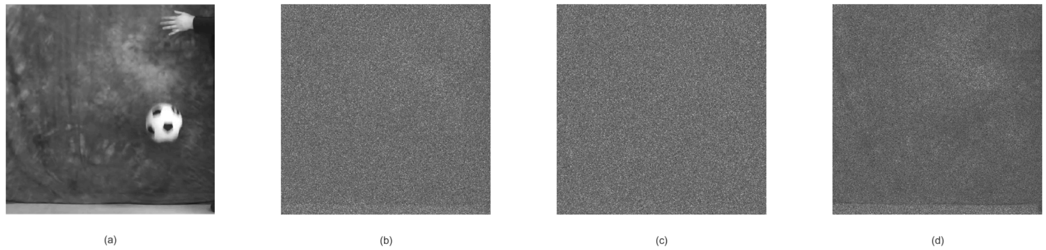

Figure 5: Reconstruction using the ℓ_1-norm optimization technique. (a)shows the original image. Figure (b) shows the reconstructed from image with Gaussian noise, with a PSNR of 15.97 dB compared to the original frame, Figure (c) shows the reconstruction when salt and pepper is added with PSNR of 15.51 dB, Figure (d) shows the reconstruction when speckle noise is added with a PSNR of 17.84

performance of Normalized Cross Correlation. The error is plotted with respect to the noise level SNR (in dB). The performance of BP degrades significantly for SNR less than 5.3 dB. While this approach shows a slight enhancement on the performance, the performance still deteriorates around 5.3 dB. Figure 6 displays measured performance of BP and compares it to NCC as a function of SNR.

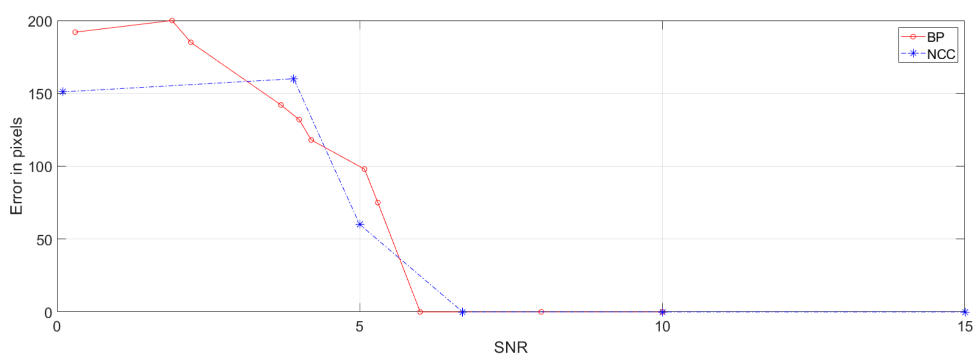

Figure 6: Comparison between BP and NCC (Normalized Cross Correlation)

3.3 OMP Using Sparse Representation

Since BP reconstruction did not show promising results in extreme conditions, the next step was to try one of the greedy reconstruction methods. In this section, we use OMP to reconstruct the frames. The following subsections will describe the data set changes required to enhance reconstruction and detection of the object.

3.3.1 Scene Based Dictionary

In this approach we applied OMP to reconstruct the noisy images. We used a similar set of images to train the noiseless dictionary. We apply this algorithm to the same noisy data and we compare the results in figure 7.

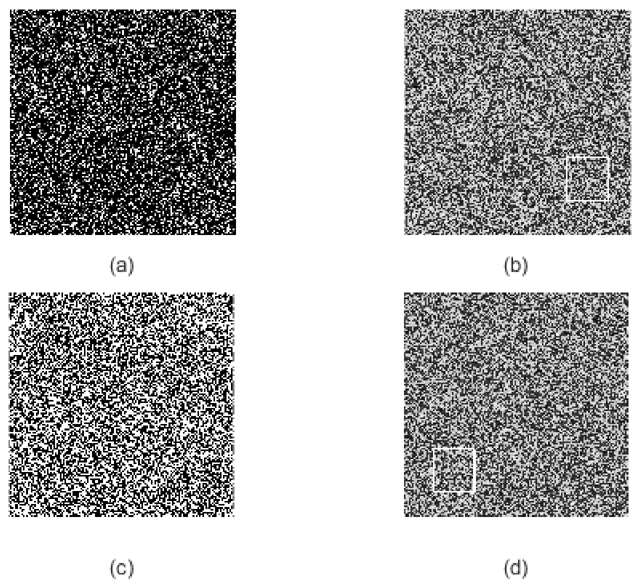

Figure 7: Comparison between distorted images and recovered scenes

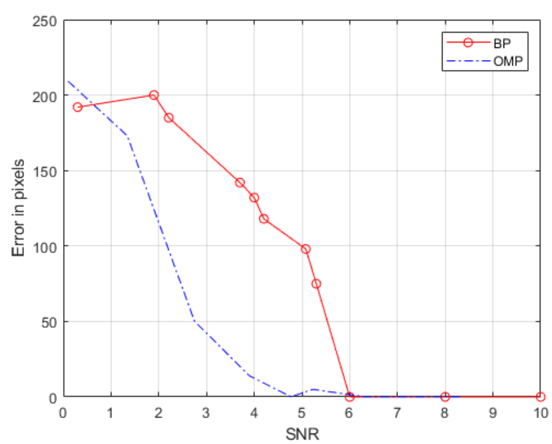

Figure 8: Performance comparison between BP and OMP using a dictionary based on the entire scene

As shown in figure 7, this approach attempts to recover the entire scene based on a prior knowledge. The detection algorithm was able to detect the target in some cases with a small error.

3.3.2 Object Based Dictionary

Given that the goal is object detection and tracking, background scene recovery is of reduced importance. Thus, the focus of the dictionary construction was switched from total scene reconstruction to identification/detection enhancements necessary to make only the target detectable. Therefore, the next step was to make the dictionary highly dependent on the object. An object based dictionary was constructed based on pristine shifted version of the target and then applied to the noisy images for target identification. The result of this change produced enhanced results, a better tracking accuracy in the results, and faster dictionary learning and reconstruction time.

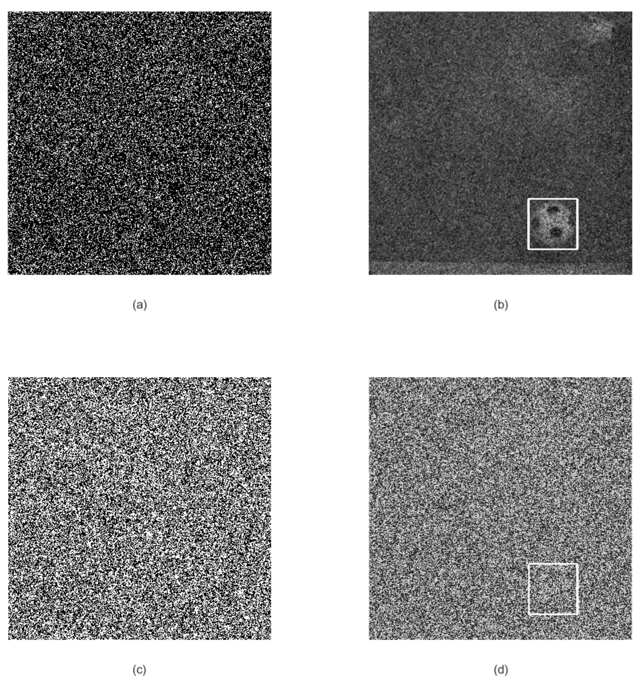

Figure 9: Sample from recovered images a) Source image with $S\&P$ noise, b) is the recovered image, c) is the noisy image with speckle noise and d) is the recovered object

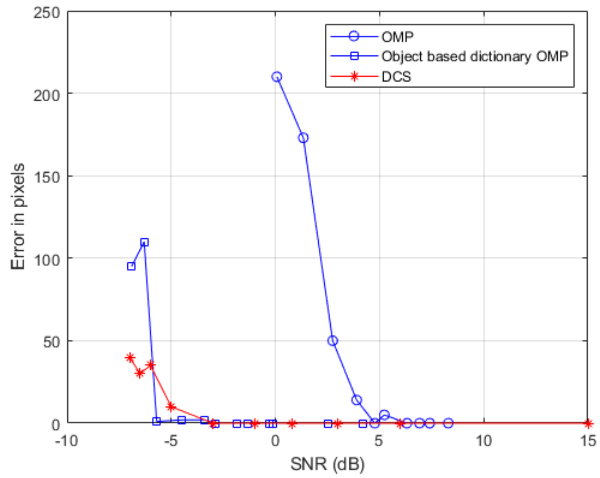

Figure 10: Comparing performance of OMP, OMP with an objsect based dictionary, and the DCS method implemented in [1]

There is a significant enhancement in performance between the two approaches implemented in this chapter, and we see that this algorithm is able to recover objects in SNR down to -5.8 dB.

This chapter summarized the methods implemented, and a comparison in performance between all approaches. First, we used a video of a bouncing tennis ball, which does not contain many features. The results were poor due to high dependencies in the object and the background. Therefore, a different object had to be picked in order to enhance the results. A soccer ball was the next choice since it had the black pentagon, which served as additional target detail. The added features enhanced the recovery as well as the detection immensely. We noticed a better object recovery when the dictionary is based solely on the targeted object. The dictionary training requires a set of parameters with a remarkable impact on both the computational time, and the performance of the process such as sparsity, the number of atoms, and patch size. Finding the appropriate set of parameters in order to find the best performing point is mandatory, in order to keep the process quick but also maintain a reliable object detection. These parameters depend on the kind of application in which Compressive Sensing is used. Here, we set the patch size to be 3×3 and the dictionary had 50 atoms in order to speed up the learning process.

CHAPTER 4

CONCLUSION

4.1 Conclusion

This thesis has primarily focused on the application of compressive sensing theory in object detection during degraded visual environments. The theory behind compressive sensing is was summarized and shown to require the following two conditions be met for successful implementation:

1. The measurement matrix ϕ needs to follow the restricted isometery property as close as possible. meaning that measurement matrix should be incoherent.

2. The reconstruction algorithm needs to be adapted to the application l_1 minimization, OMP and CoSaMP are discussed in this thesis. Only l_1 minimization and OMP where implemented. We change the parameters to enhance the results of each method.

We showed that l_1 minimization does not perform well under heavy noise, and that OMP provides better results. It was also shown that the recovery of the object can be enhanced by adding features to the object. For example, a tennis ball is harder to recover than a soccer ball under the same circumstances since it has less features. These added features will improve the detection and tracking processes as well. It was also shown that it is possible to detect objects in severely degraded visual environments or zero visibility using an off the shelf camera, rather than an infrared or LIDAR. These results will help cut the costs of object detection in many applications.

4.2 Future Work

This approach can be improved by allowing parallel computations, which will reduce the processing time significantly. Applying background subtraction maybe useful in focusing the entire CS process on the target object. Further enhancements can be done by testing other recovery techniques such as CoSaMP.

REFERENCES

[1] Lau, RC & T. K. Woodward (2015). See-through Obscurants via Compressive Sensing in Degraded Visual Environment. In Compressive Sensing IV, edited by Fauzia Ahmad, Proc. of SPIE Vol. 9484 94840F-1

[2] Candès, E.J. & Wakin, M. B. (2008). An introduction to compressive sampling. *IEEE Signal Process. Mag.*, 25(2), 21-30.

[3] Candès, E.J., Romberg, J., & Tao, T. (2006). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory*, 52(2), 489-509.

[4] Candès, E.J. & Tao, T. (2006). Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. Inf. Theory*, 52(12), 5406-5425.

[5] Donoho, D.L. (2006). Compressed sensing. *IEEE Trans. Inf. Theory*, 52(4), 1289-1306.

[6] Oppenheim, A.V & Schafer, R.W. (1989). Discrete-Time Signal Processing. Prentice Hall.

[7] D. G. Lowe (2004). Distinctive image features from scale-invariant keypoints. IJCV, 60(2):91110.

[8] Priemer, R. (1991). Introductory Signal Processing. World Scientific.

[9] Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. Journal of Basic Engineering. 82: 35.

[10] Sengupta, N, Sahidullah, Md, & Saha, G. (2016). Lung sound classification using cepstral-based statistical features. *Computers in Biology and Medicine*, 75(1), 118-129.

[11] Baraniuk, R. (2011). More is less: Signal processing and the data deluge. *Science*, 331(6018), 717-719.

[12] T. Strohmer and R. Heath, "Grassmanian frames with applications to coding and communication,"Appl. Comput. Harmon. Anal., vol. 14, no. 3, pp. 257-275, May 2003.

[13] N. Ailon and E. Liberty, "Fast dimension reduction using Rademacher series on dual BCH codes,"Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), San Francisco, California, USA, January 20-22, 2008.

[14] S. Howard, R. Calderbank, and S. Searle, "A fast reconstruction algorithm for deterministic compressive sensing using second order Reed-Muller codes,"Conference on Information Systems and Sciences (CISS), Princeton, NJ, Mar. 2008.

[15] Z. Xu, "Deterministic sampling of sparse trigonometric polynomials,"Journal of Complexity, vol. 27, pp. 133-140, 2011.

[16] N. Y. Yu, "Deterministic compressed sensing matrices from multiplicative character sequences,"45th Annual Conf. Information Sciences and Systems (CISS), Johns Hopkins University, Baltimore, MD, Mar. 2011.

[17] S. Li, F. Gao, G. Ge, and S. Zhang, "Deterministic construction of compressed sensing matrices via algebraic curves,"IEEE Trans. Inf. Theory, vol. 58, no. 8, pp. 5035-5041, Aug. 2012.

[18] H. Rauhut, "Compressive sensing and structured random matrices, Preprint. May 2010.

[19] R. Calderbank, S. Howard, and S. Jafarpour, "Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property."IEEE J. Selected Topics in Signal Processing, vol. 4, no. 2, pp. 358-374, Apr. 2010.

[20] L. Applebaum, S. D. Howard, S. Searle, and R. Calderbank, "Chirp sensing codes: deterministic compressed sensing measurements for fast recovery, Appl. Comput. Harmon. Anal. (ACHA), vol. 26, pp. 283-290, 2009.

[21] R. A. DeVore, "Deterministic constructions of compressed sensing matrices."J. of Complexity, 23, pp. 918 - 925, Aug. 2007.

[22] M. Fornasier and H. Rauhut, "Compressive sampling."Chapter in Part 2 of the Handbook of Mathematical Methods in Imaging (O. Scherzer Ed.), Springer, 2011.

[23] M. Davenport, M. Duarte, Y. Eldar, and G. Kutyniok, "Introduction to Compressed Sesning,"Chapter in Compressed Sensing: Theory and Applications, Cambridge University Press, 2012.

[24] K. Brandenburg, "MP3 and AAC explained,"in AES 17th International Conference on High-Quality Audio Coding, Sep. 1999.

[25] W. Pennebaker and J. Mitchell, "JPEG: Still image data compression standard, Van Nostrand Reinhold, 1993.

[26] D. S. Taubman and M. W. Marcellin, "JPEG 2000: Image Compression Fundamentals, Standards and Practice,"Kluwer, 2001.

[27] Moore, K., Khim, J., & Ross, E. (2018). Greedy Algorithms. Brilliant.org. Retrieved 09:49, September 7, 2018, from https://brilliant.org/wiki/greedy-algorithm/

[28] Papadimitriou, C.H. & Steiglitz, K. (1998). Combinatorial optimization: algorithms and complexity. Courier Corporation.

[29] Black, E.P. (2005). Greedy algorithm, in Dictionary of Algorithms and Data Structures [online], Vreda Pieterse and Paul E. Black, eds. 2 February 2005. (accessed 09/07/2018) Available from: https://www.nist.gov/dads/HTML/greedyalgo.html

[30] Wikepedia (2018b). Kalman Filter. `https://en.wikipedia.org/wiki/Kalman_filter` accessed 9/7/2018

[31] Wikepedia (2018c). Greedy algorithm. `https://en.wikipedia.org/wiki/Greedy_algorithm` accessed 9/7/2018

[32] Dalal, N. and Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. Proceeding CVPR '05 Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01, Pages 886-893

[33] Vocal Technology (2018). Histogram of Oriented Gradients (HOG) for Object Detection. `https://www.vocal.com/video/histogram-of-oriented-gradients-hog-for-object-detection/`

[34] J. A. Tropp and A. C. Gilbert., Signal recovery from random measurements via orthogonal matching pursuit, IEEE Trans. Inf. Theory, vol. 53, no. 12, pp. 4655-4666, 2007

[35] Wikepedia (2018a). Shannon Sampling Theorem. `https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem` accessed 9/7/2018

[36] Wikipedia (2018d). Power Law. `https://en.wikipedia.org/wiki/Power_law` accessed 10/11/2018

[37] Wikipedia (2018e). Histogram of Oriented Gradients. https://en.wikipedia.org/wiki/Histogram_of_oriented_gradients accessed 10/27/2018

[38] Clauset, Aaron (2009). "POWER-LAW DISTRIBUTIONS IN EMPIRICAL DATA". SIAM Review. 51: 661703.

[39] K. Ni, P. Mahanti, S. Datta, S. Roudenko, D. Cochran, Stability of efficient deterministic compressed sensing for images with chirps and Reed-Muller sequences, Proc. SPIE 7497, MIPPR 2009: Medical Imaging, Parallel Processing of Images, and Optimization Techniques, 74971S, Oct 30, 2009.

[40] G. Kutyniok, Compressed Sensing: Theory and Applications. Preprint, March 12, 2012.

[41] M. Rudelson and R. Vershynin, "On sparse reconstruction from fourier and gaussian measurements, Communications on Pure and Applied Mathematics, vol. 61, pp. 1025 - 1045, 2008.

[42] S. Mallat, A Wavelet Tour of Signal Processing: The Sparse Way, 3rd ed. Burlington, MA, USA: Academic, 2009.

[43] R. G. Baraniuk, "Compressive sensing [lecture notes]," IEEE Signal Process. Mag., vol. 24, no. 4, pp. 118121, Jul. 2007.

[44] J. Kang, H. Lee, K. Kim, Detection-directed sparse estimation using Bayesian hypothesis test and belief propagation. submitted on IEEE Trans. Signal Processing, Nov. 2012.

[45] D. Needell and J. A. Tropp, CoSaMP: Iterative signal recovery from incomplete and inaccurate samples, Appl. Comput. Harmon. Anal., vol. 26, pp. 301-321, 2009

[46] J. A. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk, "Random filters for compressive sampling and reconstruction," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., Toulouse, France, May 2006, p. 3.

[47] Li, Y. (2013). Empirical Recovery Performance of Fourier-based Deterministic Compressed Sensing. A thesis Presented to Lakehead University in Partial Fulfillment of the Requirement for the Degree of Master of Science in Electrical and Computer Engineering Lakehead University Thunder Bay, Ontario, Canada.

[48] Brad C. Yu, BC (2005). Tracking with Kalman Filter Based on Computer vision A modern Approach. The Australian National University. users.cecs.anu.edu.au/ bradyu/MyPapers/Tracking-BradYU.ppt Accessed 09/17/2018

[49] Rani, M., S. B. Dhok, & R. B. Deshmukh (2018). A Systematic Review of Compressive Sensing: Concepts, Implementations and Applications. IEEE Access, Volume 6, 4875 - 4894

[50] Y. Zhao, Y. H. Hu, and H. Wang, "Enhanced random equivalent sampling based on compressed sensing," IEEE Trans. Instrum. Meas., vol. 61, no. 3, pp. 579586, Mar. 2012.

[51] R. Tibshirani, "Regression shrinkage and selection via the lasso," J. Roy. Statist. Soc. B, Methodol., vol. 58, no. 1, pp. 267288, 1996.

[52] G. H. Golub and C. F. Van Loan, "Matrix analysis," in Matrix Computations, 4th ed. Baltimore, MD, USA: The Johns Hopkins Univ. Press, 2013, ch. 2, pp. 6873.

[53] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud, Theory and implementation of an analog-to-information converter us-

ing random demodulation, in Proc. IEEE Int. Symp. Circuits Syst., New Orleans, LA, USA, May 2007, pp. 19591962.

[54] I. Daubechies, M. Defrise, and C. De Mol, ”An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” Commun. Pure Appl. Math., vol. 57, no. 11, pp. 14131457, Nov. 2004.

[55] S. S. Chen, D. L. Donoho, and M. A. Saunders, (1999). Atomic decomposition by basis pursuit. SIAM J. Sci. Comput., vol. 20, no. 1, pp. 3361.