```
% Two-Step Enhancement Framework for Improving Underwater Image Quality
% Step 1: Color Correction
% Step 2: Contrast Enhancement

clc;
clear;
close all;

% Load the underwater image
inputImage = imread('underwater_image.jpg'); % Replace with your image path
figure;
imshow(inputImage);
title('Original Underwater Image');

% Step 1: Color Correction
% Apply White Balance to correct color cast
colorCorrectedImage = whiteBalance(inputImage);
figure;
imshow(colorCorrectedImage);
title('Color Corrected Image');

% Step 2: Contrast Enhancement
% Apply Contrast-Limited Adaptive Histogram Equalization (CLAHE)
contrastEnhancedImage = contrastEnhancement(colorCorrectedImage);
figure;
imshow(contrastEnhancedImage);
title('Contrast Enhanced Image');
```

```
% Save the final enhanced image
imwrite(contrastEnhancedImage, 'enhanced_underwater_image.jpg');

% Function for White Balance (Color Correction)
function correctedImage = whiteBalance(image)
    % Convert the image to double for calculations
    image = double(image);

    % Calculate the mean of each color channel
    meanR = mean(mean(image(:,:,1)));
    meanG = mean(mean(image(:,:,2)));
    meanB = mean(mean(image(:,:,3)));

    % Compute the scaling factors
    scaleR = meanG / meanR;
    scaleB = meanG / meanB;

    % Apply the scaling factors to each channel
    correctedImage(:,:,1) = image(:,:,1) * scaleR;
    correctedImage(:,:,2) = image(:,:,2);
    correctedImage(:,:,3) = image(:,:,3) * scaleB;

    % Normalize the image to the range [0, 255]
    correctedImage = uint8(correctedImage);
end

% Function for Contrast Enhancement (CLAHE)
function enhancedImage = contrastEnhancement(image)
```

```
    % Convert the image to LAB color space
    labImage = rgb2lab(image);

    % Apply CLAHE to the L channel (lightness)
    labImage(:,:,1) = adapthisteq(labImage(:,:,1), 'ClipLimit', 0.02, 'Distribution', 'rayleigh');

    % Convert the image back to RGB color space
    enhancedImage = lab2rgb(labImage);
    enhancedImage = uint8(enhancedImage * 255);
end
```

2nd code: -

```
function enhancedImage = underwaterImageEnhancement(inputImage)
    % Read the input image
    I = imread(inputImage);

    % Step 1: Color Correction using Piecewise Linear Transformation
    correctedImage = colorCorrection(I);

    % Step 2: Contrast Enhancement using Optimal Contrast Method
    enhancedImage = contrastEnhancement(correctedImage);

    % Display the original and enhanced images
    figure;
    subplot(1, 2, 1);
    imshow(I);
    title('Original Image');

    subplot(1, 2, 2);
    imshow(enhancedImage);
    title('Enhanced Image');
end

function correctedImage = colorCorrection(I)
    % Convert image to double for processing
    I = im2double(I);
```

```
    % Calculate the average intensity of each channel
    avgRed = mean(I(:,:,1), 'all');
    avgGreen = mean(I(:,:,2), 'all');
    avgBlue = mean(I(:,:,3), 'all');

    % Apply piecewise linear transformation to correct color distortion
    correctedRed = I(:,:,1) + 0.5 * (avgGreen - avgRed) * (1 - I(:,:,1));
    correctedGreen = I(:,:,2);
    correctedBlue = I(:,:,3) + 0.5 * (avgGreen - avgBlue) * (1 - I(:,:,3));

    % Ensure values are within [0,1]
    correctedRed = max(min(correctedRed, 1), 0);
    correctedBlue = max(min(correctedBlue, 1), 0);

    % Combine the corrected channels
    correctedImage = cat(3, correctedRed, correctedGreen, correctedBlue);
end

function enhancedImage = contrastEnhancement(I)
    % Convert image to double for processing
    I = im2double(I);

    % Apply histogram equalization for contrast enhancement
    enhancedImage = adapthisteq(I);

    % Convert back to uint8 for display
    enhancedImage = im2uint8(enhancedImage);
end
```