
```
%NAME: Krishna Kodali
%INST: IIT Bhubaneswar
%DATE: 23/10/2020
%CATEGORY: BTech
%BRANCH: Computer Science
%Roll Number: 17CS01008
% Assignment-04
% Image Restoration using Mean & Median Filtering
%Removing previous Buffer
clc;clear;close all;

%Question-1: Removing Salt & Pepper noise using Median Filtering

%Taing an image and adding Salt & Pepper Noise
Image = imread("Fingerprint.jpg");
N_Image = imnoise(Image, "salt & pepper");

figure
subplot(2,3,2),
imshow(Image),
title("Input Image");
subplot(2,3,4),
imshow(N_Image),
title("Noisy Image with Salt& Pepper Noise");
sgtitle("Images before & after adding noise");

[p, q] = size(Image);
M_Image = zeros(p, q);
for i = 2 : (p - 1)
    for j = 2 : (q - 1)
        subImage = N_Image(i - 1 : i + 1, j - 1 : j + 1);
        M_Image(i, j) = median(subImage(:));
    end
end

subplot(2,3,6), imshow(uint8(M_Image)),
title('After Median Filtered');
sgtitle('Median Filtering');
```

Median Filtering

Input Image



Noisy Image with Salt& Pepper Noise



After Median Filtered



```
%Question-2:Using contraharmonic mean filtering to remove Noise.
%           Show the effect of wrong choice of polarity in the order Q
CF_1 = ContraHM(N_Image, -3);
CF_2 = ContraHM(N_Image, -2);
CF_3 = ContraHM(N_Image, 0);
CF_4 = ContraHM(N_Image, 1);
CF_5 = ContraHM(N_Image, 2);

figure
subplot(2, 3, 1), imshow(N_Image),
title('Noisy Image');
subplot(2, 3, 2), imshow(uint8(CF_1)),
title('Q = -3');
subplot(2, 3, 3), imshow(uint8(CF_2)),
title('Q = -1');
subplot(2, 3, 4), imshow(uint8(CF_3)),
title('Q = 0');
subplot(2, 3, 5), imshow(uint8(CF_4)),
title('Q = 1');
subplot(2, 3, 6), imshow(uint8(CF_5)),
title('Q = 2');
sgtitle('Contraharmonic mean filter');

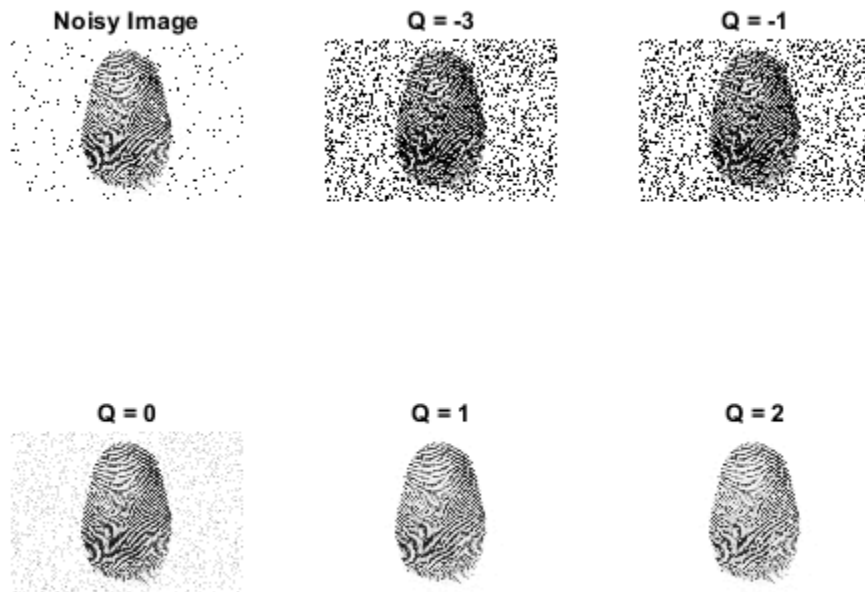
function C_Filter = ContraHM(N_Image, Q)
    [p, q] = size(N_Image);
    C_Filter = zeros(p, q);
```

```

for i = 2 : (p - 1)
    for j = 2 : (q - 1)
        subImage = double(N_Image(i - 1 : i + 1, j - 1 : j + 1));
        Num = subImage .^ (Q + 1);
        Denum = subImage .^ (Q);
        C_Filter(i, j) = uint8((sum(Num(:))) / (sum(Denum(:)))));
    end
end
end

```

Contraharmonic mean filter



Published with MATLAB® R2020a