

DESIGN OF A COMPRESSED SENSING SYSTEM FOR HYPER SPECTRAL IMAGING

By

Gabriel Eduardo Ramirez

A thesis submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

UNIVERSITY PUERTO RICO

MAYAGUEZ CAMPUS

May, 2014

Approved by:

Vidya Manian, PhD
President, Graduate Committee

Date

Luis O. Jiménez Rodríguez, PhD
Member, Graduate Committee

Date

Lizdabel Morales Tirado, PhD
Member, Graduate Committee

Date

Edgardo Lorenzo Gonzalez, PhD
Representative of Graduate Studies

Date

Pedro I Rivera Vega, PhD
Chairperson of the Department

Date

Abstract of Thesis Presented to the Graduate School
of the University of Puerto Rico in Partial Fulfillment of the
Requirements for the Degree of Master of Science

**DESIGN OF A COMPRESSED SENSING SYSTEM
FOR HYPER SPECTRAL IMAGING**

By
Gabriel Eduardo Ramirez Ibañez
May 2014

Chair: Vidya Manian PhD

Major Department: Electrical and Computer Engineering

The advent of single pixel imaging brings the promise of lower sensor costs and higher efficiency by implementing spatial compression at the same time the image is sensed. This is achieved with the use compressed sensing (CS) principles and digital micro-mirror devices. Remote sensing systems with hyper spectral imaging capabilities could benefit greatly from sensor cost reduction and more computationally efficient compression techniques. However in order to analyze the image captured with a single pixel camera it must first be reconstructed, which is a computationally intensive process. Typical remote sensing applications like surveillance or target detection where a large number of images have to be analyzed, most of which will show to be of no great interest after the fact, are ill suited for these imaging systems as most of the time would be spent reconstructing images that will later prove to be of low significance.

In an effort to bring the benefits promised by single pixel imaging closer to these applications the present work develops the theory, design and implementation of CS on hyper spectral imaging. Progressive implementations of CS are performed on images, starting with spectral compression, followed by spatial compression and culminating with the proposal of a spatial CS implementation that allows recursive 2 stage reconstructions.

The proposed system can be implemented on single pixel cameras while reducing the amount of computing power and time required by plain CS implementations to perform image reconstruction. The introduction of a partially recovered image also allows for a preliminary analysis of the image, allowing applications to determine if the image needs any further analysis before engaging in full image reconstruction. The partially reconstructed image is an arrangement of the measurements from spatially compressed image sections. The final compression stage can tackle each section as a separate image to be reconstructed; this is achieved using a structured measurement matrix.

The following chapters provide tests and experiments that compare processing times, classification statistics and error rates that point towards the systems practicality, making it an interesting option for signals with high data volumes like those found in hyper spectral imaging.

Resumen de Tesis Presentado a Escuela Graduada
de la Universidad de Puerto Rico como requisito parcial de los
Requerimientos para el grado de Maestro en Ciencias
**DISEÑO DE UN SISTEMA DE COMPRESSED SENSING
PARA IMÁGENES HYPER ESPECTRALES**

Por
Gabriel Eduardo Ramirez Ibañez
Mayo 2014

Consejero: Vidya Manian PhD

Departamento: Ingeniería Eléctrica y de Computadoras

La implementación de cámaras de un pixel trae consigo la promesa de reducciones en los costos de los sensores y una mayor eficiencia gracias a que comprimen la imagen espacialmente al mismo tiempo que hacen el sensado de la misma. Esto se logra gracias a la aplicación de los principios de compressed sensing (CS) con el uso dispositivos de micro-espejos digitales. Sistemas de sensado remoto que capturan imágenes híper espectrales podrían beneficiarse enormemente de una reducción en los costos de los sensores y de técnicas de compresión más eficientes. Sin embargo el uso de un solo pixel hace necesaria la reconstrucción de la imagen capturada antes que esta pueda ser analizada, lo cual es un proceso computacionalmente intenso. Aplicaciones típicas de sensado remoto como vigilancia o detección en las cuales un alto número de imágenes tienen que ser analizadas, la mayoría de las cuales mostraran ser de poco interés luego del hecho, hacen poco práctico el uso de este tipo de sistemas debido a que los mayores esfuerzos se harían reconstruyendo imágenes que terminarían siendo de poco significado.

En un esfuerzo por acercar los beneficios prometidos por las cámaras de un pixel a dichas aplicaciones el presente trabajo escrito desarrolla la teoría, diseño e implementación de CS en imágenes híper espectrales. La implementación de CS en imágenes se hace de forma gradual, empezando con la compresión espectral, seguida de compresión espacial y culminando en la

propuesta de una forma de compresión espacial que permite reconstrucción recursiva de la señal en dos fases.

El sistema propuesto puede implementarse en cámaras de un pixel al mismo tiempo que se reduce el tiempo y poder computacional requerido para reconstruir las imágenes. La introducción de una imagen parcialmente reconstruida permite que se haga un análisis preliminar de la imagen, de manera que las aplicaciones puedan decidir si la imagen requiere un análisis más detallado antes de embarcar en el proceso de reconstrucción completa de la imagen. La reconstrucción parcial es un arreglo hecho con las mediciones tomadas de la compresión espacial de secciones de la imagen. La etapa de reconstrucción final puede procesar cada una de las secciones como una imagen independiente, esto se logra con una matriz de medición estructurada.

Los capítulos siguientes presentan la teoría, pruebas y resultados comparando los tiempos de procesamiento, estadísticas de clasificación y tasas de error que indican que tan práctico es el sistema, volviéndolo una opción interesante para señales con altos volúmenes de datos como los encontrados en imágenes híper espectrales.

TABLE OF CONTENTS

ABSTRACT ENGLISH.....	ii
RESUMEN ESPAÑOL.....	iv
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
ACRONYMS.....	ii
1. INTRODUCTION.....	3
1.1 Motivation.....	3
1.2 Outline.....	6
2. BACKGROUND AND LITERATURE REVIEW.....	7
2.1 Hyper spectral Imaging.....	7
2.2 Compressed Sensing.....	10
2.3 Signal Sparsity.....	13
2.4 Signal Decomposition and Sparse Representations.....	15
2.4.1 Fixed Complete Ortho-Normal Basis.....	17
2.4.2 Over-complete Basis.....	20
2.5 Large Scale Linear Programming.....	27
2.5.1 Duality in LP.....	28
2.6 CS Formulation.....	28
2.6.1 Signal Recovery.....	30
2.6.2 CS implementations.....	36
2.7 Classification.....	37
2.7.1 K-Means.....	38
3. OBJECTIVES.....	40

3.1	General Objective.....	40
3.2	Specific Objectives.....	40
4.	IMPLEMENTATION	41
4.1	1 - Dimensional CS	42
4.2	2 - Dimensional CS	49
4.3	Recursive 2 - Dimensional CS	52
4.3.1	Measurement matrix construction:	54
4.3.2	Complete Compression and Reconstruction:.....	56
4.3.3	Partial Compression	56
4.3.4	Final Compression Stage	58
4.3.5	Partial Reconstruction.....	59
4.3.6	Complete Reconstruction.....	60
5.	Tests and Results	62
5.1	Partial Reconstruction	64
5.2	Full Reconstruction	67
6.	CONCLUSIONS AND FUTURE WORK.....	69
7.	REFERENCES	72

LIST OF TABLES

Table 2-1. Examples of Hyper Spectral Sensors and their imaging capabilities.....	9
Table 4-1: k -means Indian Pines - Confusion matrix.....	48
Table 4-2: k -means Compressed Indian Pines - Confusion.	48
Table 4-3: k -means Reconstructed Indian Pines – Confusion.....	48
Table 5-1. Confusion Matrix description.....	63
Table 5-2. ROC axis description.....	63
Table 5-3. Confusion Matrix for IPR: S_16_T1_D_M1_140_T2_D_M2_90	65
Table 5-4. Confusion Matrix for IPR: S_16_T1_D_M1_140_T2_W_M2_80	65
Table 5-5. Confusion Matrix for IPR: S_16_T1_W_M1_160_T2_D_M2_80	65
Table 5-6. Confusion Matrix for IPR: S_16_T1_W_M1_160_T2_W_M2_80	65
Table 5-7. Confusion Matrix for IPR: S_32_T1_D_M1_59_T2_D_M2_456	66
Table 5-8. Confusion Matrix for IPR: S_32_T1_D_M1_59_T2_W_M2_434	66
Table 5-9. Confusion Matrix for IPR: S_32_T1_W_M1_60_T2_D_M2_437	66
Table 5-10. Confusion Matrix for IPR: S_32_T1_W_M1_60_T2_W_M2_426	66
Table 5-11. Confusion Matrix for IR: S_16_T1_D_M1_140_T2_D_M2_90	67
Table 5-12. Confusion Matrix for IR: S_16_T1_W_M1_160_T2_D_M2_80	67
Table 5-13. Confusion Matrix for IR: S_32_T1_D_M1_59_T2_D_M2_456	67
Table 5-14. Confusion Matrix for IR: S_32_T1_W_M1_60_T2_D_M2_437	67
Table 6-1: Summary of the results from the tests conducted.....	69

LIST OF FIGURES

Figure 1-1 Single Pixel Camera layout [2].	4
Figure 2-1 Types of spectral imaging and electromagnetic bands by wavelength (Near Infra-Red; Short, Medium and Long, Wave Infra-Red).	8
Figure 2-2 Hyper spectral data cube deconstruction.	8
Figure 2-3 Scanning methods [16].	9
Figure 2-5: K-means pseudo code [49].	39
Figure 4-1 Indian Pines HIS.	43
Figure 4-2 Compressed Indian Pines image.	45
Figure 4-3: Reconstructed Indian Pines.	47
Figure 4-4: Indian Pines spectrogram reconstruction.	47
Figure 4-5: <i>k</i> -means classification result for Indian Pines.	49
Figure 4-6: Salinas-A scene sample bands.	52
Figure 4-7 Salinas-A scene 2-D DCT .	52
Figure 4-8: Proposed CS method.	53
Figure 4-9: <i>k</i> -means Classification on IPC .	57
Figure 5-1: RGB satellite image of a boat at sea.	62
Figure 5-2. Ground Truth Images for IPR	65
Figure 5-3. Location on a ROC of IPR classification.	66
Figure 5-4. Location on a ROC of IR classification.	68
Figure 6-1: ROC chart comparison.	69

ACRONYMS

CS	Compressed Sensing
CI	Compressive Imaging
HSI	Hyper spectral Imaging
RS	Remote Sensing
MP	Matching Pursuit
BP	Basis Pursuit
DMD	Digital Micro-mirror Device
DCT	Discrete Cosine Transform
DWT	Discrete Wavelet Transform
FFT	Fast Fourier Transform
RIP	Restricted Isometry Property
ROC	Receiver Operating Characteristic
TPR	True Positive Rate
FPR	False Positive Rate
NIR	Near Infra-Red
SWIR	Short Wave Infra-Red
LWIR	Long Wave Infra-Red

1. INTRODUCTION

1.1 Motivation

Spectral imaging systems capture light in tens or hundreds of narrow contiguous spectral bands, constructing an image for each one. These bands provide additional information on the materials in the image, making them ideal for a wide range of classification applications in fields like agriculture, geology, biology, and surveillance among many others.

Signal processing algorithms analyze the data from spectral imaging systems in order to distil specific information tailored to an application, such as mapping out the materials in the image, or pointing out a target. While more data helps algorithms reach better conclusions by providing additional information, the increase in data also means they will have to work harder sifting through it and drawing the spectral and spatial relations in the data before presenting the desired information. Aside from the limitations imposed from the large data sets, another hurdle in the proliferation of spectral imaging is the high costs of the sensors, making sensors that can capture larger regions of the spectrum several times more expensive than regular imaging sensors with the same spatial resolution.

Imaging systems that only use a single pixel sensor (instead of a sensor array) while performing spatial compression of the image at the sensing stage have been developed with the use of digital mirror devices (DMD) and compressed sensing (CS) techniques [2]. These are referred to as single pixel cameras and Figure 1-1 illustrates the main blocks used in the implementation described in [2]. Single pixel imaging starts like conventional imaging systems by capturing light from the objects in the camera field. The light beams that reach the camera are focused on the DMD's by an optical array. The DMD's then switch on or off so they reflect or deflect the

desired pixels towards or away from another optical array that focuses the beams on the single pixel sensor. The result is a measurement that corresponds to the sum of the reflected pixels. The only additional computation required is that used to store and execute the DMD's switching order and timing.

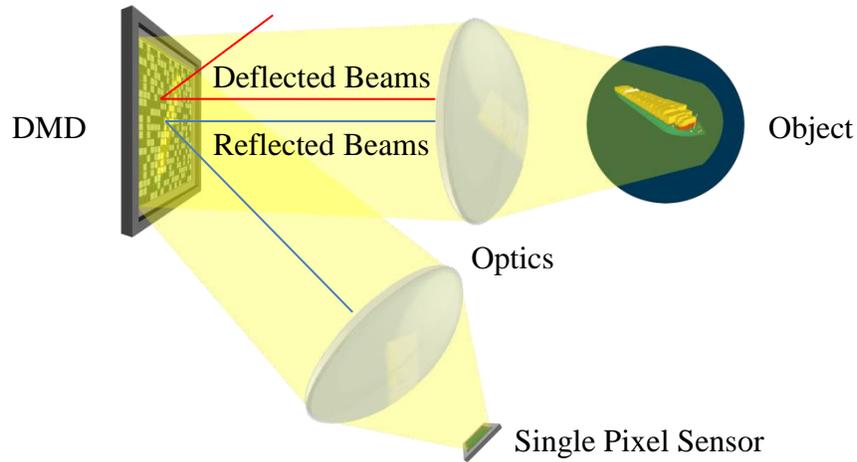


Figure 1-1 Single Pixel Camera layout [2].

The use of a single pixel sensor would reduce the cost in hyper spectral imaging applications, and the simultaneous sensing and compression of the signal would facilitate its storage and transmission in remote sensing applications where the sensor (i.e. an imaging satellite) has limited processing capabilities due to hardware and power restrictions.

CS theory is behind image reconstruction from a series of pixel measurements, making single pixel imaging possible. CS is a lossy compression method as the recovered image is an approximation of the original, the quality of the recovered image is affected by the number of measurements used for reconstruction and the selection of a basis on which to represent the image. The approximation is the result of an optimization problem that is solved with the help of linear programming algorithms. This process is computationally intensive, but has to be performed before any analysis on the image content can be made because the measurements do not present

any of the image's spatial information. The time needed to solve the optimization problem and reconstruct the image makes implementing CS in applications oriented towards image analysis such as classification or target detection less attractive, especially when the application requires the system to process a stream of images.

An example of an application ill-suited for single pixel imaging would be RS, HSI surveillance systems (i.e. a satellite scanning the oceans in search of boats) in which many images have to be analyzed, most of which are not of large interest (images of open sea without any boats). Therefore most of the extensive computation invested in image reconstruction would be wasted on images that are not of interest to the final user.

Another way to implement CS in said application and keep the pixels from mixing, thus preserving the spatial information is compressing the spectral signatures along the individual pixels. This approach would allow some analysis of the image content before having to invest in its reconstruction, while still allowing a more detailed analysis to be made by decompressing the spectral information. However it does not allow for the use of a single pixel sensor, and combining the spectrographs components would increase the cameras complexity.

A single pixel imaging scheme with a two stage reconstruction is presented in hopes of reducing the amount of processing needed before being able to analyze the image. The method introduces a partial reconstruction of the image that is less computationally intensive to reach than full reconstruction. This partial reconstruction should allow image classification to be performed in an effort to peer into the image content and determine if the image is of interest. If the partial analysis determines the image worthy of being analyzed in greater detail it should be possible to fully reconstruct the image from the partial reconstruction through additional processing.

1.2 Outline

In order to understand many of the concepts dealt with in the present document a brief introduction to the related subjects is given in the following chapter “Background and Literature Review”, including the areas of: Hyper spectral Imaging, Compressed Sensing and Classification. After the reader has been presented with the knowledge needed to understand the proposed two stage reconstruction scheme and its implications the objectives are numbered to clarify the goal of the thesis in the “Objectives” chapter. The reader is then presented with a chapter of “Implementation” where the stages required for the satisfactory completion of the objectives are numbered and described in detail. The “Results” chapter is divided into sections that mirror those in the previous chapter, displaying the outcome of the experiments proposed for each of the stages of execution. The final chapter “Conclusions and Future Work” is where the results are analyzed to produce a series of points relating the different variables and the system’s efficiency, as well as, humbly giving a direction for further development and improvement.

2. BACKGROUND AND LITERATURE REVIEW

This chapter presents fundamental concepts, and the review of previous work pertinent to this research. The concept of hyper spectral imagery is described, as well as compressed sensing, discrete cosine and wavelet transforms, and classification methods.

2.1 Hyper spectral Imaging

The human eye can capture and translate visible light into electrical signals; similarly spectral imaging systems register the information from a larger number of the electromagnetic spectrum's frequency bands and convert it to electrical signals [16]. These bands provide additional information on the objects in the sensor's field of view. Different materials reflect, absorb, transmit and emit electromagnetic radiation in distinct patterns according to their composition; these patterns can be registered by the sensor and later be used to identify them or discriminate among them. The sensors capability to capture more and narrower bands will affect the ability to distinguish the materials present in the image. Figure 2-1 illustrates the different spectral imaging types according to the number of bands captured; ranging from panchromatic that has a single but wide band, multispectral that has several to tens of bands, to hyper spectral containing hundreds of narrow bands.

Therefore spectroscopy imaging can be defined as the image acquisition of a scene or object, where each pixel in the image has a spectral radiance given by the amount of radiation (energy) arriving at the sensor at a given frequency or wavelength. The data captured is arranged in a cube whose face is the spatial coordinates (image captured at a certain frequency or band) and depth is the spectral information as shown in Figure 2-2.

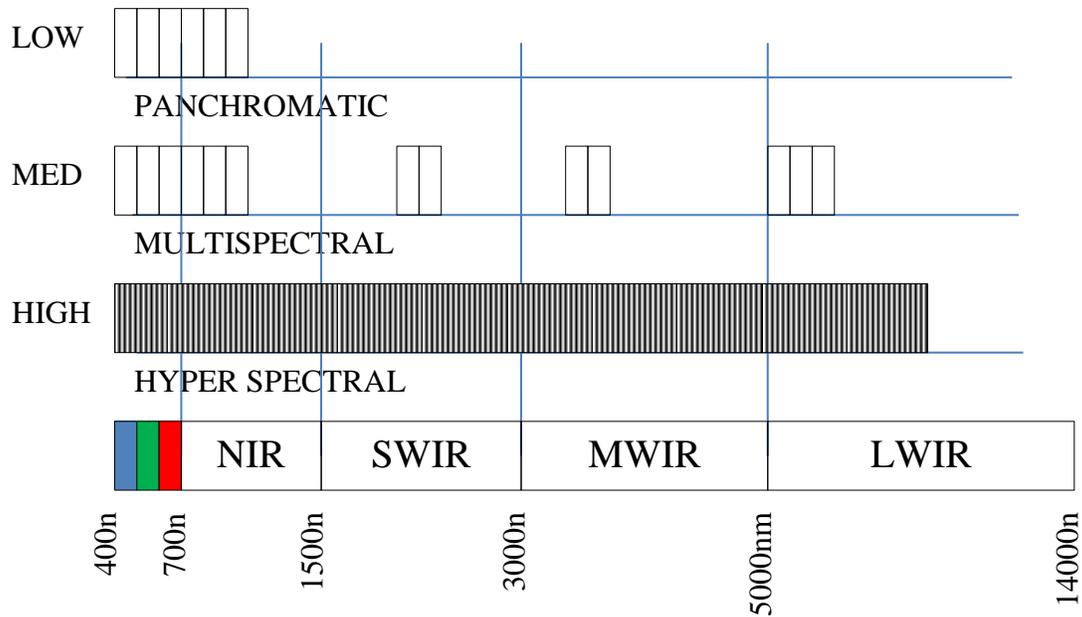


Figure 2-1 Types of spectral imaging and electromagnetic bands by wavelength (Near Infra-Red; Short, Medium and Long, Wave Infra-Red).

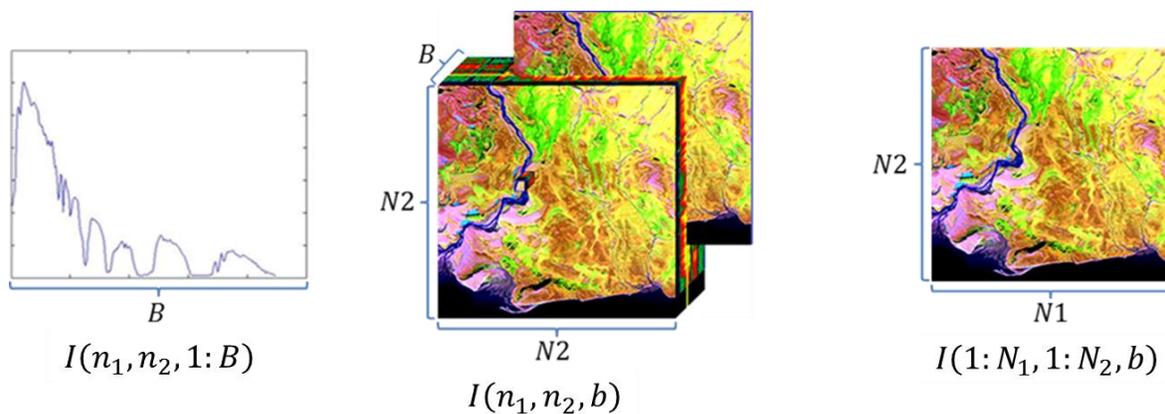


Figure 2-2 Hyper spectral data cube deconstruction.

Given the higher number of bands and therefore of data in hyper spectral images (HSI), it would be the type of spectral imaging that would benefit the most from an efficient compression scheme. For that reason it becomes the main target of this document. Among HSI applications, remote sensing (RS) has benefited from an increased level of attention in the last decades due to its potential to capture objects or properties on the earth's surface using sensors located on aircraft or satellites. The sensors elevation gives RS systems a larger field of view, characterizing

large expansions of land in a single image, making it ideal for applications in agriculture, mineralogy, surveillance and meteorology.

HSI has three main characteristics: the spatial resolution that determines the spatial size of the pixels in the image, the spectral resolution that is the wavelength width of the different frequency bands recorded by the sensor, and the radiometric resolution that could be described as the sensors sensitivity; some prominent examples of RS, HSI and their characteristics are listed in Table 2-1.

Hyper Spectral Sensor:	AVIRIS	HYDICE	HYPERION
Spectral Range (nm)	400-2500	400-2500	400-2500
Spectral Resolution (nm)	10	10	10
Spectral Samples (bands)	224	210	220
Spatial Resolution (m)	20	1-4	30
Radiometric Resolution (bits)	12	16	16

Table 2-1. Examples of Hyper Spectral Sensors and their imaging capabilities.

Most remote sensing spectral imaging systems build up the image cube by scanning through it spatially. There are four conventional methods to scan through the cube: line, whiskbroom, push-broom and filter, the first 3 are shown in Figure 2-3.

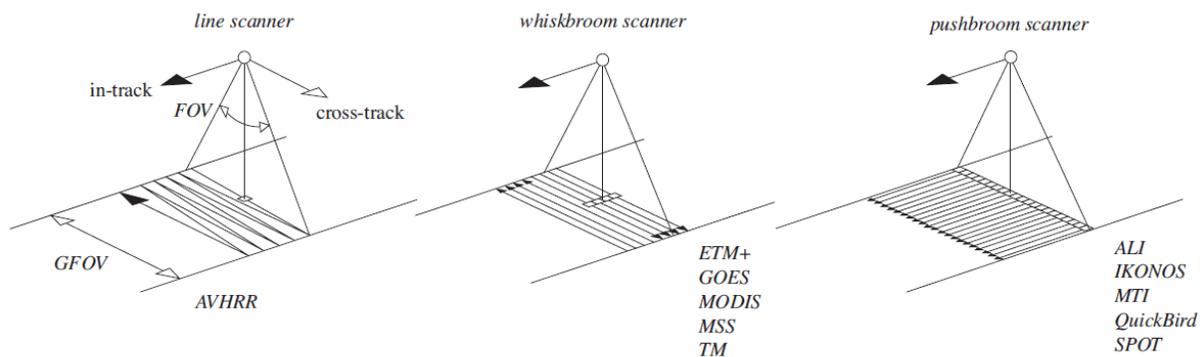


Figure 2-3 Scanning methods [16].

A single pixel camera like that shown in Figure 1-1 could capture the image as if a conventional imaging sensor array was being used, removing the need to scan the image mechanically.

2.2 Compressed Sensing

"It is vain to do with more what can be done with fewer"

William of Ockham

The Nyquist-Shannon sampling theorem establishes that in order to avoid frequency aliasing and guarantee correct reconstruction of a bandlimited signal the sampling frequency used must be at least twice that of the highest frequency component present in the signal [25]. In today's digital world it has become ubiquitous in signal acquisition protocols used in consumer audio and visual electronics, medical imaging devices, radio receivers, and many others. In the case of signals such as images that are not naturally bandlimited it plays an implicit role in the selection of an antialiasing low-pass filter to bandlimit the signal before sampling. However, no matter the source or signal type, if the signal is to be compressed, i.e. to facilitate storage, transmission or manipulation, large amounts of the data collected at the sensor will only be discarded at the compression stage. The main reason behind these implications is that sensing is traditionally done in a uniform manner, so a signal with size N would require N samples from one or more sensors (cases where the samples are temporally or spatially sensed respectively). These traditional sampling schemes have proven to be inefficient as they capture more data than information, and lead to signals with higher dimensionality (number of samples required).

Compressed Sensing or Sampling (CS) uses non-adaptive (non-signal dependent), efficient, sensing protocols designed to capture and condense the useful information in a sparse signal by correlating it with a measurement matrix A [10].

The output of a CS system is a series of measurements b_m or lineal combinations of the signals original N samples ($M \ll N$) obtained as:

$$b_m = A_m * x \quad \text{For:} \quad 0 < m \leq M \quad \text{Eq. 2-1}$$

The dimensions of A are: $M \times N$, where M is the number of columns in A or measurements, and N is the length of the columns and corresponds to the length of the signal. The Research in CS establishes that the minimum value of M is proportional to the: sparsity level of the signal, as well as to the coherence of the measurement matrix ($\mu(A) = \sqrt{N} \max_{i,j} |A_{i,j}|$). Recovering the exact signal from the M measurements is not a trivial matter as CS constructs an underdetermined set of equations that are solved by means of linear programming.

When reconstructing non-sparse signals from its measurements CS will only be able to recover an approximation of the signal. This approximation corresponds to the signals S largest coefficients when represented on a basis chosen before reconstruction. The value of S is proportional to the number of measurements taken for signal reconstruction, a larger number of measurements means that the approximation will include more coefficients and will therefore be closer to the original signal. CS is therefore a form of lossy compression, and the reconstruction error is related to the number of measurements, as well as the basis chosen to represent the signal. A basis that concentrates the signal in fewer coefficients will achieve a better approximation because the S highest coefficients will include more of the signal. An adequate basis is one that achieves a sparse representation of the signal, which is why it is referred to as the sparsifying basis.

By knowing which sparsifying basis will be used for reconstruction, and having some knowledge on the signal, it is possible to determine the minimum number of measurements needed to have the reconstruction fall within a vicinity from the original signal. This minimum measurement requirement places a roof on the level of compression that can be achieved. However, being able to estimate beforehand how close the reconstructed signal will be means applications like classification can anticipate the information loss, and determine (theoretically or by experimentation) if the available number of measurements would reach a reconstruction that allowed the analysis algorithms to reach the expected results.

The relation between the number of measurements, the recovered approximation and the sparsifying basis will be expanded on in Section 2.6, it is worth mentioning that CS literature has shown that a random binary measurement matrix with a Gaussian distribution is practical because it behaves similarly with any fixed sparsifying basis while also being signal independent.

In spite of the prospect of losing signal information and the computational requirements for signal reconstruction the use of measurements instead of samples (implementing CS) has a series of intrinsic advantages [46] that have lead researchers to continue to develop the subject. These advantages are:

- Signal independent compression that can be built into the sensor stage as demonstrated by the implementation of single pixel camera [2].
- Sampling complexity is reduced, making it easier to implement in remote systems with limited processing power.

- It is possible to estimate the minimum number M measurements required for the approximation to be within a desired range of the original signal, while allowing the reconstruction to improve with additional measurements further reducing the level of error.
- Measurements are “democratic” [46]; each one carries roughly the same amount of signal information so they are equally important or un-important (i.e. unlike compression schemes such as Principal Component Analysis (PCA) where losing one of the coefficients with higher information content will greatly impact the reconstructed signal).
- The reconstruction algorithms are flexible and independent of the acquisition process. Several sparsifying basis can be chosen to formulate the signal reconstruction equations, posing an optimization problem that can be solved with various linear programming algorithms.

2.3 Signal Sparsity

A sparse signal has very few values that are different from 0 this translates into the signal having a large amount of data (number of samples) with very few of them containing useful information (as most are 0). Sparse signals are very convenient in the field of digital signal processing as it reduces the number of operations that have to be realized when operating on the signal (sums and multiplications that involve samples with value 0 can be ignored or shortcut). Sparse signals also make information extraction and analysis easier because the relevant information is condensed in very few samples.

A great number of linear operations and methods can be easily implemented to remove the redundant data without losing the information contained in the signal, thus sparsity allows for

signals to be compressed and in general the sparser the signal the more it can be compressed (and later fully recovered satisfactorily). To illustrate this concept we present a trivial form of compression that represents the signal with only the relevant values and its positions (sample number). If a signal is sparse this can be done with very few values as most are known to be 0 or negligible by its very definition; furthermore as the signal's degree of sparsity increases the amount of data required for a good representation is reduced. However this type of compression is signal dependent, deeming it impractical for many applications.

A natural mathematical measurement of a signal's level of sparsity is its p norm (ℓ_p) for small values of p as defined in [6] and [44]:

$$\|\alpha\|_p \equiv \left(\sum_i |\alpha_i|^p \right)^{\frac{1}{p}} \quad \begin{array}{l} \text{For:} \\ 0 < \|\alpha\|_p \leq R \\ 0 < p < 2 \end{array} \quad \text{Eq. 2-2}$$

In CS literature, a signal's level of sparsity is normally measured using the ℓ_0 "norm" as defined by Donoho, it is a pseudo norm and is referred to in some literature in quotation marks as it does not satisfy the more severe requirements of a norm¹, here however we will shed their use. It is shown in the following equation as described in [6]:

$$\|f\|_{\ell_0} = \sum_{n=1}^N |f_n|_{\ell_0} \quad \begin{array}{l} \text{Where:} \\ |f_n|_{\ell_0} = \begin{cases} 0, & f_n = 0 \\ 1, & f_n \neq 0 \end{cases} \end{array} \quad \text{Eq. 2-3}$$

Real life signals, like those taken by a microphone, camera or wireless communication system are not sparse in the high dimension sampling base, but if they possess some low dimensional

¹ The strict mathematical definition of the ℓ_0 norm was established by Banach's Theory of Linear Operations.

structure, that is, they lie on or near low-dimensional subspaces, sub-manifolds, or stratifications they can be represented or decomposed sparsely in some other basis [35]. This phenomenon is a generalization of the uncertainty principle which originally relates signals sparsity in the time and frequency domains, establishing that the same signal can not be band-limited in both time and frequency domains. A textbook example of this is the ideal Dirac delta or Spike; it is the sparsest signal imaginable in the time domain but spans evenly along the entire frequency domain.

The field of digital signal processing has benefited greatly of the fact that whole categories of signals of great interest in engineering applications have sparse representations on fixed basis that are easily reached by means of linear transforms (i.e. Fourier or Wavelet transforms) and whose algorithms have reached a point of high efficiency and have been adapted to better serve specific applications and signal types [52].

2.4 Signal Decomposition and Sparse Representations

A signal can be decomposed or expressed as a linear combination of expansion functions or waveforms ψ_k . This can be expressed mathematically in the following form:

$$f(n) = \sum_{k=1}^K \alpha_k * \psi_k(n) \quad \text{Eq. 2-4}$$

Decomposing a signal gives us a different view of the signal, revealing certain characteristics that lay hidden beneath all the original data. The coefficients $\alpha_k(n)$ assigned to each of the waveforms indicate how dominant that waveforms structure is in the signal, a basis that is well paired with the signal's inherent characteristics will require a small or sparse set of coefficients $\alpha_i(n)$, thus facilitating its analysis.

A signal will rarely have a representation where most coefficients are exactly 0, which means the definition for sparse representation has to be broadened to include signals whose coefficients decay rapidly after being sorted by magnitude. This means the smallest ones can be discarded using only the largest to reconstruct an approximation of the original signal without generating large levels of error.

In order to express a signal as in Eq. 2-4 the sequence of waveforms $\{\psi_k\}$ must form a frame of the vector space of V that also contains them ($\psi_k \in V$). This guarantees that any arbitrary element \mathbf{v} of the vector space ($\mathbf{v} \in V$) can be reconstructed as:

$$\mathbf{v} = \sum_{k=1}^K c_k \psi_k \quad \text{Eq. 2-5}$$

In mathematics a frame is a set of vectors $\{\psi_k\}$ that span V (if it does not span V there will be at least one \mathbf{v} that is orthogonal to every $\{\psi_k\}$), however for it to be considered a frame $\{\psi_k\}$ must satisfy for any \mathbf{v} of V the properly called “frame condition” which was established in 1952 by Duffin Schaeffer and is described as follows:

$$A \|\mathbf{v}\|^2 \leq \sum_{k=1}^K |\langle \mathbf{v} | \psi_k \rangle|^2 \leq B \|\mathbf{v}\|^2 \quad \begin{array}{l} \text{Such that:} \\ A, B \in \mathbb{R} \\ 0 < A \leq B < \infty \end{array} \quad \text{Eq. 2-6}$$

If the elements in the frame are also linearly independent it is considered a base, which is a specific type of frame that guarantees a unique set of coefficients c_k for each \mathbf{v} . The trivial basis matrix for a N dimensional signal is the identity matrix I , this is often referred to as the standard basis, natural basis, canonical basis, or sampling basis. This last one is due to the fact that it can be used to represent the process of signal sampling by using translated Dirac functions as waveforms to indicate the signals value when sampled (i.e. sampling in time) as follows:

For:

$$1 \leq k \leq N$$

Where:

$$\alpha_k = \langle f, \psi_k \rangle$$

$$\psi_k(n) = \delta(n - k)$$

$$f(n) = \sum_{k=1}^K \alpha_k * \psi_k(n) \quad (a)$$
$$f(n) = \sum_{k=1}^K f(n) * \delta(n - k) \quad (b)$$
$$f_I = \Psi * \alpha = I * f \quad (c)$$

Eq. 2-7

2.4.1 Fixed Complete Ortho-Normal Basis

Different types of data have come to be associated to bases that have proven to be effective at information extraction of that type of data (achieving a sparse representation). In this document we are concerned with imaging, which is normally associated to discrete cosine transform (DCT) and discrete wavelet transforms (DWT). The sparse representations will be used to reconstruct the images as will be shown later when describing the CS implementation.

Discrete Cosine Transform

The Discrete Cosine Transform (DCT), was introduced by Ahmed, Natarajan and Rao in 1974 [29] and since then it has become one of the most known and used transforms in digital signal processing of image data, particularly image compression [30]. The DCT is a Fourier related transform because like the Discrete Fourier Transform (DFT) it uses sinusoids oscillating at different frequencies as waveforms, but unlike the DFT which uses both cosine and sine functions (as complex exponentials) to approximate the signal, the DCT uses only cosine functions (real component of the complex exponentials).

One of the main uses of the DCT is image compression and coding as it has a strong “energy compaction” property that locates the larger coefficients at the low frequency components of the basis [53]. The two dimensional DCT measures the correlation of the neighboring pixels in both

horizontal and vertical direction, unlike the one dimensional DCT. This makes the two dimensional DCT separable into one dimensional DCTs by first transforming the rows (columns) of the data matrix and then transforming the columns (rows) of the semi-transformed matrix [33].

Discrete Wavelet Transform

Fourier analysis of a signal gives us a signal's frequency content, describing stationary or periodical signals efficiently. However because it decomposes the signal into long (ideally infinite) cosine signals, when the signal has brief variations in time it is not the best fit as these signal characteristics become invisible. It is impossible to tell when an event took place by just looking at the Fourier transform of a signal [35]. Fourier analysis is therefore ill suited for signals with abrupt changes such as fingerprint images. The Short Time Fourier Transform (STFT) solves this inconvenience through the use of a smaller Fourier transform that slides along the signal (technique known as windowing). The STFT has a fixed window size, which means its resolution in time and frequency is fixed. The next logical step is to implement a windowing technique with variable window sizes. This makes wavelet analysis capable of showing trends, breakdown points, discontinuities in higher derivatives and self-similarity within the signal [35].

Because the discrete wavelet transform (DWT) is a linear transformation it can be just as easily described as a transformation matrix - signal vector product, and to generate the transformation matrix we use the method proposed by the authors of [26] where they describe a DWT that can be constructed from smaller filter matrices described in Eq 2-8 to 2-10.

$$\begin{aligned} \begin{bmatrix} y_h^{(k)} \\ y_g^{(k)} \end{bmatrix} &= \begin{bmatrix} \mathbf{H}^{(k)} \\ \mathbf{G}^{(k)} \end{bmatrix} y_h^{(k-1)} & \quad (a) \\ y_h^{(k-1)} &= \mathbf{H}^{(k-1)} y_h^{(k-2)} & \quad (b) \end{aligned}$$

Eq. 2-8

$$y = \begin{bmatrix} y_{hh}^{(k)} \\ y_{hg}^{(k)} \\ y_{gh}^{(k)} \\ y_{gg}^{(k)} \\ y_{hg}^{(k-1)} \\ y_{gh}^{(k-1)} \\ \vdots \\ y_{gh}^{(1)} \\ y_{gg}^{(1)} \end{bmatrix} = \mathbf{W}x = \begin{bmatrix} \left(\mathbf{H}_x^{(k)} \mathbf{H}_x^{(k-1)} \mathbf{H}_x^{(k-2)} \dots \mathbf{H}_x^{(1)} \right) \otimes \left(\mathbf{H}_y^{(k)} \mathbf{H}_y^{(k-1)} \mathbf{H}_y^{(k-2)} \dots \mathbf{H}_y^{(1)} \right) \\ \left(\mathbf{H}_x^{(k)} \mathbf{H}_x^{(k-1)} \mathbf{H}_x^{(k-2)} \dots \mathbf{H}_x^{(1)} \right) \otimes \left(\mathbf{G}_y^{(k)} \mathbf{H}_y^{(k-1)} \mathbf{H}_y^{(k-2)} \dots \mathbf{H}_y^{(1)} \right) \\ \left(\mathbf{G}_x^{(k)} \mathbf{H}_x^{(k-1)} \mathbf{H}_x^{(k-2)} \dots \mathbf{H}_x^{(1)} \right) \otimes \left(\mathbf{H}_y^{(k)} \mathbf{H}_y^{(k-1)} \mathbf{H}_y^{(k-2)} \dots \mathbf{H}_y^{(1)} \right) \\ \left(\mathbf{G}_x^{(k)} \mathbf{H}_x^{(k-1)} \mathbf{H}_x^{(k-2)} \dots \mathbf{H}_x^{(1)} \right) \otimes \left(\mathbf{G}_y^{(k)} \mathbf{H}_y^{(k-1)} \mathbf{H}_y^{(k-2)} \dots \mathbf{H}_y^{(1)} \right) \\ \left(\mathbf{H}_x^{(k-1)} \mathbf{H}_x^{(k-2)} \dots \mathbf{H}_x^{(1)} \right) \otimes \left(\mathbf{G}_y^{(k-1)} \mathbf{H}_y^{(k-2)} \dots \mathbf{H}_y^{(1)} \right) \\ \left(\mathbf{G}_x^{(k-1)} \mathbf{H}_x^{(k-2)} \dots \mathbf{H}_x^{(1)} \right) \otimes \left(\mathbf{H}_y^{(k-1)} \mathbf{H}_y^{(k-2)} \dots \mathbf{H}_y^{(1)} \right) \\ \vdots \\ \left(\mathbf{G}_x^{(1)} \right) \otimes \left(\mathbf{H}_y^{(1)} \right) \\ \left(\mathbf{G}_x^{(1)} \right) \otimes \left(\mathbf{G}_y^{(1)} \right) \end{bmatrix} x$$

Eq. 2-9

$$\begin{aligned} \mathbf{H}^{(k)} &= \begin{bmatrix} h(-1) & 0 & 0 & 0 & \dots & h(-3) & h(-2) \\ h(-3) & h(-2) & h(-1) & 0 & \dots & h(-5) & h(-4) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & h(-1) & 0 \end{bmatrix} & \quad (a) \\ \mathbf{G}^{(k)} &= \begin{bmatrix} g(-1) & 0 & 0 & 0 & \dots & g(-3) & g(-2) \\ g(-3) & g(-2) & g(-1) & 0 & \dots & g(-5) & g(-4) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & g(-1) & 0 \end{bmatrix} & \quad (b) \end{aligned}$$

Eq. 2-10

Even though the authors present even more efficient ways of performing the DWT by arranging the individual operations in a recursive fashion, we will use this initial recursion described above as our sole interest is the DWT transformation matrix. As was the case with the DCT before, the possibility of representing the DWT as a matrix vector operation makes its application to regular CS reconstruction algorithms straight forward.

2.4.2 Over-complete Basis

A spike is a signal that is maximally sparse on the sampling base, but when converted a Fourier basis its coefficients are spread over the basis. A sampled cosine signal has very few values that are zero or negligible, but when translated to a Fourier basis it becomes very sparse. These are examples of signals that have fairly straight forward sparse representations (the spike on the sampling base and the cosine on a Fourier base); however there are some signals that present greater a challenge when searching for a way to represent them sparsely. An illustrative example is a signal that is the result of adding a cosine wave and a spike or delta as shown in **Error!**

Reference source not found..

$f_1(n)$: Spike	$f_2(n)$: Cosine	$f(n) = f_1(n) + f_2(n)$	Eq. 2-11
$f_1(n) = \delta(n_0 - n)$	$f_2(n) = \sum_{i=1}^N \cos(2\pi fi) \delta(i - n)$		
$f(n) = \sum_{i=1}^N \cos(2\pi fi) \delta(i - n) + \delta(n_0 - n)$			

Because $f(n)$ is the sum of $f_1(n)$ and $f_2(n)$, for $f(n)$ to be sparse both $f_1(n)$ and $f_2(n)$ have to be sparse on whichever basis is used as reference. Analyzing the components that make up $f(n)$ we see that they are at odds when searching for a sparse representation. As mentioned before on the sampling base, the spike is maximally sparse while the cosine is not sparse at all; moving the signals to a Fourier base makes the cosine signal sparse, but eliminates all sparsity from the spike portion of the signal. Therefore neither basis makes it possible to represent $f(n)$ sparsely.

The sparse representation of each component on a different basis can be exploited by rewriting $f(n)$ as in Eq. 2-12 (a). The spike component $f_1(n)$ is sparse in the sampling base ψ_S , therefore

most of the coefficients α_{1S} will be 0. Most coefficients α_{2T} from the representation of $f_2(n)$ on the Fourier basis ψ_T are 0 as well. Therefore most coefficients α from Eq. 2-12 (b) will be 0, its level of sparsity $S = \|\alpha\|_{\ell_0}$ is just the sum of that from α_{1S} and α_{2T} , or $S = S_1 + S_2$. The combination of the coefficients also implies the combination of the basis ψ_1 and ψ_2 into a greater basis ψ that allows $f(n)$ to be sparse, and it is referred to as an over-complete basis.

$$f(n) = \sum_{i=1}^N \alpha_{1S_i} * \psi_{S_i}(n) + \sum_{j=1}^N \alpha_{2T_j} * \psi_{T_j}(n) \quad (a)$$

$$f(n) = \sum_{k=1}^{2N} \alpha_k * \psi_k(n) \quad (b)$$

Where:

$$\alpha_k = \begin{cases} \alpha_{1S_k}, & 1 \leq k \leq N \\ \alpha_{2T_{k-N}}, & N + 1 \leq k \leq 2N \end{cases}$$

$$\psi_k = \begin{cases} \psi_{S_k}, & 1 \leq k \leq N \\ \psi_{T_{k-N}}, & N + 1 \leq k \leq 2N \end{cases}$$

Eq. 2-12

Over-complete basis incorporate a number of waves greater than the size of the original signal (the signal is size N , but the basis is size $2N$). Increasing the diversity of waves to include groups with different characteristics will increase the probability of efficiently representing a signal, thus generating a highly sparse approximation. Therefore over-complete basis trade complexity at the basis for a possibly simpler (sparser) representation of the signal; making it easier to analyze a broader range of signals when extracting information, however this presents a new problem.

To illustrate the issue that arises we return once more to $f(n)$ from Eq. 2-121 as an example, it is possible to approximate $f(n)$ in the same basis ψ in any of the 3 forms shown in Eq. 2-123.

$\alpha_k = \begin{cases} \alpha_{1S_k}, & 1 \leq k \leq N \\ \alpha_{2T_{k-N}}, & N + 1 \leq k \leq 2N \end{cases} \quad S = S_1 + S_2 \quad (a)$	Eq. 2-13
$\alpha_k = \begin{cases} \alpha_{1S_k} + \alpha_{2S_k}, & 1 \leq k \leq N \\ 0, & 1 \leq k \leq 2N \end{cases} \quad S = N \quad (b)$	
$\alpha_k = \begin{cases} 0, & 1 \leq k \leq N \\ \alpha_{1S_k} + \alpha_{2S_k}, & 1 \leq k \leq 2N \end{cases} \quad S = N \quad (c)$	

The approximations in Eq. 2-13 (b) and (c) are simply those from representing both the components $f_1(n)$ and $f_2(n)$ in only one of the individual basis that compose ψ , ψ_S or ψ_T respectively. Because all 3 approximations are valid, it is up to the user to deem one of them best according to the objective (exactitude, sparsity). In this case we wish to reach a higher level of sparsity, so the clear choice is Eq. 2-13 (a).

In cases where none of the basis' components (waves) line up closely to the signal, one of many possible weighed combinations (the larger the basis the larger the number of possible combinations) must be chosen to maximize sparsity. Each possible combination has a level of sparsity (how many coefficients are required), as well as a level of error (how good the approximation is), making the search for the sparsest combination possible a topic for researchers that propose intricate methods of achieving said objective.

In summary, complete orthonormal basis will always generate a single set of coefficients for the signals representation on that basis, making the signals levels of sparsity dependent on the selection of a basis. Using a over-complete basis will not generate a unique representation of the signal, but might achieve higher levels of sparsity, however finding the sparsest group of coefficients that best fits the signal among the possible combinations belongs to a class of “computationally intractable problems, the set of NP-hard problems” [39]. The fact that optimal

solutions would become increasingly difficult as the signal size and dictionary size get larger researchers have worked on sub-optimal algorithms that achieve a good enough approximation. The theory behind these algorithms has spawned several of the concepts being applied to CS, and it is why the subject of frames, over-complete bases or over complete dictionaries is touched upon as an introduction to the algorithms: Matching Pursuit and Basis Pursuit. The literature on these algorithms refers to the frames as over-complete or redundant dictionaries composed of atoms, so we will do so as well during the following sections, keeping in mind that it is just a pseudonym for a frame composed of orthogonal waves.

MatchingPursuit

The Matching Pursuit (MP) algorithm was introduced by Mallat and Zhang [24] in 1993 as an algorithm that “decomposes any signal into a linear expansion of wave-forms (ϕ_γ) that belong to a redundant dictionary (\mathfrak{D}) of functions” by providing an “interpretation of the signal structures. If a structure does not correlate well with any particular dictionary element, it is sub-composed into several elements and its information is diluted” [24]. This is achieved by means of a greedy approach that chooses at each iteration a waveform g_{γ_0} that is best adapted to approximate the signal, or the residual of the previous stage Rf as shown in the following equivalence:

$$f = \langle f, \psi_{\gamma_0} \rangle \psi_{\gamma_0} + Rf \quad \text{Eq. 2-14}$$

Where:

$\bar{\psi}(t)$: complex conjugate of $\psi(t)$

$$\alpha = \langle f, \psi \rangle = \int_{-\infty}^{\infty} f(t) \bar{\psi}(t) dt$$

Eq. 2-15

Since ψ_{γ_0} is orthogonal to Rf , then:

$$\|f\|^2 = |f, \psi_{\gamma_0}|^2 + \|Rf\|^2 \quad \text{Eq. 2-16}$$

To minimize $\|Rf\|$ the ψ_γ that maximizes $|f, \psi_\gamma|$ must be found, to do so MP implements a optimality factor α ($0 < \alpha \leq 1$) that chooses the ψ_γ that is best or at the very least “almost the best” [24]:

$$|\langle f, \psi_{\gamma_0} \rangle| \geq \alpha \cdot \sup_{\gamma \in \Gamma} |\langle f, \psi_\gamma \rangle| \quad \text{Eq. 2-17}$$

The residual Rf is later sub-decomposed by projecting it on another vector of \mathfrak{D} that matches it as closely as possible as it was done for f .

The following is a step by step explanation of the n^{th} element of the decomposition:

$R^0 f = f$	(a)	Eq. 2-18
$ \langle R^n f, \psi_{\gamma_n} \rangle \geq \alpha \cdot \sup_{\gamma \in \Gamma} \langle R^n f, \psi_\gamma \rangle $	(b)	

The residual $R^n f$ is sub-composed into:

$R^n f = \langle R^n f, \psi_{\gamma_n} \rangle \psi_{\gamma_n} + R^{n+1} f$	(a)	Eq. 2-19
$\ R^n f\ ^2 = \langle R^n f, \psi_{\gamma_n} \rangle ^2 + \ R^{n+1} f\ ^2$	(b)	

The order m decomposition of f as a concatenated sum is:

$$\begin{aligned}
 f &= \sum_{n=0}^{m-1} (R^n f - R^{n+1} f) + R^m f = \sum_{n=0}^{m-1} \langle R^n f, \psi_{\gamma_n} \rangle \psi_{\gamma_n} + R^m f & (a) \\
 \|f\|^2 &= \sum_{n=0}^{m-1} (\|R^n f\|^2 - \|R^{n+1} f\|^2) + \|R^m f\|^2 & (b) \\
 \|f\|^2 &= \sum_{n=0}^{m-1} |\langle R^n f, \psi_{\gamma_n} \rangle|^2 + \|R^m f\|^2 & (c)
 \end{aligned}$$

Eq. 2-20

This last equation shows that although the matching pursuit decomposition is non-linear there is energy conservation like in orthonormal complete basis.

Published research has shown that the MP method does not deconstruct certain signals optimally due to the fact that if a mistake is made at the beginning of the procedure the greedy approach it implements will not correct it but try to fit the reconstruction using the remaining waves in the dictionary, making it miss when seeking sparsity or super resolution [27] and [40].

These observations on the MP's performance have been corrected with the development of the Orthogonal Matching Pursuit (OMP) by adding an additional step that solves:

$$\min_{\alpha_i} \left\| f - \sum_{i=1}^m \alpha_i \psi_{\gamma_i} \right\|_2 \quad \text{Eq. 2-21}$$

For all the m terms that have been entered up to that stage (m) to later form the residual [38] and [39].

$$\bar{R}[m] = f - \sum_{i=1}^m \alpha_i^{(m)} \psi_{\gamma_i} \quad \text{Eq. 2-22}$$

This additional step solves the issues related to MP but increases the computation time dramatically in cases that should be fairly straightforward [27].

Basis Pursuit

Basis Pursuit was introduced in 2001 by Shaobing Donoho and Saunders in [27], where they go into detail explaining the virtues and shortcomings of the different implementations for sparse signal de-composition on over complete dictionaries. They show that by implementing linear programming (LP) to the decomposition problem they can achieve higher levels of sparsity and super-resolution. This is achieved by expressing the problem of finding the sparsest coefficients as a minimization problem, where the level of sparsity is represented by the ℓ_1 norm of the signal and becomes the objective function of the optimization problem. The signal equivalence is used to generate the constraints in the formulation. The initial mathematical model is shown side by side with linear programming's standard form [41][40]:

Objective Function	Equality Constraints	Source
$\min c^T x$	$b = Ax$	Linear Programming Standard Form
$\min \ \alpha\ _{\ell_1}$	$f = \Psi\alpha$	Basis Pursuit Equations

The search for the coefficients generates a non-quadratic, convex set of equations, that require a higher effort to solve than the other decomposition methods. They implement a primal dual log barrier LP algorithm taking advantage of the advances in the field of linear programming, including the development of interior point methods that allow for a quick and efficient solution.

2.5 Large Scale Linear Programming

In a LP problem the collection of feasible points that satisfy the constraint function $x: Ax = b, x \geq 0$ delimit a convex polyhedron on R^N with a number of vertices equal to the size of the vector b (this might be different from N if the matrix A is not square). This polyhedron is also referred to as a simplex and its vertex or extreme points are defined by each of the constraints, and one of them minimizes $c^T x$. The name of the shape in R^N lends its name to the most tried and true LP algorithm: the simplex algorithm, which bounces between the vertices looking for an improvement on the objective function until it cannot further optimize (minimize) the objective function. The simplex method can be visualized as sorting through the vertices on the surface of a multidimensional object until it reaches its goal while implementing anti cyclic rules to avoid infinite loops.

However, upon visualizing the process, one might imagine that a reduction in the number of steps taken might be achieved if a path is cut through the inside of the object as a form of shortcut to re-emerge at the surface on the goal vertex. These methods are known as interior point methods, and have been of great interest for researchers since Karmarkar presented his algorithm. Even though increased efficiencies over simplex algorithms were never scientifically proven, it was widely considered at the time to provide a lower polynomial time complexity boundary than the traditional simplex methods [43]. The greater impact of Karmarkar's development however was the revitalization of research on interior point methods and algorithms, which on some case by case basis have proven faster than their simplex counterparts, especially with large problem sizes [43].

2.5.1 Duality in LP

Duality allows for the reformulation of a standard form LP problem into an equivalent representation which achieves higher computational efficiencies.

If we express a problem \mathcal{P} :

	Subject to:	
$\mathcal{P}: \max c^T x$	$A^T x \leq b$	Eq. 2-23
	$0 \leq x$	

It has an equivalent representation as \mathcal{D} :

	Subject to:	
$\mathcal{D}: \min b^T y$	$A^T y \geq c$	Eq. 2-24
	$0 \leq y$	

The equivalency holds in both directions, the dual representation of \mathcal{P} is \mathcal{D} , and \mathcal{D} has its dual representation in \mathcal{P} [50][51]. Aside from the equivalence, the weak duality principle establishes that: if \mathcal{P} is unbounded, then \mathcal{D} is infeasible; if \mathcal{D} is unbounded then \mathcal{P} is infeasible; and if \bar{x} is feasible for \mathcal{P} and \bar{y} is feasible for \mathcal{D} with $c^T \bar{x} = b^T \bar{y}$, then \bar{x} is an optimal solution to \mathcal{P} and \bar{y} is an optimal solution to \mathcal{D} [50].

2.6 CS Formulation

The measurements resulting from CS are a representation of the signal on an incomplete basis, meaning that determining α requires solving an undetermined set of equations.

As it was mentioned at the beginning of section 2.2, CS systems are capable of compressing at the sensor because they take a series of measurements instead of samples to represent a desired signal. These measurements are lineal combinations of the signal's samples and by approximating the signal to its sparse representation in some basis the samples can be recovered from fewer measurements than samples.

To illustrate the CS principle let us start with a sparse signal $x(n)$, that has a total of N values, of which only S are different from 0. The acquisition of the measurements is described mathematically in the following formula:

$$\begin{array}{l}
 b_i = \sum_{j=1}^N A_{i,j}x(j) \quad (a) \\
 b_i = \mathbf{A}_i * \mathbf{x} \quad (b)
 \end{array}
 \quad \text{Eq. 2-25}$$

Where: b_i is the measurement i , $x(j)$ is the sample j and $A_{i,j}$ is the coefficient that corresponds to the sample j for the measurement i [10].

In the case of 2 dimensional signals \mathbf{x} is a matrix and the measurement b_l is expressed as follows:

$$b_l = \sum_{i=1}^{N_{column}} \sum_{j=1}^{N_{row}} A_{l,i,j}x_{i,j} = \mathbf{A}_l \cdot \mathbf{x} \quad \text{Where: } l = (i - 1) * N_{row} + j \quad \text{Eq. 2-26}$$

The measurement matrix and signal can be rearranged as vectors, while maintaining the coefficient and sample pairs, so it can be expressed as the 1D case in Eq. 2-25. This presents an advantage when constructing the measurement vector \mathbf{y} , as it can now be written:

$$y = A * x_c \quad \text{Eq. 2-27}$$

Where y is a $M \times 1$ vector of the taken measurements, A is a $M \times N$ matrix where each row is the re-arranged measurement matrix for that sample, and x_c is the signal matrix shaped as a vector of size $N \times 1$. The signals recovery would then be achieved by solving:

$$A^{-1} * y = x_c \quad \text{Eq. 2-28}$$

If $M < N$ then the signals dimension or data has been reduced, achieving the desired compression but leaving the system in Eq. 2-28 underdetermined, which means that a large number of vectors of size N can solve Eq. 2-28, therefore determining the exact vector x_c becomes a more complicated issue.

2.6.1 Signal Recovery

In order to select the correct answer we include an additional characteristic in the solutions profile, its sparsity. Simply put the problem brought forward by CS is the reconstruction of a sparse signal from a limited number of linear measurements. It has been shown in general, the problem of constructing decoders is nontrivial as the linear system of M equations is underdetermined, and thus, admits infinitely many solutions. The decoder must then choose the “correct solution” among the infinitely many solutions. For sparse signals the problem of finding the desired solution can be expressed as an optimization problem where the objective is to maximize an appropriate measure of sparsity while simultaneously satisfying the constraints. The literature on CS and related subjects in sparse signal representations normally use the ℓ_0 pseudo norm detailed in Eq. 2-3 as a measure of a signal’s sparsity since it denotes how many of the elements are different from 0. This means the desired solution is the one that solves:

<i>Objective Function:</i>	<i>Subject to:</i>	Eq. 2-29
$\min_x \ x_c\ _0$	$A * x_c = y$	

CS would be of little practical use if it could only be used for sparse signals, so it must be extended to include signals which are not sparse when sensed. To show how this can be achieved we introduce a signal $f(n)$ which is not sparse when sampled, but which has a level of sparsity S when represented on a sparsifying basis ψ . This means that:

$$f_c = \psi * x_c$$

$$x_c = \psi * f_c \quad \text{Eq. 2-30}$$

Therefore when applying CS to f_c we get:

$y = A * f_c$	(a)	Eq. 2-31	
$y = A * \psi * x_c$	(b)		
<i>Objective Function:</i>	<i>Subject to:</i>		(c)
$\min_x \ x_c\ _0$	$y = A * \psi * x_c$		(d)

After solving Eq. 2-31 (d) for x_c , it is easy to recover f_c from it by means of Eq. 2-30(a).

Implementing the ℓ_0 norm guarantees a sparse solution, however to solve the equations involves solving a combinatorial problem which is computationally impractical for signals with a large size N . The ℓ_1 norm is a much more relaxed measurement of sparsity that has been proven to achieve the same sparse result of the ℓ_0 under certain conditions, while being much easier to use as the objective function.

Restricted Isometry Property (RIP)

In order to guarantee that the ℓ_1 norm will achieve the same results as the stricter ℓ_0 measure of sparsity the system must satisfy the restricted isometry property (RIP) [3][50].

$(1 - \delta)\ q\ _2^2 \leq \ A\Psi q\ _2^2 \leq (1 + \delta)\ q\ _2^2 \quad (a)$	Eq. 2-32
$\delta_k := \inf\{\delta: (1 - \delta)\ q\ _2^2 \leq \ A\Psi q\ _2^2 \leq (1 + \delta)\ q\ _2^2, \forall I \leq k, \forall q \in \mathbb{R}^{ I }\} \quad (b)$	

Satisfaction of the RIP condition is related to the level of mutual coherence $\mu(A, \Psi)$ between the measurement matrix A and the sparsifying basis Ψ and is:

$\text{Where:} \quad (a)$	Eq. 2-33
$\mu(A) := \max_{1 \leq i, j \leq N} \langle A_i, \psi_j \rangle \quad (b)$	
$1 \leq \mu \leq \sqrt{N} \quad (c)$	

For two normalized basis the range of μ is between 1 and \sqrt{N} , where 1 implies the basis are maximally incoherent, this is the case of the time basis $\phi_k(t) = \delta(t - k)$, and frequency or Fourier basis $\psi_j(t) = N^{-1/2} e^{i2\pi jt/N}$.

The importance of the concept of incoherence is that researchers have used it to estimate how many measurements are required to reconstruct a signal. It has been proven in [18] and [19] that random matrices with independent and identically distributed entries that follow either Gaussian or Bernoulli distribution with zero mean and variance $1/N$ satisfies the RIP condition as long as:

$$k \leq C \frac{M}{\log(N/M)} \quad \text{Eq. 2-34}$$

Researchers Candes and Romberg in [54] have drawn a simpler numerical formulation for the minimum number of samples M required for a signals correct reconstruction from:

- the signals sparsity in Ψ, S
- the signals original size N

$$M \geq S \cdot \log N \quad \text{Eq. 2-35}$$

LP Solution to CS Using ℓ_1 Norm

Due to the fact that the ℓ_1 norm is convex the problem becomes much easier to solve and now can be expressed as:

<p><i>Objective function:</i></p> $\min_x \ x\ _1$	<p><i>Constraints:</i></p> $A * f_c = A * \psi * x = y$	<p>Eq. 2-36</p>
--	---	------------------------

The ℓ_1 norm however is not linear, so the problem above these lines can not be solved by implementing regular LP algorithms, it must then be worked upon further. The following procedure is that done in [47] for the optimization of sparse signal representation using basis pursuit as both LP problems are equivalent. First it is necessary to get rid of the non-linear component in the objective function of Eq. 2-36 by introducing the variable t , such that for each element x_n of x there is an element t_n of t expressed as: $|x_n| \leq t_n$ for $1 \leq n \leq N$.

Rewriting the relation between x_n and t_n as $-t_n \leq x_n \leq t_n$ does away with the absolute value, and Eq. 2-36 can be written as:

Objective function:

$$\min t_1 + t_2 + \dots + t_N = \min \sum_{n=1}^N t_n = \min 1_N^T * t \quad \text{Eq. 2-37}$$

Where 1_N is a vector of ones of length N and t is a vector of length N composed of the individual t_n . The constraints have to be expanded into:

$A * x = y$	$x \leq t$ $-t \leq x$	Eq. 2-38
-------------	-------------------------------	----------

Finally the problem can be expressed as:

<p><i>Objective Function:</i></p> $\min 1_N^T * t$	<p><i>Constraints:</i></p> $A * x = y$ $0 \leq x + t$ $0 \geq x - t$	Eq. 2-39
--	--	----------

This problem is linear and convex, and can be solved using any suitable LP algorithm such as the simplex or interior point implementations.

By invoking the duality principle described in Eq. 2-23 and Eq. 2-24 the problem can be rewritten as:

<p><i>Objective Function:</i></p> $\max b^T * y$	<p><i>Constraints:</i></p> $A^T * y + v + w = 0$ $w - v = 1_N$ $0 \leq w$ $0 \geq v$	Eq. 2-40
--	--	-----------------

By solving the second constraint for w and placing it in the first and third constraints we get:

<p><i>Constraints:</i></p>	$A^T * y + v + 1_N + v = 0$ $0 \leq 1_N + v$ $0 \geq v$	Eq. 2-41
----------------------------	---	-----------------

Replacing $v \leq 0$ for $v' = -v$, so $v' \geq 0$ and:

<p><i>Constraints:</i></p>	$A^T * y - 2v' = 0$ $0 \leq 1_N - v'$ $0 \leq v'$	Eq. 2-42
----------------------------	---	-----------------

Finally the problem can be written as:

<p><i>Objective Function:</i></p> $\max b^T * y$	<p><i>Constraints:</i></p> $A^T * y - 2v = -1_N$ $0 \leq v \leq 1_N$	Eq. 2-43
--	--	-----------------

This LP problem can be solved using any traditional algorithm much easier than the initial CS formulation by having gotten rid of the ℓ_0 norm and solving for the unrestricted variable y .

2.6.2 CS implementations

The desire to capture images with greater detail has driven an increase in the resolution of imaging systems, which has been made possible by progressive technological improvements in imaging technology. A higher image resolution automatically triggers an increase in image size, and that coupled with the growing number of imaging systems have created a need for digital image compression methods to make their storage, transmission and analysis more practical. These compression methods rely mostly on a sparse approximation of some linear transformation operated upon the image (i.e. DCT and JPEG) and are completely independent of the sensor or source of the data.

Therefore in order to produce an image with a higher resolution a larger sensor array will be needed. If the application is trying to register low intensity signals the sensors in the array will need a higher light sensibility which is related to the sensor area (a greater area captures more energy or light at a given instant). The area of the sensor and the size of the sensor array are directly related to its cost, and can make some applications impractical or harder to implement. To make matters worse, no matter how high the complexity and costs of the imaging system (optics, sensor and electronics) most the data obtained will be discarded in a later compression stage.

Research in CS and the development of Texas Instruments DMDs have made it possible to implement a single pixel camera that can compress the image at the sensor while using the smallest possible sensor (a single pixel) [2]. This implementation uses the DMDs to perform the

product between the binary measurement matrix and the pixels or samples by having them reflect the light from the pixels towards, or away from, optics that add the contribution of the pixels completing the inner product between the image and measurement matrix. This brings additional benefits aside from those of CS, such as reduced complexity at the analog to digital converters and higher noise tolerance [2]. This scheme can also be implemented with HSI, as they operate on the same basic principle as cameras, and would process every band simultaneously. Switching the DMDs according to the measurement matrices allows it to construct the M measurements required to guarantee correct reconstruction at a later stage.

2.7 Classification

Because the targeted applications that might benefit the most from this approach are those whose end goal is classification or target detection the reconstructed images are subjected to a classification algorithm in order to compare their results with those achieved by the original image. The use of classifiers as a metric for the level of success of the signal reconstructions is done because correct classification was given more importance than the metrics normally used (i.e. quadratic error between the original and reconstructed images). This is done in hopes of more accurately measuring its performance in the targeted applications. It also takes into account that classifiers depend on several of those more traditional metrics (depending on the classifier) in order to differentiate the pixels and categorize them into groups, meaning that if the classifier groups both images in similar fashion they both share similar metrics.

Unsupervised classification is a means by which pixels in an image are assigned to spectral classes without the user having foreknowledge of the existence or names of those classes. It is performed most often using clustering methods. These procedures can be used to determine the

number and location of the spectral classes into which the data falls and determine the spectral class of each pixel. The analyst then identifies those classes a posteriori, by associating a sample of pixels in each class with available reference data, which could include maps and information from ground visits. Clustering procedures are generally computationally expensive yet they are central to the analysis of remote sensing imagery. While the information classes for a particular exercise are known, the analyst is usually totally unaware of the spectral classes, or sub-classes as they are sometimes called. Unsupervised classification is therefore useful for determining the spectral class composition of the data prior to detailed analysis by the methods of supervised classification.

2.7.1 K-Means

The k -means classification or clustering method assumes no prior knowledge of the data beyond the number of classes k present in the data. It separates the data into k groups or clusters by splitting the data space into Voronoi cells. To construct a Voronoi diagram there needs to be a seed, site or generator for each cell and each cell relates to the region where the distance to the corresponding seed is lower than that to any other seed. In the case of k -means clustering the seed is the mean of the data assigned to that class or cluster, therefore it changes when data is moved from one cluster to another at each iteration while looking for the best classification.

Because the seeds are not fixed in k -means clustering algorithms they need a way to determine which assignments are best, and it does so by minimizing the total Euclidean distance between the data points and its clusters mean. Mathematically the total sum of Euclidean distances for a cluster c_k , with $x_i \in c_k$ is:

$$J(c_k) = \sum_{x_i \in c_k} \|x_i - \mu_k\|^2 \quad \text{Eq. 2-44}$$

The total distance for a distribution C is the sum of the total distances for each cluster, k -means clustering seeks to minimize that value and it is expressed as:

$$\min J(C) = \min \sum_{k=1}^K \sum_{x_i \in c_k} \|x_i - \mu_k\|^2 \quad \text{Eq. 2-45}$$

This problem is NP hard, and is normally solved with greedy algorithms that have the following basic pseudo code shown in Figure 2-4. The greedy approach can only guarantee that the result reached is a local minimums, and depends heavily on the selection of c_k^0 , which is why many k -means implementations are run several times, using different initial clusters, and choosing the best solution among the results [49]. When applying k -means to HSI the spectrographs I_{n_1, n_2} for $1 \leq n_1, n_2 \leq N_1, N_2$ takes the place of x_i .

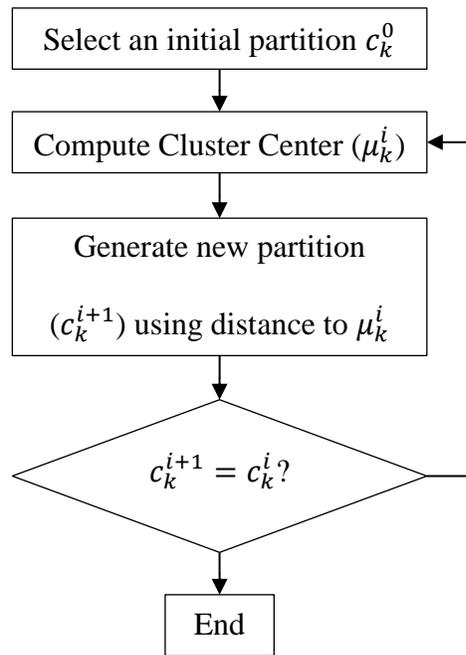


Figure 2-4: K-means pseudo code [49].

3. OBJECTIVES

3.1 General Objective

Develop a Recursive Compressed Sensing Implementation that could be used on single pixel Hyper spectral Imaging with Classification or Targeting applications.

3.2 Specific Objectives

- Compression and decompression of 1 dimensional signals using CS.
- Compression and decompression of 2 dimensional signals using CS.
- Perform 2 stage compression and decompression of 2 dimensional signals.
- Define the effects that the 2 stage compression settings have on the reconstructed images and the processing required at each of the stages.
- Analyze and compare the implications of CS's implementation on HSI with the use of filters or DMD's.

4. IMPLEMENTATION

Because HSIs form data cubes of size $N_1 \times N_2 \times B$ that can be physically interpreted either as a series of spectrographs at different spatial locations, or as a series of still images at different light bands, it is natural to compress the HSI as one of the two (independent vectors or matrices as shown in Figure 2-2). The first case would correspond to $N_1 * N_2$ vectors of length B and the second is a series of B matrices with dimensions $N_1 \times N_2$. Therefore HSIs can be compressed as a group of separate 1 or 2 dimensional signals, to establish a uniform notation we will refer to the HSI data cube as:

	<i>For:</i>	
	$0 \leq n_1 \leq N_1$	
$I(n_1, n_2, b)$	$0 \leq n_2 \leq N_2$	
	$0 \leq b \leq B$	Eq. 4-1

The first interpretation $I(n_1, n_2, :)$ will be represented by: I_{n_1, n_2} and the second $I(:, :, b)$ with: I_b . Viewing the hyper spectral image as vectors has the advantage of avoiding any spatial mixing, which might in turn allow for some level of classification on the fully compressed image. However to acquire the compressed signal or measurements (instead of the sampled signal) along the frequency bands the camera would need either: active filters that switched bands on and off for each individual pixel before reaching the sensor, or some backend signal processing to calculate them based on the sampled signal. None of these options reduces the camera's complexity, and the need for the complete spatial information means implementation of single pixel imaging would not be possible, taking away the possibility for a cheaper single pixel sensors.

Compressing the image as I_b however would allow for a single pixel sensor at the camera, but in order to do any analysis on the image it would have to be fully reconstructed, which is a large investment of computational power, which would be wasted if there were no elements of interest in the image.

Despite these limitations it is convenient to make a first approach to CS of the HSI by breaking it down into 1-D vectors as they are easier to process and visualize. Afterwards the HSI will be decomposed in 2-D signals for its compression and reconstruction, and finally the recursive approach will be tested using the methods and algorithms developed in the first two stages.

4.1 1 - Dimensional CS

The image used for this series of tests is the well-known in HSI literature Indian Pines, and is shown in Figure 4-1. It has a spatial resolution of 145×145 pixels and 200 bands and was sourced from [48] as well as its ground truth. The total amount of data (numerical values) in the image is:

$$N_1 * N_2 * B = 145 * 145 * 200 = 4205 * 10^3 \quad \text{Eq. 4-2}$$

As mentioned in a previous section the HSI is not sparse in nature, and its representations using transforms like the DCT is not strictly sparse either, however it does shift significance to a dramatically smaller number of coefficients (as shown in Figure 4-1). It is then necessary to determine how many samples are required to come close to the original signal within a defined percentage α by calculating S such that:

$$S: \sum_{s=1}^S \left\| \max_s DCT(I_{n_1, n_2}) \right\|^2 \geq \alpha * \sum_{b=1}^B \left\| I_{n_1, n_2}(b) \right\|^2 \quad \text{Eq. 4-3}$$

Where: $DCT(I_{n_1, n_2})$ is the DCT transform of the spectrogram at a pixel n_1, n_2 ; $\max DCT(I_{n_1, n_2})$ is a vector of the absolute values from the coefficients $DCT(I_{n_1, n_2})$ ordered in a descending fashion; and finally $\max_s DCT(I_{n_1, n_2})$ is the s value of said vector, corresponding to the s highest maximum.

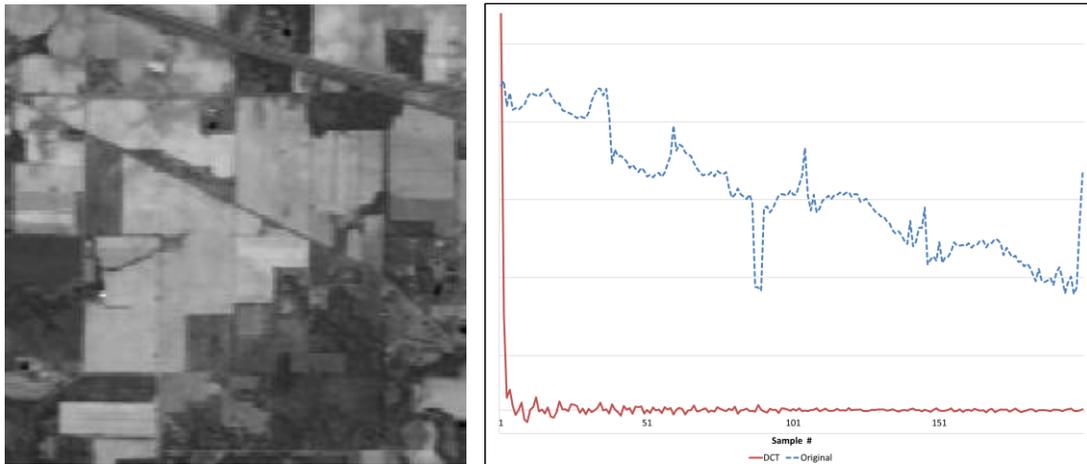


Image has 145 samples, 145 lines (145×145 pixels) and 200 bands. Sample band (left), Spectrograph at pixel 121, 65 in blue, as well as its respective DCT in red (right).

Figure 4-1 Indian Pines HIS.

To guarantee that all the components of the HSI can be reconstructed correctly the highest value of S among those calculated for all the vectors I_{n_1, n_2} is used to determine the number of measurements required for the complete image using Eq. 2-35.

The results obtained for an $\alpha = 0.995$ is $S = 15$, which produces $M \geq 79.5$. The selected number of measurements for the initial tests is $M = 90$, so the resulting measurement matrix A is of size $M \times B = 90 \times 200$, where each of the rows is used to combine the samples in the spectrogram into a measurement. The same matrix A is used for all the spectrograms

independently and the values in A are binary bits assigned randomly to guarantee incoherence with the DCT basis.

The compressed image IC is a data cube of dimensions $N_1 \times N_2 \times B = 145 \times 145 \times 90$, and the measurement m ($1 \leq m \leq 90$) of the pixel n_1, n_2 ($1 \leq n_1, n_2 \leq 145$) is calculated as:

$$IC_{n_1, n_2, m} = \sum_{b=1}^B A(m, b) * I_{n_1, n_2}(b) = A_m * I_{n_1, n_2} \quad \text{Eq. 4-4}$$

The total amount of data contained in IC is: $N_1 * N_2 * M = 145 * 145 * 90 = 1892,25 * 10^3$.

The reduction in the amount of data due to the implementation of the measurement matrix is: $100(M/B)\% = 100(90/200)\% = 45\%$.

The compressed spectrograph for a specific pixel n_1, n_2 is IC_{n_1, n_2} :

$$IC_{n_1, n_2} = A * I_{n_1, n_2} \quad \text{Eq. 4-5}$$

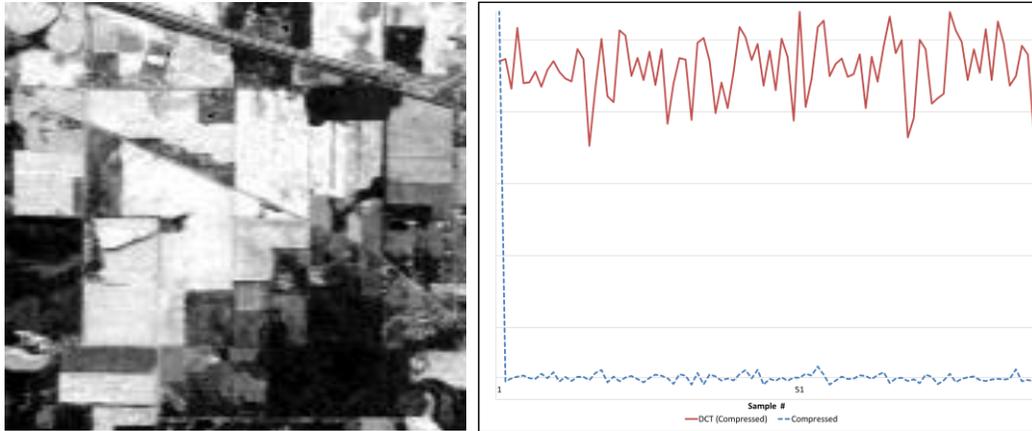
Compression of the full HSI can be done in a single operation by rearranging the spectrographs I_{n_1, n_2} into a matrix I_V of size $B \times (N_1 * N_2) = 200 \times 21025$, (where the columns correspond to each of the individual spectrograms) and calculating:

$$IC_V = A * I_V \quad \text{Eq. 4-6}$$

IC_V is a reordered version of IC , and they are related by:

$$\boxed{\begin{aligned} IC_{n_1, n_2} &= IC_V(n) \\ n &= (n_2 - 1) * N_1 + n_1 \end{aligned}} \quad \text{Eq. 4-7}$$

A sample band of the compressed Indian Pines image is shown in Figure 4-2. It should be noted that the bands that compose IC do not correspond to any specific band in the original image, as they result from linear combinations (in the case of the figure determined by A_{56}) of all the spectrograms in the image.



Compressed image IC ; Sample band 56 , IC_{56} (left), spectrograph at pixel $121, 65$ $IC_{121,65}$ as well as its DCT $DIC_{121,65}$ (right).

Figure 4-2 Compressed Indian Pines image.

The bands IC_b of the resulting compressed image are a close representation of those in the original image I_b as there has been no spatial mixing among the pixels in the compressed image. This immediacy can be observed in Figure 4-1 and Figure 4-2; the visible resemblance of the two spectrographs makes it seem possible to draw conclusions on the full image from analysis of the compressed version. The reach of this concept is tested by subjecting the compressed image to the k -means classification tool in Envi and constructing the confusion matrix using the image ground truth, resulting in a percentage of correct detection of 30.5%, which is lower than the 35.3% of the original image under the same parameters. The confusion matrices for the original image and the compressed version are shown in Table 4-1 and Table 4-2 respectively.

This result shows that some form of classification comparable to that of the original image can be implemented in this preliminary stage, allowing some decision making before investing the

computational cost required to reconstruct the complete image. However, to improve the classification from that obtained using the compressed signal IC the full sized image must be reconstructed.

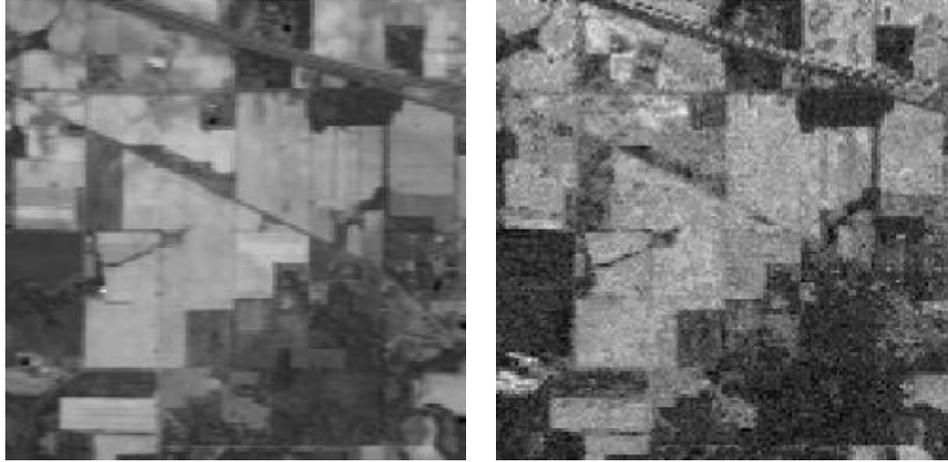
The reconstructed image is acquired indirectly by finding the sparsest solution to the set of equations formulated by the measurement matrix. The LP problem for each spectrograph is described mathematically as:

$$IR_{n_1, n_2} = IDCT * DIR_{n_1, n_2} \quad \text{Eq. 4-8}$$

$\min \ DIR_{n_1, n_2}\ _1$ $IC_{n_1, n_2} = A * IDCT * DIR_{n_1, n_2}$	Eq. 4-9
---	----------------

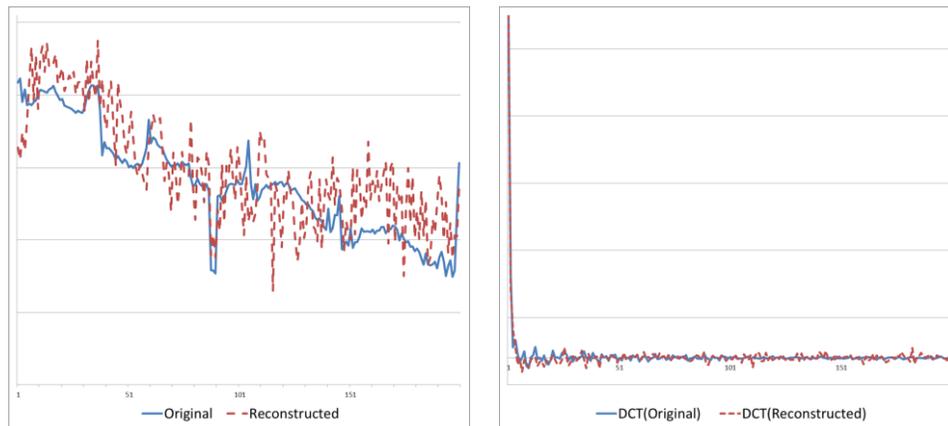
Where DIR_{n_1, n_2} is the desired sparse signal or DCT of the reconstructed spectrograph for n_1, n_2 and $IDCT$ is the matrix representation of the inverse DCT.

The approach described in 0 is used to rewrite and solve the minimization problem, finding DIR_{n_1, n_2} and using it to reconstruct the complete image. Each separate spectrograph is reconstructed individually in any order independently of the others, making this process highly parallelizable. The resulting image composed of the reconstruction of each of the spectrograms is shown in Figure 4-3 next to the original image, and Figure 4-4 compares a spectrogram of the original and reconstructed versions of the image, as well as their DCTs.



Sample band of the original Indian Pines image I_{125} (left), Sample band of the reconstructed Indian Pines image IR_{125} (right).

Figure 4-3: Reconstructed Indian Pines.



Original and reconstructed spectrograph at pixel $121, 65$, $I_{121,65}$ solid blue and $IR_{121,65}$ dashed red (left), as well as its DCT $DI_{121,65}$ solid blue and $DIR_{121,65}$ dashed red (right).

Figure 4-4: Indian Pines spectrogram reconstruction.

As another point of comparison on the implementations efficiency in classification applications the reconstructed image is operated upon by the k -means classification tool from Envi, under the same parameters as the compressed version of the image. The resulting classification accuracy was of 34.8%, and the confusion matrix resulting from it is shown in using Indian Pines ground truth, the results are detailed in Table 4-3.

Overall Accuracy of 35.3%, (3617/10249).																
Class	Ground Truth (Pixels)															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	28	0	0	1	13	0	2	237	0	0	0	0	0	0	0	0
2	0	254	45	48	0	0	0	0	0	0	44	43	0	0	0	10
3	0	296	289	1	0	0	0	0	0	250	751	63	0	0	1	3
4	0	157	45	58	1	0	0	0	0	0	1	35	0	0	0	0
5	0	0	0	0	270	21	0	0	0	0	0	0	0	583	11	0
6	0	3	0	0	39	354	0	0	12	1	9	0	0	9	96	0
7	1	60	76	48	12	0	6	0	0	156	216	24	1	0	7	0
8	17	3	0	1	83	2	20	239	0	4	11	0	0	0	7	0
9	0	74	39	11	3	0	0	0	0	61	120	111	0	0	8	0
10	0	117	100	2	0	0	0	0	0	239	208	117	0	0	2	0
11	0	201	166	8	1	0	0	0	0	178	796	45	0	0	0	0
12	0	240	61	0	1	0	0	0	0	77	254	136	0	0	0	3
13	0	1	0	9	0	224	0	0	7	0	4	0	203	20	114	0
14	0	0	0	0	48	41	0	0	0	0	0	0	0	653	125	0
15	0	22	9	50	12	88	0	2	1	6	41	19	1	0	15	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	77

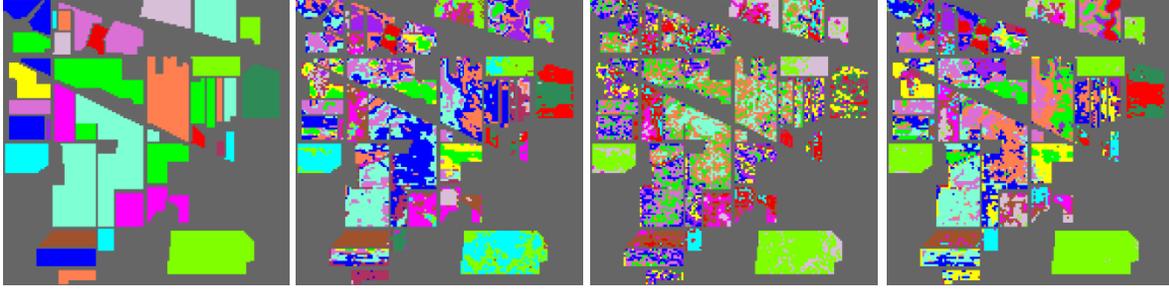
Table 4-1: *k*-means Indian Pines - Confusion matrix.

Overall Accuracy of 30.5%, (3122/10249).																
Class	Ground Truth (Pixels)															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	31	7	10	12	19	38	3	26	0	9	41	36	1	0	8	22
2	0	325	122	0	0	0	0	0	0	128	451	99	0	0	0	6
3	0	117	245	12	0	0	0	2	0	67	249	85	0	0	0	0
4	0	84	33	70	3	0	4	110	0	67	111	28	0	0	0	4
5	1	6	2	9	79	71	0	0	0	1	38	6	2	0	36	26
6	0	1	0	0	4	198	0	0	0	0	1	0	20	13	69	0
7	12	28	7	22	8	12	7	133	0	12	28	12	0	0	5	10
8	1	46	11	40	7	4	12	189	0	32	47	17	0	0	3	0
9	1	133	107	66	5	0	1	18	0	74	135	48	0	0	0	4
10	0	368	20	0	0	0	0	0	0	313	513	87	0	0	0	2
11	0	131	1	0	0	0	0	0	0	198	399	28	0	0	0	3
12	0	179	272	3	0	0	0	0	0	70	432	147	0	0	0	1
13	0	1	0	0	6	204	0	0	13	0	5	0	159	2	58	3
14	0	0	0	0	245	0	0	0	0	0	0	0	0	817	9	1
15	0	0	0	0	69	46	0	0	0	0	0	0	0	432	132	0
16	0	2	0	3	38	157	1	0	7	1	5	0	23	1	66	11

Table 4-2: *k*-means Compressed Indian Pines - Confusion.

Overall Accuracy of 34.8%, (3567/10249).																
Class	Ground Truth (Pixels)															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	21	0	0	0	4	0	6	230	0	0	4	1	0	0	0	0
2	0	244	73	0	0	0	0	0	0	235	153	36	0	0	0	9
3	0	211	298	6	0	0	0	0	0	43	335	119	0	0	1	1
4	1	203	89	145	5	0	4	0	0	104	148	37	0	0	1	0
5	9	4	1	7	122	79	1	1	8	5	10	0	6	0	42	0
6	0	1	0	11	4	190	0	0	2	0	2	0	14	0	52	0
7	0	30	11	58	19	22	8	2	0	24	64	29	1	0	14	0
8	15	1	0	1	4	0	9	245	0	0	1	0	0	0	0	0
9	0	110	50	1	0	0	0	0	0	97	95	80	0	0	0	0
10	0	186	52	0	0	0	0	0	0	313	572	32	0	0	0	1
11	0	138	222	7	2	0	0	0	0	26	640	73	0	0	0	0
12	0	289	22	0	0	0	0	0	0	125	343	138	0	0	0	9
13	0	1	0	1	5	208	0	0	10	0	4	0	184	5	74	0
14	0	0	0	0	302	9	0	0	0	0	0	0	0	1226	109	0
15	0	1	0	0	16	222	0	0	0	0	2	0	0	34	86	0
16	0	9	12	0	0	0	0	0	0	0	82	48	0	0	7	73

Table 4-3: *k*-means Reconstructed Indian Pines – Confusion.



From left to right, ground truth [48] and results for k -mean classifier using: I , IC and IR .

Figure 4-5: k -means classification result for Indian Pines.

As expected the classification rates for the reconstructed image are better than those of the fully compressed image, but are still lower than those of the original image, this is due to some loss of information in the process of compression and reconstruction.

4.2 2 - Dimensional CS

After successful implementation of CS on a HSI viewed as a group of vectors, a 2 dimensional implementation is the next incremental step towards achieving recursive 2 dimensional CS on HSI. Experiments in this regard done using the HSI Salinas-A scene of dimensions $83 \times 86 \times 204$, sourced from [48]. Taking advantage of the independence of the bands in the proposed CS method, the algorithms can be tested using a single band $b = 8$ treating it as a greyscale image which is shown in Figure 4-6. The level of sparsity is determined in a similar fashion as in the 1 dimensional case by approximating the image to a sparse signal within a tolerable level of error. Additional considerations must first be made before determining the number of samples required to approximate the signal because the signal is now a 2-D signal; chief among these: the 1-D DCT is replaced by the 2-D DCT which is more suitable for images. The resulting coefficients of the 2-D DCT are shown in Figure 4-6, these are then placed in a vector $max DCT(I_b)$ in descending order according to their absolute value. The level of sparsity is then determined by:

$$S: \sum_{s=1}^S \|\max_s DCT(I_b)\|^2 \geq \alpha * \sum_{n_2=1}^{86} \sum_{n_1=1}^{83} \|I_b(n_1, n_2)\|^2 \quad \text{Eq. 4-10}$$

For the grayscale image I_8 and an approximation level of $\alpha = 0.9995$, the resulting $S = 42$, meaning that according to Eq. 2-35, $M \geq 372.7$, in order to satisfy this condition tests were conducted for $M = 400$ and $M = 1000$. The number of measurements taken (M) versus number of samples in the original image ($N = 86 * 83 = 7138$) gives compression ratios of 5.6% and 14.01% respectively. It is interesting to note that should the other bands in the original image have lower values of S the data ratios achieved would not have to be changed, and the same measurement matrix could be used for the complete image.

The measurement matrix A is a cube of size $N_1 \times N_2 \times M$ composed of binary bit generated randomly so that it has a Gaussian distribution that is incoherent with the DCT basis.

$$IC_b(m) = \sum_{n_2=1}^{86} \sum_{n_1=1}^{83} A(n_1, n_2, m) * I_b(n_1, n_2) \quad \text{Eq. 4-11}$$

The compressed signal for the band b IC_b is a vector of length M , when working with a full HSI the same measurement matrix is used to compress all the bands independently; and the compressed signal IC will have dimensions $M \times B$ where each column corresponds to the M measurements taken for the band b . In this particular implementation it can be said for practical purposes that $B = 1$.

The signals compression was achieved in Matlab by reordering the measurement matrix into an equivalent matrix A_R of size $M \times N$, with $N = N_1 * N_2$ and reordering the matrix I_8 into a vector

of length N I_{R_8} . The M measurements can then be obtained by simply operating the matrix vector product:

$$\begin{aligned}
 IC_8 &= A_R * I_{R_8} \quad \text{Where: } I_{R_8}(n) = I_8(n_1, n_2) \\
 A_R(m, n) &= A(n_1, n_2, m) \\
 n &= n_1 + (n_2 - 1)N_1
 \end{aligned}
 \tag{Eq. 4-12}$$

In the case compression where to be performed on the totality of the B bands of the HSI I_C is:

$$\begin{aligned}
 IC &= A_R * I_R \quad \text{Where: } I_R(n) = I(n_1, n_2) \\
 A_R(m, n) &= A(n_1, n_2, m) \\
 n &= n_1 + (n_2 - 1)N_1
 \end{aligned}
 \tag{Eq. 4-13}$$

I_R has dimensions $N \times B$, and each column b corresponds to a I_{R_b} .

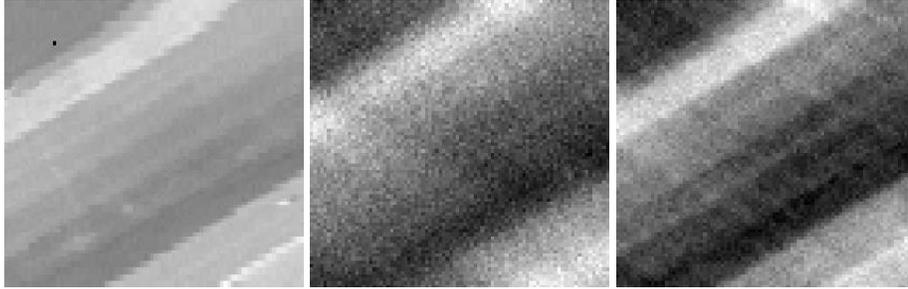
Image reconstruction is done by solving for each $1 \leq b \leq B$ the optimization problem:

$$\begin{aligned}
 &\min \|DCT2I_{b_R}\|_1 \\
 IC_b &= A * IDCT2 * DCT2I_{R_b}
 \end{aligned}
 \tag{Eq. 4-14}$$

The reconstructed image is then:

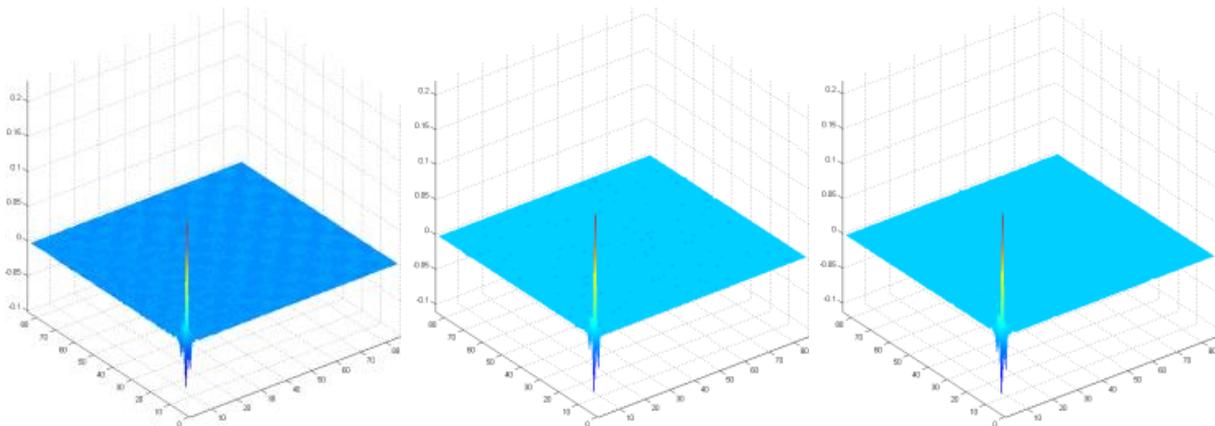
$$IR_{R_b} = IDCT2 * DCT2I_{b_R}
 \tag{Eq. 4-15}$$

The resulting reconstructed images are shown in Figure 4-6, as well as their respective 2-D DCT transforms are shown in Figure 4-7.



From left to right, sample band 8 Salinas-A scene: for the original image, reconstructions from **400** measurements and **1000** measurements.

Figure 4-6: Salinas-A scene sample bands.



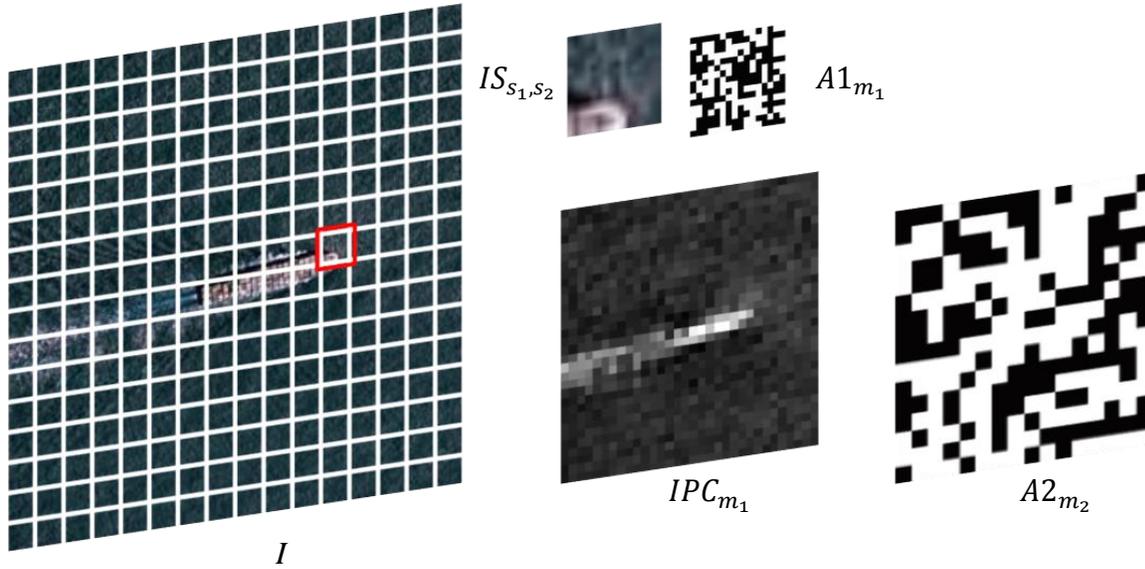
From left to right, 2-D *DCT* of Salinas-A scene: for the original image I , reconstructions IC from **400** measurements and **1000** measurements.

Figure 4-7 Salinas-A scene 2-D *DCT*.

The increased level of approximation achieved by taking additional measurements is noticeable at plain sight in Figure 4-6, however neither of the reconstructions could account for the black pixel located at 15, 13 of the original image. This is a manifestation on the *lossy* nature of CS.

4.3 Recursive 2 - Dimensional CS

Both CS applications shown in the previous sections present some disadvantage; either the real world implementation becomes more complicated or additional steps are required before any signal analysis can be done. As a way to reach a compromise between the two options while still allowing for single pixel imaging, a novel CS method based on a measurement matrix constructed as the Kroneker product of two smaller ones is proposed.



The original image is split into sections IS_{s_1, s_2} that are compressed using a measurement matrix $A1_{m_1}$. The measurements m_1 from all the sections form a partially compressed image IPC_{m_1} that is fully compressed using a second measurement matrix $A2_{m_2}$.

Figure 4-8: Proposed CS method.

When the resulting measurement matrix is used to compress an image it is the mathematical equivalent of compressing image sections with a first measurement matrix and completing the compression process with a second measurement matrix as shown in Figure 4-8.

This recursive two stage compression equivalency allows for single step compression and recursive decompression, so the first stage recovers the compressed image sections to form a partially reconstructed image suitable for analysis and decision making before full recovery of the original image.

The methods and algorithms presented in the previous sections of this chapter for the compression and decompression of HSI will now be put to use towards the proposed implementation of CS. As was the case for the 2-D implementation, the validity of the proposed methods can be tested on any 2-D signals and later extended to HSI.

4.3.1 Measurement matrix construction:

The two successive compression stages can be implemented in a single step by using a measurement matrix that is the Kroneker product of the two smaller measurement matrices. The ability to obtain the measurements in a single step makes the use of a single pixel camera straightforward.

There are two possible ways to mathematically obtain a measurement $m = (m_2 - 1) * M_1 + m_1$, the first is operating directly on the complete image by using a measurement matrix constructed from the measurement matrices $A1_{m_1}$ and $A2_{m_2}$, the second implies operating the measurement matrix $A1_{m_1}$ on sections of the image, composing a partially compressed image with the results from the sections and operating on this image with the measurement matrix $A2_{m_2}$. Both methods produce the same measurement.

The first measurement matrix will be referred to as $A1_{m_1}$ and is of size $H \times W$, where $H = N_1/S_1$ and $W = N_2/S_2$, S_1 and S_2 are determined by the user and correspond to the number of sections the image will be split into along each image axis.

The second matrix $A2_{m_2}$ is of size $S_1 \times S_2$, and it corresponds to the final compression stage, it is used to recover the partially compressed image from the final measurements.

The measurement matrix A_m is:

$$A_m = A2_{m_2} \otimes A1_{m_1} \quad \text{Eq. 4-16}$$

According to the definition of Kroneker product this is:

$$A1_{m_1} = \begin{bmatrix} a1_{1,1} & \dots & a1_{1,j} & \dots & a1_{1,W} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a1_{i,1} & \dots & a1_{i,j} & \dots & a1_{i,W} \\ a1_{i+1,1} & \dots & a1_{i+1,j+1} & \dots & a1_{i+1,W} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a1_{H,1} & \dots & a1_{H,j} & \dots & a1_{H,W} \end{bmatrix} \quad \text{Eq. 4-17}$$

$$A2_{m_1} = \begin{bmatrix} a2_{1,1} & \dots & a2_{1,j} & \dots & a2_{1,S_2} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a2_{i,1} & \dots & a2_{i,j} & \dots & a2_{i,S_2} \\ a2_{i+1,1} & \dots & a2_{i+1,j} & \dots & a2_{i+1,S_2} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a2_{S_1,1} & \dots & a2_{S_1,j} & \dots & a2_{S_1,S_2} \end{bmatrix} \quad \text{Eq. 4-18}$$

$$A_m = \begin{bmatrix} a2_{1,1} * a1_{1,1} & \dots & a2_{1,1} * a1_{1,W} & \dots & a2_{1,j} * a1_{1,1} & \dots & a2_{1,j} * a1_{1,W} & \dots & a2_{1,S_2} * a1_{1,1} & \dots & a2_{1,S_2} * a1_{1,W} \\ \vdots & \ddots & \vdots \\ a2_{1,1} * a1_{H,1} & \dots & a2_{1,1} * a1_{H,W} & \dots & a2_{1,j} * a1_{H,1} & \dots & a2_{1,j} * a1_{H,W} & \dots & a2_{1,S_2} * a1_{H,1} & \dots & a2_{1,S_2} * a1_{H,W} \\ \vdots & \ddots & \vdots \\ a2_{i,1} * a1_{1,1} & \dots & a2_{i,1} * a1_{1,W} & \dots & a2_{i,j} * a1_{1,1} & \dots & a2_{i,j} * a1_{1,W} & \dots & a2_{i,S_2} * a1_{1,1} & \dots & a2_{i,S_2} * a1_{1,W} \\ \vdots & \ddots & \vdots \\ a2_{i,1} * a1_{H,1} & \dots & a2_{i,1} * a1_{H,W} & \dots & a2_{i,j} * a1_{H,1} & \dots & a2_{i,j} * a1_{H,W} & \dots & a2_{i,S_2} * a1_{H,1} & \dots & a2_{i,S_2} * a1_{H,W} \\ \vdots & \ddots & \vdots \\ a2_{S_1,1} * a1_{1,1} & \dots & a2_{S_1,1} * a1_{1,W} & \dots & a2_{S_1,j} * a1_{1,1} & \dots & a2_{S_1,j} * a1_{1,W} & \dots & a2_{S_1,S_2} * a1_{1,1} & \dots & a2_{S_1,S_2} * a1_{1,W} \\ \vdots & \ddots & \vdots \\ a2_{S_1,1} * a1_{H,1} & \dots & a2_{S_1,1} * a1_{H,W} & \dots & a2_{S_1,j} * a1_{H,1} & \dots & a2_{S_1,j} * a1_{H,W} & \dots & a2_{S_1,S_2} * a1_{H,1} & \dots & a2_{S_1,S_2} * a1_{H,W} \end{bmatrix} \quad \text{Eq. 4-19}$$

The total number of measurements taken to fully compress the image is the product of the number of measurement matrices $A1_{m_1}$ and $A2_{m_2}$, M_1 and M_2 respectively; as the result of each measurement matrix $A1_{m_1}$ is operated on by each $A2_{m_2}$. Therefore the group of measurement matrices A forms a cube of dimensions $N_1 \times N_2 \times M$, ($M = M_1 * M_2$). The use of a measurement matrices constructed using Kroneker products also requires less storage (memory), as they can be expressed by the two smaller measurement matrices.

4.3.2 Complete Compression and Reconstruction:

As with the implementation in section 0, the M measurements can be obtained in a single step by reshaping A and I into A_R and I_R of dimensions $M \times N$ and $N \times B$ respectively so:

$$IC_R = A_R * I_R \quad \text{Eq. 4-20}$$

The previous equation produces the $M \times B$ measurements required to fully recover the original image by solving the following LP minimization problem for each band $1 \leq b \leq B$:

$$IR_{R_b} = IT2 * T2IR_{R_b} \quad \text{Eq. 4-21}$$

$\min \ T2IR_{R_b}\ _1$	Eq. 4-22
$IC_{R_b} = A_R * IT2 * T2IR_{R_b}$	

Again the image is reconstructed indirectly by first recovering the 2-D sparse representation of the image $T2IR_{R_b}$ on a transform basis (DCT2 or 2-D Wavelets), and later inverting the transform with the matrix $IT2$. The recovered image is a reordered version of IR_{R_b} .

4.3.3 Partial Compression

Partial compression is the first step in the two stage compression approach towards IC . In order to partially compress the image must first split the image into $S_1 \times S_2$ sections of size $H \times W$ for each individual band. These sections are:

$IS_{s_1, s_2, b} = I_b(i, j) \quad \text{For:} \quad \text{Where:}$	Eq. 4-23
$1 \leq s_1 \leq S_1 \quad H * (s_1 - 1) + 1 \leq i \leq H * s_1$	
$1 \leq s_2 \leq S_2 \quad W * (s_2 - 1) + 1 \leq j \leq W * s_2$	

The partially compressed image $IPC_{m_1,b}$ is formed with the A_{m_1} measurement matrix operating on each $IS_{s_1,s_2,b}$, it has size $S_1 \times S_2$, and is calculated as:

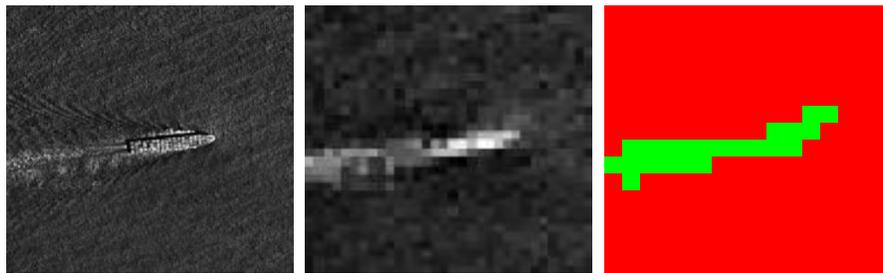
$$IPC_{m_1,b}(s_1, s_2) = \sum_{h=1}^H \sum_{w=1}^w A1_{m_1}(h, w) * IS_{s_1,s_2,b}(h, w) \quad (a)$$

$$IPC_{m_1,b}(s_1, s_2) = A1_{m_1} \cdot IS_{s_1,s_2,b} \quad (b)$$

Eq. 4-24

The number of measurement matrices $A1_{m_1}$ required for correct reconstruction of the image sections is M_1 and is determined by the least sparse among the image sections that will be compressed using $A1_{m_1}$. When it is not possible to determine the levels of sparsity of the image sections because of lack of access to the original image the value is based on previously acquired statistical knowledge of the expected image sections.

After compressing all the sections for a band b we can construct $IPC_{m_1,b}$, this partially compressed image holds great interest because it resembles the original image at that band as evidenced in Figure 4-9. Again as in the case from 4.1 it seems we can draw certain conclusions on the original image from it, in the case of target detection applications it might be possible to determine which sections correspond to the target of interest. As an initial test of this hypothesis classification of an image with only 2 classes is shown in Figure 4-9.



From left to right, sample band of the original image, rearranged partially compressed image IPC_{m_1} , result of k -means classifier on IPC .

Figure 4-9: k -means Classification on IPC .

It is also fairly plain to assume the quality of the analysis that can be done with *IPC* is inversely related to the size of the sections, as smaller sections present less spatial mixing. The lower limit in this relation is determined by the minimum section size possible of a single pixel, in which case the partially compressed image is equivalent to the original image. Section size is determined by the user with the selection of the number of sections S_1 and S_2 , this allows the user a level of control over the quality of the analysis (it will be shown later on that S_1 and S_2 also affect the time required to recover *IPR*, creating a compromise between the two).

4.3.4 Final Compression Stage

To arrive at the measurements for the fully compressed image *IC* from 4.3.2 the second measurement matrix A_2 has to be used to combine the pixels from the individual partially compressed images $IPC_{m_1,b}$. Mathematically this can be done by reshaping the measurement matrix $A_{2_{m_2}}$ into a vector $A_{2_{R_{m_2}}}$ of size $1 \times S$, $S = S_1 * S_2$ and reshaping $IPC_{m_1,b}$ into the vector $IPC_{R_{m_1,b}}$ of size $S \times 1$ and evaluating the product of the two vectors.

$$IC_{m,b} = A_{2_{R_{m_2}}} * IPC_{R_{m_1,b}} \quad \text{Eq. 4-25}$$

In order to guarantee that $IPC_{R_{m_1,b}}$ can be successfully recovered we must first have a statistical estimate of the level of sparsity of the partially compressed images. As with the case of the image sections the minimum number of measurements is determined by the least sparse among $IPC_{R_{m_1,b}}$, this level of sparsity is used to determine M_2 .

The position of the measurement product from the partially compressed image m_1,b and the measurement matrix $A_{2_{m_2}}$ in the vector IC_b is $m = (m_2 - 1) * M_1 + m_1$. The combinations, for

an individual band, of m_1 and m_2 from the groups of measurement matrices $A1$ and $A2$ respectively fill the M values of IC in the same order as the single stage compression.

4.3.5 Partial Reconstruction

Whether the partially compressed image would offer some insight on the contents of the original image or not would not be of any use if the original image were required to construct it. To make it of any practical interest it must be possible to reach IPC from the measurements of the fully compressed image IC .

To avoid confusion: the partially compressed image that is obtained by rearranging the measurements from the compressing the image sections using $A1_{m_1}$ will be $IPC_{m_1,b}$; while $IPR_{m_1,b}$ will be used to refer to the partially reconstructed image or reconstructed version of $IPC_{m_1,b}$ that is obtained from the final fully compressed measurements and $A2$.

The values of m that are produced by $1 \leq m_2 \leq M_2$ with a fixed m_1 and b are needed in order to reconstruct $IPR_{m_1,b}$. To simplify this notation we will temporarily refer to these as $IC_{m_2,b}$. After being ordered according to m_2 these measurements can be expressed as:

$$IC_{m_2,b} = A2_R * IPC_{R_{m_1,b}} \quad \text{Eq. 4-26}$$

Here: $A2_R$ is a matrix of size $M_2 \times S$, where the rows are reshaped versions of $A2_{m_2}$; $IPC_{R_{m_1,b}}$ is a vector of length S containing the same elements of $IPC_{m_1,b}$.

Expressing the measurements in that form makes it clear that it is the same case as that from section 4.2 and the process can be then inverted in the same fashion as before by solving:

$$\min \|T2IPR_{R_{m_1,b}}\|_1$$

$$IC_b = A * IT2 * T2IPR_{R_{m_1,b}}$$

Eq. 4-27

Unlike the application in 4.2, the sparsifying basis or transform is not specified but left open as $T2$ with an inverse that can be expressed in matrix form as $IT2$. The reconstructed version of IPC_b is then obtained by rearranging the elements from IPR_{R_b} into a matrix of size $S_1 \times S_2$:

$$IPR_{R_b} = IT2 * T2I_{R_b}$$

Eq. 4-28

Because the partially reconstructed image is a version of the partially compressed image it can also be used to perform analysis like the target detection shown in Figure 4-8, allowing for some decision making before investing in further reconstruction that might be wasted on image section that hold no valuable information.

4.3.6 Complete Reconstruction

In order to reconstruct IR_b the individual sections must be reconstructed and later stitched together, therefore it is necessary to first determine what measurements correspond to which sections.

The totality of the partially reconstructed images for a band b form a data cube IPR_b of dimensions $S_1 \times S_2 \times M_1$, the vector $IPR_b(s_1, s_2, :)$ of length M_1 can then be extracted from the data cube and it corresponds to the measurements from the section s_1, s_2 . Expressed as vector matrix operations:

$$IPR_b(s_1, s_2, m_1) = A1_{R_{m_1}} * IS_{R_b, s_1, s_2}$$

Eq. 4-29

And the M_1 measurements from the compression of the section IS_{b,s_1,s_2} correspond to:

$$IPR_b(s_1, s_2, :) = A1_R * IS_{R_{b,s_1,s_2}} \quad \text{Eq. 4-30}$$

The same reconstruction procedure as in 4.2 is called upon for one last time, expressing the recovery of $T2ISR_{R_{s_1,s_2,b}}$ as solution of the LP minimization problem:

$$\min \|T2ISR_{R_{s_1,s_2,b}}\|_1$$

$$IPR_b(s_1, s_2, :) = A1_R * IT2 * T2ISR_{R_{s_1,s_2,b}}$$

Eq. 4-31

As with the previous cases in order to obtain the reconstructed version of the original signal (in this case the image sections) the solution to the LP problem must be subjected to the inverse transform, so:

$$ISR_{R_{s_1,s_2,b}} = IT2 * T2ISR_{R_{s_1,s_2,b}} \quad \text{Eq. 4-32}$$

However a final step must be taken in order to recover the reconstructed version of the original image $IR_b(n_1, n_2)$, it requires the organization of the sections into a single image. This is done by using the following equivalency:

$$IR_b(n_1, n_2) = ISR_{s_1,s_2,b}(h, w) \quad \text{Where:}$$

$$n_1 = h + (s_1 - 1) * H$$

$$n_2 = w + (s_2 - 1) * W$$

$$1 \leq h \leq H$$

$$1 \leq w \leq W$$

Eq. 4-33

5. Tests and Results

The main implementation envisioned for the proposed method is the search for targets or anomalies data from large areas acquired through the use of single pixel remote sensors. It is why, in an effort perceive how the proposed method would perform in said scenario, the image selected for the definitive tests is that of a cargo ship at sea.

Specifically it is a RGB image of a cargo ship at the Gibraltar strait, it was sourced from Google maps (coordinates $35^{\circ}54'8.12''N$ $5^{\circ}44'26.32''W$) and is shown in Figure 5-1. The image was cropped to be 256×256 pixels in order to conduct tests for values of $S = 16$ and 32 without having to change or crop the image for each test.



256×256 pixels, 3 bands (RGB) and 2 object types (boat and water).

Figure 5-1: RGB satellite image of a boat at sea.

To be able to draw conclusions that are closer to those desired in the intended real world application the recovered images are subjected to the k -means classification algorithm in order to

determine and compare the classification accuracy for each configuration at each stage. In this regard the image also presents an advantage as it only contains two classes (boat and sea water).

The result of each comparison is displayed in a square matrix of size $C \times C$, in this case for two classes $C = 2$. This matrix is a confusion matrix and the columns $p_{i,j}$, for $1 \leq i \leq C$ correspond to the class j in the ground truth image. The rows $p_{i,j}$, for $1 \leq j \leq C$ are the pixels categorized as i as a result of the classification of the image. Aside from quantifying the number of pixels correctly assigned to each class, the presented confusion matrices have columns of: commission error, omission error, producer accuracy and user accuracy. Each of these is described in the following table:

Commission Error	What percentage of the pixels assigned to a class c_i are incorrect ($i \neq j$)?	$CE_{c_i} = \frac{\sum_{j=1}^{c_i-1} p_{c_i,j} + \sum_{j=c_i+1}^C p_{c_i,j}}{\sum_{j=1}^C p_{c_i,j}}$
Omission Error	What percentage of the pixels in a class c_j are not classified as such?	$OE_{c_j} = \frac{\sum_{i=1}^C p_{i,c_j} - p_{c_j,c_j}}{\sum_{i=1}^C p_{i,c_j}}$
Producer Accuracy	What percentage of the pixels in a class c_j are classified correctly?	$Prod. Acc_{c_j} = \frac{p_{c_j,c_j}}{\sum_{i=1}^C p_{i,c_j}}$
User Accuracy	What percentage of the pixels assigned to a class c_i are correct ($i = j$)?	$User. Acc_{c_j} = \frac{p_{c_i,c_i}}{\sum_{j=1}^C p_{c_i,j}}$

Table 5-1. Confusion Matrix description

To complement the information presented in the confusion matrices, the boat class (p_1) is selected as the desired target and the True Positive Rate (TPR) and False Positive Rate (FPR) are calculated accordingly for each test and plotted on a Receiver Operating Characteristic (ROC) chart. Using the data from a confusion matrix the formula for the TPR and FPR are described in Table 5-2.

TPR	What percentage of the boat pixels are classified as boat?	$\frac{p_{1,1}}{p_{1,1} + p_{2,1}}$
FPR	What percentage of the sea pixels are classified as boat?	$\frac{p_{1,2}}{p_{1,2} + p_{2,2}}$

Table 5-2. ROC axis description.

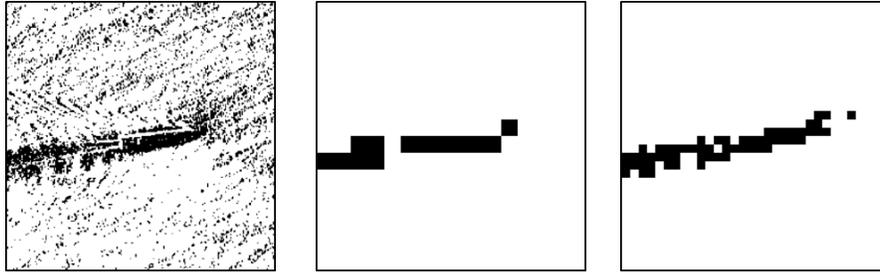
The ROC chart has 2 axis, a vertical TPR and a horizontal FPR , with $0 \leq TPR \leq 1$ and $0 \leq FPR \leq 1$. To achieve some additional clarity, the range on these axes is varied in each of the ROCs displayed. The ideal receiver or classifier on the ROC chart is located at the top left corner with a 100% TPR and a FPR of 0%. The ROC chart displays the results of each configuration relative to each other in a visual manner, making it possible to compare them at a glance by their position relative to the ideal case (0,1).

Aside from the usefulness held by the recovered images in drawing conclusions on the complete image, running times were recorded in seconds using Matlab's "*tic*", "*toc*" function as estimates of the computational cost for each reconstruction stage in order to draw conclusions on the possible existence of tradeoffs or compromises.

The tests consist of image compression using measurement matrices constructed from two smaller ones as described in Eq. 4-16, using section sizes of 16 and 8 pixels ($S = 16$ and $S = 32$ respectively). The second measurement matrices A_2 are used to perform the partial reconstruction stage as described in sections 4.3.5 and these partial reconstructions are used to recover the complete image with the measurement matrices A_1 as described in section 4.3.6.

5.1 Partial Reconstruction

Testing on the partially reconstructed image is meant to determine how good they are at providing insight to the content of the fully recovered image. The partially reconstructed images IPR are then subjected to a unsupervised k -means classifier and compared to $GT_A_S_16$ and $GT_B_S_32$ for the case of $S = 16$ and 32 respectively. These two reference images that are used as ground truths are the result of classifying the original image using the k -means classifier, and then assigning each section to the class with the highest presence in it.



From left to right: *GT_RGB*, *GT_A_S_16* and *GT_B_S_32* (all are scaled for detail).

Figure 5-2. Ground Truth Images for IPR

For the different tests conducted the partially recovered image was used to classify the signal and later compared to the classification done using the partially compressed image. The levels of accuracy for each class are used as an indicator on how close analysis done on the partially reconstructed version is to the expected analysis. Each confusion matrix is shown in the Table 5-3 to Table 5-10.

$S1 = 16, T1 = D, M1 = 140, T2 = D, M2 = 90, \text{Accuracy} = 198 / 256 = 77.34$

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	12	57	57 / 69	1 / 13	12 / 13	12 / 69	69
Sea	1	186	1 / 187	57 / 243	186 / 243	186 / 187	187
Total GT	13	243					

Table 5-3. Confusion Matrix for IPR: $S_{16}T_{1D}M_{1140}T_{2D}M_{290}$.

$S1 = 16, T1 = D, M1 = 140, T2 = W, M2 = 80, \text{Accuracy} = 227 / 256 = 88.67$

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	13	29	29 / 42	0 / 13	13 / 13	13 / 42	42
Sea	0	214	0 / 214	29 / 243	214 / 243	214 / 214	214
Total GT	13	243					

Table 5-4. Confusion Matrix for IPR: $S_{16}T_{1D}M_{1140}T_{2W}M_{280}$.

$S1 = 16, T1 = W, M1 = 160, T2 = D, M2 = 80, \text{Accuracy} = 183 / 256 = 71.484375$

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	13	73	73 / 86	0 / 13	13 / 13	13 / 86	86
Sea	0	170	0 / 170	73 / 243	170 / 243	170 / 170	170
Total GT	13	243					

Table 5-5. Confusion Matrix for IPR: $S_{16}T_{1W}M_{1160}T_{2D}M_{280}$.

$S1 = 16, T1 = W, M1 = 160, T2 = W, M2 = 80, \text{Accuracy} = 198 / 256 = 77.34$

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	13	58	58 / 71	0 / 13	13 / 13	13 / 71	71
Sea	0	185	0 / 185	58 / 243	185 / 243	185 / 185	185
Total GT	13	243					

Table 5-6. Confusion Matrix for IPR: $S_{16}T_{1W}M_{1160}T_{2W}M_{280}$.

S1 = 32, T1 = D, M1 = 59, T2 = D, M2 = 456, Accuracy = 995 / 1024 = 97.17

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	56	26	26 / 82	3 / 59	56 / 59	56 / 82	82
Sea	3	939	3 / 942	26 / 965	939 / 965	939 / 942	942
Total GT	59	965					

Table 5-7. Confusion Matrix for IPR: S_32_T1_D_M1_59_T2_D_M2_456.

S1 = 32, T1 = D, M1 = 59, T2 = W, M2 = 434, Accuracy = 1009 / 1024 = 98.53515625

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	52	8	8 / 60	7 / 59	52 / 59	52 / 60	60
Sea	7	957	7 / 964	8 / 965	957 / 965	957 / 964	964
Total GT	59	965					

Table 5-8. Confusion Matrix for IPR: S_32_T1_D_M1_59_T2_W_M2_434.

S1 = 32, T1 = W, M1 = 60, T2 = D, M2 = 437, Accuracy = 986 / 1024 = 96.29

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	55	34	34 / 89	4 / 59	55 / 59	55 / 89	89
Sea	4	931	4 / 935	34 / 965	931 / 965	931 / 935	935
Total GT	59	965					

Table 5-9. Confusion Matrix for IPR: S_32_T1_W_M1_60_T2_D_M2_437.

S1 = 32, T1 = W, M1 = 60, T2 = W, M2 = 426, Accuracy = 1007 / 1024 = 98.34

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	53	11	11 / 64	6 / 59	53 / 59	53 / 64	64
Sea	6	954	6 / 960	11 / 965	954 / 965	954 / 960	960
Total GT	59	965					

Table 5-10. Confusion Matrix for IPR: S_32_T1_W_M1_60_T2_W_M2_426.

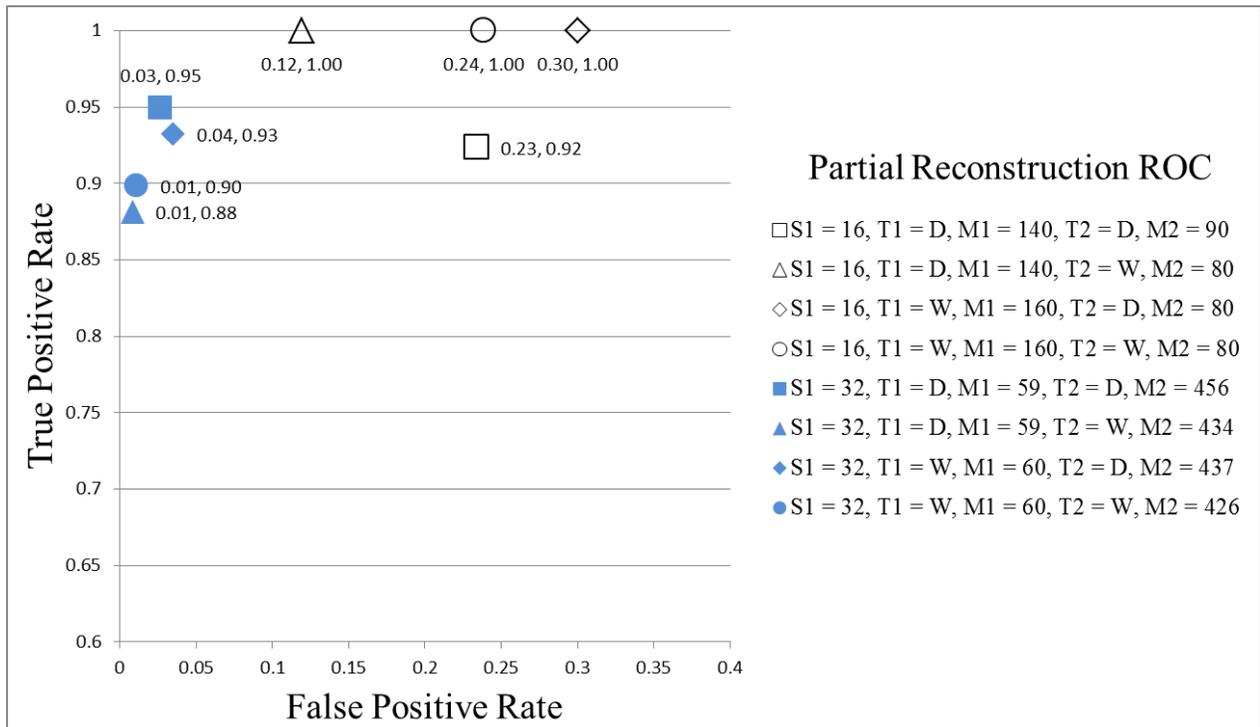


Figure 5-3. Location on a ROC of IPR classification

5.2 Full Reconstruction

Being able to gain some foresight into the contents of the signal before complete reconstruction would hold no value if the reconstructed signal has no liking to the original. It is therefore also important to compare the fully reconstructed image to the original to be sure that it can be used for classification or detection within certain margins of error. First the partially reconstructed images and the measurement matrices A_2 are used to fully recover the original image following the procedure described in section 4.3.6. Again using the classification accuracy and error as an indicator of the we present the confusion matrices for each test (Table 5-11 to Table 5-14) as well as their location on the ROC chart (Figure 5-4). This ROC chart also shows the location of a signal compressed by only taking the highest values of the sparse representation.

S1 = 16, T1 = D, M1 = 140, T2 = D, M2 = 90, Accuracy = 50746 / 65536 = 77.43

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	23511	8033	8033 / 31544	6757 / 30268	23511 / 30268	23511 / 31544	31544
Sea	6757	27235	6757 / 33992	8033 / 35268	27235 / 35268	27235 / 33992	33992
Total GT	30268	35268					

Table 5-11. Confusion Matrix for IR: S_16_T1_D_M1_140_T2_D_M2_90.

S1 = 16, T1 = W, M1 = 160, T2 = D, M2 = 80, Accuracy = 51642 / 65536 = 78.80

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	23591	7217	7217 / 30808	6677 / 30268	23591 / 30268	23591 / 30808	30808
Sea	6677	28051	6677 / 34728	7217 / 35268	28051 / 35268	28051 / 34728	34728
Total GT	30268	35268					

Table 5-12. Confusion Matrix for IR: S_16_T1_W_M1_160_T2_D_M2_80.

S1 = 32, T1 = D, M1 = 59, T2 = D, M2 = 456, Accuracy = 60575 / 65536 = 92.43

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	28497	2754	2754 / 31251	2207 / 30704	28497 / 30704	28497 / 31251	31251
Sea	2207	32078	2207 / 34285	2754 / 34832	32078 / 34832	32078 / 34285	34285
Total GT	30704	34832					

Table 5-13. Confusion Matrix for IR: S_32_T1_D_M1_59_T2_D_M2_456.

S1 = 32, T1 = W, M1 = 60, T2 = D, M2 = 437, Accuracy = 59997 / 65536 = 91.55

	Boat	Sea	Commission	Omission	Prod. Acc.	User Acc.	Total Class
Boat	28204	3039	3039 / 31243	2500 / 30704	28204 / 30704	28204 / 31243	31243
Sea	2500	31793	2500 / 34293	3039 / 34832	31793 / 34832	31793 / 34293	34293
Total GT	30704	34832					

Table 5-14. Confusion Matrix for IR: S_32_T1_W_M1_60_T2_D_M2_437.

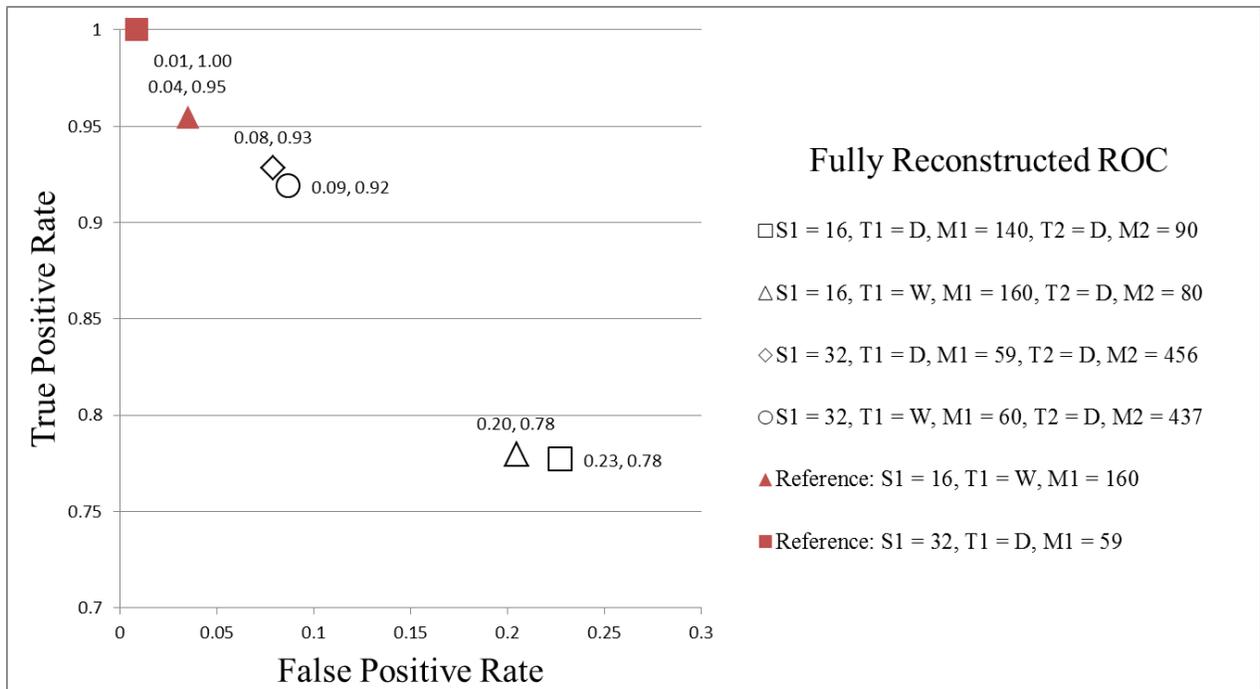


Figure 5-4. Location on a ROC of *IR* classification

6. CONCLUSIONS AND FUTURE WORK

Hardware limitations made it impossible to execute a single stage reconstruction of the image, leaving us wanting for empirical data on the processing time of said approach. It was therefore impossible to compare processing times of the recursive and direct reconstruction. However the hardware limitation does allow us to draw the conclusion that the proposed method can be executed in systems with lower processing power (system memory) than that required for direct or traditional CS reconstruction.

$\frac{H}{W}$	M_1	$\frac{M_1}{H*W}\%$	2DT	$\frac{IPR_b}{\sphericalangle ISR_b}$	Overall Accuracy	S_1 S_2	M_2	$\frac{M_2}{S_1*S_2}\%$	2DT	$\frac{IPC_b}{\sphericalangle IPR_b}$	Overall Accuracy	$\frac{M}{N}$	$IC_b \sphericalangle ISR_b$
16	140	54,68	DCT	4.69	77.43	16	90	35.16	DCT	1.97	86.33	19.23	4429.32
			80				31.25	WLT	1.28	93.75	17.09	4139.52	
	160	62,5	WLT	6.34	78.8	16	80	31.25	DCT	1.61	76.56	19.53	5641.92
			80				31.25	WLT	1.48	80.86	19.53	5579.52	
8	59	92,19	DCT	0.04	84.78	32	456	44.53	DCT	303.3	97.27	41.05	53812.29
			434				42.38	WLT	539.1	97.27	39.07	95547.12	
	60	93,75	WLT	0.04	84.39	32	437	42.68	DCT	285	96.88	40.01	51424.68
			426				41.6	WLT	532.1	99.02	39	95899.08	

Table 6-1: Summary of the results from the tests conducted.

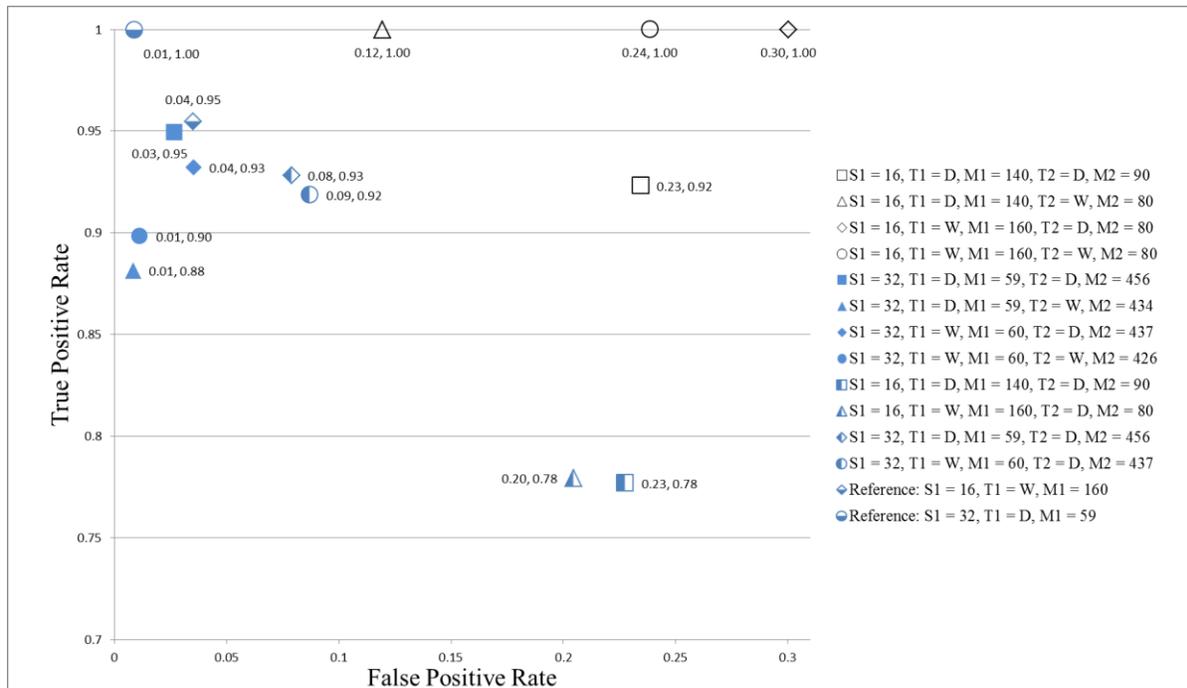


Figure 6-1: ROC chart comparison.

The results on the recursive implementation from the previous chapter can be condensed in Table 6-1 and Figure 6-1. The side by side comparison of the data from the previous table as well as the visual indication given by the ROC charts allows us to draw the following conclusions:

- The selection of S_1 and S_2 have the greatest impact on the operation of the whole system as they affect the extent to which conclusions can be drawn from the partially reconstructed image (measured through classification accuracy), as well as the processing time required to achieve said reconstruction when starting from the fully compressed measurements. The results obtained from the test with $S = 32$ produced higher levels of accuracy and better ROC charts, but took significantly longer than the tests with $S = 16$.
- The compression level also affects the quality of the analysis done on the partially compressed image as well as the processing time required for reconstruction. Lower levels of compression account for better levels of accuracy when classifying the contents of the signal. However, there is an inherent desire in any classification application to achieve the maximum reduction possible, so it makes more sense to tweak the system by adjusting the values of S_1 and S_2 while trying to keep M_1 and M_2 as low as possible.
- The complete system is highly parallelizable: the first recovery stage can be implemented simultaneously on M_1 platforms in order to obtain the partially reconstructed images, after which the second reconstruction stage could be processed in as many as $S_1 * S_2$ instances at the same time. The only bottle neck is the requirement of the totality of the results from the first reconstruction stage in order to further any analysis.
- The different reconstruction stages are independent of the transformation or sparsifying basis as long as the signal has a greater level of sparsity in said basis than that which was initially determined to be the minimum level. For example, in the cases where the number

of samples was determined using the DCT2 but the 2-D Wavelet transform presented a higher level of sparsity; the DCT2 could be switched for the 2-D Wavelet transform and achieve normal operation.

The tests conducted and conclusions drawn above would seem to guide any immediate development in this research towards achieving higher levels of efficiency, this could be done by implementing parallel processing in the reconstruction stages, as well as using finished reconstructions as starting points for the optimization algorithms (use the result from *IPR* as a starting point for *IPR*, the closer the starting point is to the optimum value the faster the algorithms will reach a solution) or testing other optimization algorithms that might seem better suited.

The literature on CS also displays algorithms [47] and methods suited for applications on noisy signals which might be better suited for real world applications like the single pixel camera.

Farther down the road it would also seem of interest to: add layers to the compression and reconstruction in an effort to increase the number of recursions and examine its effect on processing times and analysis of preliminary versions of the recovered signal; test the methods described on an implementation of the single pixel camera using a HSI sensor; or test the worth of the application of recursive CS to other types of signals aside from HSI.

7. REFERENCES

- [1] D. Donoho and J. Tanner, "Neighborliness of Randomly-Projected Simplices in High Dimensions", *Proceedings of the National Academy of Sciences of the United States*, vol. 102, no. 27, pp. 9452-9457, March 2005.
- [2] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly and R. G. Baraniuk, "Single-Pixel Imaging Via Compressive Sampling", *IEEE Signal Processing Magazine*, pp. 83-91, March 2008.
- [3] D. Donoho and M. Elad, "Optimally Sparse Representation in General (Non-Orthogonal) Dictionaries Via L1 Minimization", *Proceedings of the National Academy of Sciences of the United States*, vol. 100, no. 5, pp. 2197-2202, March 2003.
- [4] P. Tseng, "Further Results on Stable Recovery of Sparse Over Complete Representations in the Presence of Noise", *IEEE Transactions on Information Theory*, vol. 55, no. 2, pp. 888-899, February 2009.
- [5] D. Donoho and X. Huo, "Uncertainty Principles and Ideal Atomic Decomposition", *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845-2862, November 2001.
- [6] D. Donoho, "Sparse Components of Images and Optimal Atomic Decompositions", *Constructive Approximation*, vol. 17, no. 3, pp. 353-382, 2001.
- [7] E. Candes and T. Tao, "Decoding by Linear Programming," *IEEE Trans. on Information Theory*, vol. 51, no. 12, pp. 4203-4215, December 2005.
- [8] D. Wei and O. Milenkovic, "Subspace Pursuit for Compressive Sensing Signal Reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230-2249, May 2009.

- [9] E. J. Candes and T. Tao, "Near-Optimal Signal Recovery from Random Projections: Universal Encoding Strategies?", *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406-5425, December 2006.
- [10] E. J. Candès and M. B. Wakin, "An Introduction to Compressive Sampling", *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21-30, March 2008.
- [11] G. Taubock and F. Hlawatsch, "A Compressed Sensing Technique for OFDM Channel Estimation in Mobile Environments: Exploiting Channel Sparsity for Reducing Pilots", in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, Las Vegas, 2008, pp. 2885-2888.
- [12] E. W. Tramel and J. E. Fowler, "Video Compressed Sensing with Multihypothesis", in *Data Compression Conference, DCC*, Snowbird, 2011, pp. 193-202.
- [13] L. Streeter, M. Cree, R. Künnemeyer and G. R. Burling-Claridge, "Comparison of Hadamard Imaging and Compressed Sensing for Low Resolution Hyper spectral Imaging", in *Image and Vision Computing New Zealand, IVCNZ*, Christchurch, 2008, pp. 1-6.
- [14] E. Candes, "The Restricted Isometry Property and Its Implications for Compressed Sensing", *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589-592, May 2008.
- [15] J. R. Schott, *Remote Sensing: The Image Chain Approach, 2nd ed.* Oxford University Press, 2007.
- [16] R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing, 3rd ed.* Academic Press, 2007.
- [17] X. Weiyu and H. Babak, "Compressed Sensing Over the Grassmann Manifold: A Unified Analytical Framework", in *46th Annual Allerton Conference on Communication, Control, and Computing*, Urbana-Champaign, IL, 2008, pp. 562-567.

- [18] M. Stojnic, B. Hassibi and X. Weiyu, "Compressed Sensing of Approximately Sparse Signals", in *IEEE International Symposium on Information Theory*, Toronto, 2008, pp. 2182-2186.
- [19] R. Baraniuk, M. Davenport, R. DeVore and M. Wakin, "A Simple Proof of the Restricted Isometry Property for Random Matrices", *Constructive Approximation*, vol. 28, no. 3, pp. 253-263, 2008.
- [20] J. D. Blanchard, C. Cartis and J. Tanner, "Compressed Sensing: How Sharp is The Restricted Isometry Property?", *SIAM Review*, vol. 53, no. 1, pp. 105-125, 2011.
- [21] A. Cohen, W. Dahmen and R. DeVore, "Compressed Sensing and Best k-term Approximation," *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 211-231, 2009.
- [22] J. Gillan, "Airborne Visible/Infrared Imaging Spectrometer", Landscape Toolbox Wiki. Internet: http://wiki.landscapetoolbox.org/doku.php/remote_sensor_types:airborne_visible_infrared_imaging_spectrometer [09 March 2014].
- [23] C. E. Shannon, "Communication in the Presence of Noise", *Proceedings of the IEEE*, vol. 86, no. 2, pp. 447-457, February 1998.
- [24] S. G. Mallat and Z. Zhang, "Matching Pursuit With Time-Frequency Dictionaries", *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397-3415, December 1993.
- [25] M. Unser, "Sampling-50 Years After Shannon", *Proceedings of the IEEE*, vol. 88, no. 4, pp. 569-587, April 2000.
- [26] J. Vieira and W. Huiyuan, "2-D Wavelet Transforms in the Form of Matrices and Application in Compressed Sensing", in *World Congress on Intelligent Control and Automation*, Jinan, 2010, pp. 35-39.
- [27] S. S. Chen, D. L. Donoho and M. A. Saunders, "Atomic Decomposition by Basis Pursuit", *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33-61, 1998.

- [28] M. Pueschel and J. M. Moura, "Algebraic Signal Processing Theory", Cornell University. Internet: <http://arxiv.org/pdf/cs/0612077v1.pdf> [9 March 2014].
- [29] N. Ahmed, T. Natarajan and K. R. Rao, "Discrete Cosine Transform", *IEEE Transactions on Computers*, Vols. C-23, no. 1, pp. 90-93, 1974.
- [30] Z. Jianqin and C. Ping, "Generalized Discrete Cosine Transform", in *Pacific-Asia Conference on Circuits, Communications and System*, Chengdu, 2009, pp. 449-452.
- [31] Y. Rivenson and A. Stern, "Practical Compressive Sensing of Large Images", in *International Conference on Digital Signal Processing*, Santorini-Hellas, 2009, pp. 1-8.
- [32] E. Candes and J. Romberg, "Sparsity and Incoherence in Compressive Sampling", *Inverse Problems*, vol. 23, no. 3, pp. 969-985, 2007.
- [33] E. A. Grissom and P. K. Rajan, "An algorithm for fast two-dimensional discrete cosine transform", in *Southeastern Symposium on System Theory*, Cookeville, TN, 1990, pp. 646-651.
- [34] M. Misiti, Y. Misiti, G. Oppenheim and J.-M. Poggi, *Matlab Wavelet Toolbox Users Guide, 4.13 ed.* Natick, MA: The MathWorks, 2014.
- [35] M. Sifuzzaman, M. R. Islam and M. Z. Ali, "Application of Wavelet Transform and its Advantages Compared to Fourier Transform", *Journal of Physical Science*, vol. 13, pp. 121-134, 2009.
- [36] M. A. Davenport, J. N. Laska, P. T. Boufounos and R. G. Baraniuk, "A Simple Proof that Random Matrices are Democratic", ECE Department, Rice University, Houston, Technical Report, November 2009.
- [37] J. Wright, M. Yi, J. Mairaly, G. Sapiro, T. S. Huang and S. Yan, "Sparse Representation For Computer Vision and Pattern Recognition", *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031-1044, 2010.

- [38] Y. C. Pati, R. Rezaifar and P. S. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition", in *Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 1993, pp. 40-44.
- [39] G. M. Davis, S. G. Mallat and Z. Zhang, "Adaptive Time-Frequency Decompositions", *Optical Engineering*, vol. 33, no. 7, pp. 2183-2191, 2005.
- [40] R. A. Devore and V. N. Temlyakov, "Some Remarks on Greedy Algorithms", *Advances in Computational Mathematics*, vol. 5, no. 1, pp. 173-187, 1996.
- [41] G. B. Dantzig, *Linear Programming and Extensions*. Princeton, NJ: Princeton University Press, 1965.
- [42] P. Gill, W. Murray and M. H. Wright, *Numerical Linear Algebra and Optimization*. Redwood City, CA: Addison-Wesley, 1991.
- [43] I. J. Lustig, R. E. Marsten and D. F. Shanno, "Interior Point Methods for Linear Programming: Computational State of the Art", *ORSA Journal on Computing*, vol. 6, no. 1, pp. 1-14, 1994.
- [44] D. L. Donoho, "Unconditional Bases are Optimal Bases for Data Compression and for Statistical Estimation", *Applied and Computational Harmonic Analysis*, vol. 1, no. 1, pp. 100-115, 1993.
- [45] M. B. Wakin, J. Laska, M. F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly and R. G. Baraniuk, "An Architecture for Compressive Imaging", in *IEEE International Conference on Image Processing*, Atlanta, GA, 2006, pp. 1273-1276.
- [46] D. Takhar, J. N. Laska, M. B. Wakin, M. F. Duarte, D. Baron, S. Sarvotham, K. F. Kelly and R. G. Baraniuk, "A New Compressive Imaging Camera Architecture Using Optical-Domain Compression," in *Proceeding of Computational Imaging*, San Jose, CA, 2006.
- [47] K. M. Cheman, *Optimization Techniques for Solving Basis Pursuit Problems*. Masters of Applied Mathematics thesis, North Carolina State University , Raleigh, NC, 2006.

- [48] Grupo De Inteligencia Computacional, "Hyper spectral Remote Sensing Scenes", Universidad del País Vasco. Internet: http://www.ehu.es/ccwintco/index.php/Hyper_spectral_Remote_Sensing_Scenes [10 March 2014].
- [49] A. K. Jain, "Data Clustering: 50 Years Beyond K-Means", *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651-666, 2010.
- [50] J. Burke, "Math 407 Linear Programming Course Notes on Duality Theory", University of Washington, Department of Mathematics. Internet: <http://www.math.washington.edu/~burke/crs/407/notes/section4.pdf> [10 March 2014].
- [51] E. J. Candes, J. K. Romberg and T. Tao, "Stable Signal Recovery from Incomplete and Inaccurate Measurements", *Communications on Pure and Applied Mathematics*, vol. 59, pp. 1207-1223, 2006.
- [52] S. Mallat, "Sparse Representations" in *A Wavelet Tour of Signal Processing: The Sparse Way 3rd ed.* Burlington, MA: Academic Press, 2008.
- [53] C. C. Wen-Hsiung, H. Smith and S. C. Fralick, "A Fast Computational Algorithm for the Discrete Cosine Transform", *IEEE Transactions on Communications*, vol. 25, no. 9, pp. 1004-1009, 1977.
- [54] E. Candes and J. Romberg, "Signal Recovery from Random Projections," in *SPIE, Computational Imaging III*, San Jose, CA, 2005.
- [55] M. Feng, J. E. Mitchell, J. S. Pang, X. Shen and A. Wachter, "Complementarity Formulations of l_0 -norm Optimization Problems", 2013.