1 Pre-Recorded Tasks

1.1 2D-DFT

1. Why is the 2D-DFT separable? Look at the definition formula!
2. The following matrix of pixel values is given:

$$x[n_1, n_2] = \begin{bmatrix} 10 & 10 & 20 & 20 \\ 10 & 10 & 20 & 20 \\ 10 & 10 & 30 & 30 \\ 5 & 5 & 40 & 40 \end{bmatrix}$$

   (a) The 2D-DFT matrix of the signal above was calculated but 2 errors occured. Find these mistakes by taking into account the DFT-properties of symmetry and real valued elements!
   Hint: Script Page 4-25

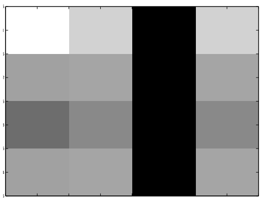
$$X[k_1, k_2] = \begin{bmatrix} 290 & -75+75j & -20-30j & -75-75j \\ -20+30j & -15-35j & 0 & 35+15j \\ -10 & 15-15j & 0 & 15+15j \\ -20-30j & 35+15j & 0 & -15+35j \end{bmatrix}$$

   (b) The 2D-DFT above was calculated by first transforming each line (separability). No errors were made in this step and the following matrix shows the result:
   Hint: Use the transformation matrix T_{DFT} for the 1-dimensional DFT of length 4.

$$X_R[k_1, k_2] = \begin{bmatrix} 60 & -10+10j & 0 & -10-10j \\ 60 & -10+10j & 0 & -10-10j \\ 80 & -20+20j & 0 & -20-20j \\ 90 & -35+35j & 0 & -35-35j \end{bmatrix}; \qquad T_{DFT} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

   Use this matrix to reconstruct the wrong elements in the 2D-DFT matrix!

   (c) The pictorial visualization of the logarithmic absolute value of the 2D-DFT transformation is as follows:



   Explain what you can derive from this picture!

2 Self-Study Matlab Tasks

2.1 Image filtering with the help of the DFT

The DFT shall now be used for filtering our images. Create a new MATLAB script for these experiments!

1. We now want to work with larger images. For example use the file `london_panorama.jpg` with 10 MPixels and convert it into a gray-scale image using the `rgb2gray`-function. As a filter use a large Gaussian blur filter `fspecial('gaussian',199,20)`. Transform both to the frequency domain with `X = fft2(x)`. What length should you use for the transformation of both? The function `padarray()` might be helpful!

2. Multiply the result element-by-element and transform them back with `y = ifft2(Y)`. What happens at the border?

3. The Gaussian filter has its center in the middle and is non-causal. Adjust your zero-padding to respect this and wrap around the non-causal part of the filter. The center of the filter needs to be at position $(1, 1)$. What happens to the filtered image?

4. The border is much better now but we still have a circular convolution in the filtering. To prevent circular convolution, pad the image to length of $(\text{image} + \text{filter} - 1)$ with additional zeros before filtering. Remove the additional border after filtering. How does the filtered result look like?

5. Now pad the image with `'symmetric'` border extension data instead of zeros. Can you see the difference?

1 Pre-Recorded Tasks

1.1 2D-DFT

1. Why is the 2D-DFT separable? Look at the definition formula!

$$X[k_1,k_2] = \sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} x[n_1,n_2]\cdot e^{-\frac{j2\pi k_1\cdot n_1}{N_1}}\cdot e^{-\frac{j2\pi k_2\cdot n_2}{N_2}} = \sum_{n_1=0}^{N_1-1} e^{-\frac{j2\pi k_1\cdot n_1}{N_1}} \underbrace{\sum_{n_2=0}^{N_2-1} x[n_1,n_2]\cdot e^{-\frac{j2\pi k_2\cdot n_2}{N_2}}}_{\substack{\text{may be calculated}\\ \text{in advance:}\\ \text{Transformation}\\ \text{of the rows}}}$$

2. The following matrix of pixel values is given:

   (a) The 2D-DFT matrix of the signal above was calculated but 2 errors occured. Find these mistakes by taking into account the DFT-properties of symmetry and real valued elements!

$$X[k_1,k_2] = \begin{bmatrix} 290 & -75+75j & \underline{-20-30j} & -75-75j \\ -20+30j & -15-35j & 0 & \underline{35+15j} \\ -10 & 15-15j & 0 & 15+15j \\ -20-30j & \underline{35+15j} & 0 & -15+35j \end{bmatrix}$$

   (b) Use this matrix to reconstruct the wrong elements in the 2D-DFT matrix!

$$X[k_1,k_2] = T_{DFT}\cdot X_R[k_1,k_2] = \begin{bmatrix} 290 & -75+75j & \underline{0} & -75-75j \\ -20+30j & -15-35j & 0 & \underline{35-15j} \\ -10 & 15-15j & 0 & 15+15j \\ -20-30j & 35+15j & 0 & -15+35j \end{bmatrix}$$

   (c) The pictorial visualization of the logarithmic absolute value of the 2D-DFT transformation is as follows:
   Explain what you can derive from this picture!
   The black vertical line at the position of the alternating coefficients indicates that there are no rapid changes in horizontal direction. In fact, you may derive directly that a subsampling by a factor of 2 is possible with no information loss. That is because there are only 2 independent frequencies in the signal and one of them (the highest part) is 0. You can also see that the DC-part has more energy than any other coefficient.

2 Self-Study Matlab Tasks

2.1 Image filtering with the help of the DFT

The DFT shall now be used for filtering our images. Create a new MATLAB script for these experiments!

1. We now want to work with larger images. For example use the file `london_panorama.jpg` with 10 MPixels and convert it into a gray-scale image using the `rgb2gray`-function. As a filter use a large Gaussian blur filter `fspecial('gaussian',199,20)`. Transform both to the frequency domain with `X = fft2(x)`. What length should you use for the transformation of both? The function `padarray()` might be helpful!

2. Multiply the result element-by-element and transform them back with `y = ifft2(Y)`. What happens at the border?

3. The Gaussian filter has its center in the middle and is non-causal. Adjust your zero-padding to respect this and wrap around the non-causal part of the filter. The center of the filter needs to be at position $(1,1)$. What happens to the filtered image?

4. The border is much better now but we still have a circular convolution in the filtering. To prevent circular convolution, pad the image to length of (image + filter − 1) with additional zeros before filtering. Remove the additional border after filtering. How does the filtered result look like?

5. Now pad the image with `'symmetric'` border extension data instead of zeros. Can you see the difference?