

Lab - 6 Discrete Wavelet Transform Watermarking

```
clc; clear; close all;
% Load Original Image
url =
'https://upload.wikimedia.org/wikipedia/en/7/7d/Lenna_%28test_image%29.png';
img = webread(url); % Correct method to read from URL
if size(img, 3) == 3
    img = rgb2gray(img);
end
% Load Watermark Image (Must be smaller than original)
watermark = imread('watermark.png');
if size(watermark, 3) == 3
    watermark = rgb2gray(watermark);
end
% Resize Watermark to Match HH Subband
[rows, cols] = size(img);
watermark = imresize(watermark, [rows/2, cols/2]);
% Apply DWT to Original Image
[LL, LH, HL, HH] = dwt2(double(img), 'haar');
% Embed Watermark in High-Frequency Subband (HH)
alpha = 0.1; % Strength of watermark
HH_watermarked = HH + alpha * double(watermark);
% Reconstruct Watermarked Image
img_watermarked = idwt2(LL, LH, HL, HH_watermarked, 'haar');
img_watermarked = uint8(img_watermarked); % Ensure values are within
displayable range
% Save and Display Watermarked Image
imwrite(img_watermarked, 'watermarked_image.png');
figure;
subplot(1, 3, 1); imshow(img); title('Original Image');
subplot(1, 3, 2); imshow(img_watermarked); title('Watermarked Image');
subplot(1, 3, 3); imshow(watermark, []); title('Watermark');
% ----- Watermark Removal -----
% Apply DWT to Watermarked Image
[LL2, LH2, HL2, HH2] = dwt2(double(img_watermarked), 'haar');
% Estimate Watermark from HH
watermark_extracted = (HH2 - HH) / alpha;
% Remove Watermark by Nullifying HH
HH2_cleaned = HH2 - alpha * double(watermark);
% Reconstruct Image
img_cleaned = idwt2(LL2, LH2, HL2, HH2_cleaned, 'haar');
img_cleaned = uint8(img_cleaned);
% Save and Display Results
imwrite(img_cleaned, 'watermark_removed.png');
figure;
subplot(1, 3, 1); imshow(img_watermarked); title('Watermarked Image');
subplot(1, 3, 2); imshow(watermark_extracted, []); title('Extracted
Watermark');
```

```
subplot(1, 3, 3); imshow(img_cleaned); title('Watermark Removed');
```

INPUT IMAGE –



Watermarked image–



Watermark-removed image-

