2012

Applications Of Compressive Sensing To Surveillance Problems

Christopher Huff
University of Central Florida

APPLICATIONS OF COMPRESSIVE SENSING TO SURVEILLANCE PROBLEMS

by

CHRISTOPHER HUFF
B.S. University of Central Florida, 2010

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Mathematics
in the College of Sciences
at the University of Central Florida
Orlando, Florida

Spring Term
2012

Major Professors: Ram Mohapatra and Qiyu Sun

ABSTRACT

In many surveillance scenarios, one concern that arises is how to construct an imager that is capable of capturing the scene with high fidelity. This could be problematic for two reasons: first, the optics and electronics in the camera may have difficulty in dealing with so much information; secondly, bandwidth constraints, may pose difficulty in transmitting information from the imager to the user efficiently for reconstruction or realization.

In this thesis, we will discuss a mathematical framework that is capable of skirting the two aforementioned issues. This framework is rooted in a technique commonly referred to as *compressive sensing*. We will explore two of the seminal works in compressive sensing and will present the key theorems and definitions from these two papers. We will then survey three different surveillance scenarios and their respective compressive sensing solutions. The original contribution of this thesis is the development of a distributed compressive sensing model.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Recent advances in technology have brought with them a great capacity for storing large amounts of data. With data sets becoming increasingly large, it is becoming difficult to analyse the data in order to make use of it. As an example, consider a network of surveillance cameras monitoring a particular area. If the number of cameras is large, it would be difficult to have a small group of people monitor them carefully. To remedy this situation one may want to have a computer program to monitor the data and tell the user when a particular event of interest is happening in the scene. An immediate issue that one would encounter in such a scenario (in addition to many computer vision related obstacles) would be that of the program parsing the large amount of video data quickly enough so as to alert the user of an event in a timely manner.

Another situation in which a great amount of data is difficult to manage can be found in signal transmission. Suppose one wanted to construct UAV (unmanned aerial vehicle) with the capability of being able to capture (very) high resolution video of the events happening on the ground below it. Assuming the UAV is able to have such a sensor attached to it (this is not a trivial consideration) the data collected by the UAV must be transmitted in order to be of use. This transmission may not always be possible, since the transmission channel will have limited bandwidth.

These two examples are among many where the a large amount of data is needed for a task, but that amount is too great to manage. This motivates one to ask the questions: is there structure in the data set that I am interested in? If so, may I exploit that structure in order to make the data easier to use?

Depending on the data one is interested in, the answers to the above questions will vary. Recently there has been a great deal of work in dealing with data sets which exhibit a characteristic, now known as sparsity. We say that a data set is sparse if most of the values in that data set are zero, or so close to zero so as to have little contribution to the overall information of the data. As a frivolous example, consider the vector

$$[1\;0\;0\;0\;1\;1\;1\;0\;0\;1\;0\;0\;0\;1\;0\;0] \tag{1.1}$$

Suppose we were interested in sensing this vector so that we may transmit it to a user. The vector has 16 entries, but only 6 of the entries are non-zero. This means that the information contained in the vector only depends on the location of the 6 non-zero entries, not the values in all 16 entries. This suggests that one may want to sense all 16 entries and then transmit the locations of the non-zero vectors. The problem with this approach is that it requires one to sense all of the data, parse all of it, and then determine the locations of the non-zero entries. This process involves many calculations, which is not desirable. This begs the following question: if we knew a priori that the vector of interest was sparse, could we take a small number of measurements and then transmit them to the user in a way such that the user could reconstruct the vector from the measurements provided? This would mean that the UAV would not be tasked with the computations described earlier.

This question has been answered affirmatively by using a technique known as compressive sensing [ET06] . The idea behind compressive sensing is as follows: given a signal $x \in \mathbb{R}^n$, one may capture $m << n$ linear measurements $y \in \mathbb{R}^m$ of x and then accurately reconstruct the original signal from y. There are conditions that must be levied on the measurement process and the signal of interest must be sparse; but with these two requirements met, compressive sensing allows one to sense and compress the signal simultaneously.

In this work we will be interested in dealing with digital images and video. It has been known for some time that natural images and videos are compressible (we will define this precisely later). This essentially means that images and videos may be represented sparsely in some basis. With the knowledge of this sparsity in hand, one needs only to devise a sensing scheme which is consistent with the theory of compressed sensing in order to enable accurate reconstruction from dramatically under-sampled data.

The work that follows is organized in the following way: first we will review some of the mathematical results which provide support for much of the literature dealing will compressive sensing. Second, we will look at some applications of compressive sensing to surveillance problems. This part of the work will demonstrate different ways in which one may find sparsity in a problem and different algorithms which may be applied a given problem. The fourth and final chapter will conclude this work with further research questions and possible directions to their solutions.

CHAPTER 2

BACKGROUND

2.1 The Mathematics of Compressive Sensing

In this section we will survey two of the most important works which developed compressive sensing into a rigorous mathematical theory. The first work we will present is entitled *Stable Signal Recovery from Incomplete and Inaccurate Measurements* [ET06], while the second is entitled *Near Optimal Signal Recovery From Random Projections: Universal Encoding Strategies?* [CT06] The former used the restricted isometry property to prove that measurements with additive noise could still be used to recover the original signal with reasonable error. The latter established the fact that compressible signals could be recovered from compressive measurements efficiently

2.1.1 Recovery From Noisy Measurements

Suppose we wish to recover a sparse vector $x_o \in \mathbb{R}^m$ from incomplete measurements $y \in \mathbb{R}^n$, $n << m$, which are subject to additive noise, e, such that $||e||_2 \leq \epsilon$. That is, $y = Ax_o + e$, where A is a matrix whose columns are the codes against which x_o is inner producted with to produce linear measurements/observations of x_o. the above problem is considered in the paper *Stable Signal Recovery from Incomplete and Inaccurate Measurements*.

The key contributions of the paper are two-fold: first, that paper was amongst the first to introduce an error model into the sparse recovery problem. Second, that paper contains a theorem which bounds the error of the recovery by a multiple of the l^2 norm of e. Before we state the major result of that paper, we need to develop the concept of the restricted isometry property.

Let $T \subset \{1, .., m\}$. Let A_T be the $n \times |T|$ submatrix of A obtained by keeping only the columns of A which correspond to the indices in T. Then we may define the S-restricted isometry constant δ_S for A which is the smallest quantity such that

$$(1 - \delta_S)||c||_2^2 \leq ||A_T c||_2^2 \leq (1 + \delta_S)||c||_2^2 \tag{2.1}$$

for all subsets T with $|T| \leq S$ and vectors $c \in \mathbb{R}^T$. We say that matrices which have an associated restricted isometry constant exhibits the restricted isometry property (RIP). With these definitions and notation in mind, we may now state the major result from [ET06]:

Theorem 1: Let S be such that $\delta_{3S} + 3\delta_{4S} < 2$. Then for any signal x_o with sparsity less that s and any perturbation e with $||e||_2 \leq \epsilon$, the solution $x^\#$ to the minimization problem:

$$\min ||x||_1 \text{subject to } ||Ax - y||_2 \leq \epsilon \tag{2.2}$$

obeys

$$||x^{\#} - x_o||_2 \leq C_S \cdot \epsilon, \tag{2.3}$$

where the constant C_S may only depend on δ_{4S}.

This theorem was is important due to the stability and error estimate provided for robustly recovering a sparse signal with additive noise.

2.1.2 Recovering A Compressible Signal

In the above theorem we have assumed that the signal of interest x_o was sparse in the canonical basis. This is not a reasonable assumption for many signals, such as natural images. To appeal to compressive sensing in the context of image acquisition we will make use of transform coding.

Suppose $I \in \mathbb{R}^m$ denotes a vectorized natural image. Then we may represent I as a sparse linear linear combination of appropriately chosen vectors. That is,

$$I = \Psi x, \tag{2.4}$$

where x is S-sparse. This representation introduces a sparse vector, but it is still not clear as to how to apply the results of compressive sensing. A reasonable question that one may ask is, for what matrix A of test vectors can we use so that the product $A\Psi = \Theta$ exhibits the RIP?

If we had such a matrix, then we would have that the solution to the minimization problem

$$\min \|x\|_1 \text{subject to } \Theta x - y = 0 \tag{2.5}$$

is the sparsest solution. We could then recover the original image via $I = \Psi x$. The answer of the above question was addressed in the paper *Near Optimal Signal Recovery From Random Projections: Universal Encoding Strategies?*

That work addressed signals whose coefficients decay like a power law in some basis. That is, if $\Psi = (\psi_j)_{j=1,...,N}$ is an orthonormal basis and $I \in \mathbb{R}^N$ is the signal of interest. Let $x_j =< I, \psi_j >$ and let us sort the vector x according to the magnitude of its elements so that $|x_k| \geq |x_{k-1}| \geq ... \geq |x_1|$. We say that x decays like a power law if there exists $C > 0$ such that

$$|x_k| \leq C \cdot k^{-1/p}, 1 \leq k \leq N \tag{2.6}$$

If p is sufficiently small ($0 \leq p \leq 1$) then we say that I is compressible.

With this type of signal in mind, we are then introduced to two principals which the measurement matrix (the role assumed by A) is to obey: the uniform uncertainty principle (UUP) and the exact reconstruction principle (ERP). Suppose that $1 \leq k \leq N$ and $\Omega = \{1, ..., k\}$. Then we will suppose that the measurement matrix $A = A_\Omega$ is a random matrix of dimension $|\Omega|$ by N. Let the number of measurements $|\Omega|$ be a random variable and denote the expected value of $|\Omega|$ by K. Further still, let R_T denote the restriction map from $\mathbb{R}^N$ to a set $T \subset \mathbb{R}^N$. Then we may define $R_T^* : T \to \mathbb{R}^N$ as the function which inserts zeros outside of T (if $x \in \mathbb{R}^N$, then $\text{supp}(R_T^* x) \subset T$). Let $A_{\Omega T} := A_\Omega R_T^*$. Then $A_{\Omega T}$ is an $|\Omega|$ by $|T|$ matrix obtained by extracting $|T|$ columns from A_Ω , where the j^{th} column is chosen if $j \in T$.

The Uniform Uncertainty Principal (UUP) [CT06]: We say that the measurement matrix A obeys the uniform uncertainty principle with oversampling factor λ if for every sufficiently small $\alpha > 0$, the following statement is true with probability greater than or equal to $1 - O(N^{-\rho/\alpha})$ for some fixed $\rho > 0$: for all subsets T such that

$$|T| \leq \alpha \cdot K/\lambda, \tag{2.7}$$

the matrix A obeys the bounds

$$1/2 \cdot K/N \leq \lambda_{min}(A_{\Omega T} * A_{\Omega T}) \leq \lambda_{max}(A_{\Omega T} * A_{\Omega T}) \leq 3/2 \cdot K/N. \tag{2.8}$$

The Exact Reconstruction Principle (ERP) [CT06]: We say that the measurement matrix A obeys the exact reconstruction principle with oversampling factor if for all sufficiently small $\alpha > 0$, each fixed subset T obeying (equation number) and each sign vector σ defined on T, $|\sigma(t)| = 1$ if $t \in T$, there exists with probability greater than $1 - O(N^{-\rho/\alpha})$ a vector $P \in \mathbb{R}^N$ with the following properties:

1. $P(t) = \rho(t)$, for all $t \in T$.

2. P is a linear combination of rows from A.

3. $|P(t)| \leq 1/2$ for all $t \in T^c$

Now that we may describe a measurement matrix A with the UUP and ERP, we may formally state the theorem which will allow us to use sparse representation to recover compressible signals.

Theorem 2 [CT06]. Let F be a measurement matrix such that the UUP and ERP hold with oversampling factors λ_1 and λ_2, respectively. Let $\lambda = \max(\lambda_1, \lambda_2)$ and assume that $K \geq \lambda$. Suppose I is a signal satisfying the compression inequality for some fixed $0 < p < 1$, and let $r := 1/p - 1/2$. Then for any sufficiently small $\alpha > 0$, any minimizer $x^{\#}$ to the problem (2.4) will obey,

$$||x - x^{\#}||_2 \leq C_{p,\alpha} \cdot R \cdot (K/\lambda)^{-r} \tag{2.9}$$

with probability $1 - O(N^{-\rho/\alpha})$.

This theorem, together with the fact that Gaussian measurement matrices obey the UUP and ERP with $\lambda = log(N)$, enables us to consider the reconstruction of large classes of signals which are sparse in some orthonormal basis. This includes the class of natural images, which are sparse in a wavelet basis. Videos may be regarded as sequences of images, and hence these results enable us to address problems of capturing videos as well.

2.2 Review of Compressive Sensing Literature

The most cited architecture for a compressive sensing camera is the single pixel camera (SPC). The SPC was developed by a team at Rice University [MB06]. The camera works as follows: light from the scene is focused through a biconvex lens onto a micro-mirror array. The mirrors in this array focus different pieces of the incoming light away from and directly at a single sensor which aggregates all of the light and renders a single value/measurement. At each instance the mirrors change their configuration so that (potentially) different set of light rays are focused on the sensor. The mirrors may be programmed to adjust themselves to be in line with any configuration. Because of this, a random Bernoulli configuration is often chosen. If we were to regard the Bernoulli configuration at each instance as a row of a matrix (vectorize the grid of mirrors), then the resulting matrix will be randomly determined with each of the rows being i.i.d. This means that the matrix will exhibit an RIP. Thus, measuring the scene with the SPC is consistent with the mathematical models in most compressive sensing literature. A graphical depiction from the University of Rice illustrates the measurement process well:

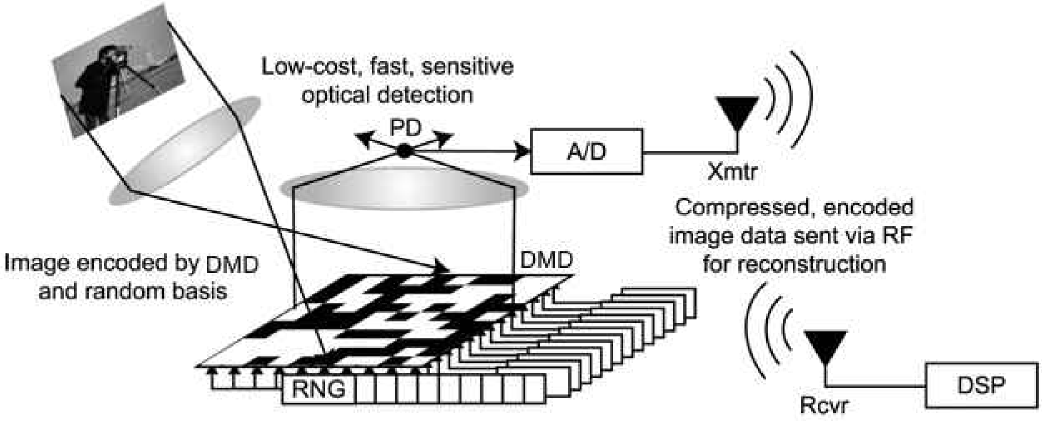

Figure 2.1: Diagram depicting how the SPC works. [MB06]

Many algorithms have been proposed to solve the l^1-minimization problem. Two popular algorithms among them are NESTA and CoSAMP [SC09] [NT08]. At its heart, NESTA is a gradient descent method. This algorithm is derived from the work of Nestrov [Nes83]. In his seminal work, Nestrov sought to minimize any sufficiently smooth convex function f on a convex set Ω_p, where the subscript is used to denote that this is the primal feasible set. Here when we say smooth, we mean that the function must be differentiable with its gradient obeying a Lipschitz condition. For the purposes of compressive sensing, our goal is to minimize $||x||_1$ subject to the constraint $||b - Ax||_2 \leq \epsilon$. The function $|| \cdot ||_1$ is convex, but not smooth. This means that we need to define a function which approximates $|| \cdot ||_1$ and is smooth.

Let us define a dual feasible set by

$$Q_d = \{u : \|u\|_\infty \le 1\}. \tag{2.10}$$

Then a reasonable such smooth function is

$$f_\mu(x) = \max_{u \in Q_d} \langle u, x \rangle - \frac{\mu}{2}\|u\|_2^2. \tag{2.11}$$

Defining $Q_p = \{x : \|b - Ax\|_2 \le \epsilon\}$, we may now reformulate the original compressive sensing minimization problem as

$$min_{x \in Q_p} f_\mu(x). \tag{2.12}$$

The NESTA algorithm may now be used to solve the above minimization problem. A rough sketch of the steps of the algorithm is as follows:

```
Algorithm 1: NESTA
input  : $x_0$
output: $\hat{x}$
$k = 0$;
while stopping criteria not met do
    $y_k \leftarrow \text{argmin}_{x \in Q_p} \frac{L_\mu}{2}\|x_k - x\|_2^2 + \langle \nabla f_\mu(x_k), x - x_k \rangle$;
    $z_k \leftarrow \text{argmin}_{x \in Q_p} \frac{L_\mu}{2}\|x - x_0\|_2^2 + \frac{\lambda}{2}\|b - Ax\|_2^2 + \langle \sum_{i \le k} \alpha_i \nabla f_\mu(x_i), x - x_k \rangle$;
    $x_k \leftarrow \tau_k z_k + (1 - \tau_k) y_k$.
```

A popular alternative to many of the gradient descent methods is known as CoSAMP. The CoSAMP algorithm is greedy in nature. The reasoning behind much of what CoSAMP is motivated by a single obstacle in signal recovery: determining the support of the largest (highest energy) components of the signal. To this end, CoSAMP estimates the support of an s-sparse signal, x, with the vector $y = A^*Ax$. The support of the s greatest components of y should be (approximately) the same as that of x. This is reasonable to assume when the sensing matrix A obeys the RIP with a small RIP constant ($\delta << 1$). After determining the estimated support of x, a least-squares estimation of x is constructed. Normally, the least-squares estimate of x would be rather inaccurate. However, we are able to use the estimated support of x and restrict the least-squares process to that set. Letting u denote the least-squares estimate, we let $a = u_s$, so that a contains only the s largest entries of u. The final steps include updating the samples and checking the halting criteria. With the convention that b denotes the noisy observations of x, a rough outline of the algorithm is presented now:

Algorithm 2: CoSAMP

```
input  : $A, b, s$
output: $\hat{x}$
$a_0 \leftarrow 0$;
$v \leftarrow u$;
$k \leftarrow 0$
while stopping criteria not met do
    $k \leftarrow k + 1$;
    $y \leftarrow A^*v$;
    $\Omega = supp(y_{2s})$;
    $T \leftarrow \Omega \cup \text{supp}(a_{k-1})$;
    $u|_T \leftarrow A_T^\dagger b$;
    $u|_{T^c} \leftarrow 0$;
    $a_k \leftarrow u_s$;
    $v \leftarrow b - Aa_k$;
```

One of the most noteworthy applications of compressive sensing is in the field of medical imaging. In *Sparse MRI: The Application of Compressed Sensing for Rapid MR Imaging*, the authors appealed to the sparsity of different types of MR images in order to lower image acquisition time or enhance image resolution via compressive sensing [LDP06]. While most MR images are sparse under a linear transformation [MP07], angiograms are sparse in image space. That is, most of the pixels in an angiogram are naturally take on very small or zero values, while a few take on high values. This means that one may randomly under-sample sample the angiogram and recover the image via an appropriate non-linear reconstruction algorithm. The authors used the SSFP angiogram data set [al04] to test their hypothesis and the results (see figure) are promising.

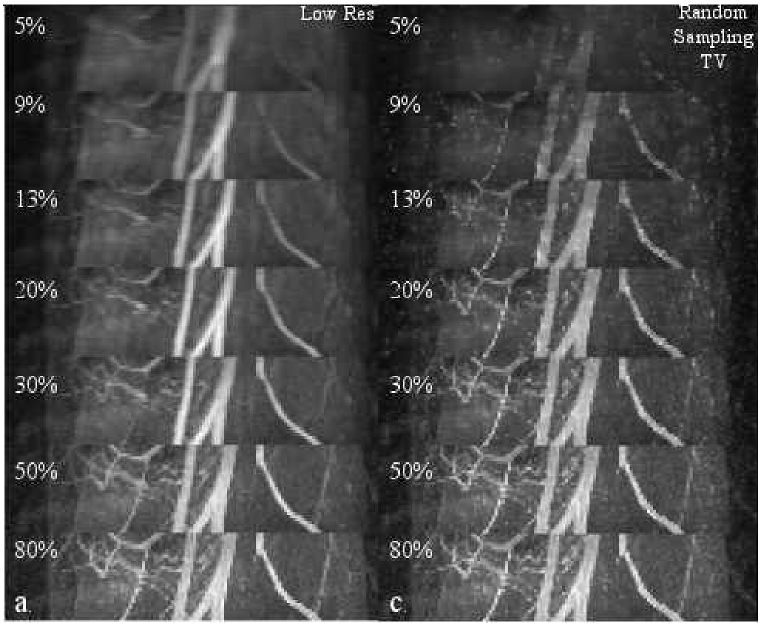

Figure 2.2: Using different data rates, we see the reconstruction given by typical imaging (left column) versus the reconstruction given by compressive sensing (right column). [LDP06]

The percentages in each reconstructed image refer to the percentage of data that was used for the reconstruction. As one can see, randomly sampling 50% of the data renders a reconstruction which is competitive with using 100% of the data with traditional imaging.

CHAPTER 3

APPLICATIONS TO SURVEILLANCE PROBLEMS

This chapter is primarily concerned with application of compressive sensing to different types of surveillance problems. The first scenario deals with a rather typical surveillance task; monitoring a parking lot. The second scenario will address the need to track motion in a video sequence. The third situation is one in which we are concerned about reconstructing a photograph of a very large land area.

The acquisition and transmission of high resolution video signal is often problematic due to the limitations of the ability of the camera to capture sufficient amounts of data and the transmission channel's bandwidth, which limits the amount of information that can be transmitted once the data is acquired. This motivates the need to develop a framework by which a scene can be sampled at a relatively low rate and then reconstructed in a way such that the video is of high quality.

There are many different types of scenes that one might capture. The type of motion in the video, the amount of the viewing area being consumed by the motion, lighting conditions, etc. For our purposes we will assume that we want to reconstruct a video in which most of the scene is static, and the lighting conditions are constant. This may seem rather restrictive upon first blush, but such scenes naturally arise in the area of surveillance (traffic cameras, UAVs, etc.). From here out we proceed with these type of surveillance applications in mind.

The first section of this chapter deals with a stationary camera capturing a dynamic scene. The second section also involves a stationary camera, but explores the idea of using compressive sensing to capture purely motion information from a scene. In the third and final section we deal with the problem of surveying a large piece of land. The contents of this final section are largely taken from a recent work written by the author of this thesis which appeared in the proceedings of an SPIE conference [HM11].

3.1 Video Reconstruction Using the LDS Model

One potential solution for compressive sensing of such video sequences was offered in *Compressive Acquisition of Dynamic Scenes* [AC10]. In the paper the authors modelled the compressive sensing of a scene in time as a linear dynamical system. The basic model of a linear dynamical system is as follows: let $\{I_t, t = 0, ..., T\}$ be a sequence of frames indexed by time t. Then we may model each frame of video $I_t \in \mathbb{R}^{\mathrm{N}}$ as

$$x_t = Cz_t,$$

where $C \in \mathbb{R}^{N \times d}$ is the observation matrix and $z_t \in \mathbb{R}^d$ is the hidden state vector. Let y_t denote the compressive measurement of x_t. That is,

$$y_t = \Phi_t x_t. \tag{3.1}$$

where Φ_t is the sensing matrix to be used at time t. At each time instance we encode the static portions of the scene as well as the dynamic portions. Let $\check{y}_t$ and $\tilde{y}_t$ denote the static and dynamic measurements, respectively. Let $\check{\Phi}$ and $\tilde{\Phi}_t$ denote the measurement matrices for the static and dynamic portions of the scene, respectively.

Then at each time instant t, we take the following measurements:

$$y_t = \begin{pmatrix} \check{y}_t \\ \tilde{y}_t \end{pmatrix} = \begin{bmatrix} \check{\Phi} \\ \tilde{\Phi}_t \end{bmatrix} I_t = \Phi_t y_t, \tag{3.2}$$

where $\check{y}_t \in \mathbb{R}^{\check{M}}$ denotes the constant measurements associated with the constant sensing matrix $\check{\Phi}$ (essentially encoding the constant motion from the scene), and $\tilde{y}_t$ denotes the dynamic measurements associated with the matrix $\tilde{\Phi}_t$.

To recover the video sequence $[x_t]$ via the LDS model, we will first solve for the state sequence $[z_t]$ and then solve for the observation matrix C (the notation $[x_t]$ denotes the matrix with columns equal to $x_t, t = 1, ..., N$). To solve for the state sequence we make the following observation: if $[x]$ lies in the column span of C, then $[\check{y}_t]$ lies in the column span of $\check{\Phi}C$. This implies that the SVD of $[\check{y}_t]$ will render an approximation of the state sequence $[\hat{z}]$. More precisely, if $\check{M} \geq d$, and $[\check{y}_t] = USV^T$, then $[\hat{z}_t] = S_d V_d^T$, where S_d is the $d \times d$ principal submatrix of S and V_d is the $T \times d$ matrix formed by columns of V corresponding to the singular values in S_d. We have that this estimate of the state sequence is reasonably accurate when x_t is compressible [ref].

Once the estimated state sequence $[\hat{x}_t]$ has been constructed, we can recover C by solving the following problem

$$\min \sum_{k=1}^{d} ||\Psi^T c_k||_1 \text{ subject to } ||\Phi_t C\hat{z} - y_t||_2 \leq \epsilon, \forall t. \tag{3.3}$$

Rather than solving this problem directly, we may use a modified CoSAMP algorithm in order to take advantage of the redundancy in the common measurements. The pseudo-code for this algorithm is provided below:

Algorithm 3: LDS CoSAMP

```
input  : $\Phi_t, \Psi y_t, \hat{z}_t, K$
output: $\hat{C}$
$\Theta_t \leftarrow \Phi_t \Psi$;
$v_t \leftarrow 0$;
$\Omega \leftarrow 0$;
while stopping criteria not met do
    $R \leftarrow \sum_t \Theta_t^T v_t \hat{z}_t$;
    $\forall k \in \{1, ..., N\}, r(k) \leftarrow \sum_{i=1}^d R^2(k,i)$;
    $\Omega \leftarrow \Omega \cup r_{2K}$;
    $A \leftarrow \text{argmin} \sum_t \|y_t - (\Theta_t)_{\cdot,\Omega} A \hat{z}_t\|_2$;
    $B_{\Omega,\cdot} \leftarrow A$;
    $B_{\Omega^c,\cdot} \leftarrow 0$;
    $\forall k \in \{1, ..., N\}, b(k) \leftarrow \sum_{i=1}^d B^2(k,i)$;
    $\Omega \leftarrow b_K$;
    $S_{\Omega,\cdot} \leftarrow B_{\Omega,\cdot}$;
    $S_{\Omega^c,\cdot} \leftarrow 0$;
    $\hat{C} \leftarrow \Psi B$;
    $v_t = y_t - \Theta_t S \hat{z}_t$;
```

This version of the CoSAMP algorithm can be interpreted as a special case of the model-based CoSAMP algorithm developed in [ref]. This interpretation offers the advantage of allowing the calculation of the number of measurements required for stable recovery by simply looking at the model sparsity of the signal. Specifically, if the sparsity of the signal(in our case $\hat{C}$) is s, then results about model-based CoSAMP guarantee that $O(s\log(Nd))$ are needed. The results in [ref] show that if the columns of C are K-sparse, then the sparsity of $\hat{C}$ is equal to dK. Thus, we need $M = O(s\log(Nd))$ measurements at each time instant in order to guarantee that the recovery will be accurate. That is, $M = O(dK\log(Nd)/T)$. This implies that as the number of frames increases, the number of measurements needed decreases.

3.1.1 Experiments with the LDS Model

The original paper which used the CS-LDS model focused on mainly on scenes which resembles changing textures. One such scene is one which contains a flame from a lighter. To show how well this model works with such a scene, we present results of different reconstructions below. In each reconstruction we vary the number of frames used. This illustrates the model's ability to allow very few measurements per frame to be used, so long as enough frames are used.

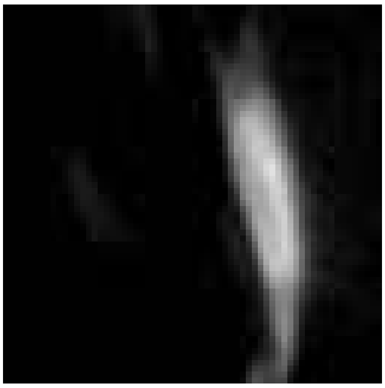

Figure 3.1: Frame 30 ground truth.

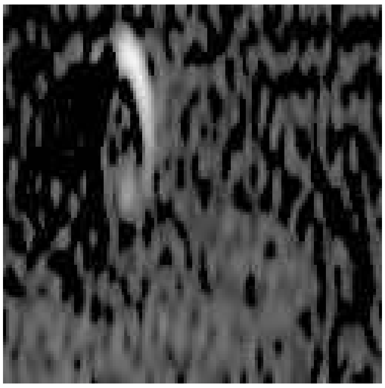

Figure 3.2: Frame 30 reconstruction with 74 frames.

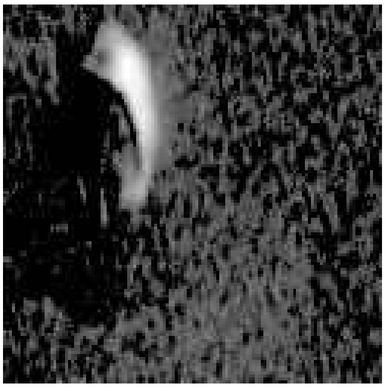

Figure 3.3: Frame 30 reconstruction with 200 frames.

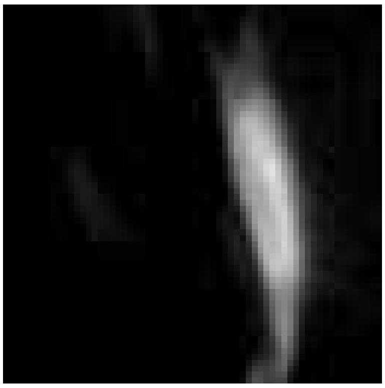

Figure 3.4: Frame 30 reconstruction with 560 frames.

For our next experiment we will consider a portion of video which captures a car passing through a static background.

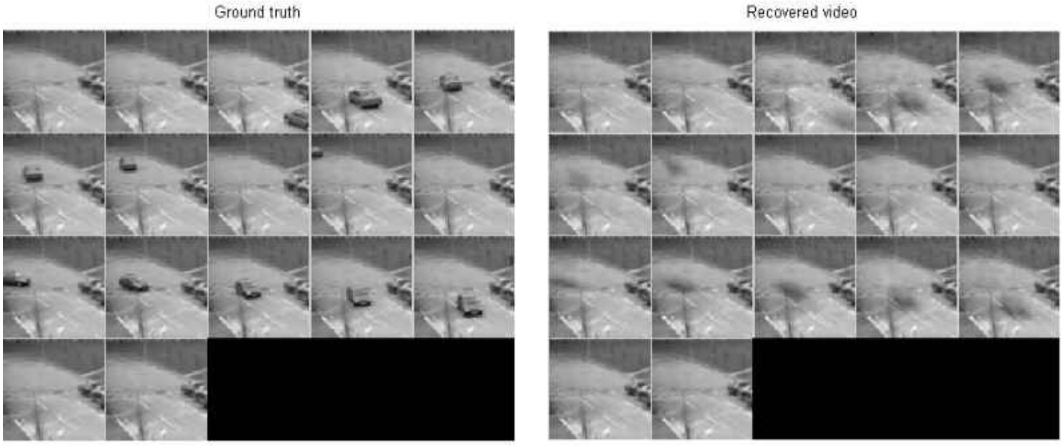

Figure 3.5: Using the CSLDS-mean model.

One notices that the static portion of the scene is reconstructed very accurately, but that the dynamic portion of the scene is hardly reconstructed at all. In fact, the cars driving by are reconstructed as faint spectres. Their positions can be gathered from the reconstruction, but their features are completely gone. In the next experiment we consider a scene with people walking around. The first example considers only a portion of the scene where the people are pacing in the same small area, turning and walking a very small distance. The second example is of the same general scene, except that now we have a person walking a significant distance through the scene.

Looking at these results, we notice that the appearance of the figures which pace around, but stay entirely within the frame, are well recovered while the person who walks off frame is poorly reconstructed (see Figures 3.6 and 3.7 in the next page), with their features being dissolved in the same way in which the features of the moving cars in the preceding experiment were. This begs the following question: why does this model reconstruct persistently visible

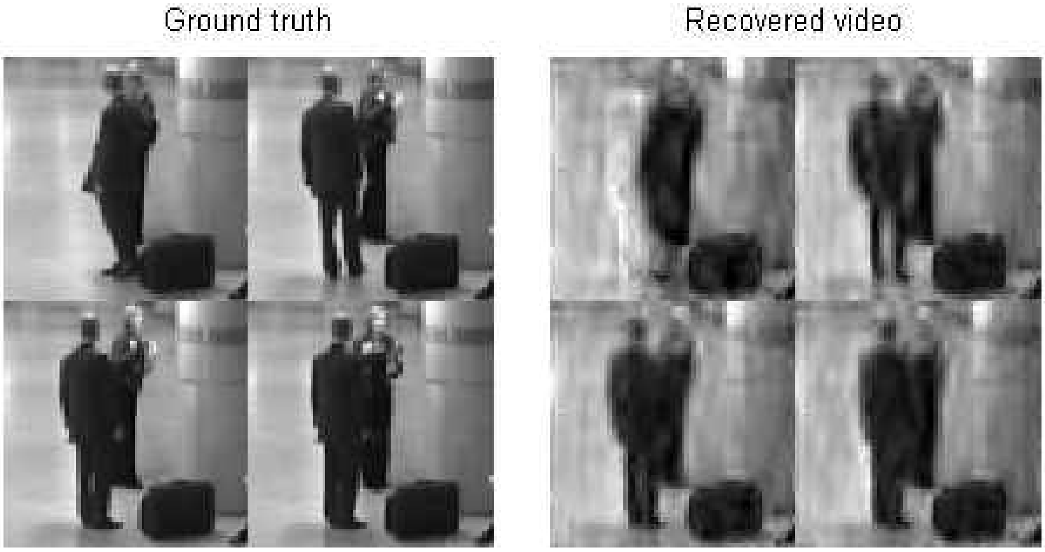

Figure 3.6: Pedestrians with little motion.

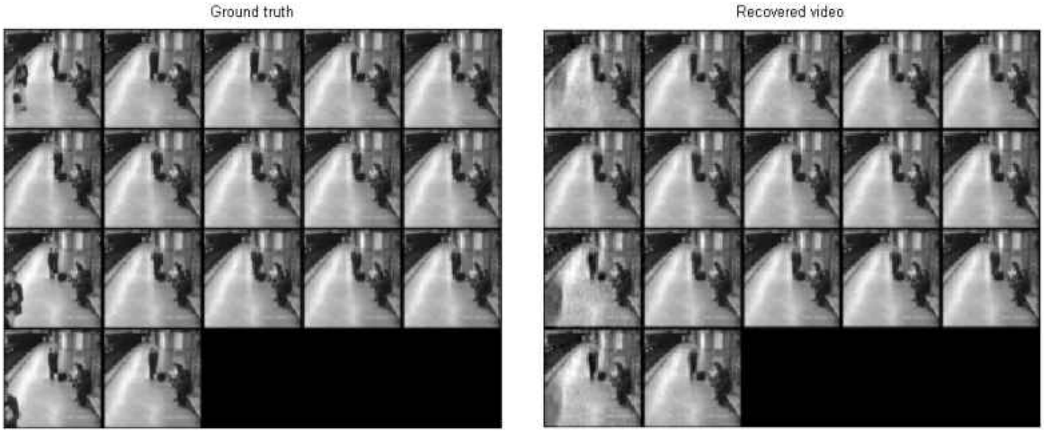

Figure 3.7: Pedestrians with significant motion.

objects well, while failing to reconstruct objects which are not always within the scene? A rigorous answer to this question is a great opportunity for further research, as this answer may lead to a better model which will be more robust to a variety of scenes.

3.2 Monitoring Motion in a Scene

In certain scenarios, the user might not be interested in what the scene looks like, but rather, what is happening in the scene. For example, one might want to know when there are moving objects in the scene and the nature of their motion, rather than the look of the scene itself. To address this surveillance concern we will demonstrate a method developed in *Compressive Sensing for Background Subtraction* [VC08]. In this work, the authors make use of the following observation: given a scene with a static background and a changing foreground, the difference image from two adjacent frames will have a higher degree of sparsity than the frames themselves.

To be more precise, let us introduce some notation. Let x_b denote the background image, x_c the current frame, and x_d the difference image, with $x_d = x_c - x_b$. Let $\mathcal{S}_d$ denote the support of the difference image. Then by parsing $\mathcal{S}_d$ one may determine the overall shape and location of motion in the frame. A conventional imaging scheme would sense x_b and x_c and then directly construct x_d. Since we are not concerned with the actual appearance of the scheme, the work needed to capture x_b and x_c is excessive. We instead seek a way to use compressive sensing to reconstruct the difference image in a way such that we never need to reconstruct x_b or x_c.

Indeed, let us observe the following:

$$y_b = \Phi x_b, \text{ and,} \tag{3.4}$$

$$y_c = \Phi x_c. \tag{3.5}$$

Therefore

$$y_b - y_c = \Phi(x_b - x_c), \text{or}, \tag{3.6}$$

$$y_d = \Phi x_d, \tag{3.7}$$

where $y_d = y_b - y_c$ denotes the difference compressive measurements. This simple idea give us a way by which to reconstruct the difference image by requiring that we only compressively sense the background and current images. Further still, when one looks at a difference image, one notices that it is mostly black. This suggests that x_d should be sparser than x_b and x_c.

Indeed, let suppose that the sparsity of the x_b and x_c is K (it is reasonable to make this assumption because of the similarities between the two images). Let K_d denote the sparsity of the difference image, x_d. Because much of the difference image will be empty, but for any motion, we may conclude that the wavelet coefficients used to represent the information contained in the static portion of the scene may be discarded. Hence, $K_d \leq K$. This means that we should be able to take few compressive measurements of x_b and x_c and still be able to reconstruct the difference image at the level of quality it would have been seen at if we took all K measurements of x_b and x_c. We will demonstrate this point empirically in the next section.

3.2.1 Experiments with Motion Tracking

In this section we will present numerical experiments which will demonstrate that a reasonable difference image may be reconstructed from the compressive measurements of the scene, and that the number of measurements required to reconstruct the difference scene is much less than the number required to reconstruct the scene itself. We will use the Coiflet wavelet basis as the sparsifying basis for each frame of video. We will recover images via the NESTA algorithm.

In our first experiment our objective is to reconstruct a scene of a parking lot with a car driving past. The field of view is 64 by 64 pixels. We will reconstruct the difference images in two ways: first we will sense the video in the traditional manner and construct the difference images from the actual image sequence, providing the ground-truth. Second, we will compressively sense the scene and construct the difference image from the difference of compressive measurements.

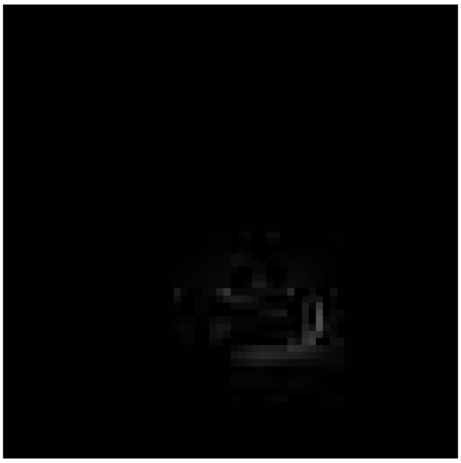

Figure 3.8: The ground truth difference image.

The results of the first experiment are presented in figures 3.7-3.10. Upon first blush, one may look at these results and be left feeling that the compressive sensing scheme does not offer much of a benefit. The amount of data the sensor needs to process is far lower with the compressive sensing scheme, but in exchange the reconstruction quality is far worse, both in terms of the appearance of the reconstruction and the error measured in terms of the L2-norm. However, when one looks closely at the reconstructed difference image one notices that the outline of a car is clearly visible and distinct from the noise. Also, the noise looks like noise. To be exact, it is clear that the errors in the reconstruction are extraneous. This gives reason

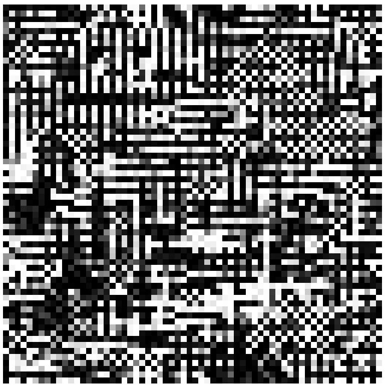

Figure 3.9: Frame 30 reconstruction using 5 percent of the data.

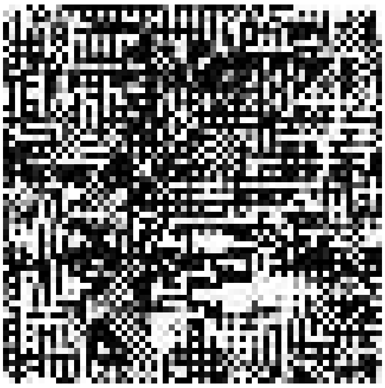

Figure 3.10: Frame 30 reconstruction using 10 percent of the data.

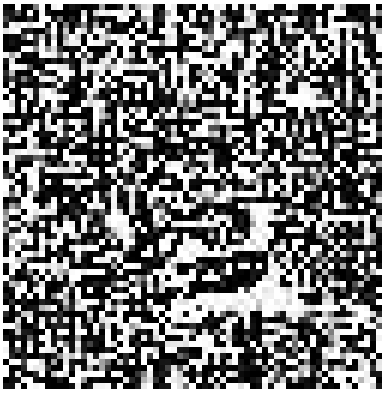

Figure 3.11: Frame 30 reconstruction using 20 percent of the data.

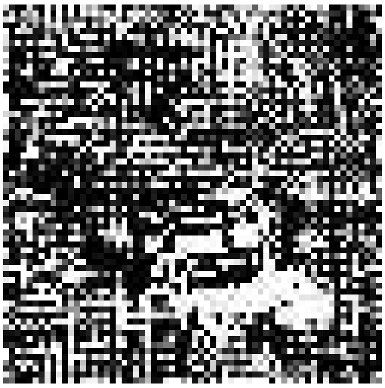

Figure 3.12: Frame 30 reconstruction using 30 percent of the data.

to believe that a filtering process may be performed on the reconstructed difference images in order to produce more accurate results.

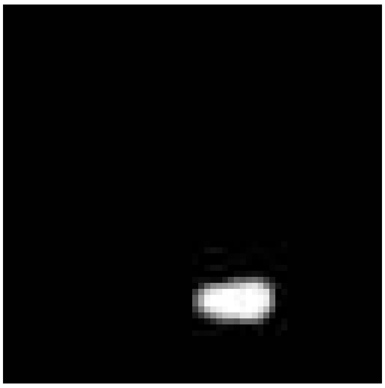

Figure 3.13: Frame 30 filtered reconstruction using 5 percent of the data.

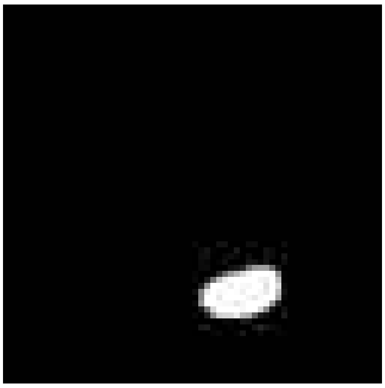

Figure 3.14: Frame 30 filtered reconstruction using 10 percent of the data.

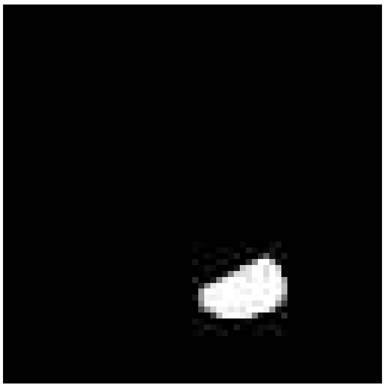

Figure 3.15: Frame 30 filtered reconstruction using 20 percent of the data.

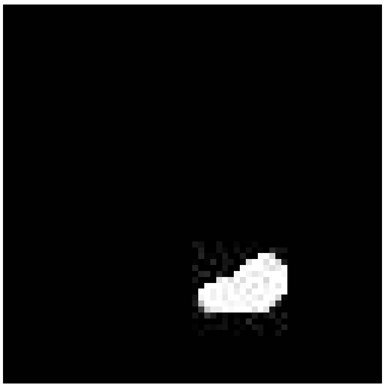

Figure 3.16: Frame 30 filtered reconstruction using 30 percent of the data.

As can be seen in figures 3.11-3.14, even a very naive thresholding technique can dramatically improve the quality of the reconstructed image. In particular, the portion of the scene with the moving car peaks high enough so that its motion is sensed correctly in every frame of video. This means that the motion sensing problem may in fact be resolved via a compressive sensing approach.

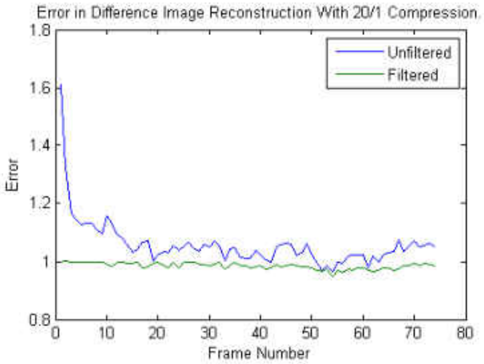

Figure 3.17: Using 5 percent of the data.

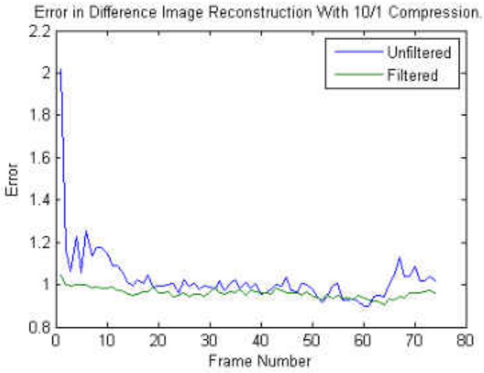

Figure 3.18: Using 10 percent of the data.

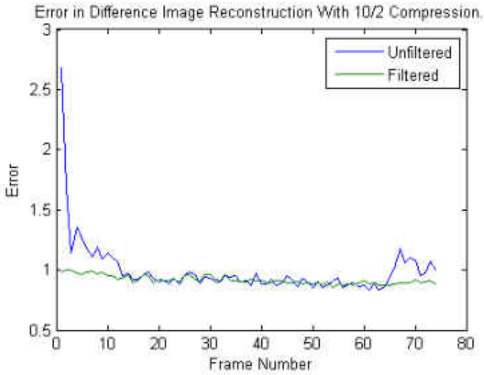

Figure 3.19: Using 20 percent of the data.

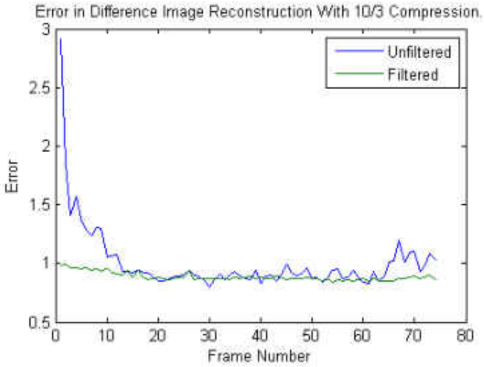

Figure 3.20: Using 30 percent of the data.

3.2.2 Using a Compressive Background Model for Object Detection

Often, it is the case that there is no new object in the scene. This implies that there is nothing of interest taking place in the scene. The above model calls for an l_1 minimization for each and every difference image. This is computationally taxing, and so it is worthwhile to investigate whether or not the minimization step really needs to be performed at every time instance. This section proposes a way of determining whether or not the scene is changing. To do this we develop a statistical model for the compressive background measurements and then use the compressive measurements directly in order to determine of a new object has entered the scene.

Suppose that we have a collection of compressive measurements of the background images. Let $y_{bi} \in \mathbb{R}^M$ denote the ith compressive measurement vector of the background of the scene with $i = 1, ...B$. Let y_b denote the mean of the background images. Let us consider the distribution of l_2 distances of the background images about their mean:

$$||y_{bi} - y_b||_2^2 = \sigma^2 \sum_{k=1}^{M} \left(\frac{y_{bi}(k) - y_b(k)}{\sigma} \right)^2 \tag{3.8}$$

If we take $M > 30$, then the central limit theorem gives us that the distribution of l_2 distances may be approximated by a Gaussian distribution. That is

$$||y_{bi} - y_b||_2^2 \sim \mathcal{N}(M\sigma^2, 2M\sigma^4). \tag{3.9}$$

Now suppose that we are comparing a test image to the mean background. Then we may derive the following distribution:

$$||y_t - y_b||_2^2 \sim \mathcal{N}(M\sigma^2 + ||\mu_d||_2^2, 2M\sigma^2 + 4\sigma^4 ||\mu_d||_2^2). \tag{3.10}$$

We can simplify matters by considering the logarithms of the l_2 distances. Using this approach we may write that

$$\log ||y_{bi} - y_b||_2^2 \sim \mathcal{N}(\mu_b, \sigma_b^2). \tag{3.11}$$

and

$$\|y_t - y_b\|_2^2 \sim \mathcal{N}(\mu_t, \sigma_t^2). \tag{3.12}$$

Our goal is to use these statistics to determine if a new object has entered a scene without having to perform a costly l_1 minimization to reconstruct the difference image. Toward this end, we learn the parameters in (3.11) via maximum likelihood estimates. With μ_b and σ_b known, we have that if σ_t^2 is sufficiently different from σ_b^2, then a simple two-sided threshold test is optimal for discriminating between another background image and an image with a new object in it [Tre68]. Thus, we say that there is a new object in the scene if

$$|\log \|y_t - y_b\|_2^2 - \mu_b| \geq a\sigma_b, \tag{3.13}$$

where a is a constant to be chosen by the user.

3.3 Monitoring a Large Track of Land

High resolution imaging sensors used in observing terrestrial activities over a very wide field-of-view will be required to produce gigapixel images at standard video rates. This data deluge affects not just the sensor but all of the processing, communication, and exploitation systems downstream. A key challenge is to achieve the resolution needed to observe and make inferences regarding events and objects of interest while maintaining the area coverage, and minimizing the cost, size, and power of the sensor system. One particularly promising approach to the data deluge problem is to apply the theory of compressive sensing, which enables one to collect fewer, information-rich measurements, rather than the many information-poor measurements from a traditional pixel-based imager.

For the wide field-of-view imaging application, Muise [Mui09] designed a compressive imaging algorithm with associated measurement kernels and has simulated results based upon a field-of-view multiplexing sensor described by Mahalanobis et al. [AB09]. These works show a viable concept for wide area imaging at high resolution. In this section, we explore concepts of collecting measurements of a wide area through multiple cameras and reconstructing the entire wide area image. This process is known as distributed compressive imaging (DCI).

Consider an N-pixel area to be sensed with multiple cameras and suppose we have limited bandwidth for communications. The bandwidth restriction precludes us from allowing for intra-camera communication. Compressive sensing theory tells us that $M = \beta \log \frac{N}{K}$ measurements are sufficient to guarantee an accurate signal recovery (here K denotes the sparsity of the area of interest). Suppose we have α cameras at our disposal and that as these cameras have overlapping fields-of-view. Then, assuming the cameras end up covering the entire area in aggregate, each camera need only take $\frac{M}{\alpha}$ compressive measurements in order to facilitate accurate signal reconstruction. The clear benefit here is that as the number of cameras increases, the amount of information each camera is responsible for acquiring decreases.

3.3.1 DCI Model

Here we propose a simple extension to the traditional compressive sensing model in order to make use of a camera ensemble. The naive DCI model is

$$\mathbf{Y} = \mathcal{P}\mathbf{B}x + \epsilon, \tag{3.14}$$

where $\mathcal{P}$ is a concatenation of the random Gaussian sensing matrices of each of the α cameras in our ensemble and $\mathbf{B}$ is the sparsity basis for the scene, x. That is,

$$\mathcal{P} = [\mathbf{P}_1, \mathbf{P}_2, \ \dots\ , \mathbf{P}_\alpha]^T = [p_1^1, p_2^1, ..., p_{k/\alpha}^1, p_1^2, ..., p_{k/\alpha}^2, ...p_1^\alpha, ..., p_{k/\alpha}^\alpha]^T.$$

Each entry of $\mathbf{Y}$ is an inner product of the image with a random projection vector p_j^i and so its form is

$$\mathbf{Y} = [\langle p_1^1, \mathbf{B}x\rangle, \langle p_2^1, \mathbf{B}x\rangle, \ \dots\ , \langle p_{k/\alpha}^1, \mathbf{B}x\rangle, \ \dots\ , \langle p_1^\alpha, \mathbf{B}x\rangle, \ \dots\ , \langle p_{k/\alpha}^\alpha, \mathbf{B}x\rangle]^T + \epsilon.$$

Our interest lies in having multiple cameras, all surveying a large region from different perspective. As such, if we take an absolute coordinate system for the entire region we model the differences in perspectives with an operator O_i so that $O_i(\mathbf{B}\alpha)$ generates the underlying scene $\mathbf{B}\alpha$ from the point of view of the ith camera. With this idea we may rewrite the observed measurements as

$$\mathbf{Y} = [\langle p_1^1, O_1(\mathbf{B}x)\rangle, \langle p_2^1, O_1(\mathbf{B}x)\rangle, \ \dots\ , \langle p_{k/\alpha}^1, O_1(\mathbf{B}x)\rangle, \ \dots\ , \langle p_1^\alpha, O_\alpha(\mathbf{B}x)\rangle, \ \dots\ , \langle p_{k/\alpha}^\alpha, O_\alpha(\mathbf{B}x)\rangle]^T.$$

For a particular perspective operator, O_i, we wish to derive the adjoint (for lack of a better term), O^*, so that

$$\langle h, O_i(y)\rangle = \langle O_i^*(h), y\rangle, \text{ for all } h, y.$$

For example, if O_i translates an image by $[a, b]$ pixels, then O_i^* would translate the measurement mask by $[-a, -b]$ pixels for an equivalent inner product. With this idea in mind we may once again rewrite our observation vector, $\mathbf{Y}$, as

$$\begin{aligned}\mathbf{Y} &= [\langle O_1^*(p_1^1), \mathbf{B}x\rangle, \langle O_1^*(p_2^1), \mathbf{B}x\rangle, \ \dots, \langle O_1^*(p_{k/\alpha}^1), \mathbf{B}x\rangle, \ \dots, \langle O_\alpha^*(p_1^\alpha), \mathbf{B}x\rangle, \ \dots, \langle O_\alpha^*(p_{k/\alpha}^\alpha), \mathbf{B}x\rangle]^T + \epsilon \\ &= \mathcal{P}^*\mathbf{B}x + \epsilon.\end{aligned}$$

Thus we take as the general DCI model

$$\mathbf{Y} = \mathcal{P}^*\mathbf{B}x + \epsilon, \tag{3.15}$$

where ϵ is included to take into consideration additive error in the sensing process. Also, unlike (3.1), this accounts for different camera perspectives. Thus we will be solving

$$\min_{\hat{\mathcal{P}}, x} \|x\|_1 \text{ subject to } \|\mathbf{Y} - \hat{\mathcal{P}}\mathbf{B}x\|_2 \leq c \tag{3.16}$$

where $\hat{\mathcal{P}} = \mathcal{P}^* + \mathcal{P}^e$, the ideal perspective operator plus an error. This alters our model for the observations to

$$\begin{aligned}
\mathbf{Y} &= \hat{\mathcal{P}}\mathbf{B}x + \epsilon \\
&= (\mathcal{P}^* + \mathcal{P}^e)\mathbf{B}x + \epsilon \\
&= \mathcal{P}^*\mathbf{B}x + \mathcal{P}^e\mathbf{B}x + \epsilon \\
&= \mathcal{P}^*\mathbf{B}x + \epsilon'
\end{aligned}$$

where our new error term is bounded by

$$\begin{aligned}
||\epsilon'||_2^2 &= ||\mathcal{P}^e\mathbf{B}x + \epsilon||_2^2 \\
&\leq ||\mathcal{P}^e||_2^2||\mathbf{B}x||_2^2 + ||\epsilon||_2^2 \\
&\leq E||\mathcal{P}^e|| + c,
\end{aligned}$$

where E is the overall energy in the image. Appealing to the result from Candes, Romberg, and Tao, we can solve (3.10) for $x^\sharp$ with the guarantee that

$$||x_{true} - x^\sharp||_2 \leq O(E||\mathcal{P}^e|| + c).$$

Although the behaviour of $E||\mathcal{P}^e||$ is difficult to characterize, there are several observations:

- When the ideal perspective estimates are known, $\mathcal{P}^\rceil = 0$ and thus $E||\mathcal{P}^e||$ is a minimum, and equation (3.4) distils down to the case studied by Candes, Romberg, and Tao.

- An iteration of (3.5) while perturbing the perspective estimates should generate a surface which has a global minimum when $\hat{\mathcal{P}} = \mathcal{P}^*$.

Hence, we are left with a procedure and an optimality criterion which theoretically should give us estimates for x and the camera perspectives by minimizing the l_1 norm of x while fitting the observed data.

3.3.2 Experiments with Large Area Monitoring

Given a wide field-of-regard image we wish to collect image projections from multiple cameras and rebuild the scene with minimal data being transmitted. Assuming that we know the perspective parameters for the multiple cameras, we have a sequence of cameras depicted by figure (3.1).

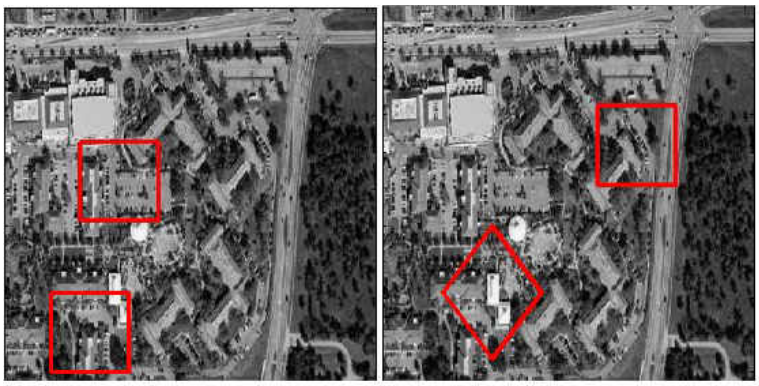

Figure 3.21: A wide field of regard image being sensed with multiple cameras.

We assume that the bandwidth of the data-link can only afford to send down 0.2% of the image over the support of its field of view. For example, if a camera generated a 128x128 image, then the amount of information transmitted for reconstruction would be approximately 24 numbers. The reconstruction from the non-compressed sensing is accomplished by observing the image, calculating the compression coefficients assuming a DCT basis set, and sending the top 0.2% of the coefficients to the reconstruction algorithm. Under this paradigm, the results of the scene reconstruction are shown in figure (3.2).

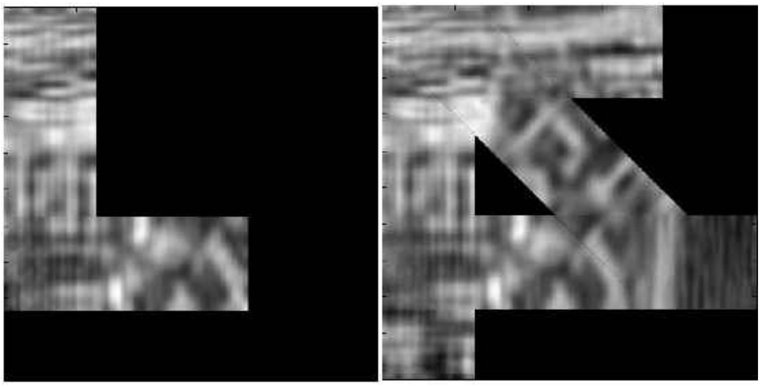

Figure 3.22: The scene being surveyed by traditional cameras with reconstruction via traditional compression.

For a distributed compressive imaging scenario, we assume the entire scene of interest can be compactly represented in a DCT basis and each individual camera would sample an image projection of a limited FOV of the scene. The projection masks should be randomized (to guarantee incoherence with the DCT basis) but should also have a notion of random sampling (as this is optimally incoherent with the DCT basis). We choose a methodology of projection mask construction as the following:

1. Randomly generate a size and location for the pixel sampling.
2. Iterate until roughly 1/4 of the pixels are contributing to the projection (this will ensure an SNR advantage through multiplexing).

An example of a projection mask used for this experiment is given in figure (3.3).

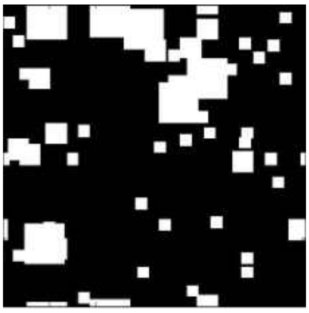

Figure 3.23: A typical projection mask.

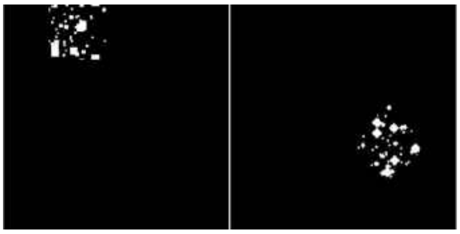

Figure 3.24: Projection masks placed in the scene's coordinate system.

With the camera perspective parameters assumed to be known, we calculate the projection mask in terms of the underlying scene coordinate system. This results in calculating the rows of the projection matrix $\mathcal{P}$. Two of these example perspective masks are given in figure (3.4).

With this calculation of the projection masks into the underlying scene coordinate system we use the STOMP [DS06] as our compressive sensing reconstruction algorithm with less than 0.2% of the underlying dimension of each camera's FOV. The results are shown in figure (3.5).

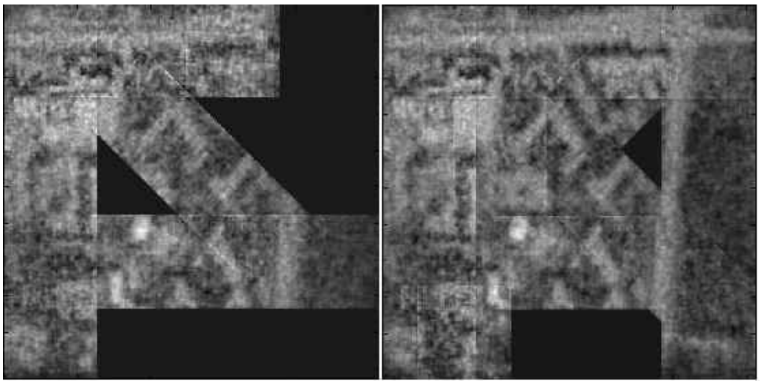

Figure 3.25: The scene being surveyed by compressive sensing cameras.

One notices that while new information is collected and transmitted, all of the areas of the underlying scene experience higher fidelity reconstruction. The final reconstruction with and without DCI are shown in figure (3.6).

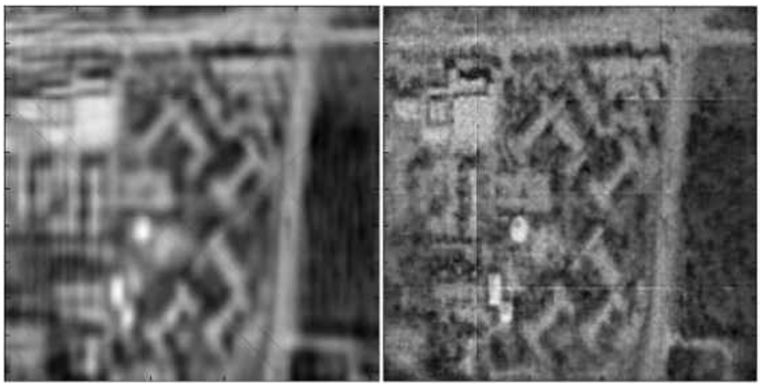

Figure 3.26: On the left is the complete reconstruction via traditional imaging. On the right is the reconstruction via compressive sensing.

One notices a very low frequency image from the standard compression which results from only 0.2% of the information being transmitted. The overall shape of the larger buildings is successfully reconstructed as well as the general large road network. With DCI, the reconstruction contains far more high-frequency content with many smaller buildings visible and the texture and shape of the trees on the right being of higher quality.

3.3.3 Multi-Camera Registration Issues

The above experiment was conducted under the assumption that the camera perspectives were known. Such information is not generally known and an image registration step would be required. For standard video cameras, this registration can be non-trivial, but solvable with standard tie-point correlation and re-sampling, or other techniques. For DCI, the imagery is unavailable to calculate correlations and we are required to register the imagery without access to the images. This problem was solved with manifold lifting techniques by Wakin [7], while we wish to test whether equation (3) gives us a general optimization criterion for estimating the camera perspectives from the image projection data stream.

Again, equation (3) suggests that the same criteria used for estimate the nonzero coefficients of a sparse model can be used to iteratively estimate the perspective parameters of our distributed compressive imaging system. To see the intuition, imagine that our perspective estimates for the cameras are incorrect. It should take more coefficients to reconstruct the incorrect scene than it would take to reconstruct the correctly registered information. Thus, finding the sparsest solution (or equivalently, the minimum l_1 solution) over all possible perspectives parameters should lead to the correct perspective estimates. We test this through a nine camera DCI test with the Lena image as described in the next section.

3.3.4 A DCI Model with Unknown Registration

In this experiment we treat the image of Lena as the field of regard and we have nine cameras surveying the image, each of which has a limited field of view. While no two cameras share the same field of view, each camera's field of view overlaps with at least one other camera's.

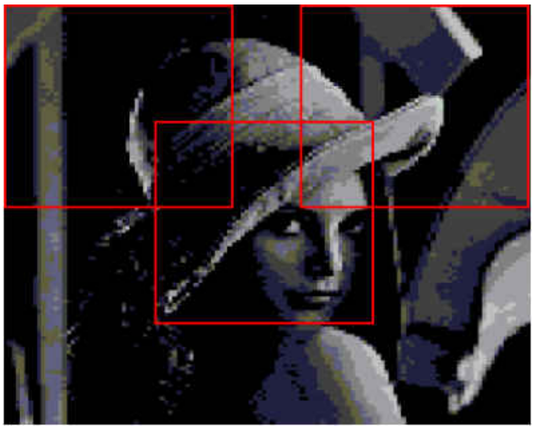

Figure 3.27: How the Lena image is being sensed.

The registration of the center camera's position is assumed to be unknown; and for the purposes of this experiment, all other camera perspective parameters are assumed to be known.

Also, although the results of our experiments should generalize to most camera perspective parameters, we test only unknown x, y translation.

With real surveillance applications in mind, it is reasonable to assume that one will have approximate camera registration parameters available. These approximate values will serve as an initial guess. Our experiment takes in the measurements from all nine cameras (the only unknown is the position of the center camera, denoted as γ), then takes in the estimate for γ, calculates the projection masks in terms of the underlying scene coordinate system, recovers an image through l_1 minimization, and saves the associated l_1 norm of the scene coefficients. We then make another estimate for γ and repeat this process, always saving the l_1 norms associated with each reconstruction. This process is meant to visualize the function from equation (3), which should give us an optimality criteria for estimating x and the camera registration parameters (which are embedded in the estimate for x).

Graphing the l_1 norms for each of the reconstructions as a function of the unknown parameter γ we have the following result in figure (8).

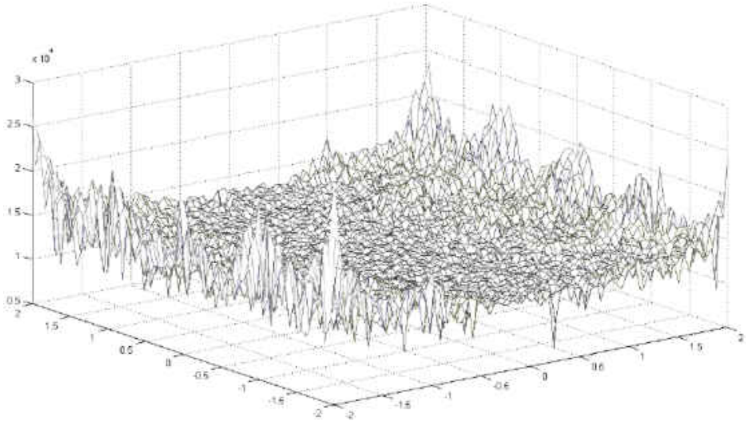

Figure 3.28: The x and y axis represent the guess for γ, while the z axis represents the l_1 norm of the reconstruction given γ. There are 81 nodes in both the x and y directions.

The noisy nature of this surface suggests that determining the optimal camera parameters based on the l_1 norm would be a difficult task. However, if we smooth this function be convolving it with a Gaussian mask of size 7x7 we gain insight into the nature of how this function behaves.

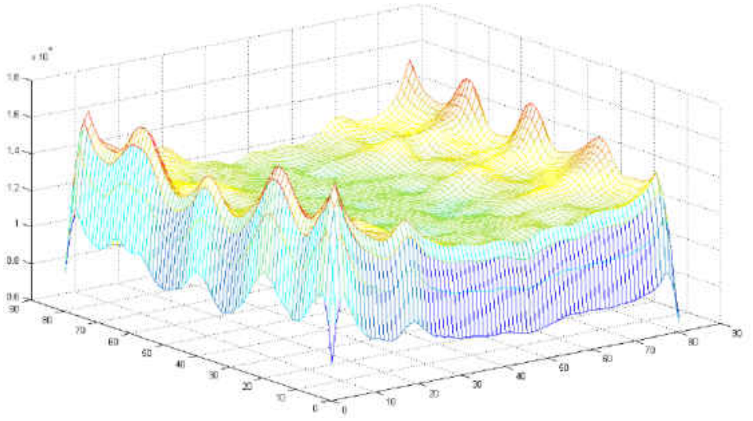

Figure 3.29: The smoothed graph of the l_1 norms as a function of γ.

This graph suggests that this function is locally quadratic. This offers one the intuition that one can find the global minimum of the function by taking a smoothed version of $||x||_1$ as our optimality criteria. To this end, for each test value of γ we solve (4) for several values close to γ and take the average value of $||x||_1$ as our objective function. The gradient of this new surface (represented in figure (9)) should now be relatively continuous and should give us insight into the possible convergence of a gradient descent algorithm. These gradients were calculated for the raw and smoothed versions of our objective function and are shown in figure (10) as arrows overlaid on an image of our objective function. The ideal perspective perspective estimates correspond to the center of each image.

The results of this experiment are promising and lend support to the argument that the minimum l_1 norm taken over different image reconstructions is minimized when the projection masks are correctly positioned within the scene's coordinate system.

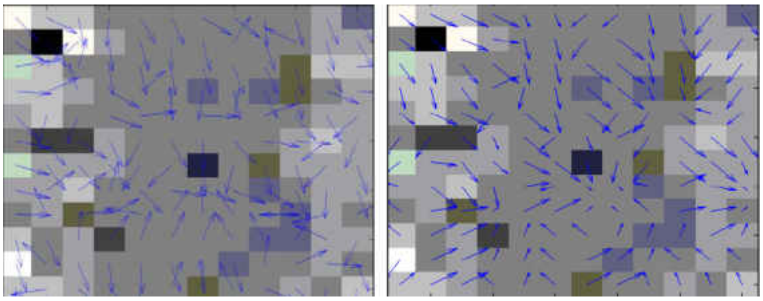

Figure 3.30: The graph on the left displays the gradient of $||x||_1$ before being smoothed. There is no indication that a gradient descent search will converge to the correct solution. The graph on the right displays the gradient of the smoothed l_1 function. There is a clear convergence to a point which is very close to the ideal perspective estimates.

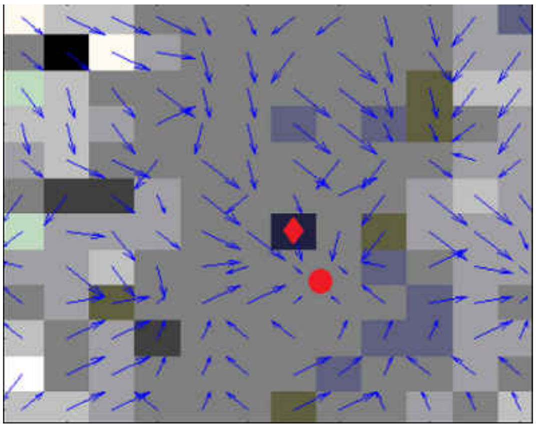

Figure 3.31: The red diamond displays the location of the true camera registration. The circle displays the location of the point that the graph's gradient converges to.

The analysis and experiment from section 3, coupled with the calculations in section 4 bode well for the concept of distributed compressive imaging. Randomized projections of limited field-of-view images seem to contain enough information to not only recover the underlying large area image, but also estimate the viewing geometry of each individual camera. This conclusion is also supported by the experiments conducted by Wakin in [ref].

CHAPTER 4

CONCLUSIONS AND FURTHER RESEARCH

The preceding work we have looked at different surveillance problems and the results that compressive sensing approaches can deliver. The LDS method is capable of reconstructing certain types of surveillance scenes with a high degree of accuracy. This model also enjoys the ability to reduce the number of measurements needed of each frame of video, so long as there is a sufficiently large number of frames available. The major drawback of this model is that it fails to reconstruct the features of dynamics which are not present in each frame. This drawback presents us with an opportunity for future research, with questions of why this model fails in these instances and whether or not it can be generalized to allow it to reconstruct additional classes of video.

In the context of motion sensing, we have presented results which show that motion information can be sensed directly by a compressive imager. The results were noisy, but the silhouettes of the moving objects were preserved. Further, we demonstrated that even a very naive filtering method could get rid of most of the noise. There are limitations to this method, however. In the scene we observed the object of interest was fairly large relative to the field of view. If the object(s) of interest was smaller, say a group of pedestrians from far above, then the pedestrian silhouettes may look like noise. As such, our filtering technique may disregard valuable motion information. One potential solution might be to use optical flow data. If one looks at the optical flow of the reconstructed sequence, surely one will observe mostly erratic motion vectors. However, the (small) portions of the scene which are actually representative of motion should still exhibit stable motion vectors. The portions of the scene associated with the smoothly changing motion vectors could be weighted heavily in a new filtering process. This will help prevent legitimate motion from being regarded as noise.

The third problem we looked at was that of wide-area surveillance. We have shown through analysis and simulation that there is significant benefit in distributed compressive imaging (DCI) to sense a very large area with significant benefits when there are severe bandwidth transmission restrictions. We have shown that the same criteria which allows compressive sensing to work (namely minimizing the L1-norm of the reconstruction coefficients) is also a viable criteria to estimate the registration parameters of the multiple cameras. It is particularly beneficial that one can take advantage of the redundancy of multiple cameras without intra-camera communications (something unattainable with traditional compression). A topic for further research is some combination of the manifold lifting algorithm developed by Wakin [Wak09] with the L1-minimization techniques outlined in this paper. This might lead to a faster method by which to accurately estimate the camera registration parameters.

Another topic for further research would be to determine an effective way to incorporate prior information about a scene into the model. This information should be used in a way that would increase the sparsity of the system (so that fewer measurements need to be taken) and/or decrease the number of iterations needed to converge to an accurate solution to the system. As an example, consider the wide-area surveillance application we discussed. Suppose that a low resolution photo of the entire track of land was available (this could be thought of as being given by a satellite with typical optics, without need of high-resolution capabilities). The resolution would be relatively poor, but the overall shape of the image could be captured.

A good question to ask, then, would be if one could use this information to speed up the reconstruction process. The CoSAMP algorithm uses a support pruning procedure. Could knowing roughly what the scene should look like help to more efficiently hone in on what the correct support of the sparse solution is? The ability to use such information would make for a novel algorithm and would contribute greatly in the applicability of compressive sensing to surveillance and imaging problems.

LIST OF REFERENCES

[AB09] V. K. Bhagavatula T. Haberfelde A. Mahalanobis, M. Neifeld and D. Brady. “Off-axis sparse aperture imaging using phase optimization techniques for application in wide-area imaging systems.” *Applied Optics*, **48**(28):5212–5224, 2009.

[AC10] R. Baraniuk A. Sankaranarayanan, P. Turaga and R. Chellappa. “Compressive acquisition of dynamic scenes.” In *Proceedings of the 11th European conference on Computer vision: Part I*, ECCV’10, pp. 129–142, Berlin, Heidelberg, 2010. Springer-Verlag.

[al04] N. Bangerter et al. “3D Fluid-Suppressed T2-Prep Flow-Independent Angiography using Balanced SSFP.” In *12th ISMRM Meeting*, 2004.

[CT06] E. Candes and T. Tao. “Near-Optimal Signal Recovery From Random Projections: Universal Encoding Strategies?” *IEEE Transactions on Information Theory*, **54**(12):5406–5425, 2006.

[DS06] I. Drori D. Donoho, Y. Tsaig and J. Starck. “Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit.” Technical report, 2006.

[ET06] J. Romberg E. Candes and T. Tao. “Stable Signal Recovery from Incomplete and Inaccurate Measurements.” *Communications on Pure and Applied Mathematics*, **59**(8):1207–1223, 2006.

[HM11] C. Huff and R. Muise. “Wide-Area Surveillance with Multiple Cameras Using Distributed Compressive Imaging.” In *SPIE*, 2011.

[LDP06] M. Lustig, D. Donoho, and J. M. Pauly. “Rapid MR imaging with Compressed Sensing and randomly under-sampled 3DFT trajectories.” In *in Proc. 14th*, 2006.

[MB06] M. Duarte D. Baron S. Sarvotham D. Takhar K. Kelly M. Wakin, J. Laska and R. Baraniuk. “An Architecture of Compressive Imaging.” In *International Conference on Image Processing*, 2006.

[MP07] D. Donoho M. Lustig and J. M. Pauly. “Sparse MRI: The application of compressed sensing for rapid MR imaging.” *Magnetic Resonance in Medicine*, **58**(6):1182–95, 2007.

[Mui09] R. Muise. “Compressive Imaging: An Application.” *SIAM Journal of Imaging Science*, **2**(4):1255–1276, 2009.

[Nes83] Y. Nestrov. “A Method for Unconstrained Convex Minimization Problem with the Rate of Convergence $O\left(\frac{1}{k^2}\right)$.” Technical report, Doklady AN USSR, 1983.

[NT08] D. Needell and J. A. Tropp. "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples." Technical report, California Institute of Technology, Pasadena, 2008.

[SC09] J. Bobin S. Becker and E. Candes. "NESTA: A Fast and Accurate First-order Method for Sparse Recovery." Technical report, California Institute of Technology, 2009.

[Tre68] H. Van Trees. *Detection, Estimation, and Modulation Theory, Part I.* John Wiley and Sons, Inc., 1968.

[VC08] M. Duarte D. Reddy R. Baraniuk V. Cevher, A. Sankaranarayanan and R. Chellappa. "Compressive Sensing for Background Subtraction." In *European Conf. on Computer Vision (ECCV)*, 2008.

[Wak09] M. Wakin. "A Manifold Lifting Algorithm for Multi-View Compressive Imaging." In *Picture Coding Symposium*, 2009.