```matlab
1  % Read the input image
2  img = imread('rgb.jpeg');
3
4  % Convert the image to grayscale if it is RGB
5  gray_img = rgb2gray(img);
6
7  % Convert the image to double for calculations
8  gray_img = double(gray_img);
9
10 % Get the size of the image
11 [rows, cols] = size(gray_img);
12
13 % Sobel Operator Kernels
14 sobel_x = [-1 0 1; -2 0 2; -1 0 1];
15 sobel_y = [-1 -2 -1; 0 0 0; 1 2 1];
16
17 % Prewitt Operator Kernels
18 prewitt_x = [-1 0 1; -1 0 1; -1 0 1];
19 prewitt_y = [-1 -1 -1; 0 0 0; 1 1 1];
20
21 % Roberts Operator Kernels
22 roberts_x = [1 0; 0 -1];
23 roberts_y = [0 1; -1 0];
24
25 % Initialize edge detection outputs
26 sobel_edges = zeros(rows, cols);
27 prewitt_edges = zeros(rows, cols);
28 roberts_edges = zeros(rows, cols);
29
30 % Padding the image for boundary handling
31 padded_img = padarray(gray_img, [1 1], 'replicate');
32
33 % Sobel and Prewitt Convolution (3x3 Kernels)
34 for i = 2:rows+1
35     for j = 2:cols+1
36         % Extract the 3x3 region
37         region = padded_img(i-1:i+1, j-1:j+1);
38
39         % Sobel Gradient
40         Gx_sobel = sum(sum(region .* sobel_x));
41         Gy_sobel = sum(sum(region .* sobel_y));
42         sobel_edges(i-1, j-1) = sqrt(Gx_sobel^2 + Gy_sobel^2);
43
44         % Prewitt Gradient
45         Gx_prewitt = sum(sum(region .* prewitt_x));
46         Gy_prewitt = sum(sum(region .* prewitt_y));
47         prewitt_edges(i-1, j-1) = sqrt(Gx_prewitt^2 + Gy_prewitt^2);
48     end
49 end
```

```matlab
% Roberts Convolution (2x2 Kernels)
padded_img_roberts = padarray(gray_img, [1 1], 'replicate');
for i = 1:rows
    for j = 1:cols
        % Extract the 2x2 region
        region = padded_img_roberts(i:i+1, j:j+1);

        % Roberts Gradient
        Gx_roberts = sum(sum(region .* roberts_x));
        Gy_roberts = sum(sum(region .* roberts_y));
        roberts_edges(i, j) = sqrt(Gx_roberts^2 + Gy_roberts^2);
    end
end

% Normalize the results to the range [0, 255]
sobel_edges = uint8(255 * mat2gray(sobel_edges));
prewitt_edges = uint8(255 * mat2gray(prewitt_edges));
roberts_edges = uint8(255 * mat2gray(roberts_edges));

% Display the results
figure;

subplot(2, 2, 1);
imshow(uint8(gray_img));
title('Original Grayscale Image');

subplot(2, 2, 2);
imshow(sobel_edges);
title('Sobel Operator');

subplot(2, 2, 3);
imshow(prewitt_edges);
title('Prewitt Operator');

subplot(2, 2, 4);
imshow(roberts_edges);
title('Roberts Operator');
```