ECE 285 – Assignment #6

Writing part

---

1 Exercise 1 (Properties of moving average filters)

Let $\nu \in \mathbb{R}^n$, with $n = 2p+1$, be a 1d real periodical convolution kernel $\nu_k = \nu_{n+k}$, such that

- $\nu_{-k} = \nu_k$ for all k,
- $\nu_k > 0$ for all k,
- $\sum_{k=-p}^{p} \nu_k = 1$.

Denote by $\hat{\nu}$ the discrete Fourier transform of ν: $\hat{\nu}_k = \sum_{l=-p}^{p} \nu_l e^{-i2\pi\frac{lk}{n}}$.

1. Show that $\hat{\nu}_0 = 1$.
2. Show that $\|\hat{\nu}\|_\infty = 1$. (recall that $\|\hat{\nu}\|_\infty = \max_k |\hat{\nu}_k|$)
3. Show that $\hat{\nu}_k = \sum_{l=-p}^{p} \nu_l \cos\left(2\pi\frac{lk}{n}\right)$.
4. Show that $\hat{\nu}_k \in (-1, 1]$. (Hint: use *Proof by contradiction*)
5. Show that $\|y * \nu\|_2 \leqslant \|y\|_2$. (Use that $y * \nu = Hy$ with $H = F^{-1}\begin{pmatrix}\hat{\nu}_0 & & & \\ & \hat{\nu}_1 & & \\ & & \ddots & \\ & & & \hat{\nu}_{n-1}\end{pmatrix}F$)
6. Show that $P = \lim_{n\to\infty} H^n$ has only 0 and 1 eigenvalues. How is P called?
7. Assume $|\hat{\nu}_k| < 1$ for all $k \neq 0$. Show that for all k, $(y * \underbrace{\nu * \ldots * \nu}_{n\to\infty \text{ times}})_k = \frac{1}{n}\sum_{l=0}^{n-1} y_l$. Conclude.

Practical part

---

2 Exercise 2 (Spectral analysis)

1. Create a script `analyse_spectrum_simple.m` that creates an image with an horizontal sinusoidal `a`:

```
x = (1:256)';
y = pi / 16 * ones(256, 1);
a = sin(x * y');
```

Visualize this image and the amplitude of its spectrum in one single figure as

```
figure
subplot(1, 2, 1)
imagesc(a)
colormap(gray);
axis image
subplot(1, 2, 2)
imagesc(abs(fftshift(fft2(a))));
colormap(gray);
axis image
```

Explain what you observe: number of bright points, their locations, the distance between them. Use the 'data cursor' to inspect their values.

2. Repeat the same analysis for

   - a vertical sinusoidal:

     ```
     xx = 3*pi/32 * ones(256, 1);
     yy = (1:256)';
     a = sin(xx * yy');
     ```

   - a variant of the vertical sinusoidal:

     ```
     x3 = (3 + 1/8) * pi/32 * ones(256, 1);
     a = sin(x3 * yy');
     ```

   - a diagonal sinusoidal

     ```
     a = sin(x * y' + xx * yy');
     ```

   - a square

     ```
     a = zeros(256, 256);
     a(63:191, 63:191) = 1;
     ```

   - a sinusoidal in a window

     ```
     a = a .* sin(x * y');
     ```

3. Download `assignment6.zip` and extract the data:

   - `assignment6/fftgrid.m`
   - `assignment6/house.png`
   - `assignment6/lady.png`
   - `assignment6/map.png`
   - `assignment6/montreuil.png`

   Create a script `analyse_spectrum_images.m` that loads the image `a = house`. Display this image and the logarithm of its spectrum as

   ```
   imagesc(log(abs(fftshift(fft2(a)))));
   colormap(gray);
   axis image
   ```

   What are the horizontal, vertical and oblique structures that you observe?

4. Repeat with the image `map`, `montreuil` and `lady`, and interpret what you observe.

5. For the three images, perform a spatial sub-sampling by a factor 2 in each direction

   ```
   a = a(1:2:end, 1:2:end);
   ```

   Inspect their spectrum, repeat with a factor 4, and explain how aliasing impacts these images.

3 Exercise 3 (Spectral convolution)

1. Create in **imkernel2fft.m**, a function

   ```
   function lambda = imkernel2fft(nu, n1, n2, separable)
   ```

   that creates a $n_1 \times n_2$ complex valued array `lambda` corresponding to the frequential response of the convolution kernel function with impulse response `nu`. The format of `nu` is the same as the one for `imconvolve_spatial` and is determined by the optional argument `separable` (default: `'none'`).

   Hint: use `fftgrid`.

2. Create in **imconvolve_spectral.m**, a function

   ```
   function xconv = imconvolve_spectral(x, lambda)
   ```

   that implements the convolution in the Fourier domain of x with a convolution kernel whose frequential response is given by λ.

   Hint: Don't forget to take the real part.

3. Create a script `test_imconvolve_spectral.m` that tests on `x = lady` your new convolution function. Compare the results with `imconvolve_spatial` for different kernel functions using periodical boundary conditions. Check that the error between the two is zero (up to machine precision).

4. Create a script `test_imconvolve_compare.m` that compares the computation time of `imconvolve_spatial` and `imconvolve_spectral` for $s = 0$ to 8 where $s_1 = s_2 = s$. Make the comparison when the spatial convolution is separable (e.g., box) and non-separable (e.g., exponential). Average the computation time on 50 runs. Draw the curves of average computation time as a function of s (do not forget about title, axis names, and legend). Check that your curves are consistent with the ones bellow. Repeat with `x = house` and interpret the results.