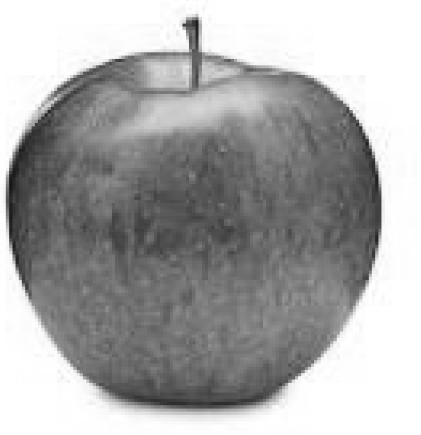
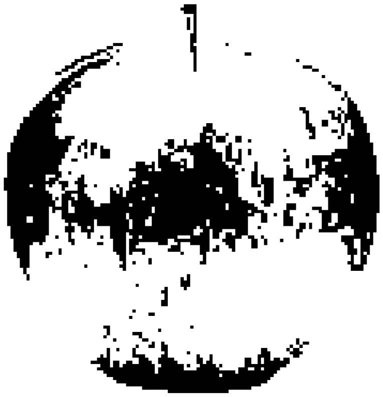
```
% Read the input image
img = imread('apple.png');

% Convert the image to grayscale if it's not already
if size(img, 3) == 3
    img_gray = rgb2gray(img);
else
    img_gray = img;
end

% Display original image
figure, imshow(img_gray), title('Original Image');
```



Original Image

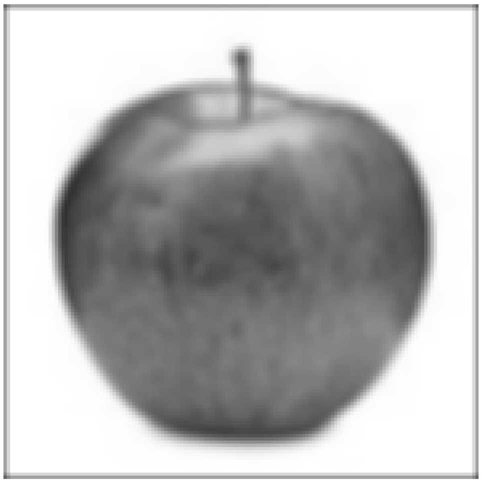
```
% Create a binary mask for the region of interest (ROI)
% Here, we will manually threshold the image to create a binary mask.
threshold = 100; % Adjust this value according to the image intensity
binary_mask = img_gray > threshold;

% Display the binary mask
figure, imshow(binary_mask), title('Binary Mask (ROI)');
```

Binary Mask (ROI)

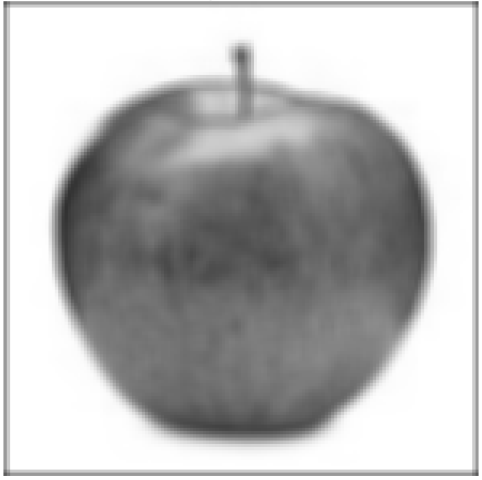


```
% Apply Low-Pass Filters
% Gaussian Low-Pass Filter
h_gaussian = fspecial('gaussian', [5 5], 2); % 5x5 kernel, sigma = 2
img_gaussian = imfilter(img_gray, h_gaussian);

% Average Low-Pass Filter
h_average = fspecial('average', [5 5]); % 5x5 kernel
img_average = imfilter(img_gray, h_average);

% Display Low-Pass Filter Results
figure, imshow(img_gaussian), title('Gaussian Filtered ');
```

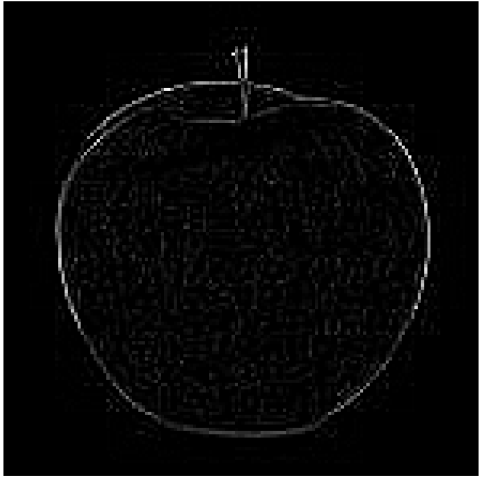
Gaussian Filtered



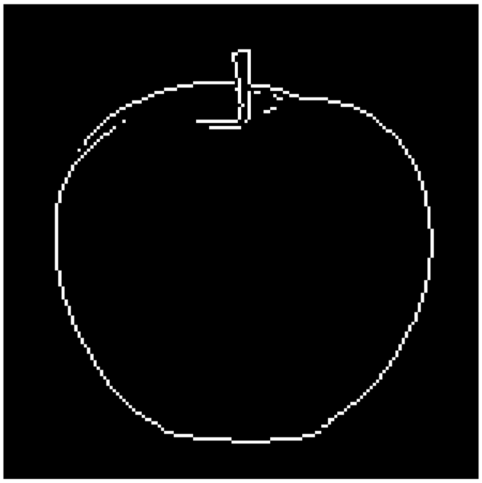
```
figure, imshow(img_average), title('Average Filtered ');
```


Average Filtered

```
% Apply High-Pass Filters
% Laplacian High-Pass Filter
h_laplacian = fspecial('laplacian', 0.2); % Sensitivity factor 0.2
img_laplacian = imfilter(img_gray, h_laplacian);

% Prewitt High-Pass Filter
img_prewitt = edge(img_gray, 'prewitt');

% Display High-Pass Filter Results
figure, imshow(img_laplacian), title('Laplacian Filtered ');
```


Laplacian Filtered

```
figure, imshow(img_prewitt), title('Prewitt Filtered ');
```



Prewitt Filtered