IMAGE PROCESSING AND APPLICATIONS

ENGI 9804

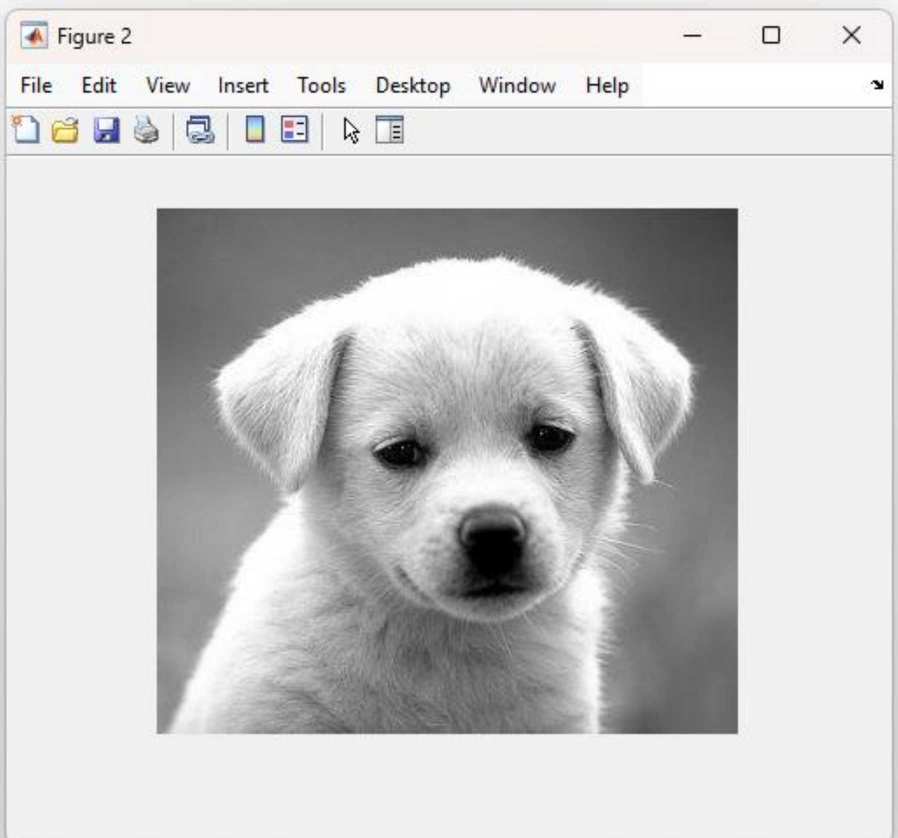
LAB -3

Group Number: 29
Kazi Salith Ur Rahman: 202291994
Nayem Al Tareq: 202293442

Question 1:

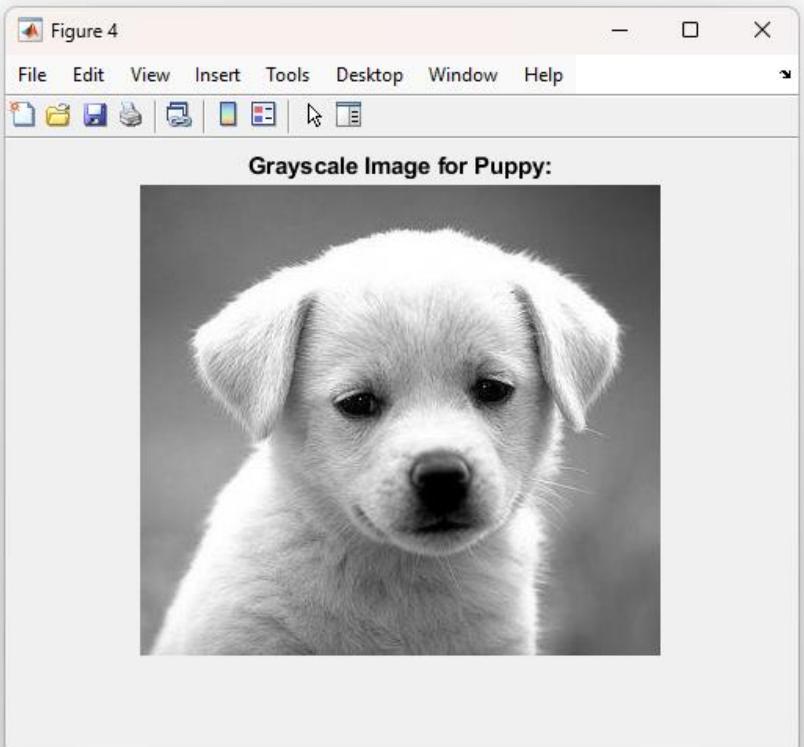
Question 2:

```matlab
% Group - 29
%Members: Kazi Salith Ur Rahman & Nayem Al Tareq

F_rgb = imread('puppy.jpg');
F = rgb2gray(F_rgb);
figure;
imshow(F);
```



Question 3:

```matlab
% Group - 29
%Members: Kazi Salith Ur Rahman & Nayem Al Tareq

F_rgb = imread('puppy.jpg');
F = rgb2gray(F_rgb);
figure;
imshow(F);
title('Grayscale Image for Puppy:');
im_size = size(F);
P = 2 * im_size(1);
Q = 2 * im_size(2);
disp(['Row: Padding size: ', num2str(P)]);
disp(['Column: Padding size: ', num2str(Q)]);
```

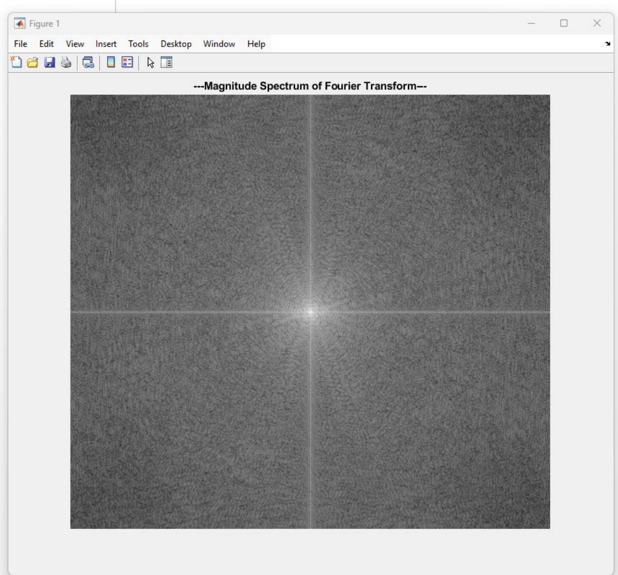


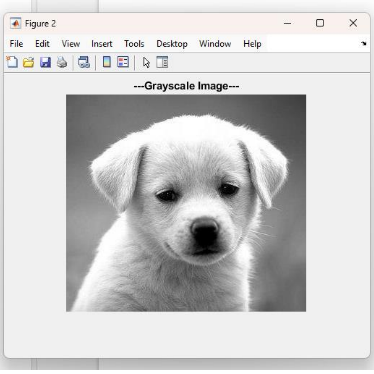
Row: Padding size: 600

Column: Padding size: 664

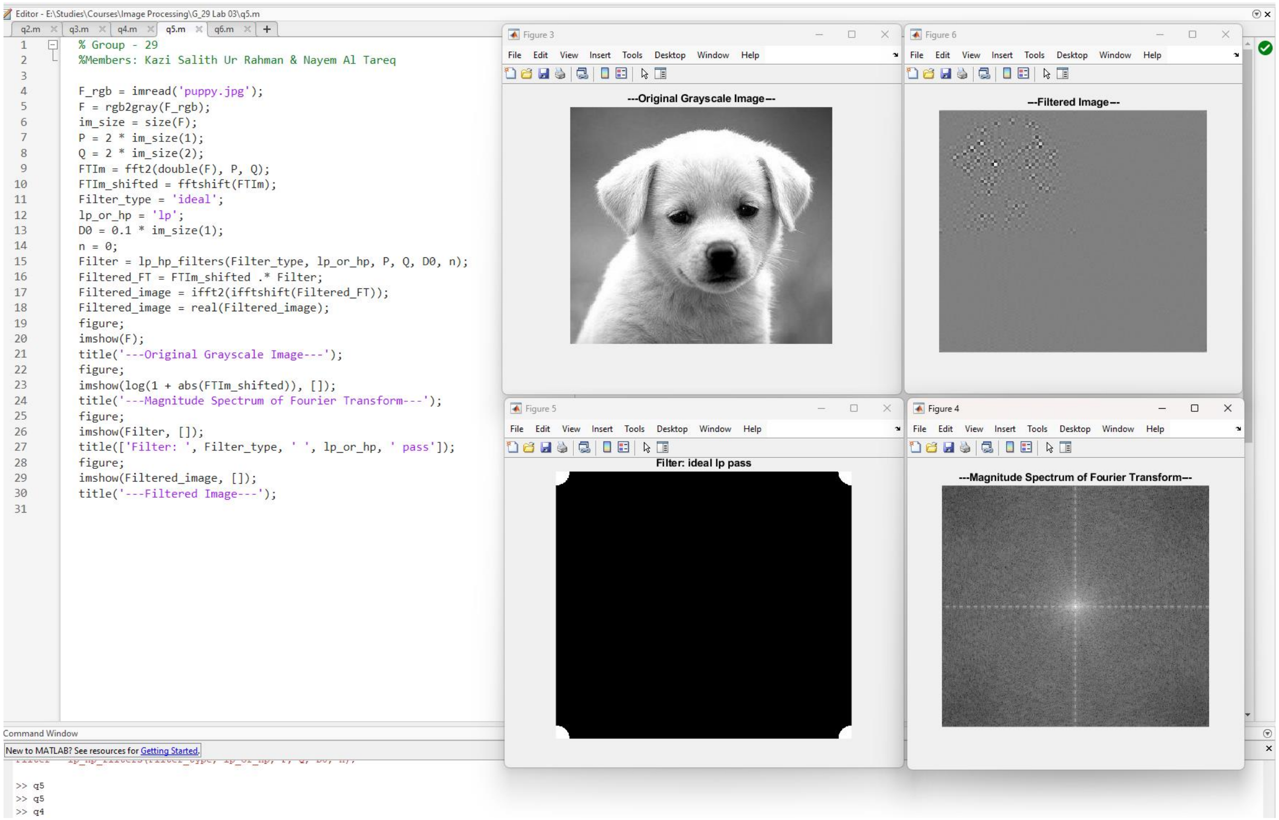
Question 4:

```matlab
% Group - 29
%Members: Kazi Salith Ur Rahman & Nayem Al Tareq

F_rgb = imread('puppy.jpg');
F = rgb2gray(F_rgb);
im_size = size(F);
P = 2 * im_size(1);
Q = 2 * im_size(2);
FTIm = fft2(double(F), P, Q);
FTIm_shifted = fftshift(FTIm);
magnitudeSpectrum = log(1 + abs(FTIm_shifted));
figure;
imshow(magnitudeSpectrum, []);
title('---Magnitude Spectrum of Fourier Transform---');
figure;
imshow(F);
title('---Grayscale Image---');
disp(['Padding size for rows: ', num2str(P)]);
disp(['Padding size for columns: ', num2str(Q)]);
```

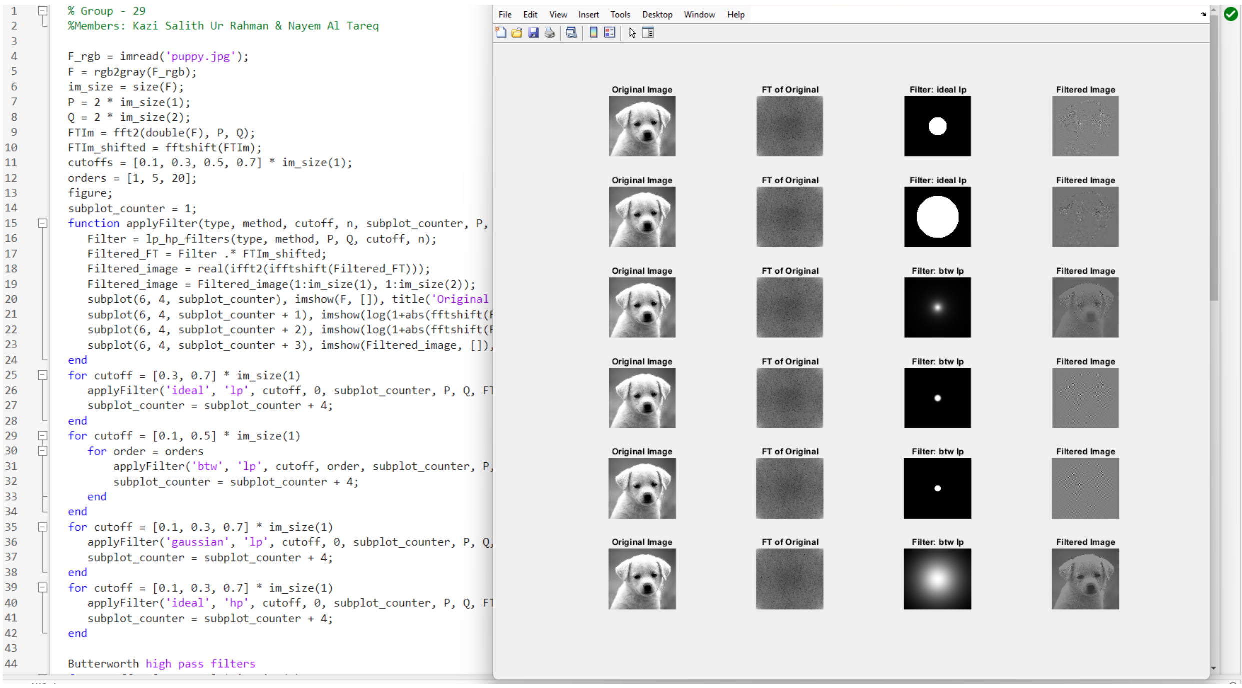




Question 5:


Editor - E:\Studies\Courses\Image Processing\G_29 Lab 03\q5.m
q2.m
q3.m
q4.m
q5.m
q6.m
% Group - 29
%Members: Kazi Salith Ur Rahman & Nayem Al Tareq
F_rgb = imread('puppy.jpg');
F = rgb2gray(F_rgb);
im_size = size(F);
P = 2 * im_size(1);
Q = 2 * im_size(2);
FTIm = fft2(double(F), P, Q);
FTIm_shifted = fftshift(FTIm);
Filter_type = 'ideal';
lp_or_hp = 'lp';
D0 = 0.1 * im_size(1);
n = 0;
Filter = lp_hp_filters(Filter_type, lp_or_hp, P, Q, D0, n);
Filtered_FT = FTIm_shifted .* Filter;
Filtered_image = ifft2(ifftshift(Filtered_FT));
Filtered_image = real(Filtered_image);
figure;
imshow(F);
title('---Original Grayscale Image---');
figure;
imshow(log(1 + abs(FTIm_shifted)), []);
title('---Magnitude Spectrum of Fourier Transform---');
figure;
imshow(Filter, []);
title(['Filter: ', Filter_type, ' ', lp_or_hp, ' pass']);
figure;
imshow(Filtered_image, []);
title('---Filtered Image---');
Figure 3
---Original Grayscale Image---
Figure 6
---Filtered Image---
Figure 5
Filter: ideal lp pass
Figure 4
---Magnitude Spectrum of Fourier Transform---
Command Window
New to MATLAB? See resources for Getting Started.
>> q5
>> q5
>> q4

Question 6:



Question 7:

Increasing the cut-off frequency radius of an Ideal Low Pass Filter allows more high-frequency components of an image to pass through, resulting in a smoother image with less removal of high-frequency details. Conversely, for an Ideal High Pass Filter, increasing the cut-off frequency radius filters out more low-frequency components, enhancing the edges and fine details in the image, but potentially introducing more noise.

Question 8:

As the order of a Butterworth filter increases, the transition between the passband and the stopband becomes more abrupt. This allows higher-order Butterworth filters to more accurately separate frequencies, making them behave similarly to an ideal filter. A higher-order Butterworth Low Pass Filter with the same cut-off frequency will permit low-frequency components to pass while more effectively attenuating high-frequency components.

Similarly, a higher-order Butterworth High Pass Filter will more effectively attenuate low frequencies, allowing high frequencies to pass more clearly.

Question 9:

Gaussian filters are known for several distinct performance characteristics compared to Ideal and Butterworth filters, making them well-suited for specific image processing tasks. One of the key features of Gaussian filters is their smooth frequency transition, unlike the sharp cutoff seen in Ideal filters. The abrupt changes in Ideal filters can create pronounced ringing artifacts (Gibbs phenomenon), whereas the Gaussian filter's exponential decrease helps avoid these artifacts, making it highly suitable for applications where preserving visual fidelity is crucial, such as in medical imaging or high-quality graphics.

While Butterworth filters aim for a maximally flat frequency response in the passband to minimize distortions, they may not preserve edges as effectively as Gaussian filters. Despite causing more overall image blurring, Gaussian filters often manage to preserve edges better in many practical scenarios. This characteristic is advantageous for reducing image noise but might be a drawback where sharp detail retention is necessary.

Unlike Butterworth filters, which can exhibit oscillations in their frequency response due to their design goal of avoiding ripples in the passband and stopband, Gaussian filters provide a monotonic decrease. This behavior produces a natural attenuation effect, often yielding more aesthetically pleasing results and being less disruptive, particularly in non-technical applications such as art and non-destructive testing.

Gaussian filters generally require less computational effort than high-order Butterworth filters. The computational complexity of Butterworth filters increases with the filter order, impacting the required resources for processing. Conversely, Gaussian filters maintain a consistent computational profile, irrespective of the smoothing extent, facilitating their implementation in systems with stringent real-time processing requirements.

The adaptability of Gaussian filters is another strength. The standard deviation of the Gaussian function can be adjusted to control the extent of smoothing, making Gaussian filters versatile for a wide range of applications—from simple blurring and smoothing tasks to more complex operations like those used in scale-space representation in computer vision.

Gaussian filters excel in scenarios where minimal artifact introduction is vital. Their inherent properties prevent the introduction of artifacts typically seen with the Ideal filter’s sharp transitions or the Butterworth filter's potential overshoots in the frequency response. These characteristics make Gaussian filters particularly useful in contexts that require a balance between detail smoothing and edge preservation. Their application extends beyond typical image processing tasks, influencing fields such as time-series analysis, 3D modeling, and feature extraction methodologies where gradual frequency attenuation is beneficial.

In summary, Gaussian filters often provide a preferable choice for a broad spectrum of image processing applications due to their balance of edge preservation, smooth frequency attenuation, computational efficiency, and minimal artifact introduction. These features make them particularly effective in environments where both visual quality and processing performance are critical.

References:

- Google
- ChatGPT
- Gemini
- Lecture Notes