University of Engineering & Technology, Mardan

Department of Electrical Engineering

Assignment #03

Subject: EE411 Digital Image Processing — Instructor: Dr. Irfan Khan

Due Date: 22/05/2025 — Total Marks: 20 [CLO 2: PLO 3, C4]

NOTE: Copied quizzes will render zero marks.

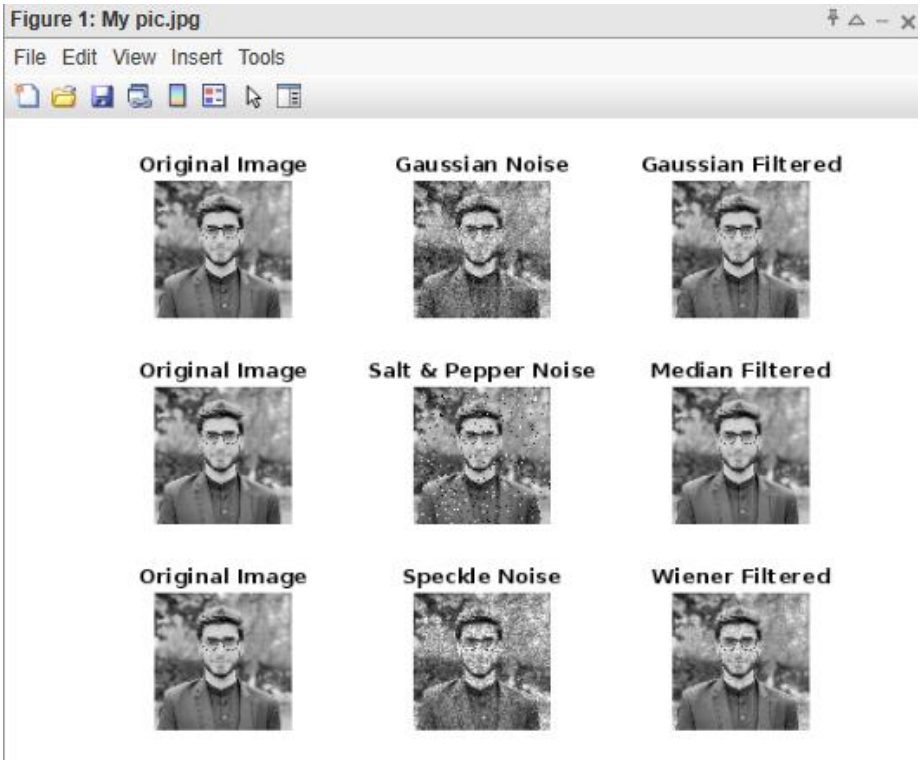
1. Implement using MATLAB different noise removal algorithms suitable for various types of noise, such as, Gaussian noise, Salt & Pepper Noise, and speckle noise
2. Analyze the performance of the noise removal algorithms using a diverse set of test images with known noise characteristics.

Matlab code:

```
close all
clear all
clc
%% List of grayscale test images
image_list = {'My pic.jpg', 'img.jpeg', 'img1.jpeg'};
num_images = length(image_list);

%% Initialize result storage
results = [];

for i = 1:num_images
   % Read and convert to grayscale double
   original = im2double(imread(image_list{i}));
   if size(original,3) == 3
      original = rgb2gray(original);
   end

   % Add known noise
   gaussian_noisy = imnoise(original, 'gaussian', 0, 0.01);
   sp_noisy = imnoise(original, 'salt & pepper', 0.05);
   speckle_noisy = imnoise(original, 'speckle', 0.04);

   % Denoise using suitable filters
   gaussian_filtered = imgaussfilt(gaussian_noisy, 1);          % For Gaussian noise
   median_filtered_sp = medfilt2(sp_noisy, [3 3]);              % For Salt & Pepper noise
   wiener_filtered_speckle = wiener2(speckle_noisy, [5 5]);     % For Speckle noise

   % Compute PSNR and SSIM
   psnr_g = psnr(gaussian_filtered, original);
   ssim_g = ssim(gaussian_filtered, original);
```

```
    psnr_sp = psnr(median_filtered_sp, original);
    ssim_sp = ssim(median_filtered_sp, original);

    psnr_speckle = psnr(wiener_filtered_speckle, original);
    ssim_speckle = ssim(wiener_filtered_speckle, original);

    % Store results
    results = [results; {image_list{i}, psnr_g, ssim_g, ...
                         psnr_sp, ssim_sp, psnr_speckle, ssim_speckle}];

    % Optional: Display sample result for each image
    figure('Name', image_list{i});
    subplot(3,3,1), imshow(original), title('Original Image');
    subplot(3,3,2), imshow(gaussian_noisy), title('Gaussian Noise');
    subplot(3,3,3), imshow(gaussian_filtered), title('Gaussian Filtered');

    subplot(3,3,4), imshow(original), title('Original Image');
    subplot(3,3,5), imshow(sp_noisy), title('Salt & Pepper Noise');
    subplot(3,3,6), imshow(median_filtered_sp), title('Median Filtered');

    subplot(3,3,7), imshow(original), title('Original Image');
    subplot(3,3,8), imshow(speckle_noisy), title('Speckle Noise');
    subplot(3,3,9), imshow(wiener_filtered_speckle), title('Wiener Filtered');
end

%% Create and display results table
T = cell2table(results, 'VariableNames', ...
    {'Image', 'PSNR_Gaussian', 'SSIM_Gaussian', ...
     'PSNR_SaltPepper', 'SSIM_SaltPepper', ...
     'PSNR_Speckle', 'SSIM_Speckle'});
disp(T);
```

Result:



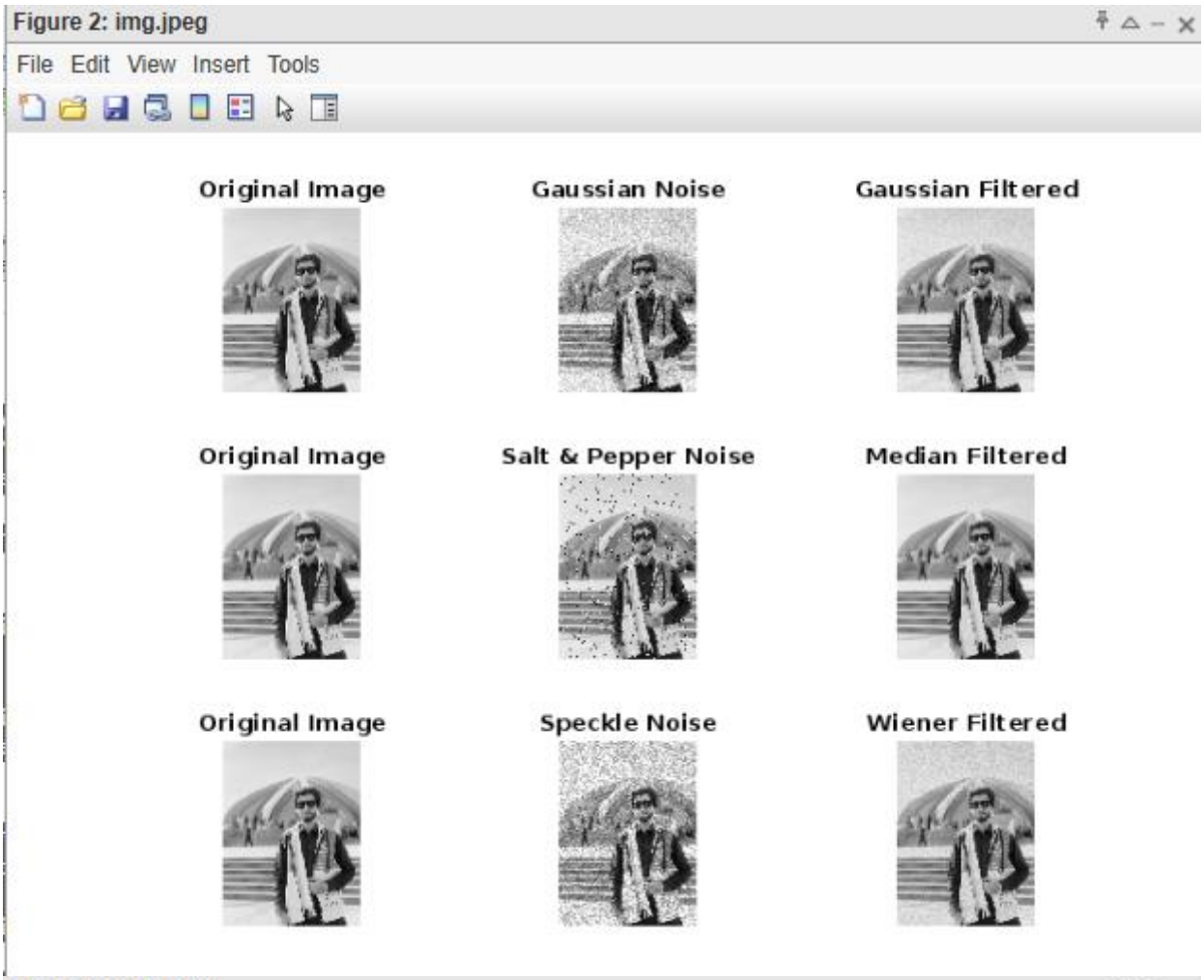


Figure 2: img.jpeg
File Edit View Insert Tools
Original Image
Gaussian Noise
Gaussian Filtered
Original Image
Salt & Pepper Noise
Median Filtered
Original Image
Speckle Noise
Wiener Filtered

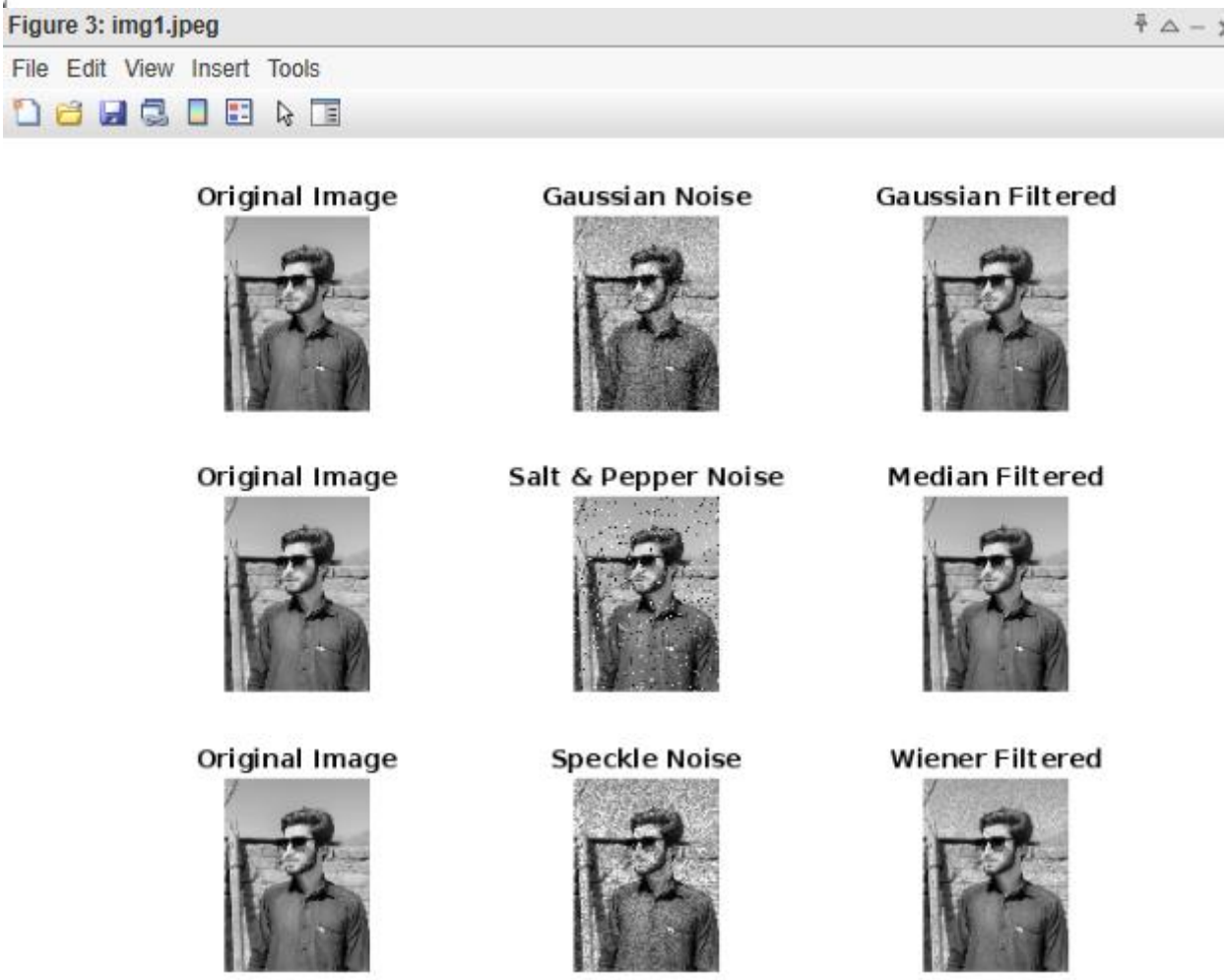


Figure 3: img1.jpeg
File Edit View Insert Tools
Original Image
Gaussian Noise
Gaussian Filtered
Original Image
Salt & Pepper Noise
Median Filtered
Original Image
Speckle Noise
Wiener Filtered