# Assignment 1: CS 754, Advanced Image Processing

**Group Details:**
Vinit Awale (18D070067)
Piyush Bharambe (18D070019)

## Question 1

- Let $\boldsymbol{\theta}^\star$ be the result of the following minimization problem (BP): $\min \|\boldsymbol{\theta}\|_1$ such that $\|\boldsymbol{y} - \boldsymbol{\Phi\Psi\theta}\|_2 \leq \varepsilon$, where $\boldsymbol{y}$ is an $m$-element measurement vector, $\boldsymbol{\Phi}$ is a $m \times n$ measurement matrix ($m < n$), $\boldsymbol{\Psi}$ is a $n \times n$ orthonormal basis in which $n$-element signal $\boldsymbol{x}$ has a sparse representation of the form $\boldsymbol{x} = \boldsymbol{\Psi\theta}$. Notice that $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\eta}$ and $\varepsilon$ is an upper bound on the magnitude of the noise vector $\boldsymbol{\eta}$.

  Theorem 3 we studied in class states the following: If $\boldsymbol{\Phi}$ obeys the restricted isometry property with isometry constant $\delta_{2s} < \sqrt{2} - 1$, then we have $\|\boldsymbol{\theta} - \boldsymbol{\theta}^\star\|_2 \leq C_1 s^{-1/2} \|\boldsymbol{\theta} - \boldsymbol{\theta_s}\|_1 + C_2 \varepsilon$ where $C_1$ and $C_2$ are functions of only $\delta_{2s}$ and where $\forall i \in \mathcal{S}, \boldsymbol{\theta_{si}} = \theta_i; \forall i \notin \mathcal{S}, \boldsymbol{\theta_{si}} = 0$. Here $\mathcal{S}$ is a set containing the indices of the $s$ largest magnitude elements of $\boldsymbol{\theta}$.

  A curious student asks the following questions: '(1) It appears that the upper bound on $\|\boldsymbol{\theta} - \boldsymbol{\theta}^\star\|_2$ is reduced as $s$ increases, which goes against the very premise of compressed sensing. How do we address this apparent discrepancy? (2) It also appears that the error bound is independent of $m$. How do you address this? (3) Now consider that I gave you another theorem (called Theorem 3A), which is the same as Theorem 3 except that it requires that $\delta_{2s} < 0.1$. Out of Theorem 3 and Theorem 3A, which is the more useful theorem? Why? (4) It appears that if I set $\varepsilon = 0$ in BP, I can always reduce the upper bound on the error even if the noise vector $\boldsymbol{\eta}$ has non-zero magnitude. Am I missing something? If so, what am I missing?'

**Answer:**

1. The upper bound on $\|\boldsymbol{\theta} - \boldsymbol{\theta}^\star\|_2$ does not reduce as $s$ increases
   Although, the term $s^{-1/2}\|\boldsymbol{\theta} - \boldsymbol{\theta_s}\|_1$ decreases as the sparsity $s$ increases, we also have to consider the constants $C_1$ and $C_2$ which are increasing functions of $\delta_{2s}$. We also know that $\delta_{2s}$ is an increasing function of the sparsity ($s$) (This can be thought intuitively as with increasing s, the bounds on $||A\theta||_2$ should be larger in terms of $||\theta||_2$). We know that the composition of two increasing functions is also increasing and hence we can conclude that the constants $C_1$ and $C_2$ are also increasing functions of $s$. Hence the two effects of decreasing $s^{-1/2}\|\boldsymbol{\theta} - \boldsymbol{\theta_s}\|_1$ and the increase in $C_1$ and $C_2$ are counter-acting each other. Moreover, the problems that are under consideration are not equivalent. As, the sparsity increases we need more measurements of the same signal for the restricted isometry to be satisfied. Hence, as $s$ increases, we need more measurements of the same signal. Hence, the claim that the upper bound on $\|\boldsymbol{\theta} - \boldsymbol{\theta}^\star\|_2$ does not reduce as $s$ increases is not valid.

2. The error bound is not independent of $m$. For the exact reconstruction of a $s$-sparse signal, we need $m$ measurements of the same signal. which has the following constraint.

$$m > O(slog(\frac{n}{s}))$$

Without m satisfying the above constant, the matrix will not satisfy the RIP condition. Hence, the error bound is not independent of $m$.

3. The number of matrices which will satisfy the constraints of Theorem 3 are more than that satsfying the constraints of Theorem 3A. If we have a theorem which assures the same implication for more number of cases then it is more useful. Hence in this case Theorem 3 is more powerful than Theorem 3A.

4. The implications of Theorem 3 are based on the assumptions that

$$||y - \mathbf{\Phi\Psi}\boldsymbol{\theta}||_2 \leq \varepsilon$$

where the $\varepsilon$ is choosen based on the noise level. Hence, if we set $\varepsilon = 0$ in BP even in the presence of a non-zero noise level, we may obtain some x, however the guarantees of Theorem 3 cannot be applied as the conditions asked by Theorem 3 are not satisfied.

## Question 2

- We will prove why the value of the coherence between $m \times n$ measurement matrix $\mathbf{\Phi}$ (with all rows normalized to unit magnitude) and $n \times n$ orthonormal representation matrix $\mathbf{\Psi}$ must lie within the range $[1, \sqrt{n}]$ (both 1 and $\sqrt{n}$ inclusive). Recall that the coherence is given by the formula $\mu(\mathbf{\Phi}, \mathbf{\Psi}) = \sqrt{n} \max_{\substack{i \in \{0,1,\ldots,m-1\} \\ j \in \{0,1,\ldots,n-1\}}} |\mathbf{\Phi}^i \mathbf{\Psi}_j|$. Proving the upper bound should be very easy for you. To prove the lower bound, proceed as follows. Consider a unit vector $\boldsymbol{g} \in \mathbb{R}^n$. We know that it can be expressed as $\boldsymbol{g} = \sum_{k=1}^{n} \alpha_k^i \mathbf{\Psi}_{\boldsymbol{k}}$ as $\mathbf{\Psi}$ is an orthonormal basis. Now prove that $\mu(\boldsymbol{g}, \mathbf{\Psi}) = \sqrt{n} \max_{i \in \{0,1,\ldots,n-1\}} \frac{|\alpha_i|}{\sum_{j=1}^{n} \alpha_j^2}$. Exploiting the fact that $\boldsymbol{g}$ is a unit vector, prove that the minimal value of coherence is attained when $\boldsymbol{g} = \sqrt{1/n} \sum_{k=1}^{n} \mathbf{\Psi}_{\boldsymbol{k}}$ and that hence the minimal value of coherence is 1. [10 points]

**Answer:**
First we prove the upper bound. For this first consider two vector $v_i, v_j \in \mathbb{R}^{n \times 1}$ such that both the vectors have a unit norm (in 2-norm), i.e. $||v_i||_2 = 1$ and $||v_j||_2 = 1$. Now, we know that for any such two vectors a function given by the following equation is a valid inner product.

$$f(v_i, v_j) = v_i^T v_j = \langle v_i, v_j \rangle$$

Hence, using Cauchy-Schwarz inequality, we know that the upper bound on the innner product between any such two vectors is given by,

$$|\langle v_i, v_j \rangle| \leq |\sqrt{\langle v_i, v_i \rangle \langle v_j, v_j \rangle}|$$

Now, since we know that

$$\langle v_i, v_i \rangle = ||v_i||_2^2 = 1$$

and

$$\langle v_j, v_j \rangle = ||v_j||_2^2 = 1$$

we can show that the upper bound on the inner product is

$$|\langle v_i, v_j \rangle| \leq |\sqrt{1}| = 1 \tag{1}$$

Now, we know that the rows of the measurement matrix ($\mathbf{\Phi}^i \in \mathbb{R}^{1 \times n}$ ) are unit normalized i.e

$$||\mathbf{\Phi}^i||_2 = 1 \qquad \forall i \in \{0, 1, \ldots, m-1\} \tag{2}$$

Also the matrix $\mathbf{\Psi}$ is orthonormal, i.e. its column vectors ($\mathbf{\Psi} \in \mathbb{R}^{n \times 1}$ are mutually orthogonal and unit normalized, i.e.

$$||\boldsymbol{\psi}_j||_2 = 1 \qquad \forall j \in \{0, 1, \ldots, n-1\} \tag{3}$$

Since, $\mathbf{\Phi}^i$ is a row vector and $\mathbf{\Psi}_j$ is a column vector, we can apply Cauchy-Schwarz inequality on them considering $\mathbf{\Phi}^i\mathbf{\Psi}_j$ as the inner product. Hence, using Cauchy-Schwarz on the row-vector of the measurement matrix and the columns of the matrix $\mathbf{\Psi}$, we know that the upper bound on the inner product between any two vectors is given by,

$$
\begin{aligned}
&|\mathbf{\Phi}^i, \mathbf{\Psi}_j| \leq 1 \qquad \forall i \in \{0, 1, \ldots, m-1\} \qquad \forall j \in \{0, 1, \ldots, n-1\} \\
\implies &\sqrt{n} \max_{\substack{i \in \{0,1,\ldots,m-1\} \\ j \in \{0,1,\ldots,n-1\}}} |\mathbf{\Phi}^i\mathbf{\Psi}_j| \leq \sqrt{n} \quad \ldots \quad \text{(Using (1), (2) and (3))}
\end{aligned}
$$

$$
\implies \mu(\mathbf{\Phi}, \mathbf{\Psi}) \leq \sqrt{n} \tag{4}
$$

Now, let us prove the lower bound of the given inequality.
As $\mathbf{\Psi}$ is an orthonormal basis we can represent the $i^{th}$ row of the measurement matrix as,

$$
(\mathbf{\Phi}^i)^T = \sum_{k=1}^{n} \alpha_k^i \mathbf{\Psi}_k \qquad \forall i \in \{0, 1, \ldots, m-1\}
$$

The transpose is taken as $\mathbf{\Phi}^i$ is a row vector and $\mathbf{\Psi}_k$ is a column vector. Hence, we can write the above equation as,

$$
\mathbf{\Phi}^i = \sum_{k=1}^{n} \alpha_k^i \mathbf{\Psi}_k^T
$$

Also, since $\mathbf{\Phi}^i$ is unit normalized, we know that,

$$
\sum_{k=1}^{n} (\alpha_k^i)^2 = 1 \tag{5}
$$

Hence, we can write the coherence between $\mathbf{\Phi}^i$ and $\mathbf{\Psi}_k$ as,

$$
\begin{aligned}
\mu(\mathbf{\Phi}, \mathbf{\Psi}) &= \sqrt{n} \max_{\substack{i \in \{0,1,\ldots,m-1\} \\ j \in \{0,1,\ldots,n-1\}}} \left|\mathbf{\Phi}^i\mathbf{\Psi}_j\right| \\
&= \sqrt{n} \max_{\substack{i \in \{0,1,\ldots,m-1\} \\ j \in \{0,1,\ldots,n-1\}}} \left|(\sum_{k=1}^{n} \alpha_k^i \mathbf{\Psi}_k^T)\mathbf{\Psi}_j\right| \\
&= \sqrt{n} \max_{\substack{i \in \{0,1,\ldots,m-1\} \\ j \in \{0,1,\ldots,n-1\}}} \left|\alpha_j^i\right|
\end{aligned}
$$

Now, since we want to find the lower bound on $\mu(\mathbf{\Phi}^i\mathbf{\Psi}_j)$ and from equation (5) we know the constraints that each column of the measurement matrix should follow, we can claim that the coherence will be minimized when all the coefficients $\alpha_j^i$ of the $i^{th}$ row of the measurement matrix are equal.

Hence, the coefficients of that $i^{th}$ row of the measurement matrix are given by,

$$
\alpha_j^i = \pm\frac{1}{\sqrt{n}} \qquad \forall j \in \{0, 1, \ldots, n-1\}
$$

The minimized value of the coherence in this case can be given by,

$$
\mu_{min}(\mathbf{\Phi}, \mathbf{\Psi}) = \sqrt{n} \left|\pm\frac{1}{\sqrt{n}}\right|
$$

$$\implies \mu_{min}(\mathbf{\Phi}, \mathbf{\Psi}) = 1 \tag{6}$$

Hence, using equation (5) and (6) we can write,

$$1 \leq \mu(\mathbf{\Phi}, \mathbf{\Psi}) \leq \sqrt{n}$$

Hence proved

# Question 3

- Compressive sensing reconstructions involve estimating a sparse signal $\mathbf{x} \in \mathbb{R}^n, n \gg 2$ from a vector $\mathbf{y} \in \mathbb{R}^m (m \ll n)$ of compressed measurements of the form $\mathbf{y} = \mathbf{\Phi x}$ where $\mathbf{\Phi} \in \mathbb{R}^{m \times n}$ is the measurement matrix (assume there is no noise). Now answer the following questions, from first principles. **Do not merely quote theorems or algorithms.**

  1. If it is known that $\mathbf{x}$ has only 1 non-zero element and that the other elements are zero, can you uniquely estimate $\mathbf{x}$ if $m = 1$? If yes, how? If not, why not? Now further suppose, you knew beforehand the index (but not the value) of the non-zero element of $\mathbf{x}$? Does this help you any further? If yes, how? If not, why not?
  2. If it is known that $\mathbf{x}$ has only 1 non-zero element and that the other elements are zero, can you uniquely estimate $\mathbf{x}$ if $m = 2$? If yes, how? If not, why not?
  3. If it is known that $\mathbf{x}$ has only 2 non-zero elements and that the other elements are zero, can you uniquely estimate $\mathbf{x}$ if $m = 3$? If yes, describe an algorithm that is guaranteed to estimate it accurately. If not, explain why not, and explain whether there are any special instances of $\mathbf{\Phi}$ for which unique estimation is possible?
  4. Repeat part (c) with $m = 4$. [1+2+3+4=10 points]

**Answer:**

## a)

$\mathbf{y} = \Phi\mathbf{x}$, $y \in R^m$, $x \in R^n$, $\Phi \in R^{mxn}$

If m=1, we can calculate $\frac{y}{\Phi_j}$ for each $1 \leq j \leq n$. But since we do not know at which index the non zero element of x is, we cannot find the unique solution of x but we will have a set of finite possible solutions.

If the index (i) of the non zero element is also known, we can uniquely find x if $\Phi_i \neq 0$. If $\Phi_i = 0$, we cannot.

## b)

Let the index of the non zero element of x be j.

$$y_1 = \phi_{1j} x_j$$

$$y_2 = \phi_{2j} x_j$$

$$\frac{y_1}{y_2} = \frac{\phi_{1j}}{\phi_{2j}} \tag{1}$$

We can find the column j in $\Phi$ for which the above equation 1 is satisfied. Then we can easily calculate

$$x_j = \frac{y_1}{\phi_{1j}}$$

However, there can be multiple columns in $\Phi$ which can satisfy equation-1. So for unique solution, no two columns of $\Phi$ should have elements in same ratio.

## c)

Let i and j be the indices of the 2 non-zero elements of x.

$$y_1 = \phi_{1i}x_i + \phi_{1j}x_j$$

$$y_2 = \phi_{2i}x_i + \phi_{2j}x_j$$

$$y_3 = \phi_{3i}x_i + \phi_{3j}x_j$$

To uniquely find x, it is necessary that any 4 columns of $\Phi$ are linearly independent. But since we only have 3 rows, any 4 columns will be linearly dependent. This is because any 4 vectors in 3D space are always linearly dependent (which can be proved by the argument that- if they were linearly independent then we can take any 3 which will also be linearly independent, and use as basis to express the remaining vector as a linear combination of those 3 and hence the contradiction) Thus it is not possible to uniquely find x for any such $\Phi$

## d)

If m=4 and x has 2 non zero elements, we can uniquely find x if any 4 columns of $\Phi$ are linearly independent.

$$y_1 = \phi_{1i}x_i + \phi_{1j}x_j \tag{2}$$

$$y_2 = \phi_{2i}x_i + \phi_{2j}x_j \tag{3}$$

$$y_3 = \phi_{3i}x_i + \phi_{3j}x_j \tag{4}$$

$$y_4 = \phi_{4i}x_i + \phi_{4j}x_j \tag{5}$$

To find x, we can do the following,-

- Consider the set of all pairs of columns (i,j) of $\Phi$. loop over the set.
- For the measurements say $y_1$ and $y_2$, find the solution $(x_i, x_j)^T$ by taking the inverse of the subset 2x2 matrix.(here solving equations 2 and 3)
- Check if the solution satisfies the equations of the remaining two measurements (say equations 4 and 5 here). If it does, we have found the unique solution and so end loop.

# Question 4

- Consider compressive measurements of the form $\boldsymbol{y} = \boldsymbol{Ax} + \boldsymbol{v}$ for sensing matrix $\boldsymbol{A}$, signal vector $\boldsymbol{x}$, noise vector $\boldsymbol{v}$ and measurement vector $\boldsymbol{y}$. Consider the problem P1 done in class: Minimize $\|\boldsymbol{x}\|_1$ w.r.t. $\boldsymbol{x}$ such that $\|\boldsymbol{y} - \boldsymbol{Ax}\|_2 \leq e$. Also consider the problem Q1: Minimize $\|\boldsymbol{Ax} - \boldsymbol{y}\|_2$ w.r.t. $\boldsymbol{x}$ subject to the constraint $\|\boldsymbol{x}\|_1 \leq t$. Prove that if $\boldsymbol{x}$ is a unique minimizer of P1 for some value $e \geq 0$, then there exists some value $t \geq 0$ for which $\boldsymbol{x}$ is also a unique minimizer of Q1. Note that $\|\boldsymbol{x}\|_1$ and $\|\boldsymbol{x}\|_2$ stand for the L1 and L2 norms of the vector $\boldsymbol{x}$ respectively. [15 points] (Hint: Consider $t = \|\boldsymbol{x}\|_1$ and now consider another vector $\boldsymbol{z}$ with L1 norm less than or equal to $t$).

**Answer:**

**P1:** $\underset{x}{minimize}\ \mathbf{x}_1$; s.t. $\mathbf{y}\text{-}\mathbf{Ax}_2 \leq e$

**Q1:** $\underset{x}{minimize}\ \mathbf{y}\text{-}\mathbf{Ax}_2$; s.t. $\mathbf{x}_1 \leq t$

Now, consider $\mathbf{x}$ to be the unique minimizer for problem P1 for some $e \geq 0$

Let $t = \mathbf{x}_1$

Consider vector $\mathbf{z} \in R^n$, $\mathbf{z} \neq \mathbf{x}$ such that $\mathbf{z}_1 \leq t$

Now since $\mathbf{z}_1 \leq \mathbf{x}_1 = t$ and x is unique minimizer of P1, $\mathbf{z}$ should not satisty the constraint of P1. Thus,

$$\mathbf{y}\text{-}\mathbf{Az}_2 > e \geq \mathbf{y}\text{-}\mathbf{Ax}_2 \tag{1}$$

Above equation implies that $\mathbf{x}$ is the unique minimizer of Q1 problem.

Hence it is proved that, if $\mathbf{x}$ is a unique minimizer of P1 for some $e \geq 0$ then there exists $t \geq 0$ for which $\mathbf{x}$ is also the unique minimizer of Q1

# Question 5

- Here is our mandatory Google search question. Note that this is the only question for which you can perform a google search to get the answer. Your task is to search for a research paper which applies compressed sensing in any one application not covered in class. Some examples include air quality monitoring, optical microscopy, or any other. Answer the following questions briefly:

  1. Mention the title of the paper, where and when it was published, which venue (name of journal or conference or workshop) and include a link to the paper.
  2. Very briefly describe the hardware architecture used in the paper. You may refer to figures from the paper itself.
  3. What reconstruction techhnique or cost function does the paper adopt for the sake of compressive reconstruction in this application? [3+4+4=10 points]

**Answer:**

## a)

**Title:** Compressive Scanning Transmission Electron Microscopy
**Authors:** D. Nicholls, A. Robinson, J. Wells, A. Moshtaghpour, M. Bahri, A. Kirkland and N. Browning
**Date:** 22 Dec 2021
**Venue:** ICASSP 2022
**Link:** https://arxiv.org/pdf/2112.11955

## b) Hardware architecture

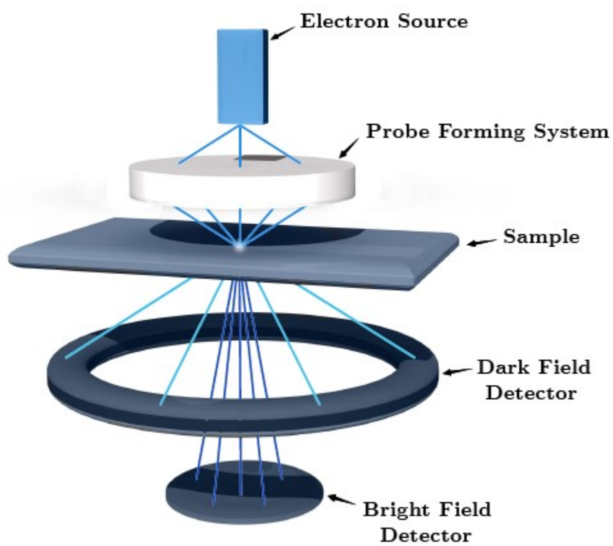



Figure 1: Scanning Transmission Electron Microscopy (STEM)

As shown in figure 1, STEM consists of an electron source, a probe forming system, a scan coil system, and an image forming system. The emitted electrons are condensed by the probe forming system. The probe is raster scanned over the sampling area. The authors used a line hop sampling scheme in which subsampling randomly the adjacent locations to the probe's line trajectory is performed. Fewer measurements are taken using this scheme to avoid the electron overdose problem. The transmitted electrons are then detected by a bright field detector. The formed image is the intensity of the resulting electron wave-function at each pixel, corresponding to a certain set of scan coordinates.

## c) Image reconstruction technique

The model is,

$$\mathbf{y} = \mathbf{P}_{\Omega}\mathbf{x} + \mathbf{n}$$

,where $\mathbf{P}_{\Omega} \in \{0, 1\}^{MxN}$ is the mask operator depending on the subsampling scheme and $\mathbf{n}$ is the noise.

The authors performed dictionary learning adopting a Bayesian non-parametric method called Beta Process Factor Analysis(BPFA) for reconstruction of the images. Each image patch is assumed to be sparse in a shared dictionary ($x_i = D\alpha_i$). BPFA approach allows to infer D, $\alpha$ and the noise statistics.Here BPFA is applied on problem P2 in a regularised form or the LASSO. BPFA prunes unneeded elements, and updates the sparsity pattern by using the posterior distribution of a Bernoulli process. It updates the weights and the dictionary from their Gaussian posteriors. These updates are done using Expectation Maximisation(EM). EM involves an expectation step to form an estimation of the latent variables ($\alpha_i s$),and a maximisation step to perform a maximum likelihood estimation to update other parameters.

## Question 6

- In class, we studied a video compressive sensing architecture from the paper 'Video from a single exposure coded snapshot' published in ICCV 2011 (See `http://www.cs.columbia.edu/CAVE/projects/single_shot_video/`). Such a video camera acquires a 'coded snapshot' $E_u$ in a single exposure time interval $u$. This coded snapshot is the superposition of the form $E_u = \sum_{t=1}^{T} C_t \cdot F_t$ where $F_t$ is the image of the scene at instant $t$ within the interval $u$ and $C_t$ is a randomly generated binary code at that time instant, which modulates $F_t$. Note that $E_u$, $F_t$ and $C_t$ are all 2D arrays. Also, the binary code generation as well as the final summation all occur within the hardware of the camera. Your task here is as follows:

  1. Read the 'cars' video in the homework folder in MATLAB using the 'mmread' function which has been provided in the homework folder and convert it to grayscale. Extract the first $T = 3$ frames of the video.
  2. Generate a $H \times W \times T$ random code pattern whose elements lie in $\{0, 1\}$. Compute a coded snapshot using the formula mentioned and add zero mean Gaussian random noise of standard deviation 2 to it. Display the coded snapshot in your report.
  3. Given the coded snapshot and assuming full knowledge of $C_t$ for all $t$ from 1 to $T$, your task is to estimate the original video sequence $F_t$. For this you should rewrite the aforementioned equation in the form $\boldsymbol{Ax} = \boldsymbol{b}$ where $\boldsymbol{x}$ is an unknown vector (vectorized form of the video sequence). Mention clearly what $\boldsymbol{A}$ and $\boldsymbol{b}$ are, in your report.
  4. You should perform the reconstruction using Orthogonal Matching Pursuit (OMP). For computational efficiency, we will do this reconstruction patchwise. Write an equation of the form $\boldsymbol{Ax} = \boldsymbol{b}$ where $\boldsymbol{x}$ represents the $i^{th}$ patch from the video and having size (say) $8 \times 8 \times T$ and mention in your report what $\boldsymbol{A}$ and $\boldsymbol{b}$ stand for. For perform the reconstruction, assume that each $8 \times 8$ slice in the patch is sparse or compressible in the 2D-DCT basis. Carefully work out the error term in the OMP algorithm, and explain this in your report!
  5. Repeat the reconstruction for all overlapping patches and average across the overlapping pixels to yield the final reconstruction. Display the reconstruction and mention the relative mean squared error between reconstructed and original Data:, in your report as well as in the code.
  6. Repeat this exercise for $T = 5, T = 7$ and mention the mention the relative mean squared error between reconstructed and original Data: again.
  7. **Note: To save time, extract a portion of about $120 \times 240$ around the lowermost car in the cars video and work entirely with it. In fact, you can show all your results just on this part. Some sample results are included in the homework folder.**
  8. Repeat the experiment with any consecutive 5 frames of the 'flame' video from the homework folder.

**Answer:**

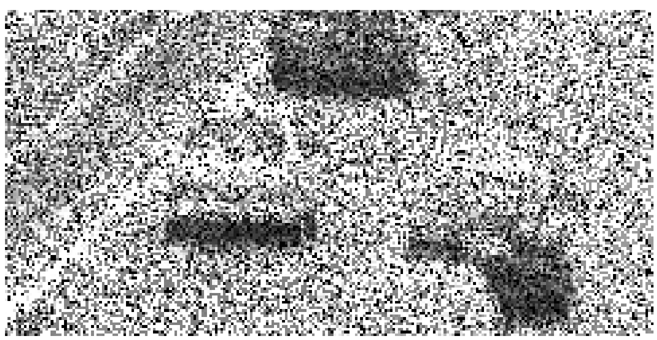
1. The first T = 3 frames of the given 'cars.avi' video resized to $120 \times 240$ around the lowermost car are shown below

Frame 1

Frame 2

Frame 3

2. The coded snapshot computed using the equation $E_u = \sum_{t=1}^{3} C_t \cdot F_t$ (where $F_t$ is the frame at time T, $C_t$ is the code at time T) is shown below.



Coded snapshot

3. The coded snapshot is computed using the following equation,

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

where $F_t$ is the frame at time t, $C_t$ is the binary code at time t. In our case the frames $F_t$ is the original signal and the coded snapshot is the compressed measurement. We have to represent the above equation in the form $y = Ax$ We can do so by vectorizing the signals.
To vectorize $F_t$ we can convert video frame to a vector by stacking columns one the other. All the video frames upto time T are then stacked one below the other to form a single column vector. Hence,

$$x = \text{Vectorized video frames} \in \mathbb{R}^{HWT \times 1}$$

The compressed measurement can be considered to be the vectorized version of the coded snapshot. This is obtained by stacking columns one the coded snapshot to form a single column vector. Hence,

$$y = \text{Vectorized coded snapshot} \in \mathbb{R}^{HW \times 1}$$

We can consider the matrix A to be of the form,

$$A = [\ diag(C_1(:))\ |\ diag(C_2(:))\ |\ ...\ |\ diag(C_T(:))] \in \mathbb{R}^{HW \times HWT}$$

i.e. we first stack the columns of the matrix $C_t$ to form a single vector, then construct a diagonal matrix with the diagonal entries as the vectorized version of the code $C_t$. Finally we stack the columns of the diagonal matrix formed from $C_t \forall t$ to form the matrix $A$.

4. In this problem we have added noise ($\eta$) from Gaussian distribution with $\sigma = 2$. Hence, with high proability the magnitude of $\eta$ will lie within 3 standard deviations from the mean. Hence, the error term in the OMP algorithm is can be set to be greater than $9m\sigma^2$ i.e.,

$$\epsilon \geq 9m\sigma^2$$

where $m$ is the number of measurements. Since, we are performing the reconstruction on $8 \times 8$ patches, we can assume that the number of measurements m are 64. Also, in the code we have converted the image pixel to range of 0 to 1. Hence, the sigma value should be modified to be 2/255 to account for the range of pixel values. Hence, the lower bound on epsilon should be set to,

$$\begin{aligned} \epsilon &\geq 9m\sigma^2 \\ &\geq 9 * 64 * 2/(255^2) = 0.0354 \end{aligned}$$

In the code we have set $\epsilon$ to be 0.1 to reduce the computational time for the reconstruction.

5. For T= 3, the RMSE between the original Data: and the reconstructed Data: is **0.0114**. The reconstructed video frames along with the original video frames are shown below.

Original Data: frame 1

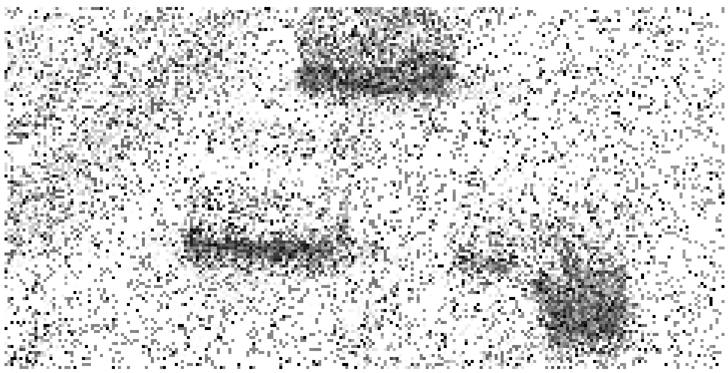
Reconstructed Data: frame 1

Original Data: frame 2

Reconstructed Data: frame 2

Original Data: frame 3 Reconstructed Data: frame 3

6. **For T = 5**
   The coded snapshot of the video for T= 5 is as follows



Coded snapshot for T = 5

The RMSE between the original Data: and the reconstructed Data: is **0.0193**.
The reconstructed video frames along with the original video frames are shown below.

Original Data: frame 1 Reconstructed Data: frame 1

Original Data: frame 2

Reconstructed Data: frame 2

Original Data: frame 3

Reconstructed Data: frame 3

Original Data: frame 4

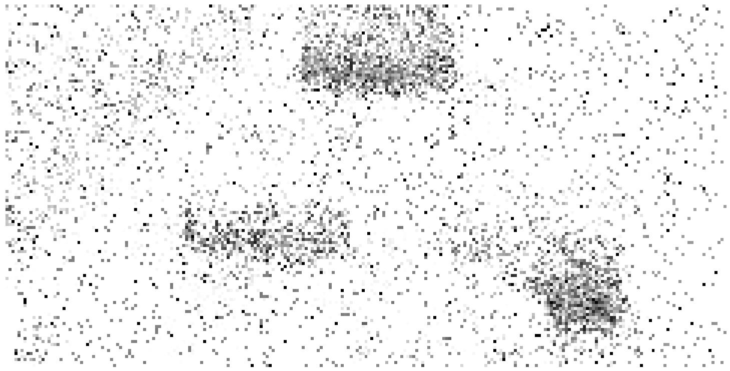
Reconstructed Data: frame 4

Original Data: frame 5

Reconstructed Data: frame 5

**For T = 7**

The coded snapshot of the video for T= 7 is as follows



Coded snapshot for T = 7

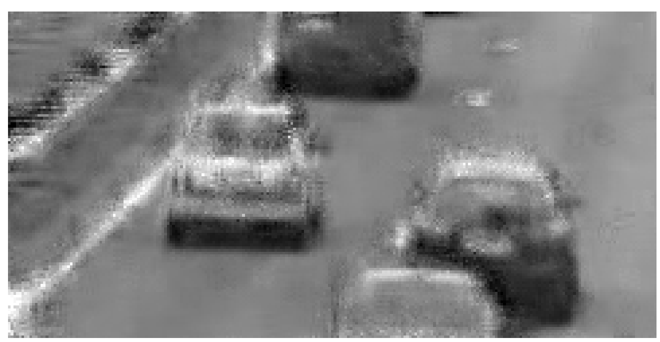
The RMSE between the original Data: and the reconstructed Data: is **0.0301**.

The reconstructed video frames along with the original video frames are shown below.

Original Data: frame 1

Reconstructed Data: frame 1

Original Data: frame 2



Reconstructed Data: frame 2

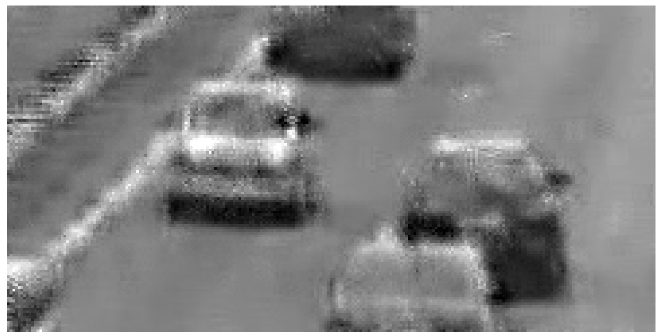
Original Data: frame 3

Reconstructed Data: frame 3

Original Data: frame 4

Reconstructed Data: frame 4

Original Data: frame 5

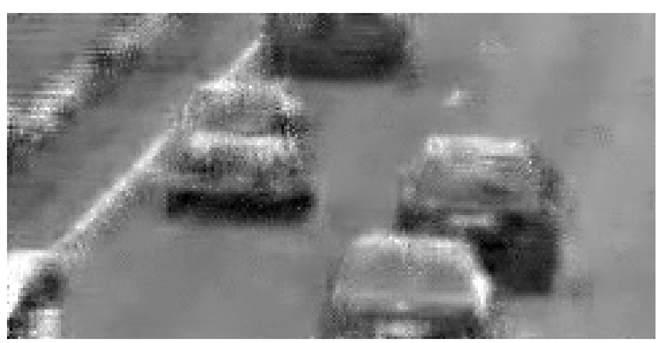


Reconstructed Data: frame 5

Original Data: frame 6

Reconstructed Data: frame 6

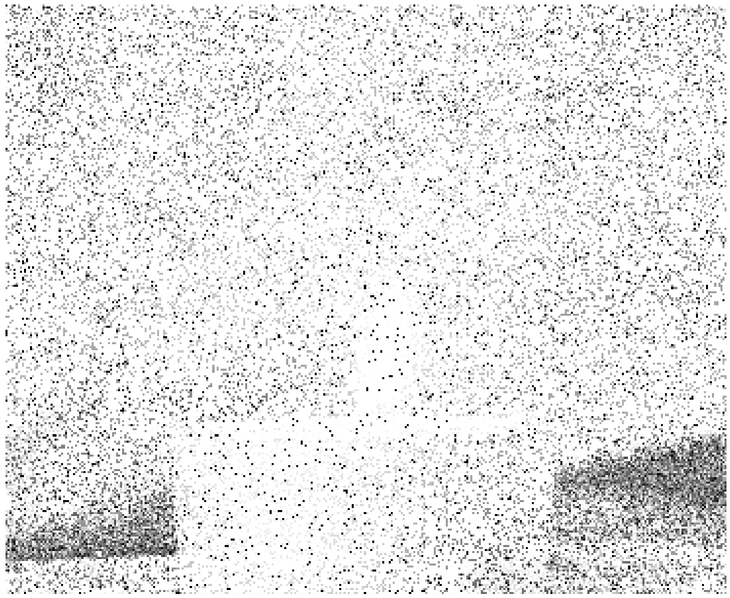
Original Data: frame 7



Reconstructed Data: frame 7

7. For the results on the 'cars.avi' video we have only considered the lower $120 \times 240$ pixels of the video.

8. **Results for the Flame Video** Note: We have considered the first 5 frames of the video. Also for this part we have considered the entire image frame for the reconstruction. The coded snapshot of the video for T= 5 is as follows



Coded snapshot for T = 5

The RMSE between the original Data: and the reconstructed Data: is **0.0011**.

The reconstructed video frames along with the original video frames are shown below.

Original Data: frame 1

Reconstructed Data: frame 1

Original Data: frame 2

Reconstructed Data: frame 2

Original Data: frame 3

Reconstructed Data: frame 3

Original Data: frame 4

Reconstructed Data: frame 4

Original Data: frame 5

Reconstructed Data: frame 5