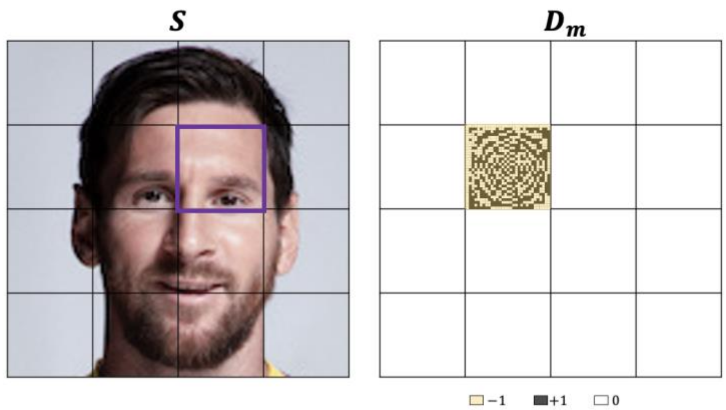
<u>Capturing Messi's Billion-Dollar Image with Compressed Sensing using DMDs</u>

In this exercise, you will understand the concept of a sparsifying basis and reconstruct Messi's **grayscale** image from a compressed representation acquired using a single pixel camera. This camera has a digital micromirror device (DMD) followed by a single image sensor. With this combination, it is possible to obtain inner products of the image with any matrix whose entries belong to {1,-1,0}. The image here will be reconstructed patch-by-patch, where each patch is of size 32x32.



The measurements used to reconstruct the image are of the form

$$y_m = \langle \boldsymbol{S}, \boldsymbol{D}_m \rangle + v_m,$$

where $v_m \sim \mathcal{N}(0, \sigma^2)$. Here, we use a variance of $\sigma^2 = 10^{-3}$. Note that both $\boldsymbol{S}$ and $\boldsymbol{D}_m$ are real-valued matrices of size 32x32.

You have the following tasks for this exercise:

a) **Identify a good sparsifying basis**: Create a plot that shows the mean squared error (MSE-dB) in the K-sparse approximation of the image when the 2D-DCT (32×32) and the Identity (32×32) are used as the sparsifying bases. The MSE should be calculated by averaging the results over 16 different 32×32 patches across the image. Analyze the two curves in this plot to determine which of the two sparsifying bases is more suitable for this image.

b) **Reconstruct the image from single pixel measurements using ISTA and OMP**:
For each patch, $M < 1024$ measurements will be acquired by applying random weights at the DMD. Note that the DMD here is of size 128×128. To acquire single pixel measurements for the patch of interest, these weights are chosen uniformly at random from {1,-1} at the locations within the patch and are set to 0 outside the patch. Based on your conclusion from exercise a), use an appropriate sparsifying dictionary and implement ISTA and OMP for sparse recovery of all the 32×32 patches from the compressed representation.

Deliverables:

- Matlab/ Octave/ Python implementations for exercises a) and b).

- Plot showing the MSE (dB) in the K-sparse approximation.
- For ISTA and OMP, show the reconstructed 128x128 image and plot the PSNR in dB for subsampling ratios of 20%, 40%, 60% and 80%.
- A short paragraph summarizing your conclusions based on the exercise.

Useful information and tips:

- Download image from brightspace [link].
- Convert image to grayscale for further processing. See Matlab implementation below:

```
Img=imread('MESSI_image.jpeg');
Img=im2double(Img);
Img=Img(:,:,1); % grayscale image
```

- We use $\boldsymbol{X}$ to denote the 2D-DCT transform of an $N \times N$ image $\boldsymbol{S}$. The relation between $\boldsymbol{X}$ and $\boldsymbol{S}$ is given by $\boldsymbol{S} = \boldsymbol{U}\boldsymbol{X}\boldsymbol{U}^T$, where $\boldsymbol{U}$ is the $N \times N$ DCT matrix defined in the Appendix of Homework 1. You can also obtain the matrix $\boldsymbol{U}$ using the function dctmtx(N) in Matlab.
- To speed up the computations, you could use the same CS matrix to acquire measurements of all the 16 patches in the image. Further, you may also try to run sparse recovery in parallel for the 16 patches.
- To avoid feeling overwhelmed as the deadline approaches, consider following this timeline to help you pace your progress.
  - Complete exercise a) after Lecture 7
  - Setup measurement model after Lecture 7.
  - Implement ISTA-based sparse recovery using the above measurement model after lecture 8.
  - Implement OMP-based sparse recovery after lecture 9.