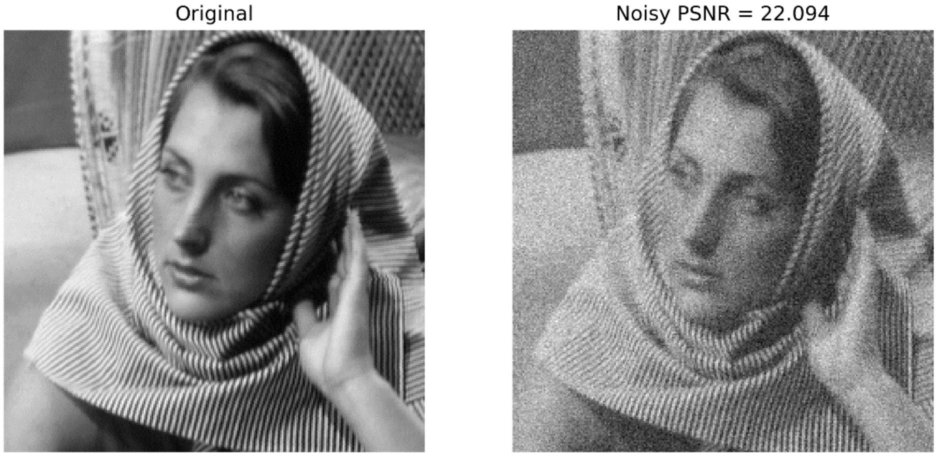
# Final Project Report

## Submitted by:

Hamza Khatieb 315398933
Alexander Shender 328626114

## Part A: Data Construction and Parameter-Setting:



Insert PSNR value of noisy image: 22.094

## Part B: DCT Dictionary

How is the error constraint satisfied for each patch in batch mode?

Just as explained in the Project task on the website, to satisfy the constraint, a calculation must be made on residuals for each patch individually to decide on its cardinality.
For each patch, in the cumulative sum array of an ascending array of residuals we are looking for the first value that exceeds $\epsilon^2$.
This means that the residual for this atom $D^T y_i \leq \epsilon^2$.
And that residuals for each of the next atoms (in the ascending sorted array) are:

$$D^T y_j \geq \epsilon^2 \ ; j = [(i+1) \dots n^2], where\ n^2 - number\ of\ atoms$$

We are obliged to choose to keep all of those atoms in order to nullify these residuals in the sparsely represented signal. This is required to keep the representation MSE according to the constraint:

$$\|y_i - D\alpha_i\|_2^2 \leq \epsilon^2$$

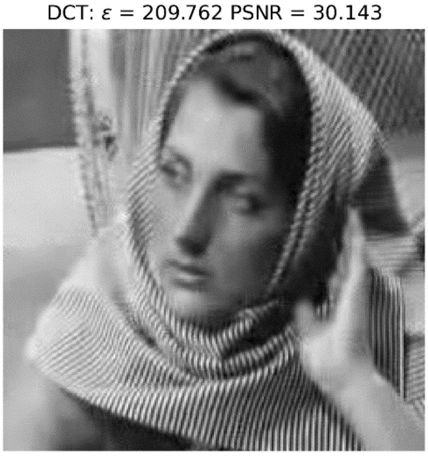
Insert average MSE of the reconstruction: 416.27

Insert average number of non-zeros of the reconstruction: 6.12

Discuss the obtained values below:

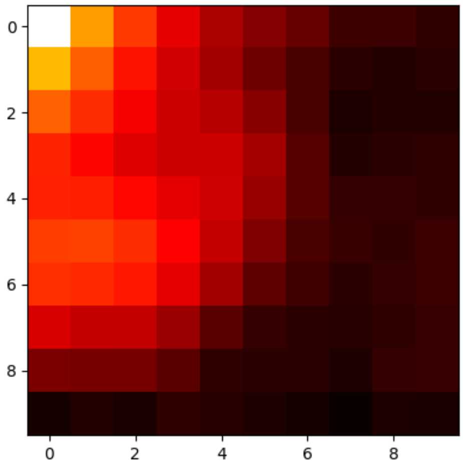
We can see that we have reached the average cardinality of 6.12, which may sound not bad, but there is yet nothing to compare to. But given that the dictionary is of size 100, this may be a reasonable compression.

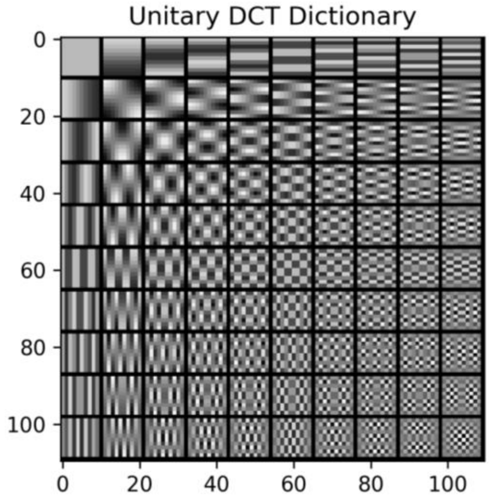
DCT reconstructed image:



Enter epsilon value: patch_size * sigma * sqrt(1.1) = 209.762
Enter PSNR of reconstruction: 30.143
Discuss the obtained results below:

We can observe that the reconstruction is not bad (visually). We can see different artifacts in the image, given that our dictionary is generic and may not accommodate closely all the various patterns in this specific image.
Out of interest we have created the following heatmap showing which atoms are used "the most". – By simply summing the of absolute values of coefficients of each atom, and using log10.

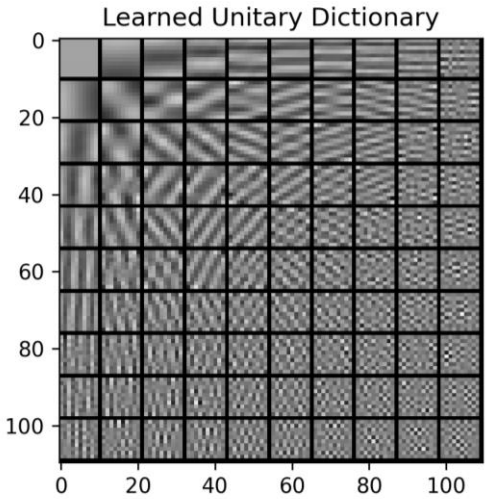




We can see that the atoms with vertical stripes are used a lot at predicted, and that the usage of the dictionary is not "uniform" – there are atoms which are barely used, but could help to represent the signal more sparsely with less cardinality.

## Part C: Procrustes Dictionary Learning

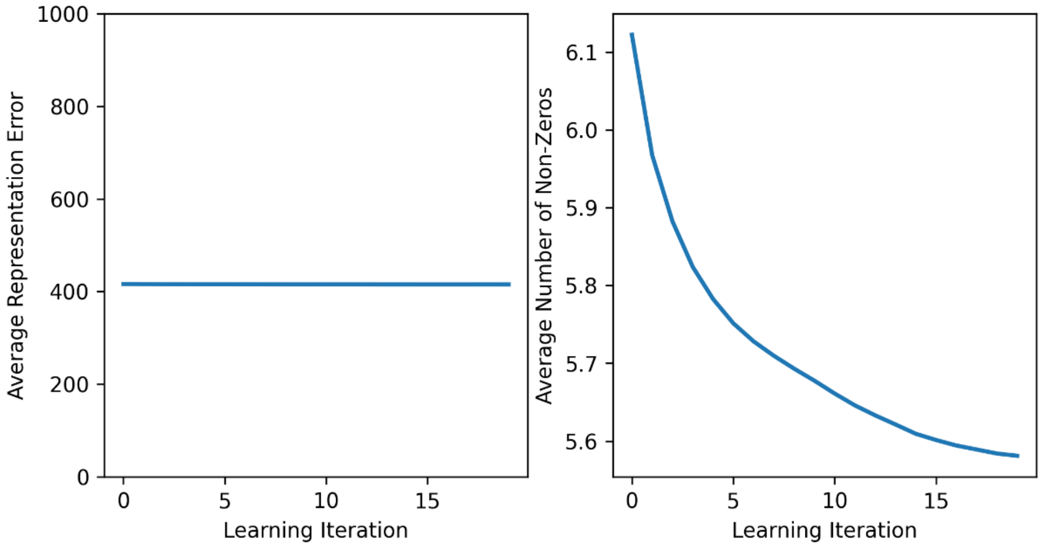
The obtained learned dictionary:



Discuss the obtained dictionary:

As in the Mid-term project, the learned Dictionary is much more 'domain-specific'. It has learned to represent better the specific patches that are present in the image that we want to represent sparsely.

Average MSE and number of nonzeros as a function of the iteration:



Discuss the obtained curves:

We can see that while the Representation Error almost stays the same, the average cardinality decreases. The solution becomes more dense over time.

Procrustes reconstructed image:

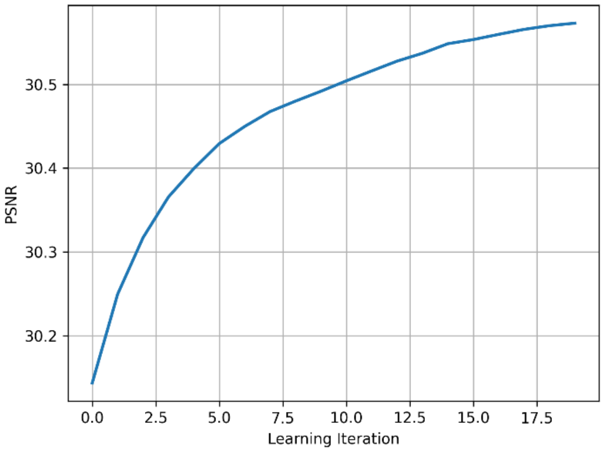
Enter epsilon value: 209.762

Enter PSNR of reconstruction: 30.579

Discuss the obtained results below:

Compare the results of the DCT dictionary and the learned dictionary:

We can see increase in the PSNR increasing from the case when just the DCT dictionary was used. Indeed, if we plot the PSNR against the iterations, we obtain the following graph:



Where we can clearly see improvement over iterations.

## Part D: SOS boosting

SOS-boosted reconstructed image:

Enter epsilon value: 230.738
Enter rho value: 1
Enter PSNR of reconstruction: 30.740

Discuss the obtained result and compare to DCT and learned dictionary:

We can see another slight improvement of the PSNR, compared to the learned dictionary. To compare the restorations visually we plot both restored (learned, learned + boosting):

Unitary: $\varepsilon$ = 209.762 PSNR = 30.579

Unitary SOS: $\varepsilon$ = 230.738 PSNR = 30.740

And it seems that the changes are too slight to be seen visually, but can be seen in PSNR improvement.

When we plot the differences between the two images (in absolute value, and inversed), we see the following picture, where we clearly see changes in various regions of the image, but mainly in the ones with the most texture.

Differences between the Learned and SOS boosted restoration