Sensing Matrix Design Techniques

BTP-1 Report

Ameya Anjarlekar
170070013

Supervised By

Prof. Ajit Rajwade and Prof. Manoj Gopalkrishnan

Department of Electrical Engineering
Indian Institute of Technology, Bombay
Powai, Mumbai - 400 076.

December 7, 2020

Contents

1 Introduction

1.1 Compressed Sensing

Compressed sensing is a signal processing technique for efficiently acquiring and reconstructing a signal, by finding solutions to underdetermined linear systems. The signals are generally undersampled below Nyquist rate and then a lossy reconstruction is performed. For reconstruction, the sparsity of signals in certain bases is exploited. Generally for images, these bases could be DCT (Discrete Cosine Transform), Wavelet, Fourier or even dictionaries learned using training data. For reconstruction, algorithms such as OMP or LASSO are generally used.
Let x be the signal and Φ be the sensing matrix. Then $y = \Phi \cdot x$ represent the measurements which we obtain. Since x is a sparse signal in some basis, let $x = D\theta$ where D is the basis and θ is the sparse vector. Then the compressed sensing problems tries to find θ using y and $A = \Phi \cdot D$

1.2 Mutual Coherence and RIP

For efficient compressive recovery, the choice of dictionary and sensing matrix should satisfy certain conditions. CS theory states that Φ and D should be "incoherent" with each other. The coherence between Φ and D is defined as

$$\mu(\Phi, D) = \sqrt{n} \cdot max_{1 \leq j \leq m, 1 \leq i \leq n} | < \Phi^j, D_i > | \tag{1}$$

We need this coherence to be as minimum as possible. However, the bounds provided by mutual coherence are very loose. To get tighter bounds, RIP (Restricted Isometric Property) can be used. For integer S = 1,2,. . . ,n, the restricted isometry constant (RIC) δ_s of a matrix A of size m by n is the smallest number such that for any S-sparse vector θ, the following holds

$$(1 - \delta_s)||\theta||^2 \leq ||A \cdot \theta||^2 \leq (1 + \delta_s)||\theta||^2 \tag{2}$$

This provides tighter bounds, however calculations of RIC are computationally expensive. Through this work, we propose two approaches, 1) Using weighted coherence approach and 2) A faster version of the L_∞ approach. Using these sensing matrix metrics, we obtain sensing matrices which would improve the reconstruction loss compared to existing approaches.

2 Weighted Mutual Coherence

2.1 Recent Work and Proposed Approach

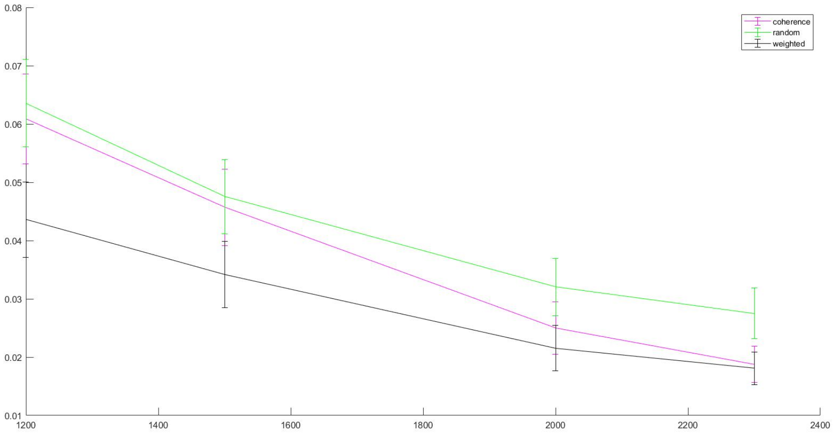
Various learning algorithms are proposed in literature for efficient sensing matrix design. [1] uses training data to estimate the position of atoms for a given image reconstruction problem. The weighted Mutual Coherence algorithm designed in [2] proposes a way to effectively reconstruct video frames by using a weighted mutual coherence approach through designing weights from preceding video frames. We extend this approach to effectively reconstruct images from the ORL dataset using some part of the data as training data.

The mutual coherence minimization approach proceeds by minimizing $||A^T \cdot A - I||$, whereas in the weighted mutual coherence approach we minimize $||(A^T \cdot A - I) \otimes (W^T W)||$ where W is a nx1 row vector of weights. The weights are calculated by summing up the values in the DCT basis of the training images. The training images (x_t) can be represented through a 2D-DCT matrix by $x = D\theta_t$ where D is the 2D-DCT matrix. The weights are then calculated by summing up over all the absolute values of θ_t.

This approach works because we now give more priority to positions (also referred as atoms) in the sparse vector which are more probable. Since the images in the dataset are related i.e. are iid samples from some probability distribution, the atoms in the sparse vectors would also be related. Thus, providing weights would help in better reconstruction.

2.2 Simulations

We use 36 images (with 10 emotions each) for training and then test on the remaining 4 images of the ORL Dataset. The images are corrupted by adding AWGN with std deviation 5. We reshape the ORL images to a size of 56x46 and perform reconstruction over the entire image. The following plot shows the relative norm error vs measurements for the case of 1)Random Gaussian sampling matrix 2) Sampling matrix obtained by minimizing mutual coherence 3) Sampling matrix obtained by minimizing weighted mutual coherence.



The following result show the comparison between the reconstructed images obtained from Gaussian sensing matrix and Weighted mutual coherence based sensing matrix for

1) Measurements = 100 (out of 154)

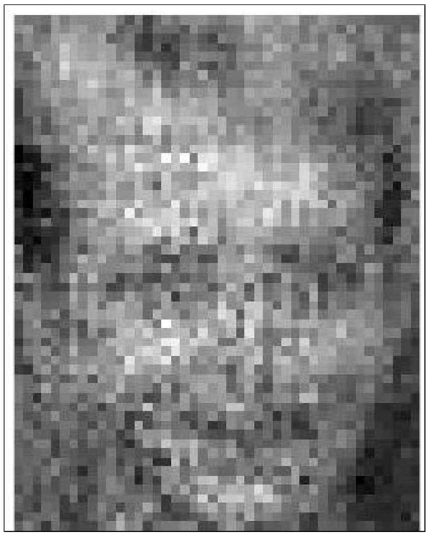


Figure 1: Gaussian sensing matrix

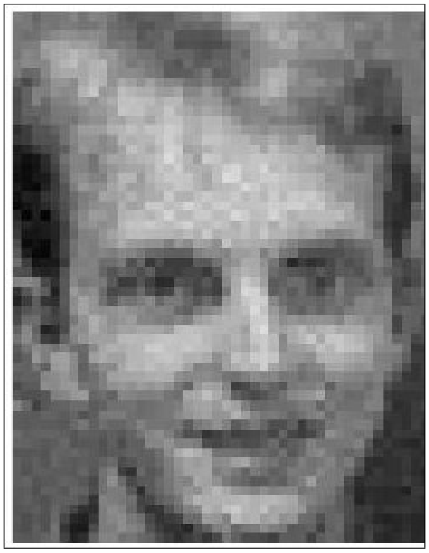


Figure 2: Weighted mutual coherence based sensing matrix

2) Measurements = 70 (out of 154)

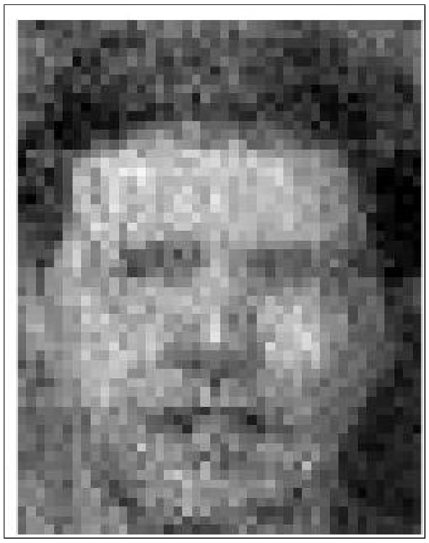


Figure 3: Gaussian sensing matrix

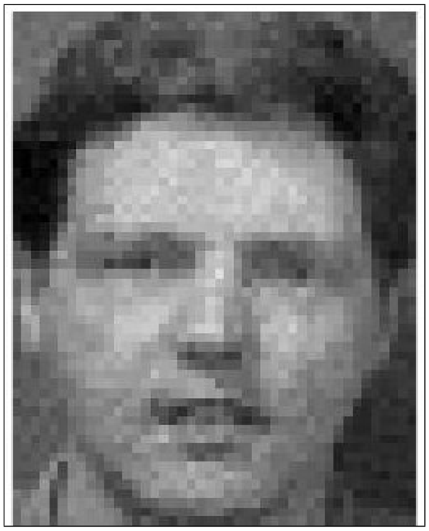


Figure 4: Weighted mutual coherence based sensing matrix

2.3 Handling rotations

We further extend this approach to reconstruct rotated versions of the test images. We consider two approaches. The first approach uses the training data for generating weights as before. The second approach also considers rotations of the training images to generate weights. We see that the second approach performs marginally better than the first one. We also find that the reconstruction error increases with the angle of rotation of the test image.

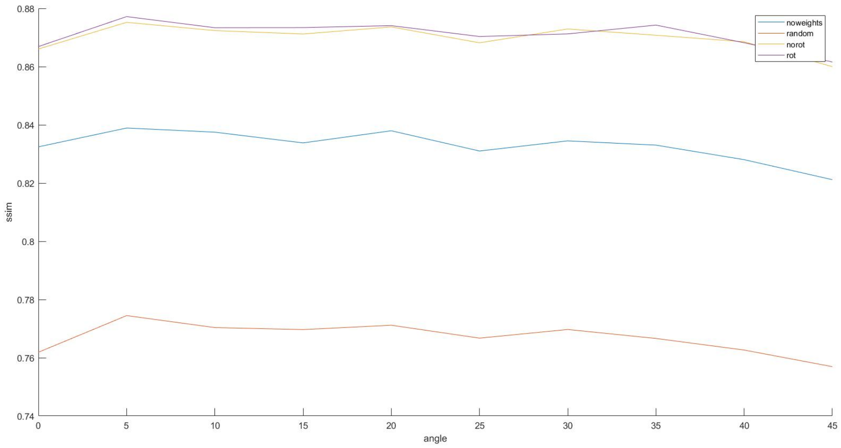


Figure 5: SSIM

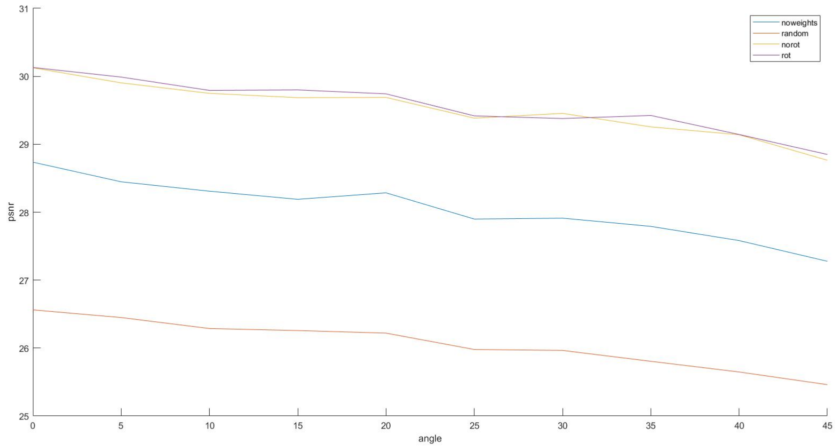


Figure 6: PSNR

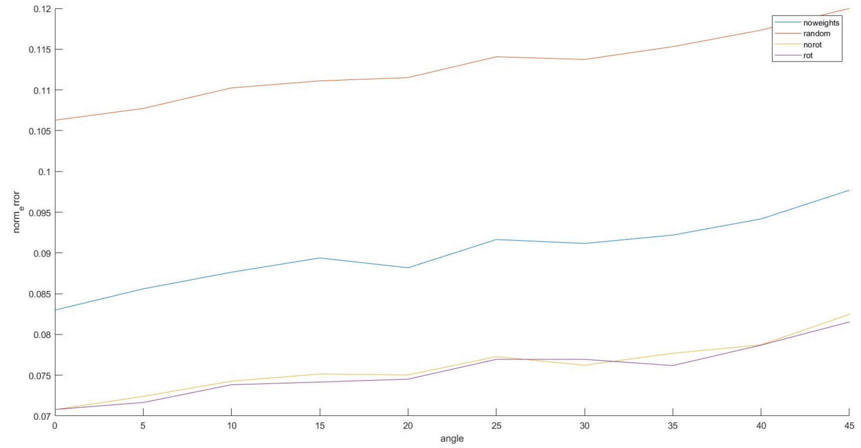


Figure 7: Relative Norm error

We also find that clipping the weights to an upper bound improves the reconstruction error. This is because excessive emphasis on a certain atoms would lead to a lesser emphasis on other atoms. Thus, differentiating the more probable atoms from the less probable atoms would become difficult.

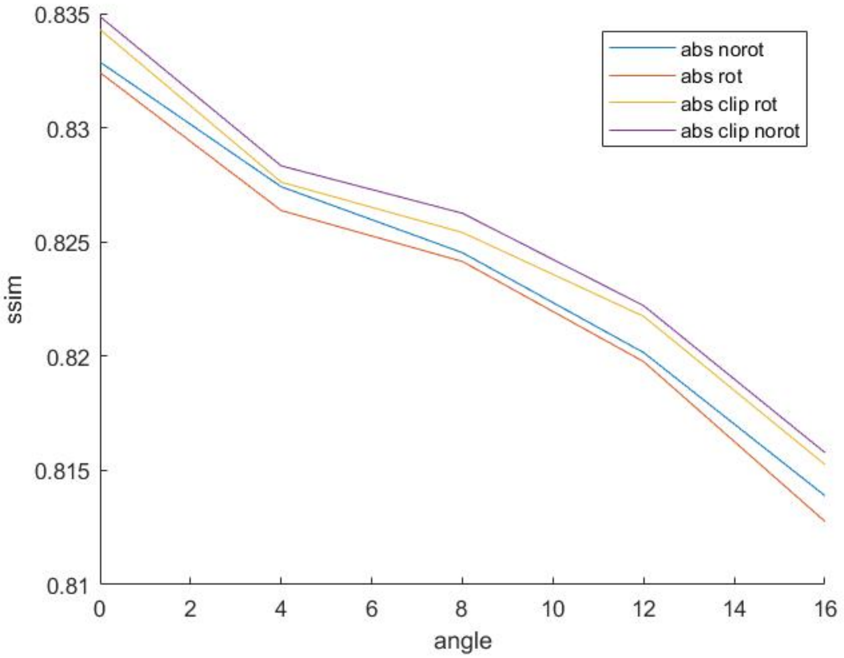


Figure 8: SSIM-clipping

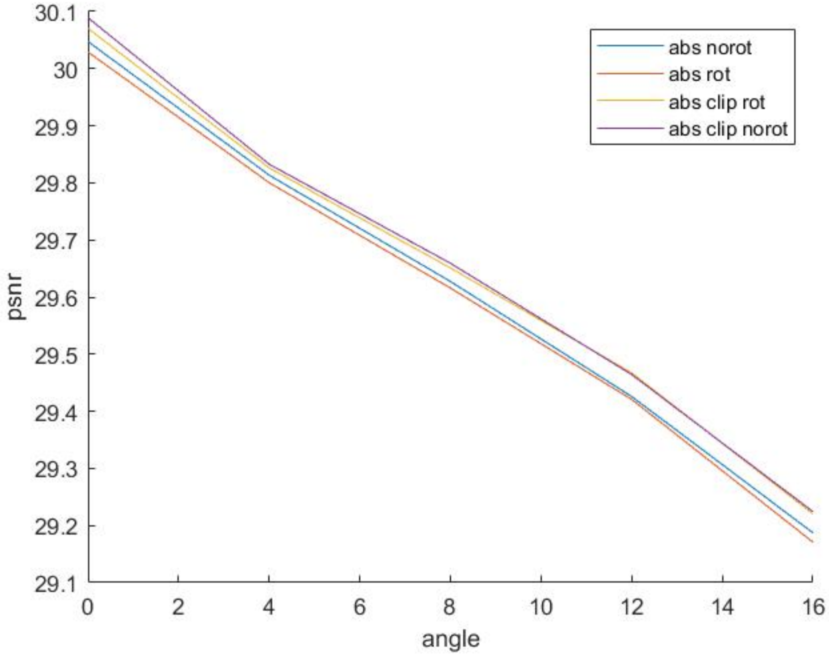


Figure 9: PSNR-clipping

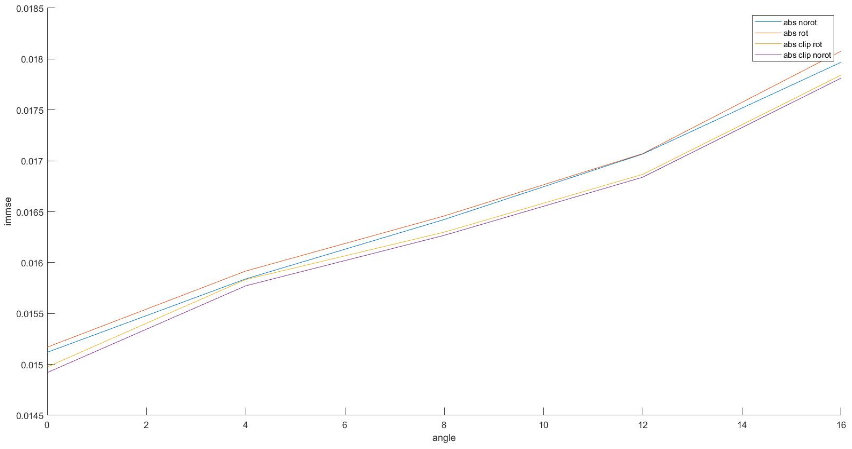


Figure 10: IMMSE-clipping

2.4 Handling scaling

We also consider the case of reconstructing scaled versions of the test images. Similar to handling rotations, we consider two approaches. The first approach uses the training data for generating weights as before. The second approach also considers scalings of the training images to generate weights. We see that in this case, the second approach works much better than the first one. The following plots depict the IMMSE error vs scaling factor for both the cases.

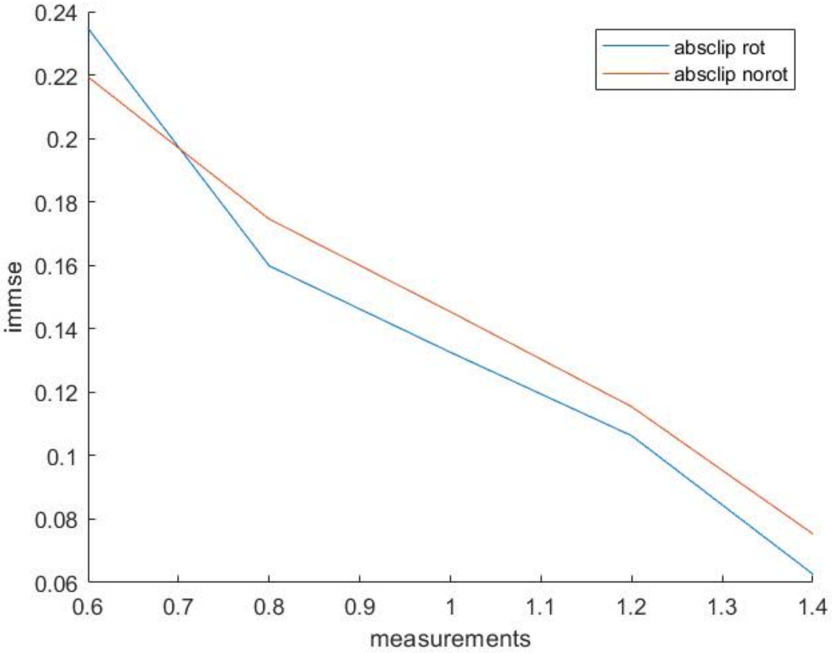


Figure 11: Measurements=300

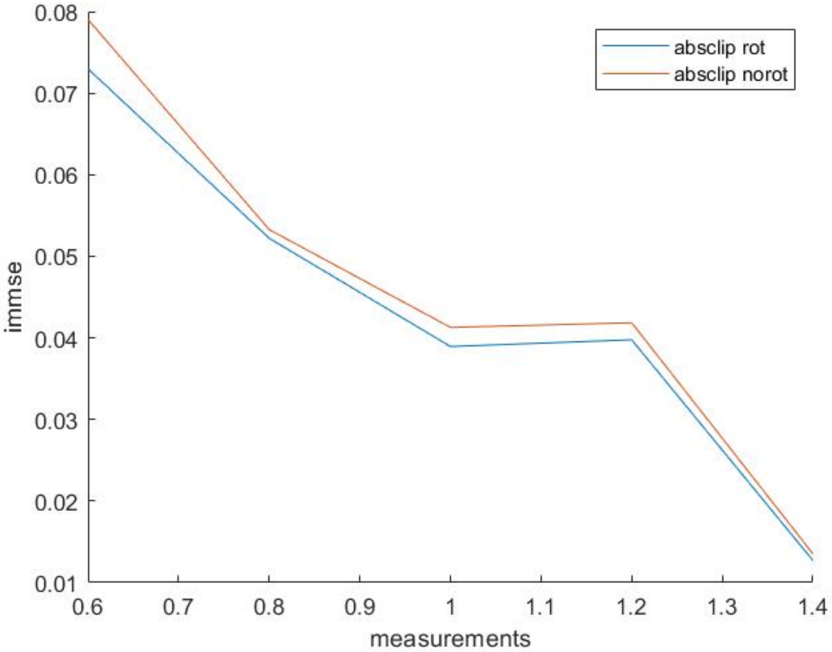


Figure 12: Measurements=700

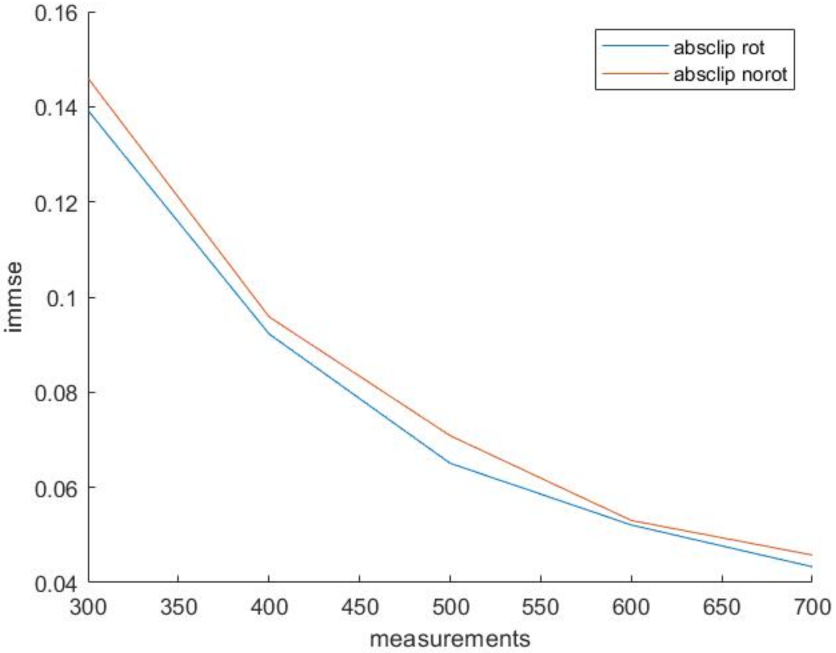


Figure 13: IMMSE vs Measurements

3 L_∞ approach

3.1 Recent Work

[3] proposes an approach which uses the $L_\infty norm$ to develop verifiable sufficient conditions and computable performance bounds of L_1-minimization based sparse recovery algorithms. We extend this work to develop faster bounds using L_0-minimization based sparse recovery algorithms. The paper proposes that:
Suppose x is k-sparse and the noise w satisfies $||w||_- \leq \epsilon$ for the basis pursuit denoising. Then,

$$||\hat{x} - x||_\infty \leq \frac{2\epsilon}{\omega_-(A, 2k)} \tag{3}$$

where - can represent any whole number and

$$\omega_-(Q, s) = min_{z:\frac{||z||_1}{||z||_\infty} \leq s} \frac{||Q \cdot z||_-}{||z||_\infty} \tag{4}$$

For calculating $\omega_-(Q, s)$ the following minimization problem is proposed in [2]

$$\omega_-(Q, s) = min_{i,\lambda} ||Q_i - Q(:, -i) \cdot \lambda||_- \quad s.t. ||\lambda||_1 \leq s - 1 \tag{5}$$

where s is the sparsity of the signal and Q(:,-i) denotes all the columns of Q except the ith column.

3.2 Proposed Approach

This procedure to compute ω is computationally complex [4]. Thus, using this procedure to find optimal sensing matrices is not feasible (since that would require computing ω several times. Therefore, we propose a faster algorithm for solving the following optimization problem.
Instead of taking the minimum over all possible i, we propose to use the mean over all i. This will facilitate easier computation of the gradient to update the sensing matrix. We also consider solving the L_0 problem as an approximation to the L_1 problem and also take - as 2. Thus, our optimization problem becomes

$$\omega_2(Q, s) = min_{i,\lambda} ||Q_i - Q(:, -i) \cdot \lambda||_2 \quad s.t. ||\lambda||_0 \leq s - 1 \tag{6}$$

The following problem can now be greedily solved using OMP. More accurate techniques such as taking all possible combinations of vectors such that $||\lambda||_0 \leq s-1$ can also be used but that will then be computationally tedious.
After calculating $\omega_2(Q, s)$ we would then use a gradient descent approach to maximize $\omega_2(Q, s)$ for finding the best sensing matrix.

3.3 Observations

We use 36 images (with 10 emotions each) for training and then test on the remaining 4 images of the ORL Dataset. The images are corrupted by adding AWGN with std deviation 5. We reshape the ORL images to a size of 56x46 and then perform non-overlapping patch reconstruction with patches of size 14x11. We run the gradient descent approach over 100 iterations at a learning rate of 0.01. Following are some of the image reconstruction results through this approach.

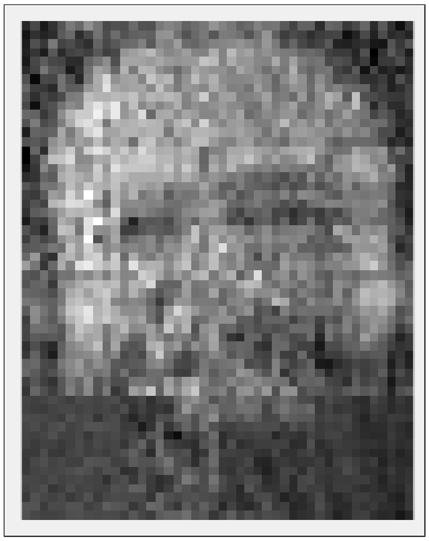


Figure 14: 0 iterations

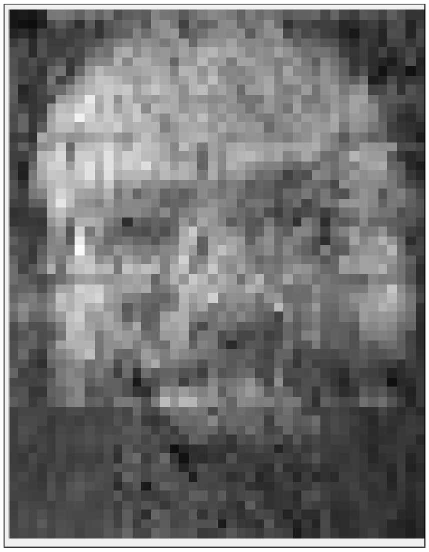


Figure 15: 100 iterations

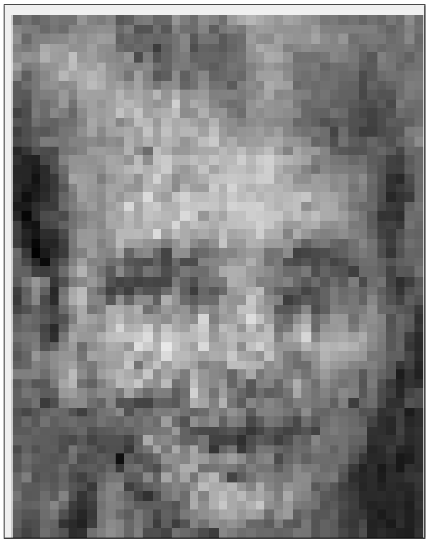


Figure 16: 0 iterations

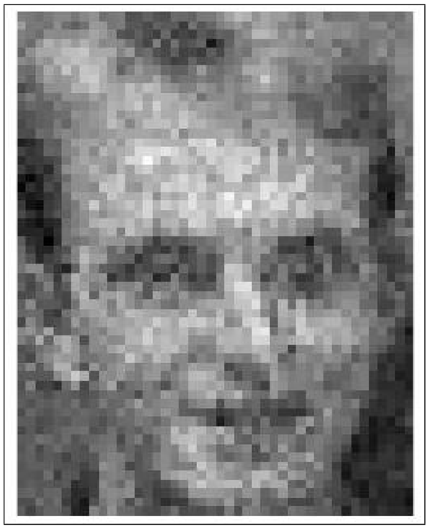


Figure 17: 100 iterations

The following is a plot showing the change in IMMSE error over 100 iterations.

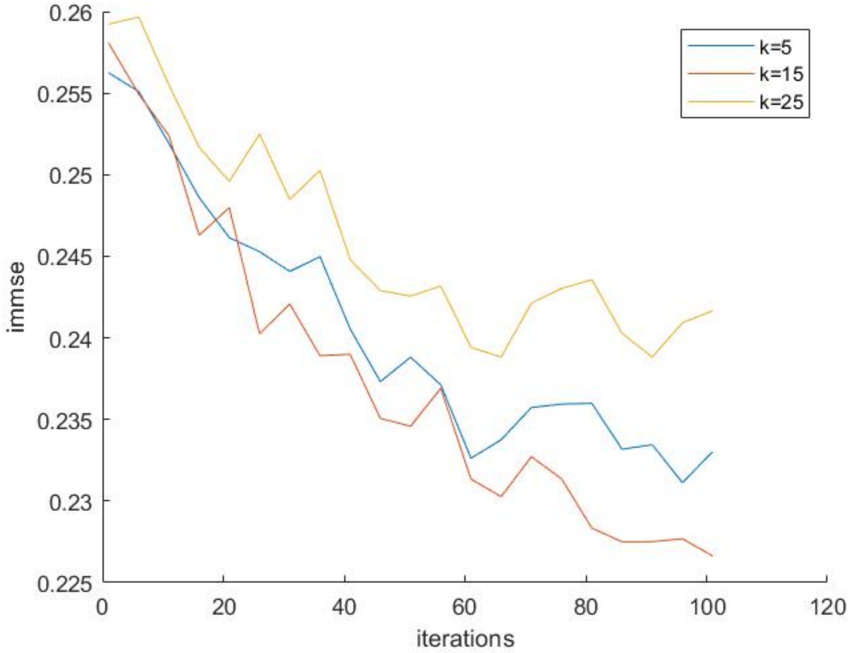


Figure 18: Measurements = 70

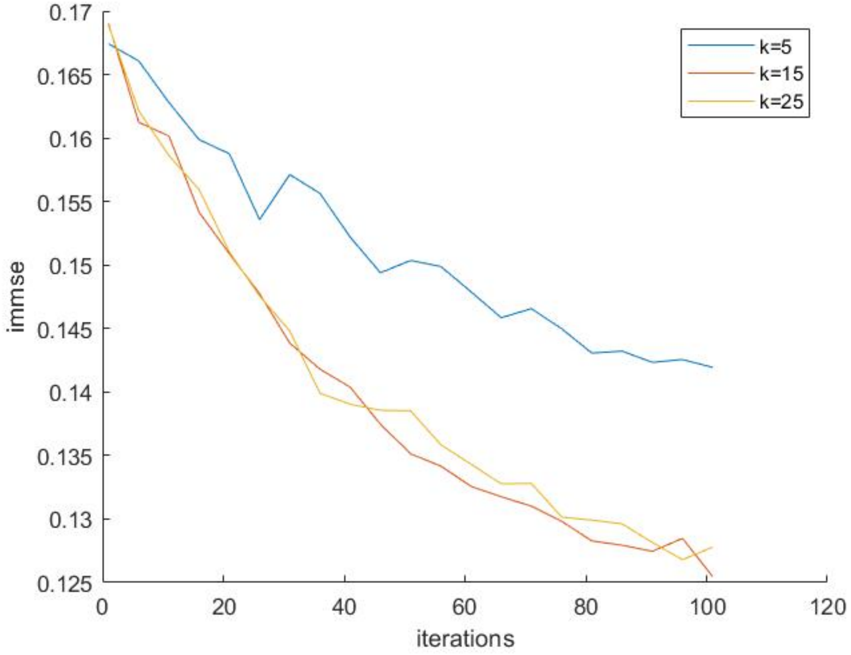


Figure 19: Measurements = 100

4 Comparison between both the approaches

In this section, we compare the results between both the approaches.
The following table compares the IMMSE error obtained from both the approaches for the same initial sensing matrix.

Table 1: Weighted mutual coherence vs L_∞ approach

measurements	IMMSE (Weighted mutual coherence)	IMMSE (L_∞)
70	0.0371	0.2266
100	0.0368	0.1255
120	0.0539	0.0819

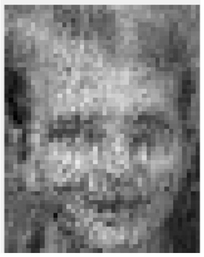


Figure 20: Random Gaussian Sensing matrix

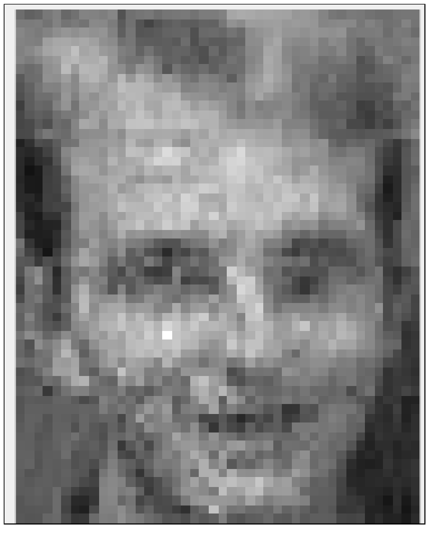


Figure 21: Sensing matrix obtained from L_∞ approach

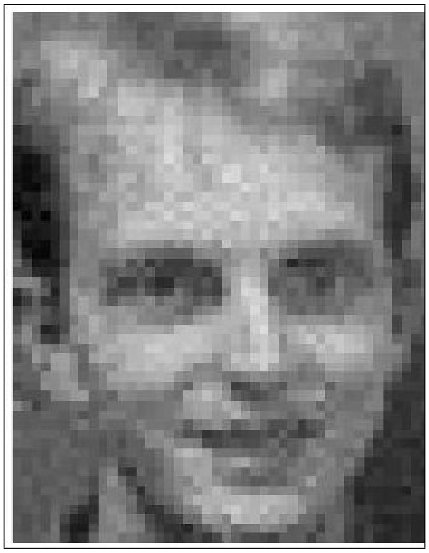


Figure 22: Sensing matrix obtained from Weighted mutual coherence approach

5 Future Work

1) We believe that the results from the L_∞ approach could be improved further with better optimization techniques. Currently, projected gradient descent is used for optimization. Further work could be done to find better optimization algorithms for this approach.

2) The sensing matrix design approaches could further be extended to find better sparse binary matrices for the group's work in group testing for COVID-19 [5].

3) Current work has focused largely on practical aspects. We further aim to provide theoretical results to support these findings.

References

[1] L. Baldassarre, Y. Li, J. Scarlett, B. Gözcü, I. Bogunovic, and V. Cevher, "Learning-based compressive subsampling," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 809–822, 2016.

[2] B. Li, L. Zhang, T. Kirubarajan, and S. Rajan, "Projection matrix design using prior information in compressive sensing," *Signal Processing*, vol. 135, pp. 36 – 47, 2017.

[3] G. Tang and A. Nehorai, "Computable performance bounds on sparse recovery," *IEEE Transactions on Signal Processing*, vol. 63, no. 1, pp. 132–141, 2015.

[4] G. Tang and A. Nehorai, "Semidefinite programming for computable performance bounds on block-sparsity recovery," *IEEE Transactions on Signal Processing*, vol. 64, no. 17, pp. 4455–4468, 2016.

[5] S. Ghosh, A. Rajwade, S. Krishna, N. Gopalkrishnan, T. E. Schaus, A. Chakravarthy, S. Varahan, V. Appu, R. Ramakrishnan, S. Ch, M. Jindal, V. Bhupathi, A. Gupta, A. Jain, R. Agarwal, S. Pathak, M. A. Rehan, S. Consul, Y. Gupta, N. Gupta, P. Agarwal, R. Goyal, V. Sagar, U. Ramakrishnan, S. Krishna, P. Yin, D. Palakodeti, and M. Gopalkrishnan, "Tapestry: A single-round smart pooling technique for covid-19 testing," *medRxiv*, 2020.