CompE565, Semester 2022 HW 1

Prepared by

Dustin Nguyen, John Sadiq

E-mail: dnguyen2468@sdsu.edu, jsadiq0352@sdsu.edu

Electrical and Computer Engineering

San Diego State University

Table of Contents

List of figures

List of Tables

List all tables in the document here

Introduction

Discuss the need for image and video coding from the point of storage and transmission bandwidth

Discuss:

What is the domain of this work?

What is the problem addressed? Why is this problem important?

Where does this work fit in the big picture? Provide references to related literature.

Outline the rest of the report

Procedural Section

1. Read and display the image using Matlab

With the given photo named “Landscape Image File.jpg”, I read the image file using ‘imread’ then using the ‘imshow’ command to display the original image. Then output the original image.

2. Display each band (Red, Green, Blue)

Separating the red green and blue from the landscape image file and displaying each out.

3. Convert the image into YCbCr color space.

Separating the Y, Cb, and Cr from the Image.

4. Display each band separately (Y, Cb, Cr)

Displaying the Y, Cb, and Cr from the last stage.

5. Subsample Cb and Cr bands using 4:2:0 and display both

Subsampling 4:2:0 by reducing half of the x & y pixels

6. Upsample & display Cb and Cr bands using

a) Linear Interpolation: Calculate and fill in the missing pixels. For odd-numbered rows, each missing pixel is calculated by taking the average of the two adjacent pixels in the same row (left & right). For even-numbered rows, each missing pixel is calculated by

taking the average of the two adjacent pixels in the same column (above & below).

b) Row or Column Replication: “For odd-numbered rows, each missing pixel will be copied from the previous columns. For even-numbered rows, the odd-numbered rows are added to the subsequent even-numbered rows.”(Image Compression in Matlab)

7. Convert the image into RGB format

Converts the YCbCr image format to RGB.

8. Display the original and the reconstructed image.

After this whole process display the image, it is expected to look the same as the original

9. Comment on the visual quality of the reconstructed images

Looking at both upscaled images from their original size, they seemed to be identical to each other. I believe there are differences in quality but human eyes can't detect those differences, at least without zooming in.

10. Measure MSE between the original and reconstructed images

Calculate the MSE using the formula provided then divide by the (N * M)

11. Comment on the compression ratio achieved by subsampling Cb and Cr

Calculate MSE using the formula provided Then divide by (N*M): subtract the reconstructed image with the original image and get the power of 2 Take the sum and divide by (N*M) Display the result.

Results

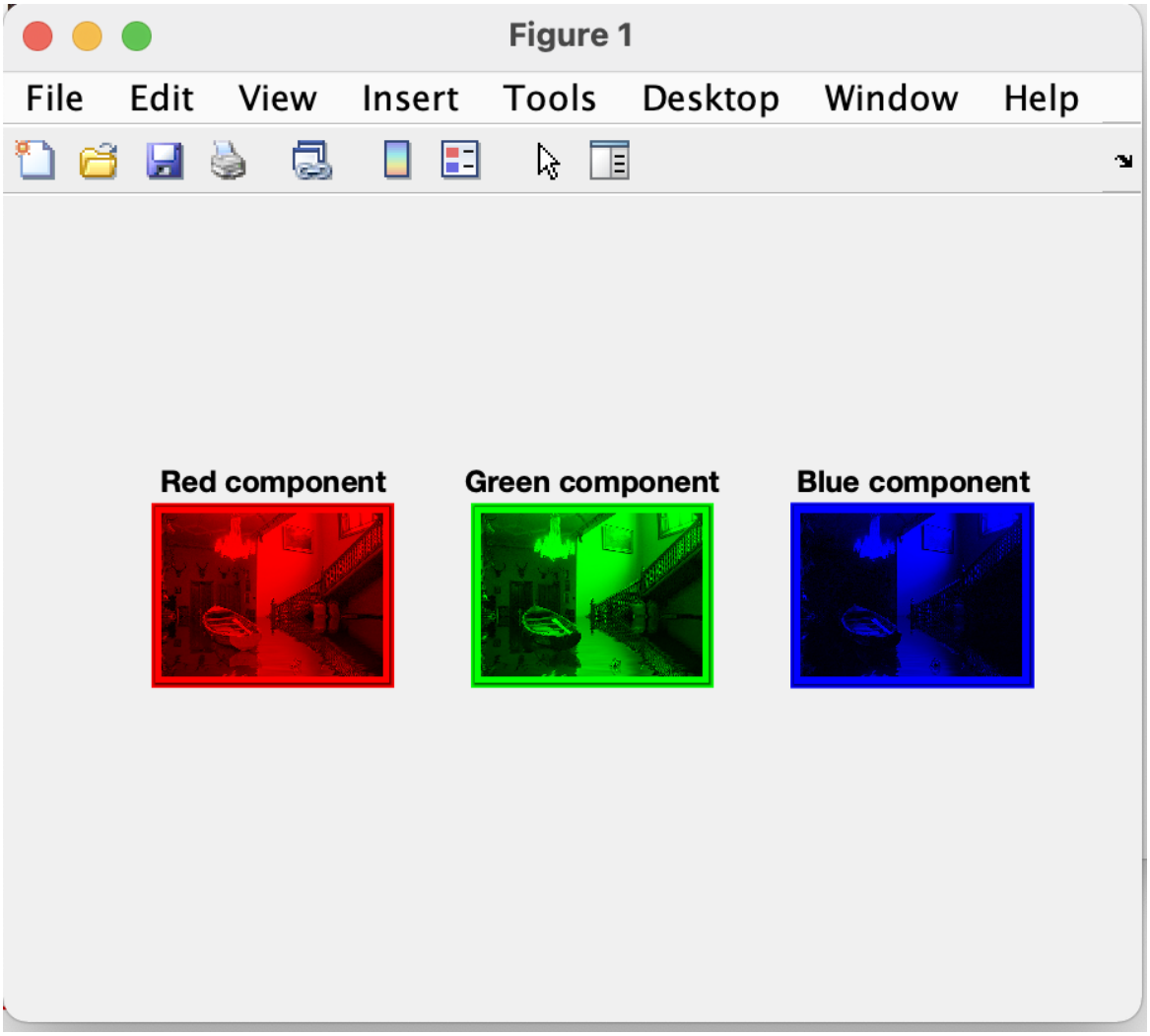


Figure 1
File Edit View Insert Tools Desktop Window Help
Red component
Green component
Blue component

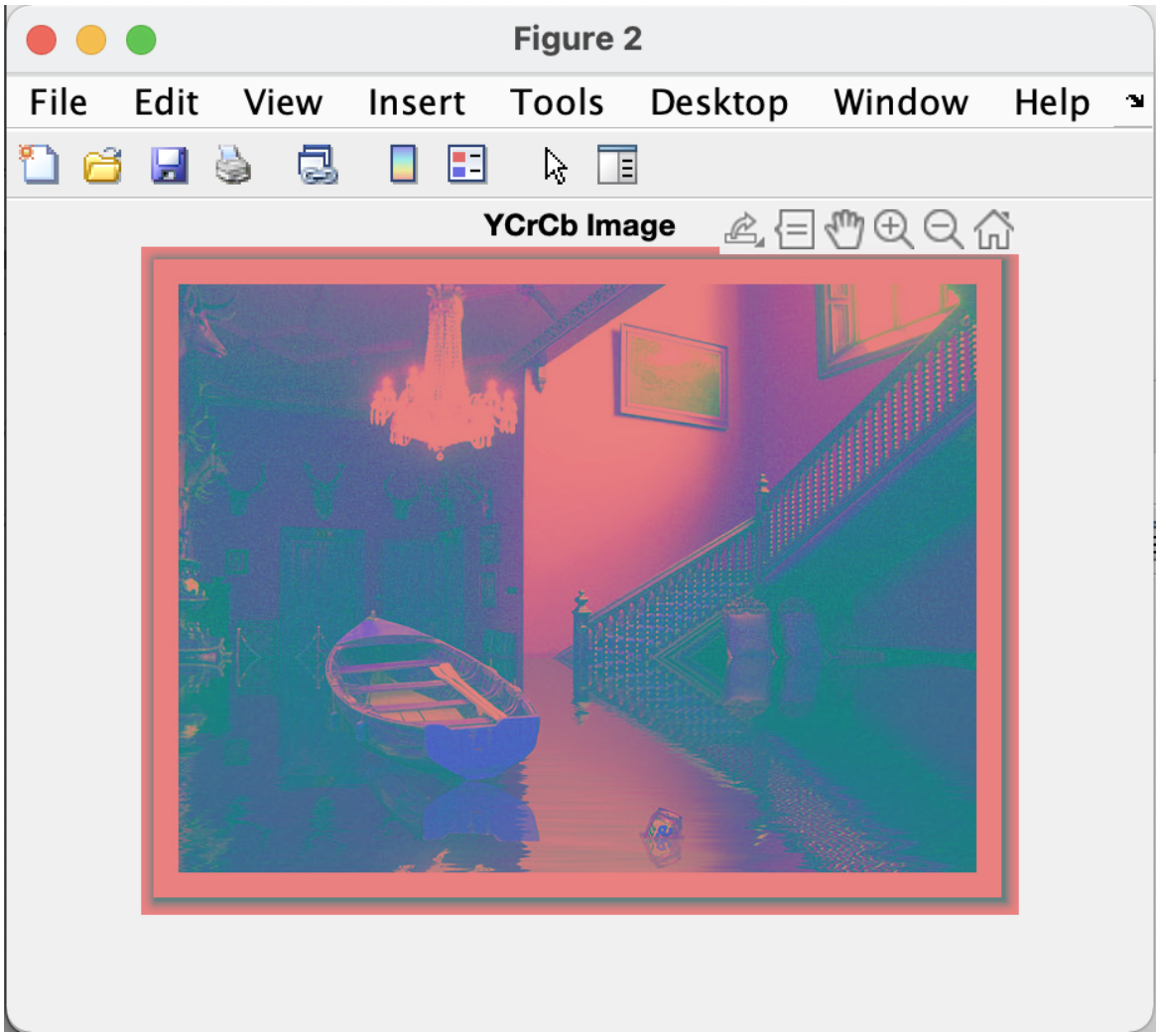


Figure 2
File Edit View Insert Tools Desktop Window Help
YCrCb Image

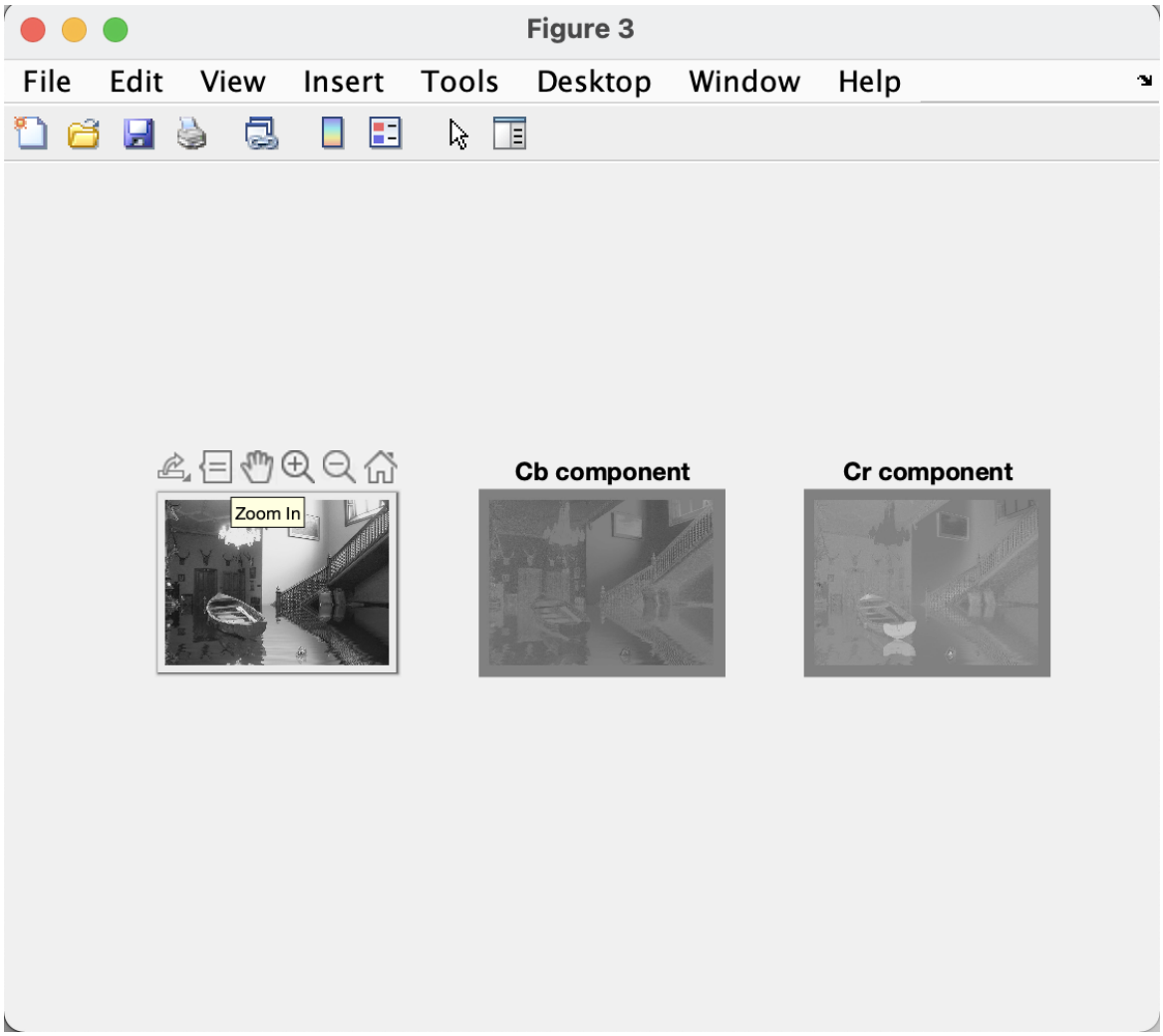


Figure 3
File
Edit
View
Insert
Tools
Desktop
Window
Help
Zoom In
Cb component
Cr component

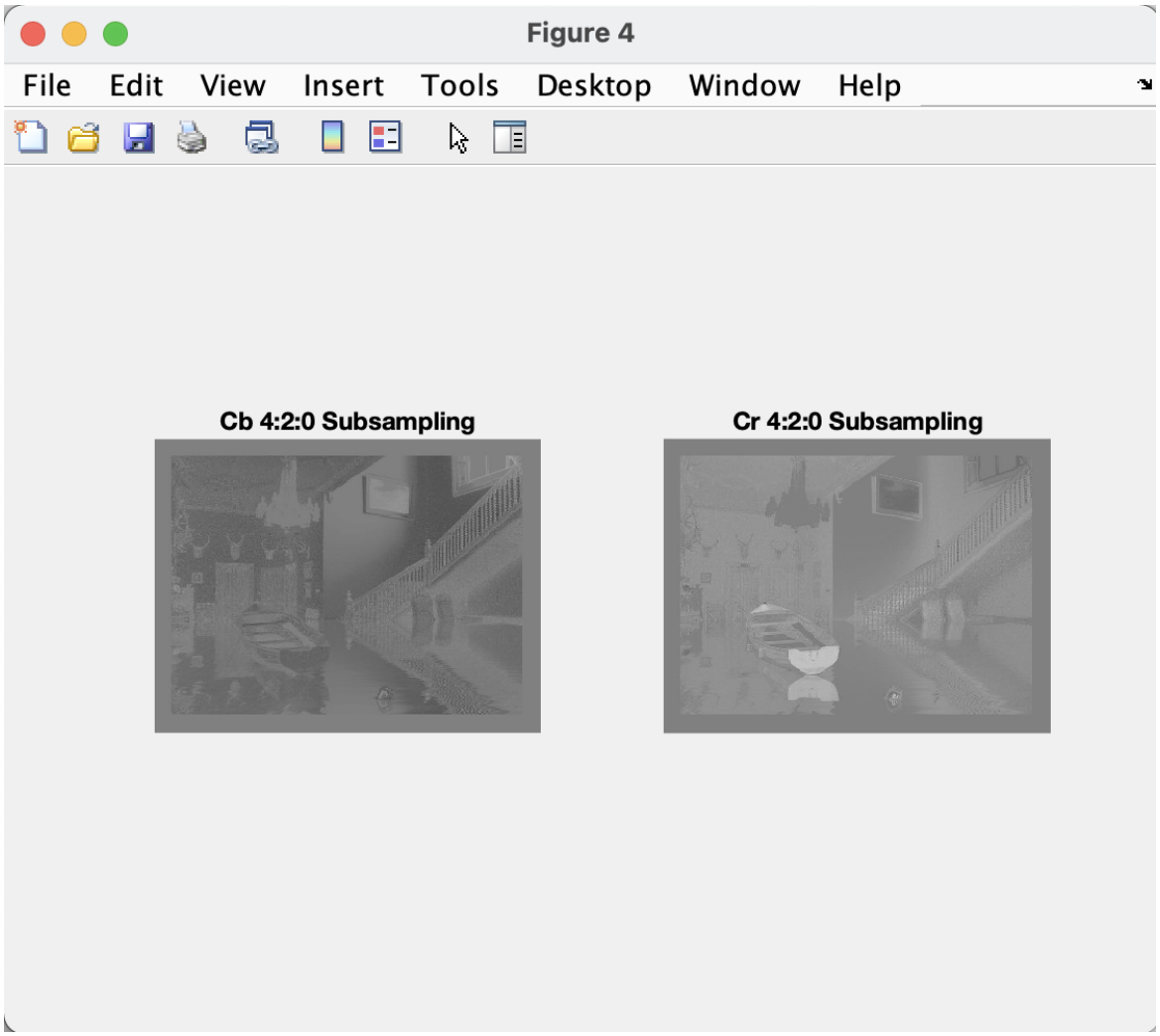


Figure 4
File Edit View Insert Tools Desktop Window Help
Cb 4:2:0 Subsampling
Cr 4:2:0 Subsampling

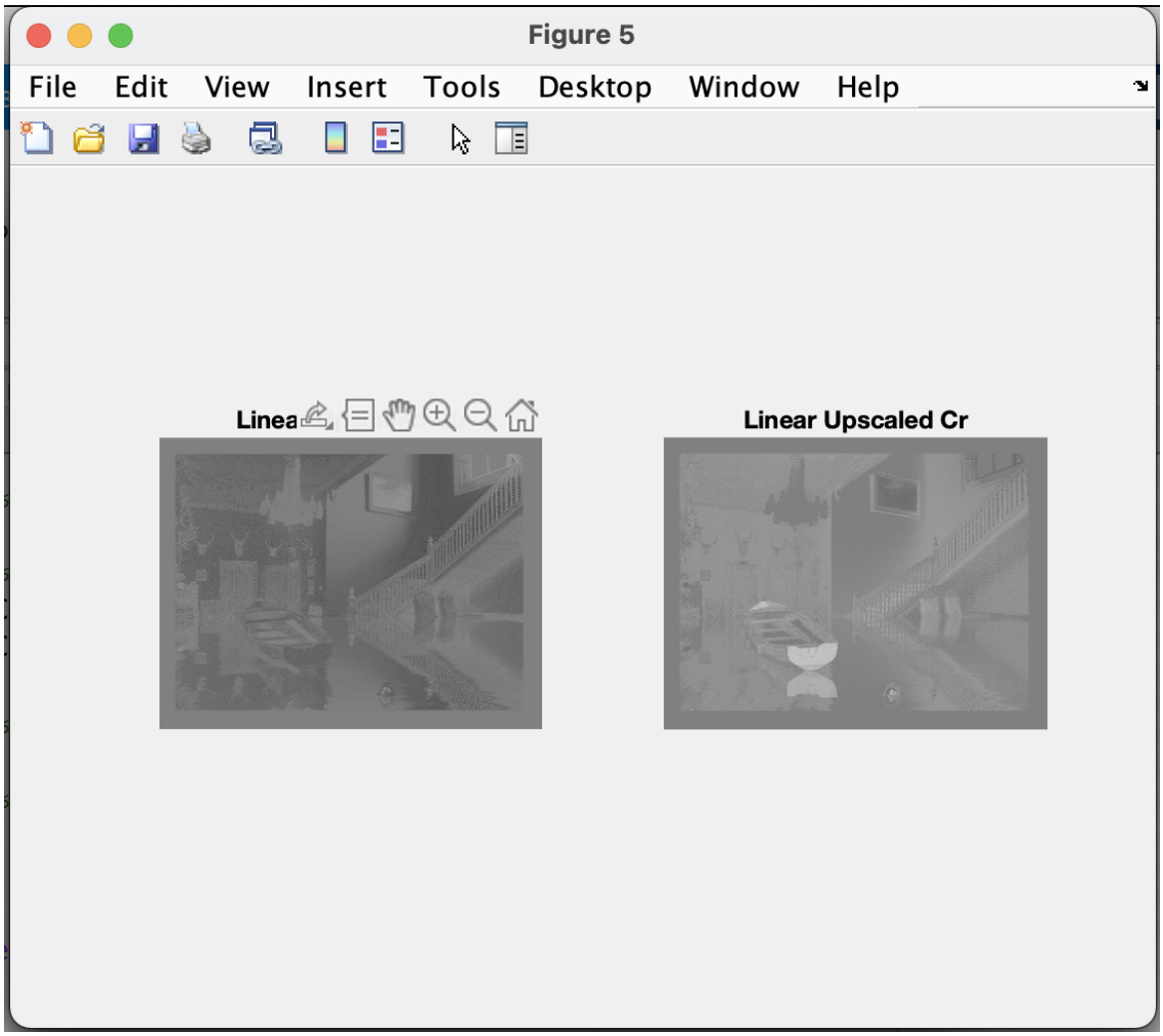


Figure 5
File Edit View Insert Tools Desktop Window Help
Linea
Linear Upscaled Cr

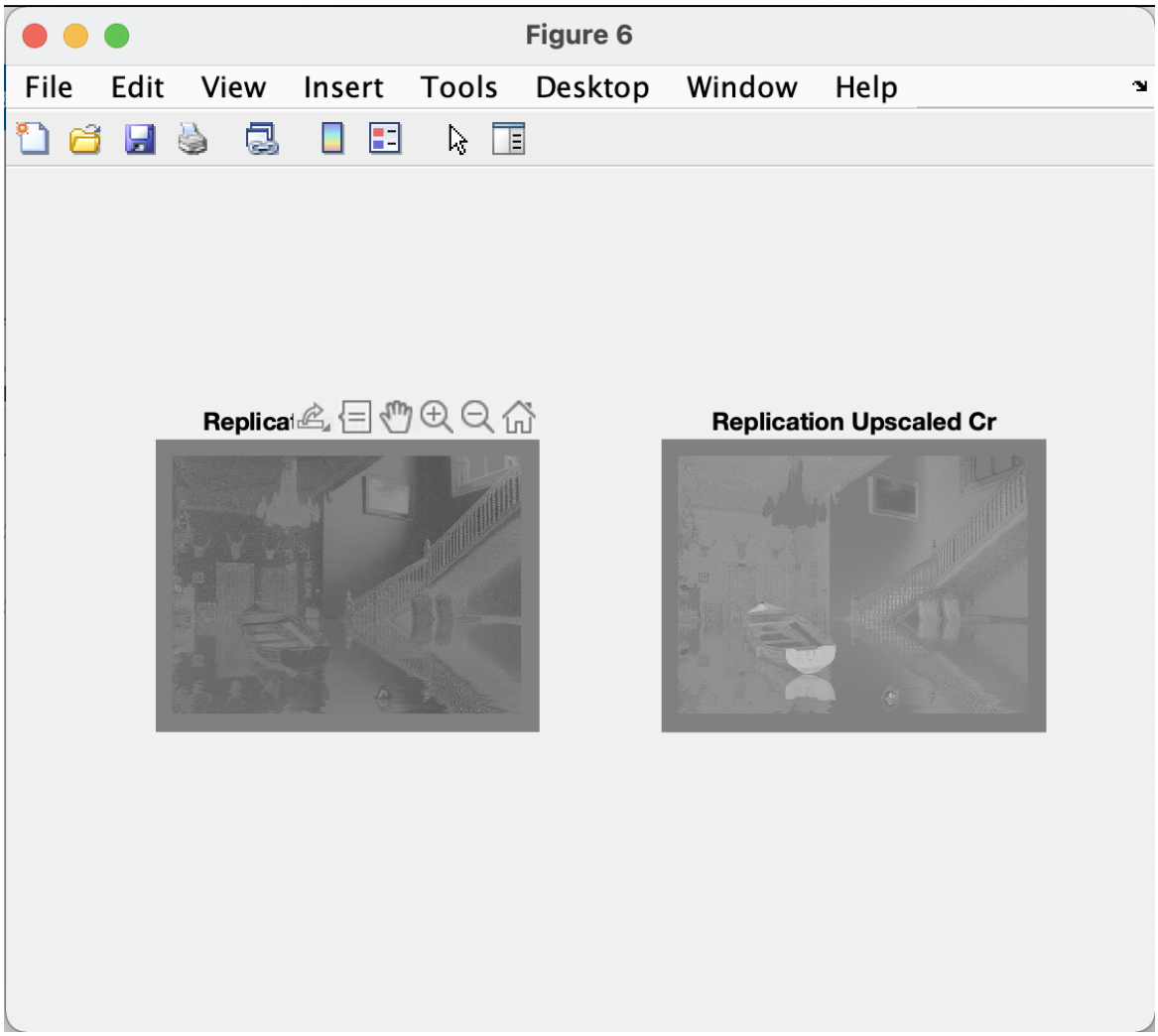
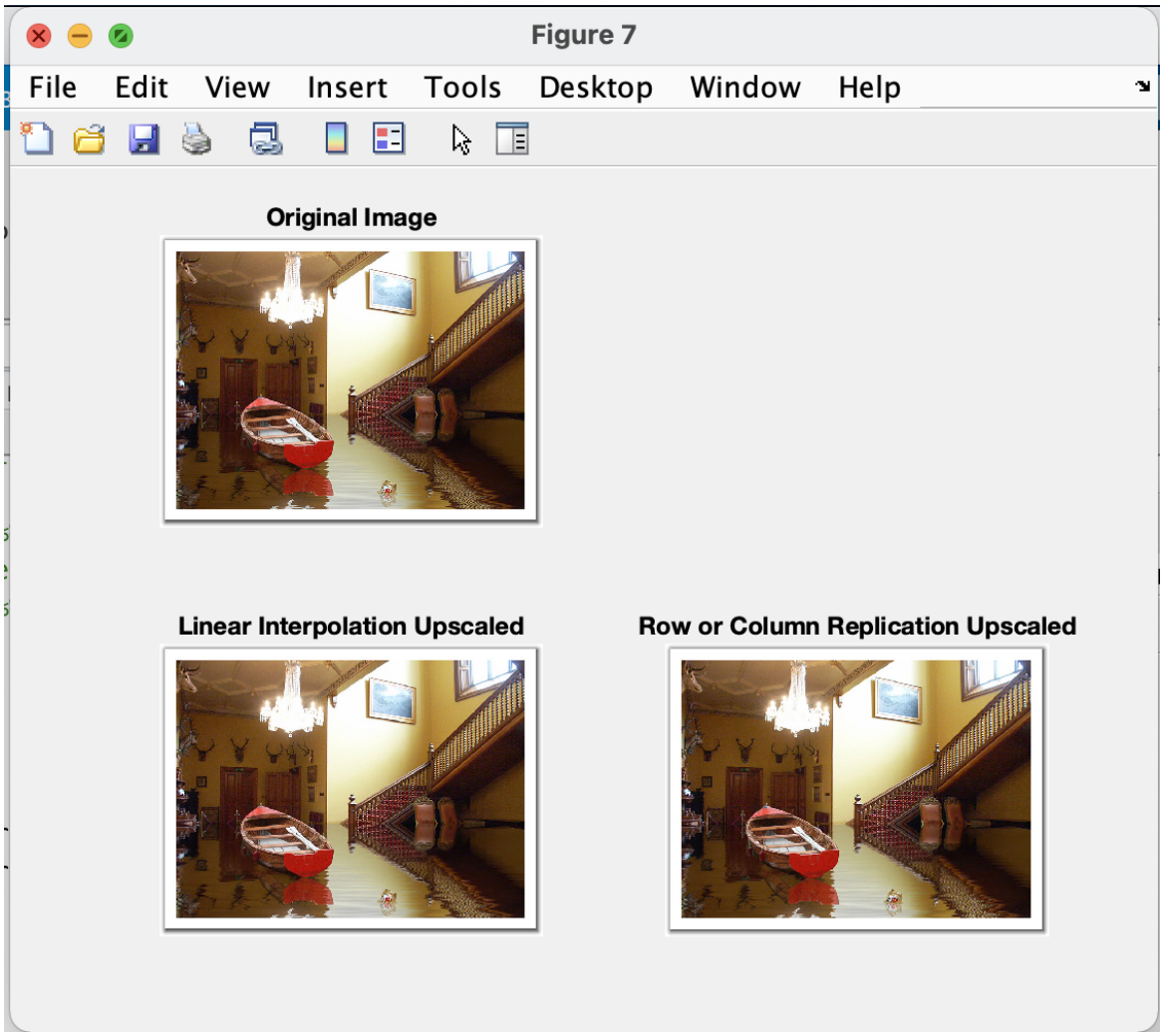


Figure 6
File Edit View Insert Tools Desktop Window Help
Replica
Replication Upscaled Cr




```
>> HW1
MSE for Cb:
    6.1288

MSE for Cr:
    3.7768
```

Conclusion

I and my partner have done a few different processes in this homework assignment, we have compressed an image and remade the same image back to its original size using Matlab, as part of the assignment we converted the image to YCbCr color space, subsampling using 4:2:0 and upsampling using linear interpolation and Simple row-column. As well as measure MSE between the original and reconstructed images.

References

Place all references using the format given below.

[1] https://www.ijser.org/researchpaper/Image-Compression-in-MATLAB-.pdf

Source Code: Please upload the .m file separately as explained in coding_ rules.htm

```
%CompE565 Homework 1

%Feb. 13, 2022

%Names: Dustin Nguyen, John Sadiq

%IDs: 824783655, 825764388

%Emails: dnguyen2468@sdsu.edu, jsadiq0352@sdsu.edu

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 1: Display Original Image

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

I = imread('Flooded_house','jpg');      %Read image into I

% figure,imshow(I)                      %Show Image

% title('Original Image','FontSize',18);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 2: Display each band (Red, Green, Blue)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Red = I(:,:,1);             %Get red component into Red

Green = I(:,:,2);           %Get green component
```

```
Blue = I(:,:,3);            %Get blue component

z = zeros(size(Red));       %Creates zero values

IRed = cat(3,Red,z,z);      %Creates a matrix with red values

IGreen = cat(3,z,Green,z);  %Creates a matrix with green values

IBlue = cat(3,z,z,Blue);    %Creates a matrix with blue values

%Display each matrix/band

% figure(1);

% subplot(1,3,1);

% imshow(IRed);

% title('Red component','FontSize',12);

% subplot(1,3,2);

% imshow(IGreen);

% title('Green component','FontSize',12);

% subplot(1,3,3);

% imshow(IBlue);

% title('Blue component','FontSize',12);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 3: Convert the image into YCbCr color space
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

YCbCr = rgb2ycbcr(I);   %Convert using rgb2ycbcr

% figure(2),imshow(YCbCr);

% title('YCrCb Image','FontSize',12);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 4: Display each band separately (Y,Cb,Cr)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Get each components

Y = YCbCr(:,:,1);

Cb = YCbCr(:,:,2);

Cr = YCbCr(:,:,3);

% figure(3);

% subplot(1,3,1);

% imshow(Y) %Display Y band

% title('Y component','FontSize',12);

% subplot(1,3,2);

% imshow(Cb) %Display Cb band

% title('Cb component','FontSize',12);
```

```
% subplot(1,3,3);

% imshow(Cr) %Display Cr band

% title('Cr component','FontSize', 12);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 5: Subsample Cb and Cr bands using 4:2:0 and display both

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Subsampling 4:2:0 by reducing half of the x & y pixels

SubCb = Cb(1:2:end, 1:2:end);

SubCr = Cr(1:2:end, 1:2:end);

% figure(4);

% subplot(1,2,1);

% imshow(SubCb);

% title('Cb 4:2:0 Subsampling','FontSize', 12);

% subplot(1,2,2);

% imshow(SubCr);

% title('Cr 4:2:0 Subsampling','FontSize', 12);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 6: Upsample & display Cb and Cr bands using
```

```
% 6.1: Linear interpolation

% 6.2: Simple row or column replication

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%6.1 - Upsampling using linear interpolation

%Filling in rows and columns for Cb1 and Cr1 using the adjacent rows &

%columns (average)

YCbCr2(:,:,1) = YCbCr(:,:,1);

YCbCr2(1:2:535,1:2:703,2) = SubCb(:,:);

YCbCr2(1:2:535,1:2:703,3) = SubCr(:,:);

YCbCr2(1:2:535,2:2:702,2:3) = (double(YCbCr2(1:2:535,1:2:701,2:3)) ...

  + double(YCbCr2(1:2:535,3:2:703,2:3)))/2;

YCbCr2(1:2:535,704,2:3) = YCbCr2(1:2:535,703,2:3);

YCbCr2(2:2:534,:,2:3) = (double(YCbCr2(1:2:533,:,2:3)) ...

  + double(YCbCr2(3:2:535,:,2:3)))/2;

YCbCr2(536,:,2:3) = YCbCr2(535,:,2:3);

% figure(5);

% subplot(1,2,1);

% imshow(YCbCr2(:,:,2));
```

```
% title('Linear Upscaled Cb','FontSize', 12);

% subplot(1,2,2);

% imshow(YCbCr2(:,:,3));

% title('Linear Upscaled Cr','FontSize', 12);

%6.2 - Upsampling using row or column replication

% YCbCr3 = YCbCr;

% YCbCr3(1:2:535,2:2:704,2:3) = YCbCr3(1:2:535,1:2:703,2:3);

% YCbCr3(2:2:536,:,2:3) = YCbCr3(1:2:535,:,2:3);

% figure(6);

% subplot(1,2,1);

% imshow(YCbCr3(:,:,2));

% title('Replication Upscaled Cb','FontSize', 12);

% subplot(1,2,2);

% imshow(YCbCr3(:,:,3));

% title('Replication Upscaled Cr','FontSize', 12);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 7: Convert the image into RGB format

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
RGB2 = ycbcr2rgb(YCbCr2);

RGB3 = ycbcr2rgb(YCbCr3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 8: Display the original and the reconstructed image

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% figure(7);

% subplot(2,2,1);

% imshow(I);

% title('Original Image','FontSize', 12);

% subplot(2,2,3);

% imshow(RGB2);

% title('Linear Interpolation Upscaled','FontSize', 12);

% subplot(2,2,4);

% imshow(RGB3);

% title('Row or Column Replication Upscaled','FontSize', 12);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 9: Comment on visual quality of the reconstructed images

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Looking at both upscaled images from its original size, they seemed to be

%identical to each other, I believed there are differences in quality but

%human eyes can't detect those differences, at least without zooming in.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 10: Measure MSE between the original and reconstructed images

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculate MSE using the formula provided

% Then divide by (NM):

double CbMSE;

double CrMSE;

CbMSE = 0;

CrMSE = 0;

% subtract the reconstructed image with original image and get power of 2

% and take the sum and divide by (NM)

for row = 1:1:536

  for col = 1:1:704

  CbMSE = CbMSE + (double(YCbCr2(row,col,2)) ...

     - double(YCbCr(row,col,2))).^2;
```

```
    CrMSE = CrMSE + (double(YCbCr2(row,col,3)) ...

        - double(YCbCr(row,col,3))).^2;

    end

end

%Display Results

disp('MSE for Cb: ');

CbMSE = CbMSE / (704536);

disp(CbMSE);

disp('MSE for Cr: ');

CrMSE = CrMSE / (704536);

disp(CrMSE);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Question 11: Comment on the compression ratio achieved by subsampling Cb and Cr

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculate MSE using the formula provided

%Then divide by (N*M):

%subtract the reconstructed image with original image and get power of 2

%Take the sum and divide by (N*M)
```

```
%Display the result:
```