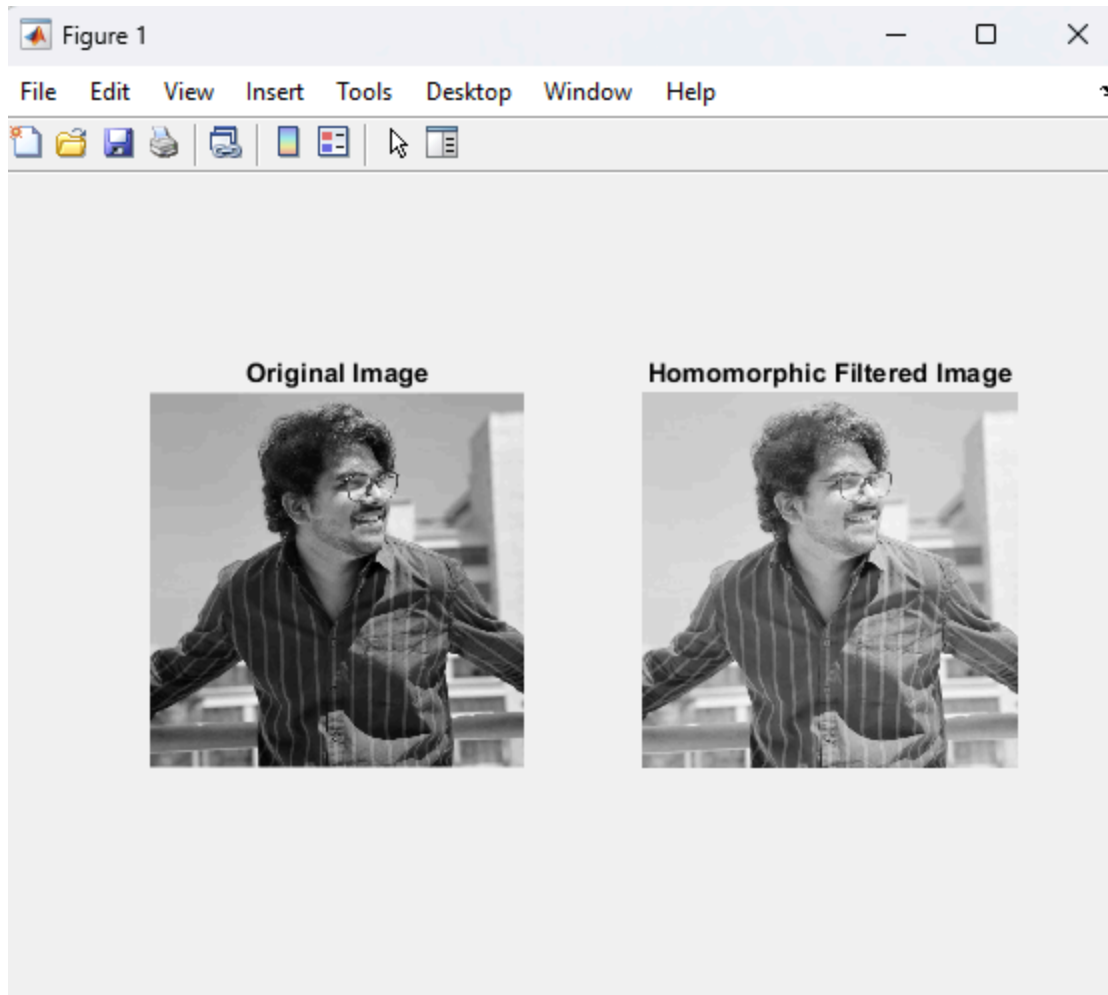


LAB-7  
HOMOMORPHIC FILTERING

SRIKAR GANESWARAM  
21BEC1230

```
clc;
close all;
clear all;
d = 10; % Cutoff frequency
d2 = d^2; % Square of cutoff frequency
f = double(rgb2gray(imread("C:\Users\dsplab\Desktop\21BEC1230\SRIKK.jpeg")));
l = log(1 + f); % Logarithmic transformation
z = fft2(l);
[m, n] = size(f);
b = zeros(m, n);
h = zeros(m, n);
for i = 1:m
    for j = 1:n
        b(i, j) = sqrt((i - m / 2)^2 + (j - n / 2)^2); % euclidian distance
        h(i, j) = exp(-b(i, j)^2 / (2 * d2)); % Gaussian filter
    end
end
L = 0.5; % Gamma low value
H = 1.5; % Gamma high value
filter = L + (H - L) * h;
s = z .* filter;
g = abs(iff2(s)); % inverse fourier transformation
e = exp(g) - 1; % inverse the logarithmic transformation
subplot(1, 2, 1);
imshow(f, []);
title('Original Image');
subplot(1, 2, 2);
imshow(e, []);
title('Homomorphic Filtered Image');
```



```

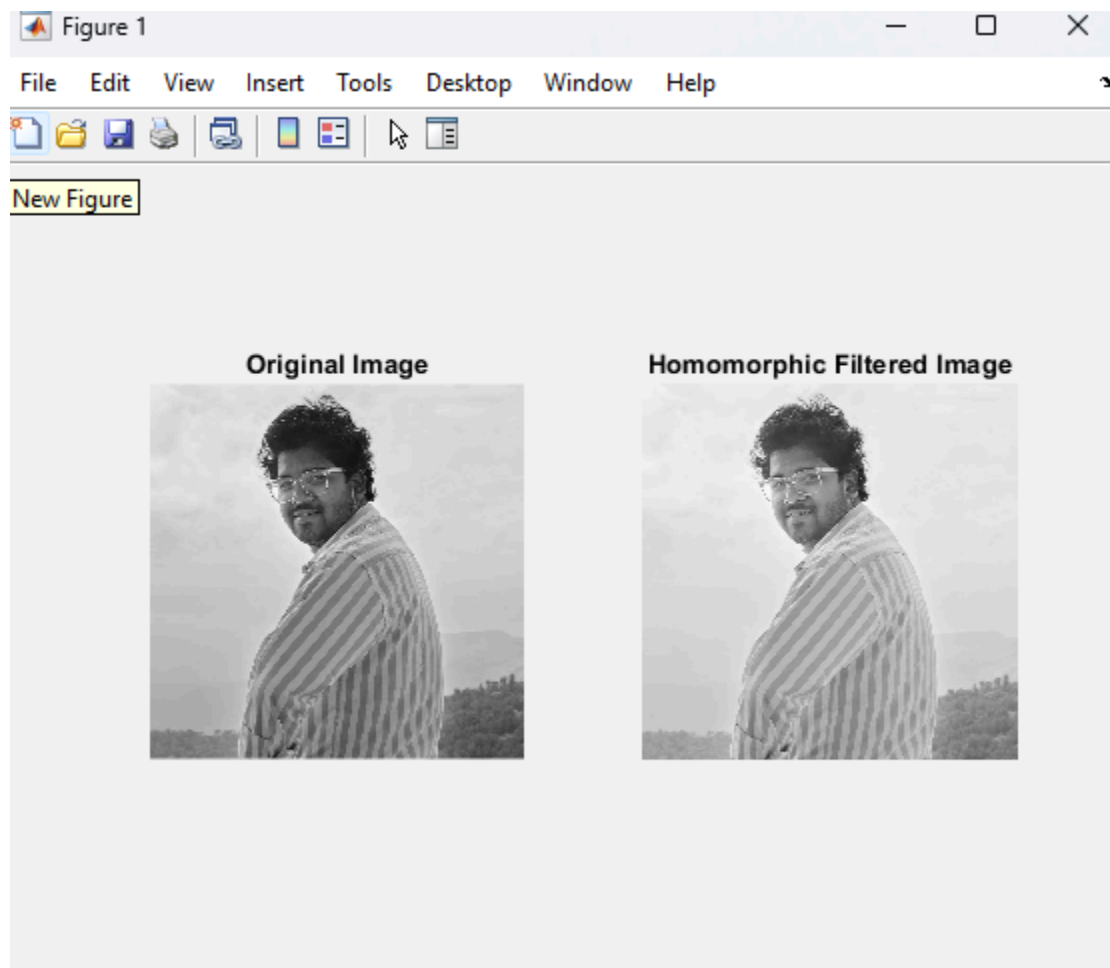
clc;
close all;
clear all;
d = 10; % Cutoff frequency
d2 = d^2; % Square of cutoff frequency
f = double(rgb2gray(imread("C:\Users\dsplab\Desktop\21BEC1230\SRIKK.jpeg")));
l = log(1 + f); % Logarithmic transformation
z = fft2(l);
[m, n] = size(f);
b = zeros(m, n);
h = zeros(m, n);
for i = 1:m
    for j = 1:n
        b(i, j) = sqrt((i - m / 2)^2 + (j - n / 2)^2); % euclidian distance
        h(i, j) = exp(-b(i, j)^2 / (2 * d2)); % Gaussian filter
    end
end

```

```

end
L = 0.5; % Gamma low value
H = 1.5; % Gamma high value
filter = L + (H - L) * h;
s = z .* filter;
g = abs(iff2(s));%inverse fourier transformation
e = exp(g) - 1;%inverse the logarithmic transformation
subplot(1, 2, 1);
imshow(f, []);
title('Original Image');
subplot(1, 2, 2);
imshow(e, []);
title('Homomorphic Filtered Image');

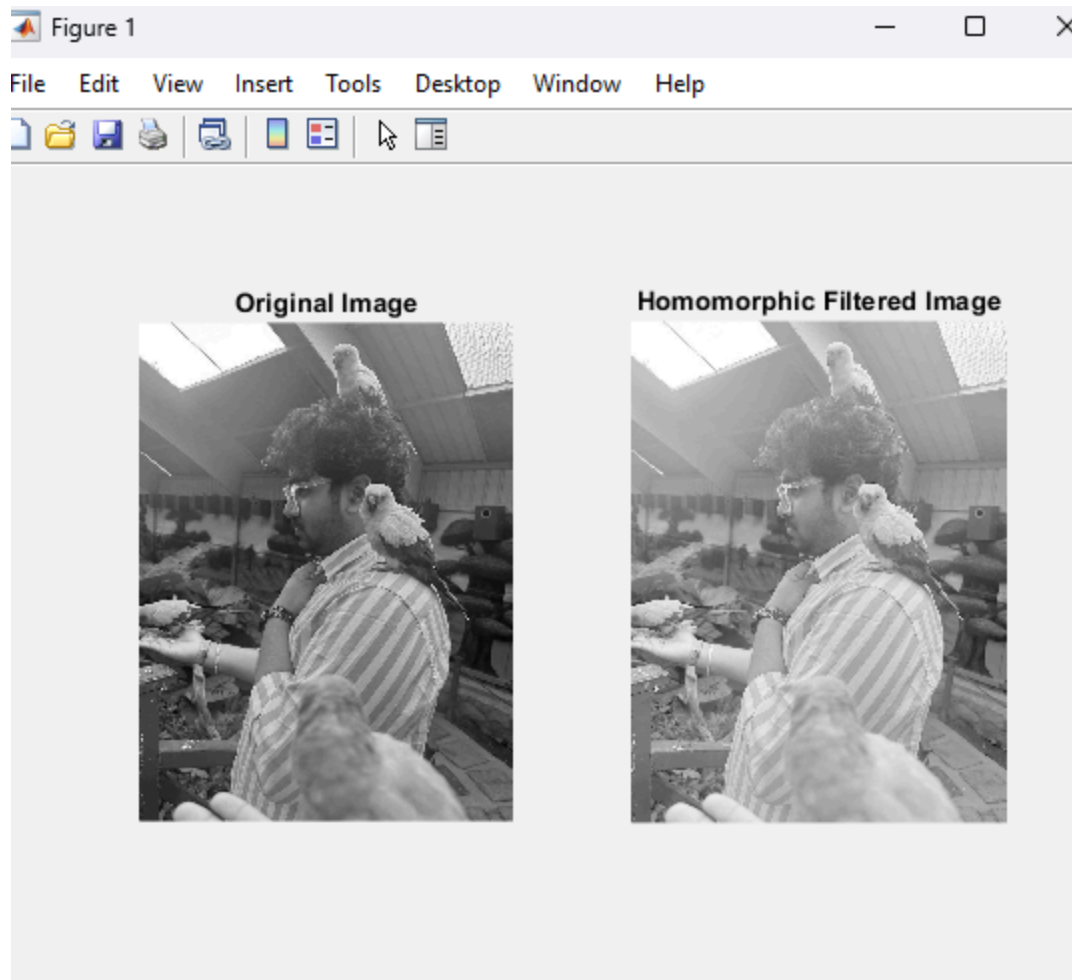
```



```

clc;
close all;
clear all;
d = 10; % Cutoff frequency
d2 = d^2; % Square of cutoff frequency
f = double(rgb2gray(imread("C:\Users\dsplab\Downloads\SRIK2.jpeg")));
l = log(1 + f); % Logarithmic transformation
z = fft2(l);
[m, n] = size(f);
b = zeros(m, n);
h = zeros(m, n);
for i = 1:m
    for j = 1:n
        b(i, j) = sqrt((i - m / 2)^2 + (j - n / 2)^2); %eucledian distance
        h(i, j) = exp(-b(i, j)^2 / (2 * d2)); % Gaussian filter
    end
end
L = 0.5; % Gamma low value
H = 1.5; % Gamma high value
filter = L + (H - L) * h;
s = z .* filter;
g = abs(ifft2(s)); %inverse fourier transformation
e = exp(g) - 1; %inverse the logarithmic transformation
subplot(1, 2, 1);
imshow(f, []);
title('Original Image');
subplot(1, 2, 2);
imshow(e, []);
title('Homomorphic Filtered Image');

```

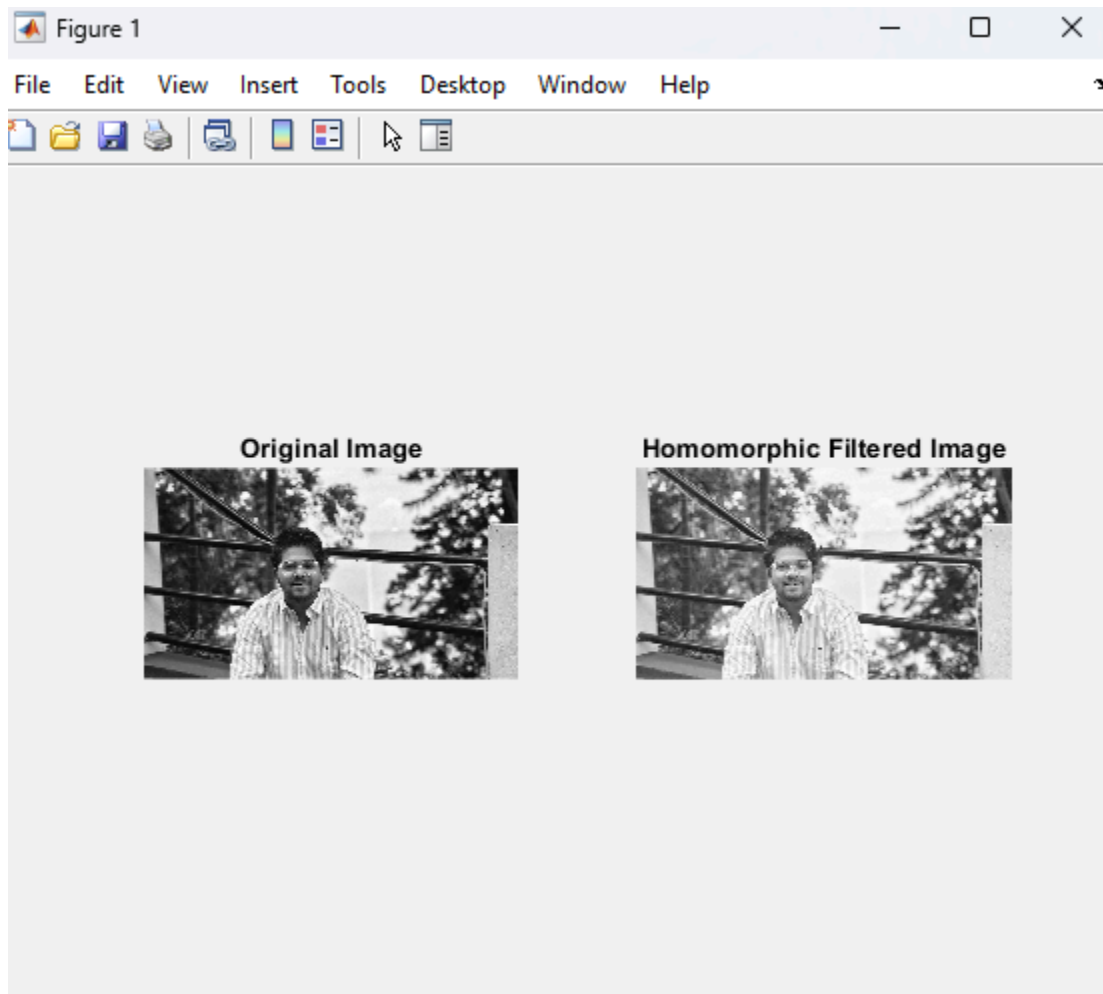


```

clc;
close all;
clear all;
d = 10; % Cutoff frequency
d2 = d^2; % Square of cutoff frequency
f = double(rgb2gray(imread("C:\Users\dsplab\Downloads\SRIK2.jpeg")));
I = log(1 + f); % Logarithmic transformation
z = fft2(I);
[m, n] = size(f);
b = zeros(m, n);
h = zeros(m, n);
for i = 1:m
    for j = 1:n
        b(i, j) = sqrt((i - m / 2)^2 + (j - n / 2)^2); %eucledian distance
        h(i, j) = exp(-b(i, j)^2 / (2 * d2)); % Gaussian filter
    end
end
end
L = 0.5; % Gamma low value

```

```
H = 1.5; % Gamma high value
filter = L + (H - L) * h;
s = z .* filter;
g = abs(ifft2(s));%inverse fourier transformation
e = exp(g) - 1;%inverse the logarithmic transformation
subplot(1, 2, 1);
imshow(f, []);
title('Original Image');
subplot(1, 2, 2);
imshow(e, []);
title('Homomorphic Filtered Image');
```



```

clc;
close all;
clear all;
d = 10; % Cutoff frequency
d2 = d^2; % Square of cutoff frequency
f = double(rgb2gray(imread("C:\Users\dsplab\Downloads\SRIK3.jpeg")));
l = log(1 + f); % Logarithmic transformation
z = fft2(l);
[m, n] = size(f);
b = zeros(m, n);
h = zeros(m, n);
for i = 1:m
    for j = 1:n
        b(i, j) = sqrt((i - m / 2)^2 + (j - n / 2)^2); %eucledian distance
        h(i, j) = exp(-b(i, j)^2 / (2 * d2)); % Gaussian filter
    end
end
L = 0.5; % Gamma low value
H = 1.5; % Gamma high value
filter = L + (H - L) * h;
s = z .* filter;
g = abs(ifft2(s)); %inverse fourier transformation
e = exp(g) - 1; %inverse the logarithmic transformation
subplot(1, 2, 1);
imshow(f, []);
title('Original Image');
subplot(1, 2, 2);
imshow(e, []);
title('Homomorphic Filtered Image');

```

