Digital Image Processing HW3

Hejung Yang
School of Electrical and Electronic Engineering, Yonsei University.

problem 1: fft and frequency spectra

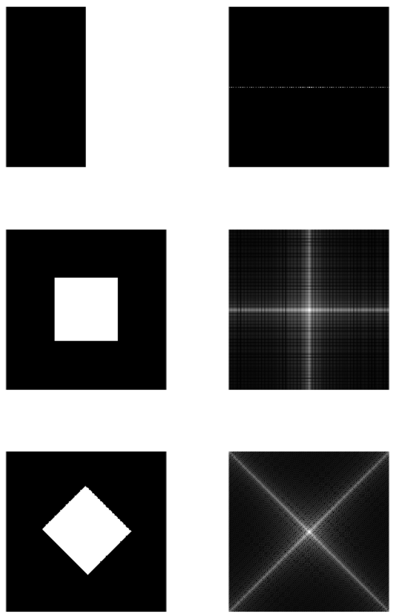

Figure 1: result of prob1.m. each row shows the original image and its magnitude within frequency domain.

Figure1 contains the result of running prob1.m. In Figure1, each row contains the spatial domain image and magnitude of its fourier transform output. Fourier transform is the transformation which converts spatial (pixel) domain image into complex-valued frequency domain image. Given each image from the first column of Figure1, 2-dimensional fft (fast fourier transform) is applied followed by shifting the zero frequency component to the center of the image. Extracting magnitude from the complex-valued frequency domain image ω can be calculated as follows:

$$\omega = \mathcal{F}(image)$$

$$magnitude(\omega) = \sqrt{Real(\omega)^2 + Imag(\omega)^2}$$

Second column of Figure1 shows each magnitude of given ω. Log scale is applied on magnitude for better visualization.

At first row, the input image consists of half black and half white dividing the image vertically. From the fact that the image is constant with respect to the vertial direction and abrupt change occurs along horizontal direction, especially at the center of the image, it can be expected that the frequency component is concentrated onto the horizontal line. It can be seen in the magnitude plot that it surely contains horizontal frequency elements, which stands for vertical edge.

At second row, given input image of white square on the center, it can be seen from the shifted 2-d fft magnitude result that the image contains both horizontal and vertical frequency components. In third row, it can be visually shown that rotating the spatial domain image also results the frequency domain image to be rotated accordingly.

Another characteristic that can be found in second, third row of Figure1 is that in frequency domain the magnitude intensity fluctuates along the lines. It can be understood from the fact that the fourier transform of a box filter is a sinc function (Figure2). Slicing the second spatial domain image in Figure1 across the horizontal/vertical directions generates 1-d box-shaped signal, so it can be expected that the resulting frequency domain output of the given image will have fluctuations corresponding to the crossings of a sinc function.

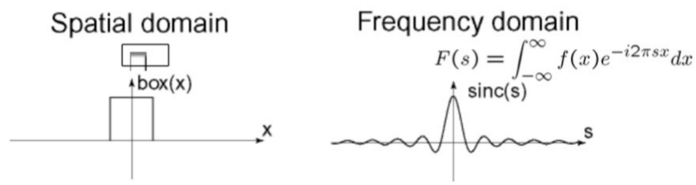

Figure 2: box filter in spatial domain and the corresponding frequency domain function

problem 2: phase and magnitude

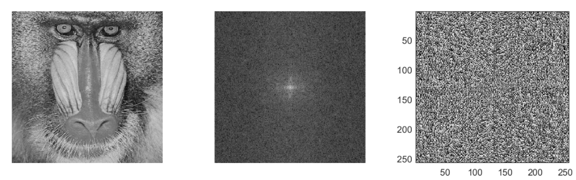

Figure 3: result of hw3_2.m. leftmost: original image, middle: log-scaled magnitude of the given image, rightmost: phase of the image

hw3_2.m reads image, calculate magnitude/phase after applying 2 dimensional fft onto the image, shifting the zero frequency component to the center of the image. As in problem 1, given complex-valued frequency domain image ω, magnitude, phase can be calculated as follows:

$$\omega = \mathcal{F}(image)$$

$$magnitude(\omega) = \sqrt{Real(\omega)^2 + Imag(\omega)^2}$$

$$phase(\omega)) = tan^{-1}\frac{Imag(\omega)}{Real(\omega)}$$

For better visualization, plotted magnitude is log-scaled. Figure3 is the magnitude/phase plot of the image. It can be seen that most of the magnitude intensity is concentrated on the center of the image. Also, phase image seems to have random values.

problem 3: phase vs. magnitude

hw3_3.m reads mandrill and clown image, swaps their phase information, then reconstruct the image. Given magnitude *mag* and phase *ph*, complex-valued frequency domain image ω and the corresponding spatial domain image can be calculated as follows:

$$\omega = mag * (cos(ph) + sin(ph) * 1j)$$

$$image = \mathcal{F}^{-1}(\omega)$$

It can be simply seen that the new ω has magnitude *mag* and phase *ph*:

$$magnitude(\omega) = \sqrt{Real(\omega)^2 + Imag(\omega)^2}$$

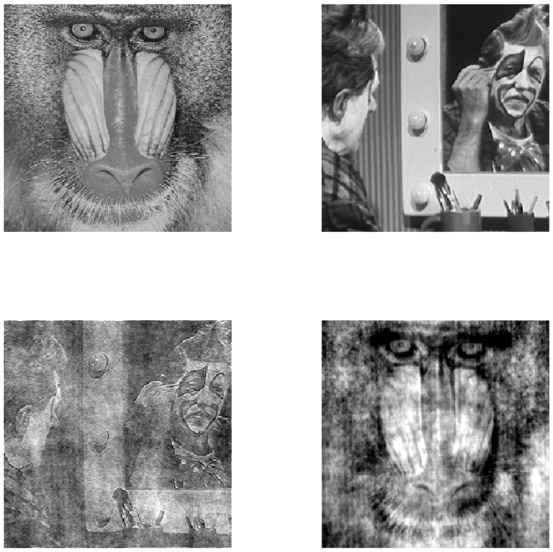

Figure 4: result of hw3_3.m. top-left: original mandrill image, top-right: original clown image, bottom-left: reconstructed image having magnitude of mandrill and phase of clown, bottom-right: reconstructed image having magnitude of clown and phase of mandrill

$$
\begin{aligned}
&= \sqrt{(mag * cos(ph))^2 + (mag * sin(ph))^2} \\
&= \sqrt{mag^2 * (cos^2(ph) + sin^2(ph))} \\
&= \sqrt{mag^2} \\
&= mag \\
phase(\omega) &= tan^{-1} \frac{Imag(\omega)}{Real(\omega)} \\
&= tan^{-1} \frac{sin(ph)}{cos(ph)} \\
&= tan^{-1}(tan(ph)) \\
&= ph
\end{aligned}
$$

Result of swapping the phase can be found in Figure4. It can be seen that the bottom-left image, consists of mandrill's magnitude and clown's phase, looks much similar to clown than mandrill. Conversely, bottom-right image which is from clown's magnitude and mandrill's phase looks like mandrill imge.

Figure5 shows the magnitude/phase of the original/transformed images. It can be seen that only the phase information is swapped while having same magnitude. From the figures, it is shown that phase information is more important than magnitude in terms of recognizing the object semantic and its structure; magnitude is applied so that the overall image intensity level is adjusted to the target image.

In bottom-left of Figure4, though the image is perceived as the clown image, it lost some overall intensity informations compared to the original clown. The magnitude of the original and transformed image can be checked from the second, third row of Figure5. It can be seen that the transformed image has less horizontal/vertical frequency components than the original image. Thus the intensity variations from the original clown image is smoothed out. Same goes with the bottom-right of Figure4. By having magnitude of bigger horizontal/vertial frequency components, the output mandrill has more intensity variations compared to the original one.

It can be also seen that the details are lost due to the phase-magnitude mismatch. Comparing top-right and bottom-left of Figure4, details such as stripes on the wall are found to be lost. Even for the mandrill case, detailed fur is hard to be recognized compared to the original image.

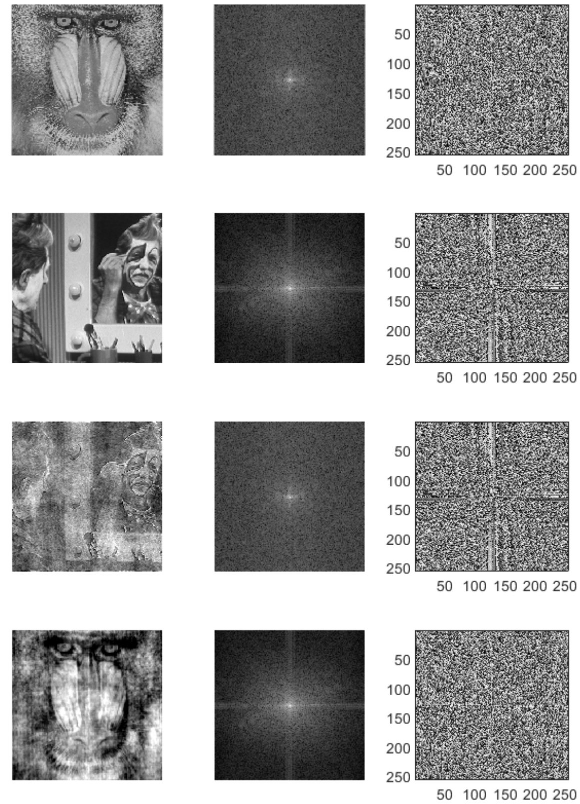

Figure 5: magnitude, phase of images in Figure4. Each row stands for top-left(mandrill), top-bottom(clown), bottom-left and bottom-right image in Figure4 in their order

problem 4: notch filter

hw3_4.m reads pattern.tif, transforms the image into complex-valued frequency domain, filters out grid pattern by applying notch filter and inverse-transform the filtered frequency domain image to be able to be seen in spatial domain.

Notch filter is a type of band-stop filter that attenuates frequencies within a specific range while passing all other frequencies unchanged. The range to be masked out can be decided manually, or with intensity threshold.

Figure7 shows the result of applying notch filter onto the noisy image. From the magnitude of the original image, it can be seen that there exists outstanding vertial/horizontal lines which partially represent the grid pattern overlaid onto the original image. Thus notch filter is designed to mask out the vertical/horizontal lines excluding the center, as can be seen in top-middle of Figure6. After applying the filter onto the frequency domain of the original image, it can be seen that the grid pattern is smoothed out. The filtered image can be found in bottom-right of Figure7.

Figure8 contains the groundtruth grid pattern and desired filtering result. The groundtruth grid pattern is constructed from pattern.tif using public paint applications. It can be seen from the middle row of the Figure8 that the grid pattern mainly consists of vertical/horizontal elements within the magnitude perspective. Subtracting original image from the grid pattern generates cleaner image, with having lower intensity on vertical/horizontal lines from the view of magnitude.

Figure6 shows the difference between the notch filter which is applied to Figure7 and the groundtruth filter. It can be seen that both the notch filter and the groundtruth filter contains vertial/horizontal line components within its magnitude plot but there exists other frequency components contributing to the grid pattern in the groundtruth plot whereas no other components

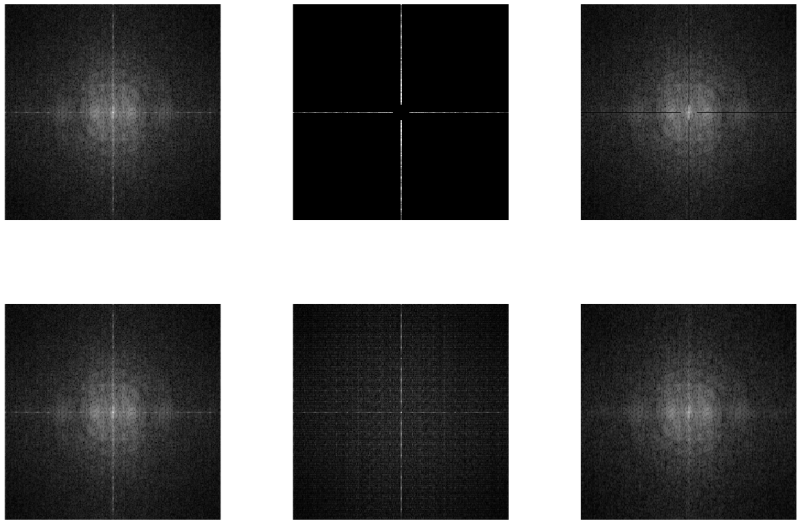

Figure 6: comparison between the notch filter and the groundtruch grid pattern. top row: original image, notch filter applied when generating Figure7 and result after applying the filter respectively. bottom row: same as above except the filter is from the groundtruch grid pattern

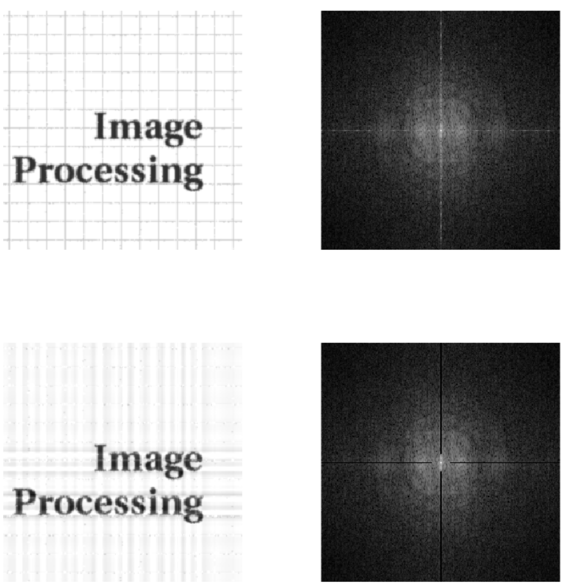

Figure 7: result of hw3_4.m. top row: original image and its magnitude in frequency domain respectively. bottom row: after applying notch filter onto the frequency domain image and its resulting magnitude

exist in the notch filter. Current notch filter output generates blurred artifacts (bottom-left of Figure7), but further improvement would be achieved when considering the detailed frequency pattern within the grid onto the notch filter.

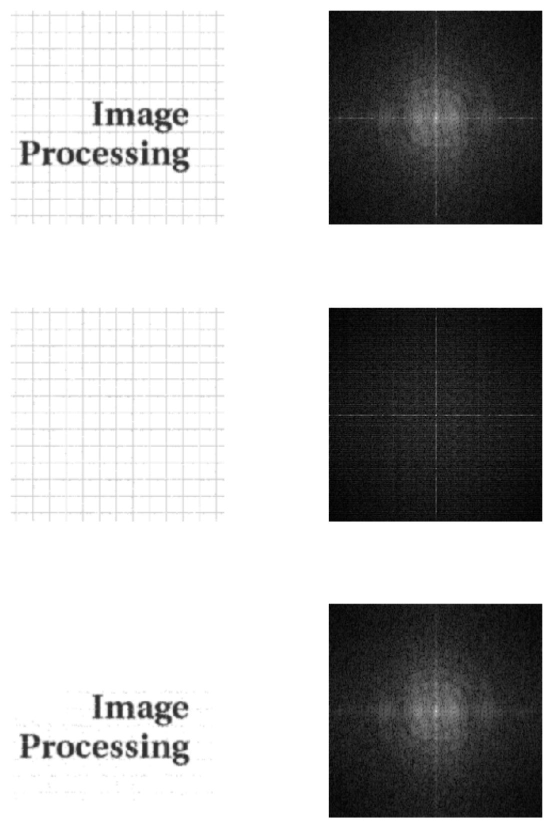

Figure 8: image and magnitude plot for components within the original image. top row: original image and its magnitude, middle row: grid-only image with its magnitude, bottom row: after subtracting grid from the original image and its magnitude