TNM087 Bildbehandling - 2014

Reiner Lenz

Lab 3: Operations in the Fourier domain

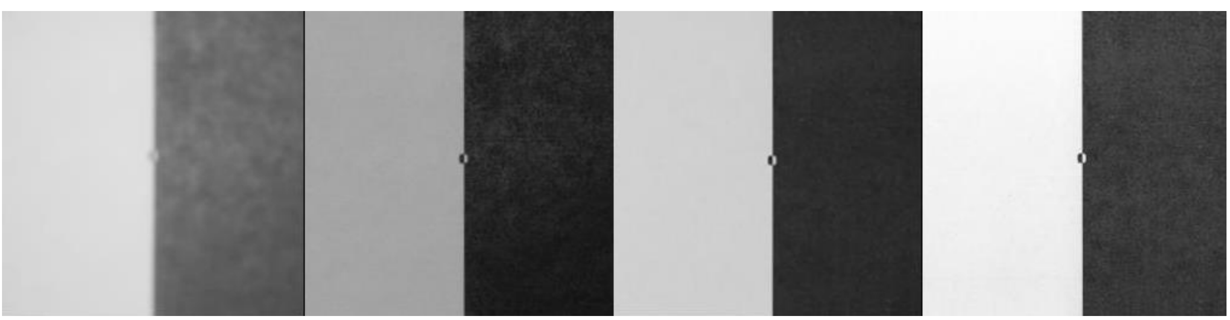
All images you find on S:\TN\M\TNM087-MT\Labs\Lab3

For background information about imaging models, blurring and Fourier optics read Szeliski: 2.3.1 Sampling and aliasing, 10.1.4 Optical blur (spatial response) estimation. Fourier optics (modeling optical systems like lenses with the help of Fourier theory) is described in Goodman: Introduction to Fourier optics. For a applications of Fourier transforms in general see Bracewell: The Fourier transform and its applications.

Part 1: Comparison of different optical systems

Overview:

The four images (left to right) HalfHolga.jpg, HalfCanon.jpg,HalfSony.jpg and HalfScanner.jpg show the same object. Images are generated with a Holga, a Canon and a Sony lens and a scanner.



One can see that the Holga image is not as sharp as the other three and the goal of this part of the lab is to use Fourier analysis to measure the quality of the four images based on their sharpness.

Sharpness values based on line-measurements:

(A) Read in the four images. As a pre-processing step you may want to convert them to a double gray-value image and normalize the gray value image so that minimum and maximum grayvalues are 0 and 1.

(B) Define a window which is 256 pixels wide and consists of a number of lines (decide yourself how many lines and where they are located but make sure that they do not include the center patch with the inverted black/white pattern). You now have four images which are all white on one and black on the other side. Call them EdgeCanon, EdgeHolga, EdgeScanner and EdgeSony. Sum each of them over all lines so that you have vectors SumEdgeCanon, SumEdgeHolga, SumEdgeScanner and SumEdgeSony which are all 256 long.

(C) Place each of the vectors SumEdgeCanon, SumEdgeHolga, SumEdgeScanner and SumEdgeSony in a larger vector with (zero) black pixels (read the description of zero-padding and its effects in the GW textbook, Sec. 4.6).

(D) Use the functions fft and fftshift to compute the Fourier transform of the padded vectors constructed in the previous step. Call these vectors FFT1EdgeCanon, FFT1EdgeHolga, FFT1EdgeScanner and FFT1EdgeSony. Plot the real parts, the imaginary parts and the absolute value. Explain the symmetry properties (relation between function values f(-u) and f(u)) of the results.

(E) Read the description of fftshift and find out where the dc component (corresponding to frequency 0, see page 246 in GW) is.

(F) Normalize vectors FFT1EdgeCanon, FFT1EdgeHolga, FFT1EdgeScanner and FFT1EdgeSony by dividing them by their DC-components and call the resulting vectors vectors NFFT1EdgeCanon, NFFT1EdgeHolga, NFFT1EdgeScanner and NFFT1EdgeSony. Explain why this is useful.

(G) Sharper images are characterized by larger high-frequency contents. Investigate NFFT1EdgeCanon, NFFT1EdgeHolga, NFFT1EdgeScanner and NFFT1EdgeSony, and compute a sharpness-measure for each of the images. Motivate your choice of sharpness measure by describing what it does. Use the absolute value of the normalized Fourier transform and a weight function. Motivate the form of you weight function and collect the sharpness values in a table.

OPTIONAL: *Next use the full images*

Use zero padding to embed the images into 512x512 images such that the center point is located at pixel (256,256) and such that the border pixels are zero. This gives you images EdgeCanon2, EdgeHolga2, EdgeScanner2 and EdgeSony2. Now do the same as in the last lab: Compute the 2D FFT, take absolute values, generate a polar coordinate system and compute the average values over rings. In more detail:

a) Compute the 2D FFT (using fft2) and rearrange the result using fftshift (mask out the center of the original images if you want)
b) Compute the absolute value (or squared absolute value) of the image and find out where the origin (ox,oy) of the Fourier transform is. Call this image A. This gives you the dc-component A(cx,xy) and depends on the average intensity value(see Sec. 4.6.5). Compensate for intensity variations by dividing the Fourier transform with the dc-component.
c) Plot the absolute value of the Fourier transform along one of the coordinate axes (A(cx,:)) for all four original images.
d) Now use meshgrid and cart2pol to compute the averages of the absolute value of the Fourier transform A.Use a few (100?) discrete values of the radial variable for averaging. Plot the first few values. Use the same processing steps as in Lab2.
e) Define a sharpness measure and compute the sharpness value for all for images. Which one is sharpest? Give an interpretation of the results, based on the sharpness values and the images.
f) If we assume that the cameras are linear systems that can be described by a convolution then we can use the relation between convolution in the image domain and the multiplication of the Fourier transforms to simulate a blurred (Holga) image from a sharp (Sony) image as follows. Write the Fourier transform of image k as F_k the Fourier transform of the object as O and the Fourier transform of camera k as T_k. The convolution theorem gives the equation

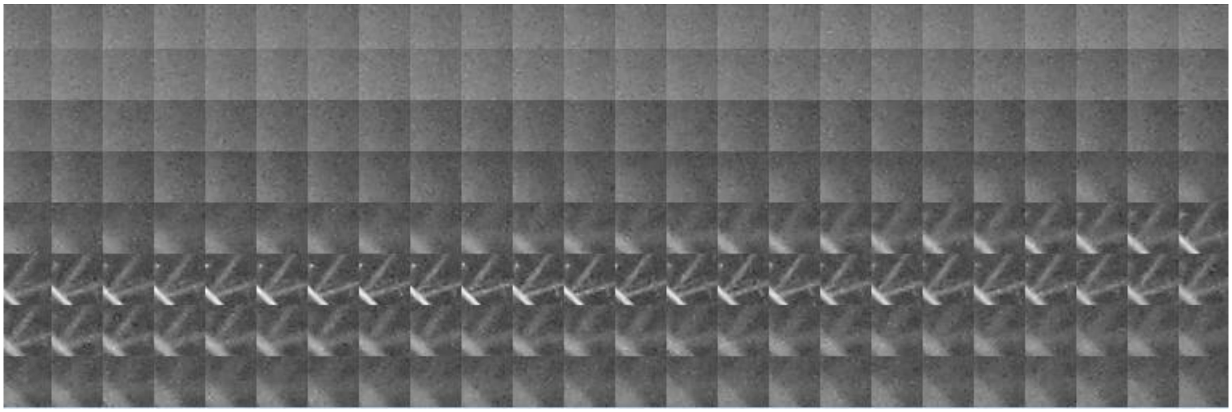
   $$F_k = T_k O$$

   One could try to simulate image 2 from image 4: $F_2 = T_2 O = T_2 F_4 / T_4 = (T_2/T_4) F_4$ with $T_2/T_4 = F_2/F_4$ The ratio T_2/T_4 describes the difference between the two imaging systems and can be computed from the measured images F_k. Replacing F_4 with the Fourier transform of the image of another object one could try to simulate Holga images from Sony images.

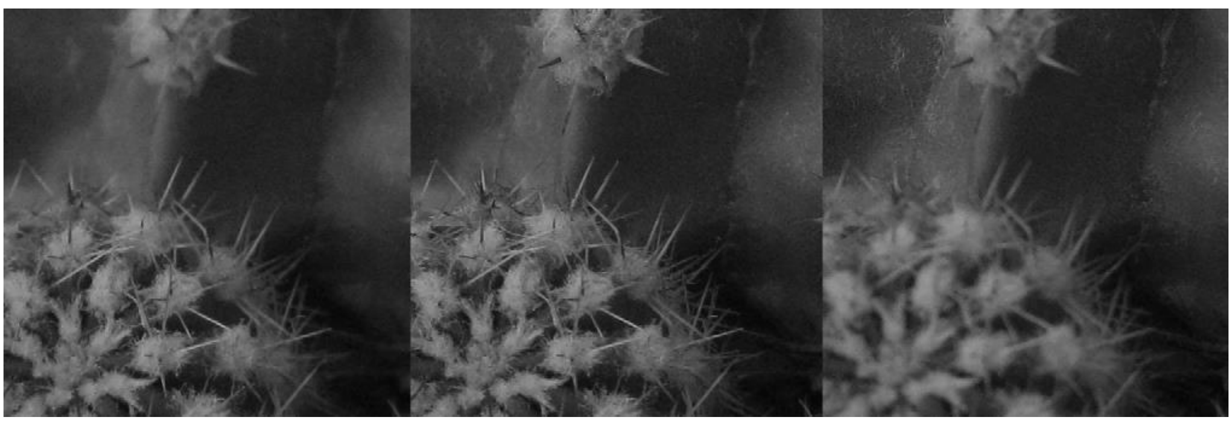
   Try to find out if it works for the given images you used to compute the Fourier transforms. Next try to rotate the Fourier F_4 to obtain $R(F_4)$ and see if the end result $(T_2/T_4)R(F_4)$ is what you expect and if not try to explain why it does not work and how you can make it work.

Part 2: Autofocus with Fourier Transforms

Overview: The mat-file winsuint8 contains small patches of size 32x32 pixels from a focus sequence consisting of 192 images.



These patches are parts of larger images. Some of them you find as Ik.jpg (k is the index in the sequence. I120.jpg is thus image 120). Here are three original images (120,132 and 150)



(A) Use the same steps as in the previous part 1 to compute the absolute values of the Fourier transforms of all 192 patches. Compensate mean intensity shifts dividing with the dc-component. Next compute the average radial frequency content of a patch by computing the weighted sum of (R.*A) where R is a matrix with the radial values and A is the matrix with the absolute value of the Fourier transform.

(B) Using these weighted sums and try to find out which patch as in focus (i.e. has the highest frequency content). Plot the values of these weighted sums.

Extra task: Divide the images Ik.jpg in small patches (say of size 32x32) and find for every patch the image in which this patch is in focus. Select the patch from the sharpest image and combine the patches into a new image in which hopefully all patches are in focus. This is the basic idea behind total-focus images.