

```

%The Kuwahara filter is a non-linear smoothing filter used in image
processing, particularly for edge-preserving smoothing. The filter divides
the neighborhood of each pixel into multiple overlapping sub-regions
(typically 4 quadrants) and computes the mean and variance for each region.
% The pixel is then replaced by the mean of the region with the smallest
variance, preserving the edges by avoiding smoothing over regions of high
variance (which usually correspond to edges).
% Phase 1: Load the input image
input_image = imread('dog.jpg'); % Replace with the actual image file
grayscale_image = rgb2gray(input_image); % Convert to grayscale (if RGB)
grayscale_image = double(grayscale_image); % Convert to double for
accurate calculations

% Phase 2: Define the Kuwahara filter size (e.g., 5x5)
filter_size = 5;
half_size = floor(filter_size / 2);

% Get image dimensions
[rows, cols] = size(grayscale_image);

% Initialize output image
output_image = zeros(rows, cols);

% Phase 3: Apply Kuwahara filtering
for i = 1+half_size:rows-half_size
    for j = 1+half_size:cols-half_size
        % Define the 4 overlapping regions (quadrants)
        region1 = grayscale_image(i-half_size:i, j-half_size:j); % Top-left
        region2 = grayscale_image(i-half_size:i, j:j+half_size); % Top-
right
        region3 = grayscale_image(i:i+half_size, j-half_size:j); % Bottom-
left
        region4 = grayscale_image(i:i+half_size, j:j+half_size); % Bottom-
right

        % Calculate the mean and variance for each region
        mean1 = mean(region1(:)); var1 = var(region1(:));
        mean2 = mean(region2(:)); var2 = var(region2(:));
        mean3 = mean(region3(:)); var3 = var(region3(:));
        mean4 = mean(region4(:)); var4 = var(region4(:));

        % Create an array of variances and corresponding means
        variances = [var1, var2, var3, var4];
        means = [mean1, mean2, mean3, mean4];

        % Find the region with the smallest variance
        [~, min_index] = min(variances);

        % Assign the pixel to the mean of the region with the smallest
variance

```

```
        output_image(i, j) = means(min_index);  
    end  
end  
  
% Phase 4: Display the results  
figure;  
subplot(1, 2, 1), imshow(uint8( grayscale_image)), title('Original Grayscale  
Image');  
subplot(1, 2, 2), imshow(uint8(output_image)), title('Kuwahara Filtered  
Image');
```

