

# CS754 Assignment 1 Report

Arpon Basu  
Shashwat Garg

Spring 2022

## Contents

<b>1</b>	<b>Problem 1</b>	<b>4</b>
1.1	Trend of Error Bound with $s$ . . . . .	4
1.2	Dependence of error bound on $m$ . . . . .	4
1.3	Change in requirement of $\delta_{2s}$ . . . . .	4
1.4	Setting $\epsilon = 0$ . . . . .	5
<b>2</b>	<b>Problem 2</b>	<b>6</b>
2.1	Lower Bound . . . . .	6
2.2	Upper Bound . . . . .	6
<b>3</b>	<b>Problem 3</b>	<b>7</b>
3.1	$m = 1$ . . . . .	7
3.2	$m = 2$ . . . . .	7
3.3	$m = 3$ . . . . .	7
3.4	$m = 4$ . . . . .	8
<b>4</b>	<b>Problem 4</b>	<b>10</b>
<b>5</b>	<b>Problem 5</b>	<b>11</b>
5.1	Hardware Architecture . . . . .	11
5.2	Reconstruction Techniques . . . . .	11
<b>6</b>	<b>Problem 6</b>	<b>12</b>
6.1	Generating the snapshot . . . . .	12
6.2	What are $A$ and $b$ ? . . . .	13
6.3	Pathwise Reconstruction . . . . .	13
6.4	Error in OMP . . . . .	14
6.5	Results . . . . .	15
6.6	Coded Snapshots- Cars . . . . .	15
6.7	Coded Snapshots- Flame . . . . .	15
6.7.1	Cars, $T=3$ . . . . .	16
6.7.2	Cars, $T=5$ . . . . .	17
6.7.3	Cars, $T=7$ . . . . .	18

6.7.4	Flame, $T=5$ . . . . .	20
-------	------------------------	----

## Introduction

Welcome to our report on CS754 Assignment 1. We have tried to make this report comprehensive and self-contained. We hope reading this would give you a proper flowing description of our work, methods used and the results obtained. Feel free to keep our code scripts alongside to know the exact implementation of our tasks.

We have referred to some sites on the web for finding the MATLAB implementations (generic documentation pages) and the same has been added in the references section.

In many places, to better give context to the place from which the questions could have arisen, some theoretical discussions have been engaged in.

Hope you enjoy reading the report. Here we go!

## 1 Problem 1

We have been given the following problem-

Let  $\theta^* : \min \|\theta\|_1$  such that  $\|y - \Phi \Psi \theta\|_2 \leq \varepsilon$ , where  $x = \Psi \theta$  and  $y = \Phi x + \eta$ .  $\varepsilon$  is an upper bound on the magnitude of the noise vector  $\eta$ .

Also, Theorem 3 states-

If  $\Phi$  obeys the restricted isometry property with isometry constant  $\delta_{2s} < \sqrt{2} - 1$ , then we have  $\|\theta - \theta^*\|_2 \leq C_1 s^{-1/2} \|\theta - \theta_s\|_1 + C_2 \varepsilon$  where  $C_1$  and  $C_2$  are functions of only  $\delta_{2s}$  and where  $\forall i \in S, \theta_{si} = \theta_i; \forall i \notin S, \theta_{si} = 0$ .

### 1.1 Trend of Error Bound with $s$

This is not a discrepancy. In reality, we are ignoring the change in  $C_1$  and  $C_2$ . The point is, we are only focusing on the effect of  $s^{-1/2}$  and  $\|\theta - \theta_s\|_1$ . We must also see the changes in  $C_1$  and  $C_2$ , which increase as the value of  $\delta_{2s}$  changes.

As the value of  $s$  increases, due to reducing sparsity, we observe that (the bound on)  $\delta_{2s}$  also increases. Intuitively, this is because an increase in  $s$  implies an increase in the required number of linearly independent columns required in the  $\Phi \Psi$  matrix. This is unlikely as  $s$  increases. An increase in  $\delta_{2s}$  leads to an increase in the value of  $C_1$  and  $C_2$ .

Thus we have two opposing behaviors and we cannot claim that the error bound improves as the sparsity measure,  $s$  increases in value.

### 1.2 Dependence of error bound on $m$

$m$  does not directly affect the error bound but it has several other indirect implications.

First of all, we require our matrix  $A = \Phi \Psi$  to obey the RIP, for which we usually construct our sensing matrix  $\Phi$  using Gaussian or Bernoulli entries randomly. The probability that this happens is very high in case  $m \geq CS \log(n/S)$ .

Apart from this,  $m$  also affects the value of  $\epsilon$  which we can take in case of uniform Gaussian noise in the measurements. In such cases, a good approximation of  $\epsilon$  is given by 3 standard deviations around the mean, which effectively translates into  $\epsilon \geq 9m\sigma^2$ .

Finally, when  $m$  increases, we observe that the rank of the matrix  $A$  increases, which would make it satisfy the RIP with a higher probability. This would indicate a drop in the value of  $\delta_{2s}$ , indicating a drop in the constants  $C_1$  and  $C_2$  and thus improving the error bound.

Thus,  $m$  has a lot of indirect implications in the error bound.

### 1.3 Change in requirement of $\delta_{2s}$

Theorem 3 is better than Theorem 3A. Theorem 3 provides the same guarantees for more relaxed values of  $\delta_{2s}$ .

For large  $\delta_{2s}$  we can work with comparatively less sparse signals. For values of  $\delta_{2s} \geq 0.1$ , Theorem 3A doesn't guarantee anything while the Theorem 3 works well for all values

till 0.4142.

Also, for values of  $\delta_{2s} \leq 0.1$ , both the theorems give the exact same reconstructions.

Thus, Theorem 3 is much more useful.

#### 1.4 Setting $\epsilon = 0$

This is not the correct way to solve the problem in presence of noise.

By setting the  $\epsilon$  to 0, we have ignored the noise and thus our  $\theta$  obtained from such a method will be incorrect. We might encounter several problems. We might not be able to converge on a value, because a solution does not even exist in this case of constraints. We might overfit our measurements to include noise which would result in the  $\theta$  not being the sparsest and correct measurement.

Instead, we should choose an  $\epsilon$  greater than the noise magnitude, for good results.

## 2 Problem 2

We note that this problem has two parts: One for demonstrating that the upper bound of  $\mu(\Phi, \Psi)$  is  $\sqrt{n}$ , and one for demonstrating that the lower bound of  $\mu(\Phi, \Psi)$  is 1. We shall deal with both parts separately.

### 2.1 Lower Bound

*Proof.* Since  $\Psi$  is an orthonormal matrix, its column vectors  $\Psi_1, \Psi_2, \dots, \Psi_n$  form an orthonormal basis for  $\mathbb{K}^n$ , and consequently we can express  $\mathbf{g}$  as  $\sum_{k=1}^n \alpha_k \Psi_k$ . Also,  $\|\mathbf{g}\|_2 = \sqrt{\sum_{j=1}^n \alpha_j^2}$ , following which we get  $\mathbf{g}_{\text{normalized}} = \sum_{k=1}^n \frac{\alpha_k}{\sqrt{\sum_{j=1}^n \alpha_j^2}} \Psi_k$ . Thus  $\mu(\mathbf{g}, \Psi) = \sqrt{n} \cdot \max_{i \in [n]} |\mathbf{g}_{\text{normalized}}^T \Psi_i| = \sqrt{n} \cdot \max_{i \in [n]} \frac{|\alpha_i|}{\sqrt{\sum_{j=1}^n \alpha_j^2}}$ . WLOG assuming that  $|\alpha_1| = \max_{i \in [n]} |\alpha_i|$ , we get that

$$\mu(\mathbf{g}, \Psi) \geq \sqrt{n} \cdot \max_{i \in [n]} \frac{|\alpha_i|}{\sqrt{\sum_{j=1}^n |\alpha_j|^2}} = \sqrt{n} \frac{|\alpha_1|}{\sqrt{n\alpha_1^2}} = 1$$

as desired. Note that equality is achieved iff  $\alpha_1 = \alpha_2 = \dots = \alpha_n$ .  $\square$

### 2.2 Upper Bound

*Proof.* Directly borrowing from the previous proof,  $\mu(\mathbf{g}, \Psi) = \sqrt{n} \cdot \max_{i \in [n]} |\mathbf{g}_{\text{normalized}}^T \Psi_i|$ . Now, by the Cauchy Schwartz inequality, for any two vectors  $\mathbf{u}, \mathbf{v}$ , we have  $|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\|_2 \|\mathbf{v}\|_2$ , where  $|\langle \cdot, \cdot \rangle|$  is the usual Euclidean inner product. Applying it to the coherence relation yields  $|\mathbf{g}_{\text{normalized}}^T \Psi_i| \leq \|\mathbf{g}_{\text{normalized}}\|_2 \|\Psi_i\|_2 = 1$ , and consequently we obtain that  $\mu(\mathbf{g}, \Psi) \leq \sqrt{n}$ , as desired. Note that equality is achieved iff  $\mathbf{g}$  is parallel to some column vector  $\Psi_i$  of  $\Psi$ .  $\square$

### 3 Problem 3

This is a problem based on the basic concepts of linear algebra and the concepts of basis vectors, null-space property and independent vectors.

We have a compressed measurement  $y$ , of the form  $\mathbf{y} = \Phi \mathbf{x}$  where  $\Phi \in \mathbb{R}^{m \times n}$  is the measurement matrix and  $\mathbf{x} \in \mathbb{R}^n$ ,  $n \gg 2$ . Consequently,  $\mathbf{y} \in \mathbb{R}^m$  ( $m \ll n$ ). Our aim is to figure out  $\mathbf{x}$ .

#### 3.1 $m = 1$

Now observe that  $\Phi$  is essentially a row vector and  $\mathbf{x}$  is a column vector, thus, even if only a element of  $\mathbf{x}$  was non-zero, we still do not know which index of  $\Phi$  and  $\mathbf{x}$  combine to give the value of  $y$ , which is effectively a scalar in this case.

On the other hand, if we know the value of index, then the vector  $\mathbf{x}$  is deducible. If the index is  $i$ , then the elements of  $\mathbf{x}$  are zero for all indices  $j \neq i$  and equal to  $y/\Phi_i$  for the  $i^{th}$  index.

Thus, answer of the first part is NO, and the second part is YES.

#### 3.2 $m = 2$

Yes! It is possible in this case, if  $\mathbf{x}$  has only one non-zero element.

Since  $\mathbf{x}$  has only one non-zero entry, say index  $i$ , only a single column of  $\Phi$  plays a role in getting the value of  $y$ .

Thus, effectively,

$$\begin{aligned} y^1 &= \Phi_i^1 \cdot x^i \\ y^2 &= \Phi_i^2 \cdot x^i \end{aligned}$$

This implies that the ratio of the values in the  $y$  vector, matches the ratio of the values in some column of the  $\Phi$  matrix.

If our  $\Phi$  matrix is sufficiently randomly generated, we can always uniquely find such an index.

After that, it is just like the previous question. If the index found is  $i$ , then the elements of  $\mathbf{x}$  are zero for all indices  $j \neq i$  and equal to  $y^1/\Phi_i^1$  or  $y^2/\Phi_i^2$  for the  $i^{th}$  index.

Again, please note that the success of this method depends on the fact that all columns of  $\Phi$  are linearly independent, which is a valid assumption given the nature of  $\Phi$ .

#### 3.3 $m = 3$

Here, we have a 2-sparse  $\mathbf{x}$  vector and 3 measurements. Effectively, we need to solve the following linear system-

$$y^1 = \Phi_i^1 \cdot x^i + \Phi_j^1 \cdot x^j$$

$$\mathbf{y}^2 = \Phi_i^2 \cdot \mathbf{x}^i + \Phi_j^1 \cdot \mathbf{x}^j$$

$$\mathbf{y}^3 = \Phi_i^3 \cdot \mathbf{x}^i + \Phi_j^1 \cdot \mathbf{x}^j$$

Note that we do not have any idea about which indices the  $i$  and  $j$  might be.

Just as we looked at the ratio in the previous part, here we need to look at the span of the two column vectors  $\Phi_i$  and  $\Phi_j$ . If the vector  $\mathbf{y}$  lies in the span of the two vectors, then we can find an appropriate solution to the values of vector  $\mathbf{x}$ .

But there is a problem. Note that the choice of  $i$  and  $j$  is not unique. We know that the  $m \ll n$ . Thus, given that  $\Phi_i$  is a random matrix, there is a high chance that we can choose another set of 2 vectors with indices  $k$  and  $l$  such that  $\mathbf{y}$  lies in the span of  $\Phi_k$  and  $\Phi_l$ .

In fact, given a large enough  $n$ , it is certain that there will exist 4 vectors (of dimension 3) which are linearly dependent and none of the coefficients ( $\alpha_i, \alpha_j, \alpha_k, \alpha_l$ ) is zero. This means that we will always have 2 ways of representing our  $\mathbf{y}$  vector. This implies that, we cannot always get the  $\mathbf{x}$  vector in this case.

Surprisingly, without knowing the indices at which  $\mathbf{x}$  is non-zero, we cannot even generate a specific  $\Phi$  such that we would have the vector  $\mathbf{x}$ . This is because such a  $\Phi$  would require most of the columns to be zero in atleast 2 places out of 3, such that we cannot get 4 linearly dependent columns with all  $\alpha_i, \alpha_j, \alpha_k, \alpha_l$  as non-zero. This would ensure that there is always a unique pair of indices that calculate  $\mathbf{y}$ , but again, it is not possible in the current scenario.

### 3.4 $m = 4$

This problem very closely resembles the null-space theorem covered in the lectures. We have a 2-sparse vector and a matrix with  $m = 4$ . If we can ensure that any 4 columns are linearly independent then we have a solution.

If you are wondering about whether we might fail due to 5 such columns being linearly dependent, note that in that case, we would have a 2 column representation equal to a 3 column representation. But a 3 column representation is useless in this case. Thus, we can avoid it.<sup>1</sup>

The solution is as follows-

We assume that the matrix is of rank = 4. This implies that a unique solution exists. We follow the following algorithm.

- Iterate over all pairs of columns in the matrix  $\Phi$ .
- Check if the vector  $\mathbf{y}$  lies in the span of the two column vectors in focus.

As long as the rank = 4, there exists only a single solution for the algorithm. Otherwise, we have two 2-sparse vectors as solution. This implies that their difference ie. a 4-sparse

---

<sup>1</sup>The previous case, for  $m = 3$  fails because in that case, any 4 columns are linearly dependent. Thus, we can get a 2 column representation equal to another 2 column representation, which leads to multiple solutions.



vector lies in the null-space of  $\Phi$ . This is a contradiction to our assumption. Hence proved.

Thus, the answer to this part is YES (mostly, since there is a certain randomness involved in creation of these matrices).

## 4 Problem 4

We provide a short and sweet proof for this problem below.

*Proof.* We set up some notation first: Let

$$\begin{aligned} \mathbf{x}^* &:= \arg \min_{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq e} \|\mathbf{x}\|_1 \\ l_1 &:= \min_{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq e} \|\mathbf{x}\|_1 \\ f(t) &:= \arg \min_{\|\mathbf{x}\|_1 \leq t} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \end{aligned}$$

Then note that if  $t < l_1$ ,  $f(t) > e$ : Why? Because if  $f(t) \leq e$ , then one would have that there exist  $\mathbf{x}$  with L1-norm lesser than  $l_1$  which nevertheless make the L2 norm of  $(\mathbf{y} - \mathbf{A}\mathbf{x}) \leq e$ , which contradicts the minimality of  $\mathbf{x}^*$ .

Also note that  $f(l_1) \leq e$ , since we have a witness  $\mathbf{x}^*$  for which the L1-norm is equal to  $l_1$  and  $\|\mathbf{y} - \mathbf{A}\mathbf{x}^*\|_2 \leq e$  due to the premise of the problem P1. Thus it's clear that all minimizers of  $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2$  under the conditions  $\|\mathbf{x}\|_1 \leq l_1$  actually lie only on the “sphere”  $\|\mathbf{x}\|_1 = l_1$  (because for points strictly to the interior of this “sphere”,  $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2$  assumes values larger than  $e$ ). Now, if it so happens that there exists another  $\mathbf{z}$  (such that  $\|\mathbf{z}\|_1 = l_1$ ) such that  $\|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2 \leq e$ , then we note that the *uniqueness* of  $\mathbf{x}^*$  is violated, as  $\mathbf{z}$  also becomes a minimizer for P1 (along with  $\mathbf{x}^*$ ) since it satisfies the premise of P1 (which is  $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq e$ ).

Thus, we have that for  $t = \min_{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq e} \|\mathbf{x}\|_1$ , both the problems P1 and Q1 have the **same unique minimizer**.

Hence proved. □

## 5 Problem 5

Paper Details	
Title of the Paper	A Configurable Energy-Efficient Compressed Sensing Architecture With Its Application on Body Sensor Networks
Link of the paper	<a href="#">Click Here</a>
Publisher	IEEE
Publication Date	February 2016
Publication Venue	IEEE Transactions on Industrial Informatics, Vol. 12, No. 1

### 5.1 Hardware Architecture

The primary goal of this paper is that given a finite amount of energy, how best can we *configure* our hardware so that we get larger and larger amounts of reconstruction accuracy on lesser and lesser amounts of energy. Note that the term energy, in our context, can be assumed to be synonymous to the amount of computational resources available at our disposal, and thus the paper addresses a very important issue of extracting performance under low resources.

To that end, it proposes a new hardware architecture as shown here:

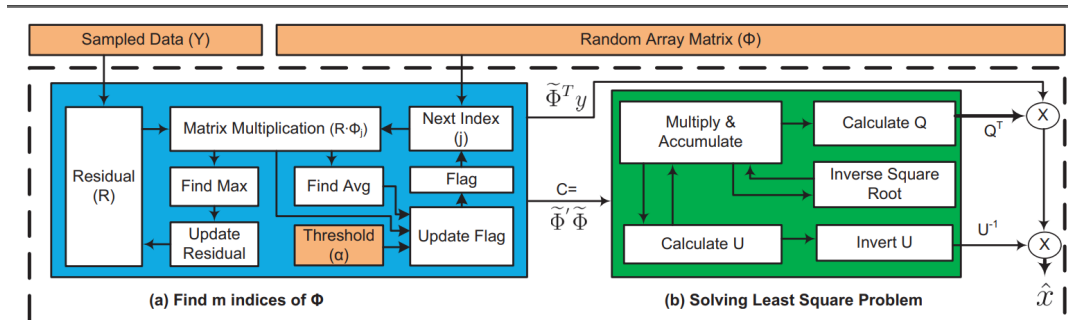


Fig. 3. Detailed architecture diagram of Improved OMP reconstruction algorithm. (a) This block iterates 8 times to solve the optimization problem eliminating columns of  $\Phi$  those are poorly correlated to  $Y$ . Operates at 85 MHz. (b) Finds inverse of a  $8 \times 8$  matrix after finding 8 columns of  $\Phi$ . Operates at 69 MHz.

This paper combines two important ideas, one of which has been already discussed in the class: We define a parameter  $M$ , which denotes the rate at which the continuous stream of analog data is sampled and **then randomly encoded**. The second idea is that of **quantization**, which basically argues that analog signals should be **digitally quantized** based on a given *bit resolution*  $b$ . It then proposes that we design a configurable hardware which takes as parameters  $(M, b)$

### 5.2 Reconstruction Techniques

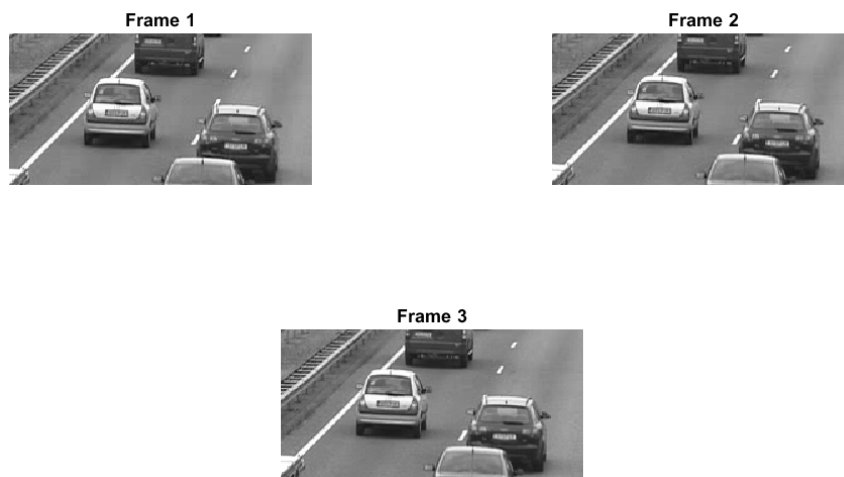
In this paper, the authors propose a novel algorithm they term **RapQCS**. The objective of this algorithm is to achieve as good a performance as possible, given that our energy budget (ie:- our computational resources) is constrained.

## 6 Problem 6

Finally, a coding problem! Here, we have to simulate compressive sensing using artificial tools to mimic the hardware architecture and omp algorithm used for processing the snapshot. Let's begin!

### 6.1 Generating the snapshot

We begin with reading the `cars.avi` video. We use the `mmread` functions provided to extract the required frames. We convert them to grayscale using the `rgb2gray()` function. This is followed by extracting the  $120 \times 240$  subsection from each frame. Finally we have our frames, which look like this.



Now, we use random bernoulli matrices to obtain the coded snapshots of the frames. Using the equation  $E = \sum_{t=1}^T C_t \cdot F_t$ , and adding zero mean Gaussian random noise of standard deviation 2 to it, we get the following snapshot.

**Coded snapshot with noise for T=3**



## 6.2 What are $A$ and $b$ ?

We already have the coded snapshot. Assume that  $E$  and  $F_t$  are vectorised. This is as follows-

$$E = \sum_{t=1}^T C_t \cdot F_t$$

We can convert the  $C_t$  vector into a diagonal matrix, which will convert the dot-product into a matrix multiplication. Thus, we can say that  $\Phi = \text{diag}(C_t)$ . This gives us the following-

$$E = \sum_{t=1}^T \Phi_t F_t$$

Finally, we know that  $F_t$  is sparse in DCT basis, thus we have a DCT basis representation of the vector  $F_t$ . Thus, we have, where  $\Psi$  stands for the DCT matrix-

$$F_t = \Psi \Theta_t$$

Finally, we can combine the above equations to get-

$$E = \sum_{t=1}^T \Phi_t \Psi \Theta_t$$

$$E = \Phi \Psi \Theta$$

where  $\Phi$  stands for  $[\Phi_1 | \Phi_2 | \dots | \Phi_T]$  and  $\Theta$  stands for  $[\Theta_1^t | \Theta_2^t | \dots | \Theta_T^t]^t$

Comparing with our equation,  $Ax = b$ , we have-

- $b$  is the vector we have measured, and equals the vector  $E$ . It is of dimension  $\mathbb{R}^{HW}$
- $x$  is the vector which we need to estimate. It is denoted by the vector  $\Theta$  and is basically the information about all the frames at once. Subsequently, it is of dimension  $\mathbb{R}^{HWT}$ .
- Finally,  $A$  is the product of the matrices  $\Phi$  and  $\Psi$ . It is of dimension  $\mathbb{R}^{HW \times HWT}$ .

This way, we are able to convert our problem into the form  $Ax = b$ .

## 6.3 Pathwise Reconstruction

Now suppose we are dealing with patches instead of the whole image. Still, we have the same notion of  $A$  and  $b$ .

Let the height and width of a patch be  $h$  and  $w$ . Let the number of frames be  $T$ .

- $A = \Phi \Psi$ , where  $A$  is of dimension  $\mathbb{R}^{hw \times hwT}$ .
- $\Phi$  is formed by horizontal concatenation of the diagonal matrices of the individual bernoulli coded matrices. It is of dimension  $\mathbb{R}^{hw \times hwT}$ .
- $\Psi$  is the DCT coefficient matrix, obtained by the following code in matlab-

$$\Psi = \text{kron}(\text{dctmtx}(p)', \text{dctmtx}(p)');$$

This is because we need the 2D-DCT basis for reconstruction. Our signals are sparse in 2D-DCT.

- $b$  again is the signal which we are measuring. It is of the dimension  $\mathbb{R}^{hw}$ .
- Finally,  $x$  is the signal which we need to estimate. It is the combination of all the frames in question. Thus, it is of dimension  $\mathbb{R}^{hwT}$ .

#### 6.4 Error in OMP

We have added a Gaussian noise of standard deviation 2 to each pixel of the coded snapshot. Let us deal with one patch at a time.

We know that the size of a patch  $= m = p * p$ . Assuming that the error dies out within 3 standard deviations, we have the combined error of the image as follows-

$$||\eta||^2 = 9 * \sigma^2 * m$$

. According to Compressive sensing theory, our epsilon (error magnitude) should always be greater than the noise. Thus, with a patch size of 8\*8 and noise from Gaussian distribution with std=2, we have-

$$\epsilon \geq 9 * 4 * 64$$

$$\epsilon \geq 2304$$

Since we are measuring the error in terms of  $\text{norm}(\eta)$  rather than  $|\eta|^2$ , we have the error metric in the OMP as-

$$||\eta||_2 \leq \epsilon$$

, where  $\epsilon = \sqrt{2304} \approx 48$

Using this value of error term, we observe that the OMP algorithm completes in around 5-20 iterations for each image and we also get good results. This indicates the correctness of choosing a correct  $\epsilon$ .

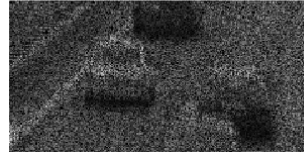
## 6.5 Results

### 6.6 Coded Snapshots- Cars

Coded snapshot with noise for T=3



Coded snapshot with noise for T=5

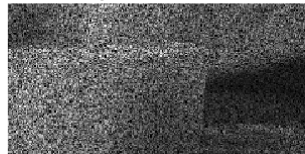


Coded snapshot with noise for T=7



### 6.7 Coded Snapshots- Flame

Coded snapshot with noise for T=5



Results of T=3 and T=7 for the `flame.avi` were not asked for, but are included in the results folder, if you wish to see them.

### 6.7.1 Cars, $T=3$

We observed a Relative Mean Squared Error = 0.0111



**Reconstruction for  $T=3$ , frame number=1**



**Reconstruction for  $T=3$ , frame number=2**



**Reconstruction for  $T=3$ , frame number=3**





### 6.7.2 Cars, $T=5$

We observed a Relative Mean Squared Error = 0.0191



### 6.7.3 Cars, $T=7$

We observed a Relative Mean Squared Error = 0.0277



Reconstruction for  $T=7$ , frame number=1



Reconstruction for  $T=7$ , frame number=2



Reconstruction for  $T=7$ , frame number=3



Reconstruction for  $T=7$ , frame number=4



Reconstruction for  $T=7$ , frame number=5



**Frame 6****Reconstruction for  $T=7$ , frame number=6****Frame 7****Reconstruction for  $T=7$ , frame number=7**

#### 6.7.4 Flame, $T=5$

We observed a Relative Mean Squared Error = 0.0012

Frame 1



Reconstruction for  $T=5$ , frame number=1



Frame 2



Reconstruction for  $T=5$ , frame number=2



Frame 3



Reconstruction for  $T=5$ , frame number=3



Frame 4



Reconstruction for  $T=5$ , frame number=4



Frame 5



Reconstruction for  $T=5$ , frame number=5

