Lecturer: Prof. Henning Sprekeler
Assistant: Owen Mackwood

Visual perception

April 21, 2016

The solutions for these exercises (comprising source code, discussion and interpretation as an IPython Notebook) should be handed in before **April 28 at 8am** through the Moodle interface (in emergency cases send them to `owen.mackwood@bccn-berlin.de`). The solutions will be discussed in the computer course on Monday, May 2.

Exercise 1: Frequency content of natural images and the Fourier transform

Natural images have interesting statistics that have been extensively studied. In particular, they have a power spectral density (PSD) that decays inversely with the square of the spatial frequency. Their PSD can be modeled by $\mathrm{PSD}(\mathbf{k}) \sim 1/(\epsilon + ||\mathbf{k}||^2)$, where $\mathbf{k}$ is the wave vector whose length $||\mathbf{k}||$ is spatial frequency and whose direction describes orientation. $\epsilon > 0$ ensures that the PSD does not diverge for low frequencies. Note that the the PSD says nothing about the phase, and we will not consider it here.

1. Generate a 1-dimensional "image" (500 pixels) whose PSD has the same dependence on spatial frequency as natural images. This can be done by constructing an appropriate filter, and applying it to white noise. We will do this in the frequency domain, so (i) generate a vector of random numbers (ii) Fourier transform it (hint: use *numpy.fft.fft*). Now we need a filter that will suitably decrease the power in the high frequencies. So (iii) create a filter with the same shape as the PSD mentioned above, for the appropriate frequencies (hint: use *fft.fftfreq* to determine the frequency associated with each Fourier component) and multiply that filter with the frequency domain image. Finally (iv) apply the inverse Fourier transform (hint: *fft.ifft*). Plot the resulting "image" and its PSD (hint: *fft.fftshift* is useful for the PSD plot). What is the effect of ϵ? (hint: as you vary ϵ be sure to include a normalization factor to give the resulting spectra comparable total power).

2. Use the same technique to generate a 2-dimensional image (500x500 pixels) that has the same power spectrum as a natural images (hint: use *fft.fft2* and *fft.ifft2*; there is no such thing as *fft.fftfreq2*, you can use *fft.fftfreq* in combination with *meshgrid* instead). Does the result look natural? Given that it has a natural PSD, discuss what you think this procedure for generating natural images is missing.

3. Load the so-called "hybrid image" that is provided on Moodle (*hybrid.png*) and filter out (i) high frequency and (ii) low frequency visual components. To do this, generate a 2D-Gaussian filter the same size as the image (hint: use *scipy.signal.gaussian* and *numpy.outer*). To generate the complementary high-pass filter, use a technique known as spectral inversion (in the 1D case, the high pass filter kernel is obtained from the low-pass by $x_{hp}[n] = \delta[n] - x_{lp}[n]$). Perform the convolution (apply the

filters) in the frequency domain. Why does generating the filter in this manner work? Is there an equivalent approach that doesn't require computing the high-pass filter kernel? Plot the resulting filtered images. How did the filtering process change the visual impression? Can you find a relation to what is happening when you look at the image close-up or from a distance?

Hint: The module *pyplot* offers functions for reading and displaying images.

If you want to see more of these images, you find a collection at `http://cvcl.mit.edu/hybrid_gallery/gallery.html`.

Exercise 2: "Natural input" for cell in the early visual system

During natural viewing conditions, we hardly ever keep our eyes still. Eye movements are sorted into 3 categories: (i) saccades: relatively large ($\sim 10°$ visual angle) and sudden changes of the fixation point, occurring roughly 3 times per second, (ii) micro-saccades and drift: small movements around the fixation point, a lot smaller in size ($\sim 10'$ visual angle) than saccades. (iii) There is also an even smaller and very rapid movement called tremor, which we neglect here. If you want to read more on the role of small eye movements in perception, check out Martinez-Conde et al., 2004, available from: `http://smc.neuralcorrelate.com/publications`.
Cells in the primary visual cortex have receptive fields with a size of $0.5 - 2°$. The goal of this exercise is to get a feeling for the visual input those cells receive during natural viewing conditions.

1. Pick one of the "natural images" provided on Moodle (*nat_images.zip*) and read it in as a matrix.

2. Write a function that extracts small rectangular patches of size L_x, L_y from a given location x, y of the image and make sure it works properly – you will need this function in the following tasks.

3. Generate a sequence of 10 images that mimics **micro-saccades** by drawing receptive field patches from a sequence of fixation points x_t, y_t with size $L_x = L_y = 25$ px (we define this as 1° visual angle). Generate the sequence of fixation points by adding little jumps $(\Delta x_t, \Delta y_t)$ to the last fixation point. Start from a random position far from the image border. Draw the micro-saccadic jump from a Gaussian distribution of width $\sigma = 4$ px (∼ 10').

4. Generate a another sequence of 10 images that mimics **saccades** by drawing patches from a sequence of fixation points x_t, y_t, again by starting from a random position and picking the same receptive field size ($L_x = L_y = 25$ px). Draw the saccadic jump from a Gaussian distribution of width σ^* that corresponds to 10° visual angle. If a jump lands outside the image, draw a new one. Optionally, replay the stored image sequence with a random delay that results in an average inter-saccadic interval of 1/3 s.

5. Qualitatively, what is the most notable difference when comparing the image sequences generated from saccades and micro-saccades? How could this difference be quantified in terms of a mathematical measure?

6. Pick a different "natural image" and repeat tasks 3. and 4. Do you observe differences regarding the resulting image sequences? What could be the reason?