Table of Contents

```
clc;
clear;
close all;
```

1. Read Image

```
I = imread("thala.jpeg");
I = im2uint8(I);

figure, imshow(I), title("Original RGB Image");
```

Original RGB Image

2. RGB -> YCbCr

```
Iycbcr = rgb2ycbcr(I);

Y  = Iycbcr(:,:,1);
Cb = Iycbcr(:,:,2);
```

```
Cr = Iycbcr(:,:,3);
```

3. Force Even Dimensions

```
[H,W] = size(Y);
H2 = floor(H/2)*2;
W2 = floor(W/2)*2;

Y  = Y(1:H2,1:W2);
Cb = Cb(1:H2,1:W2);
Cr = Cr(1:H2,1:W2);
```

4. Chroma Subsampling (4:2:0)

```
Cb_ds = Cb(1:2:end,1:2:end);
Cr_ds = Cr(1:2:end,1:2:end);
```

5. Convert Y to double

```
Y = double(Y);
```

6. Level-1 DWT

```
[LL1,LH1,HL1,HH1] = dwt2(Y,'haar');
```

7. Level-2 DWT (on LL1)

```
[LL2,LH2,HL2,HH2] = dwt2(LL1,'haar');
```

8. Thresholding (Quantization)

```
T1 = 20;
T2 = 30;

LH1(abs(LH1)<T1)=0;
HL1(abs(HL1)<T1)=0;
HH1(abs(HH1)<T1)=0;

LH2(abs(LH2)<T2)=0;
HL2(abs(HL2)<T2)=0;
HH2(abs(HH2)<T2)=0;
```

9. Inverse DWT (IMPORTANT FIX)

First reconstruct LL1

```
LL1_rec = idwt2(LL2,LH2,HL2,HH2,'haar',size(LL1));

% Then reconstruct full Y
```

```
Y_rec = idwt2(LL1_rec,LH1,HL1,HH1,'haar',size(Y));

Y_rec = uint8(min(max(Y_rec,0),255));
```
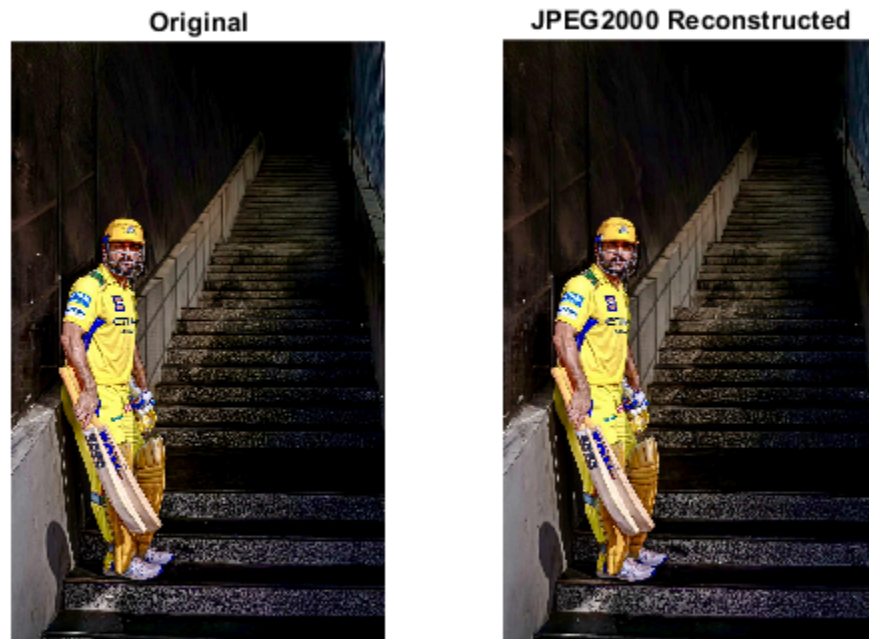
10. Chroma Upsampling

```
Cb_rec = imresize(Cb_ds,[H2 W2],'bilinear');
Cr_rec = imresize(Cr_ds,[H2 W2],'bilinear');
```

11. Recombine & Convert to RGB

```
Iycbcr_rec = cat(3,Y_rec,Cb_rec,Cr_rec);
I_rec = ycbcr2rgb(Iycbcr_rec);
```

12. Comparison

```
figure;
subplot(1,2,1), imshow(I(1:H2,1:W2,:)), title("Original");
subplot(1,2,2), imshow(I_rec), title("JPEG2000 Reconstructed");
```


13. Quality Metrics

```
mse = mean((double(I(1:H2,1:W2,:)) - double(I_rec)).^2,'all');
psnr_val = 10*log10(255^2/mse);
```

```
fprintf("MSE  = %.4f\n",mse);
fprintf("PSNR = %.2f dB\n",psnr_val);

MSE  = 33.4455
PSNR = 32.89 dB
```

Published with MATLAB® R2021a