

**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**VIỆN ĐIỆN TỬ - VIỆN THÔNG**



**BÁO CÁO THÍ NGHIỆM**  
**MÔN: ĐA PHƯƠNG TIỆN**

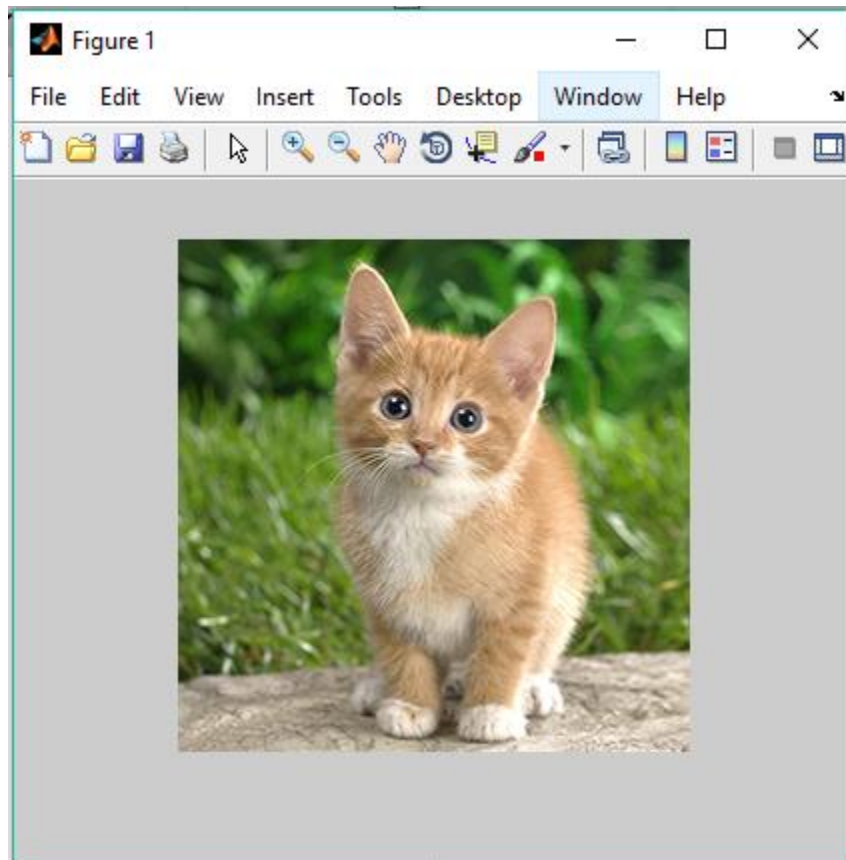
**Sinh viên:** Đỗ Xuân Sơn  
**MSSV:** 20133299  
**Lớp:** Điện tử 06 – K58  
**Mã lớp TN** 666060

# **BÀI 1: CÁC THAO TÁC XỬ LÝ ẢNH TRONG MATLAB**

## **1. Đọc file ảnh bằng Matlab:**

Code: (đề file ảnh, audio và video trong ổ D)

```
A = imread('D:\image.jpg');  
imshow(A);
```



## **2. Hiển thị 3 thành phần R, G, B và vẽ histogram của 3 thành phần:**

- **3 thành phần R, G, B:**

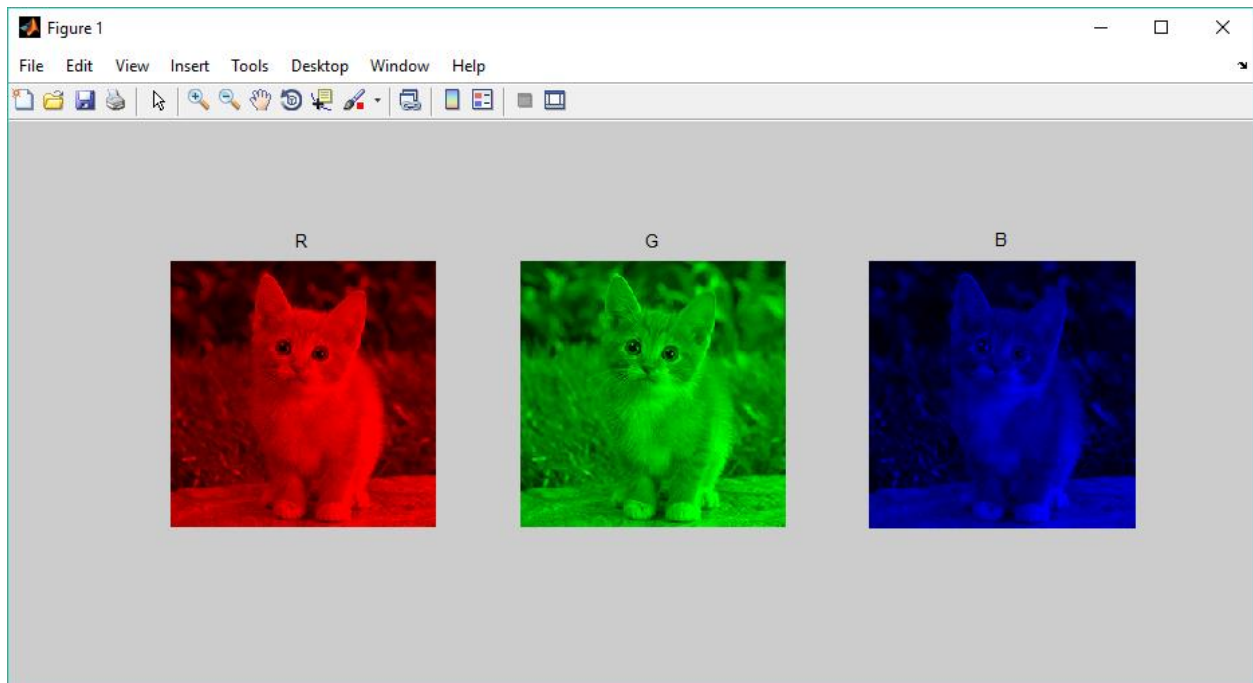
Code:

```
B=A; C=A; D=A;
```

```
B(:, :, 2)=0; B(:, :, 3)=0;
```

```
C(:, :, 1)=0; C(:, :, 3)=0;
```

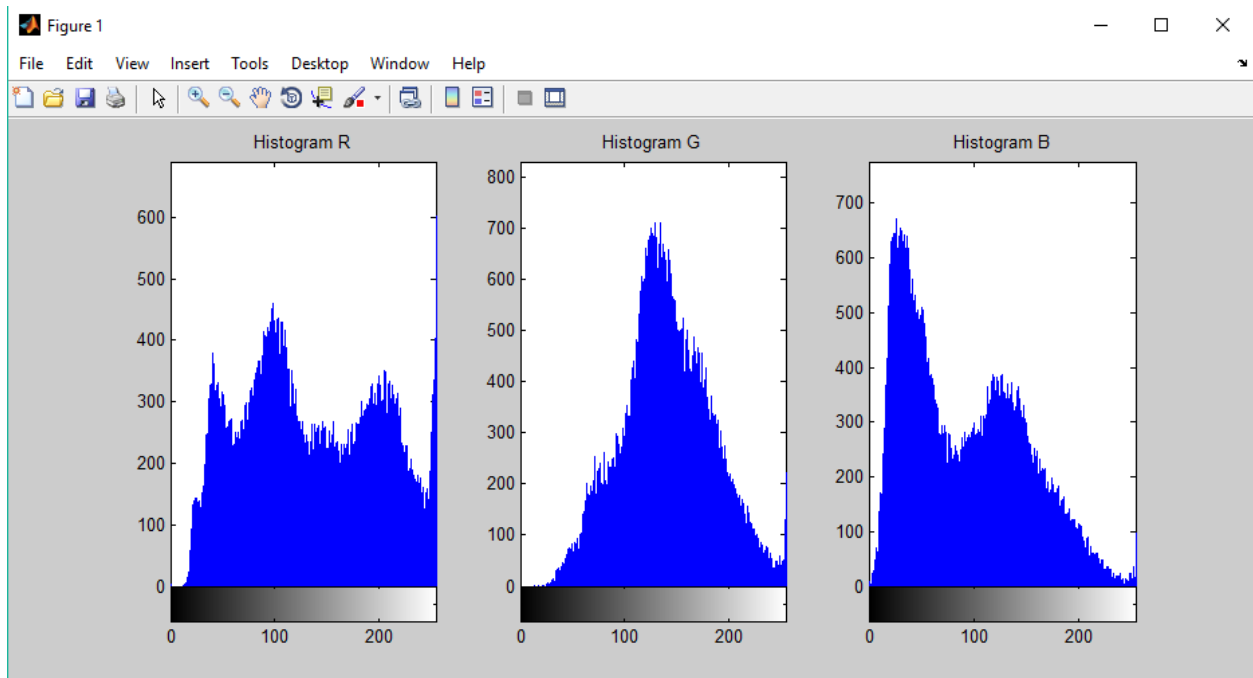
```
D(:,:,1)=0; D(:,:,2)=0;  
  
subplot(1,3,1); imshow(B); title('R');  
  
subplot(1,3,2); imshow(C); title('G');  
  
subplot(1,3,3); imshow(D); title('B');
```



- **Histogram của 3 thành phần trên:**

Code:

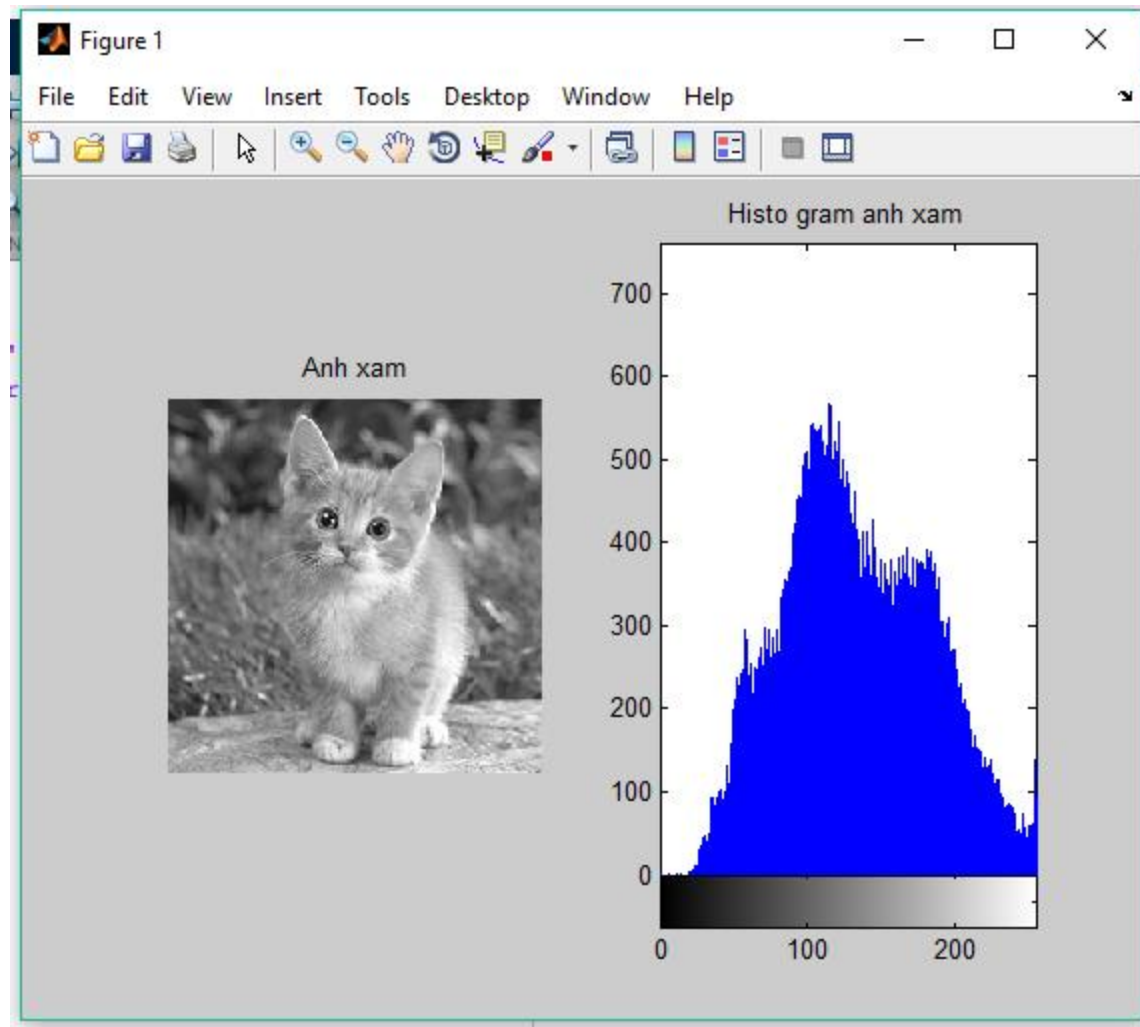
```
subplot(1,3,1); imhist(A(:,:,1)); title('Histogram R');  
subplot(1,3,2); imhist(A(:,:,2)); title('Histogram G');  
subplot(1,3,3); imhist(A(:,:,3)); title('Histogram B');
```



### 3. Chuyển ảnh chân dung thành ảnh xám và vẽ histogram:

Code:

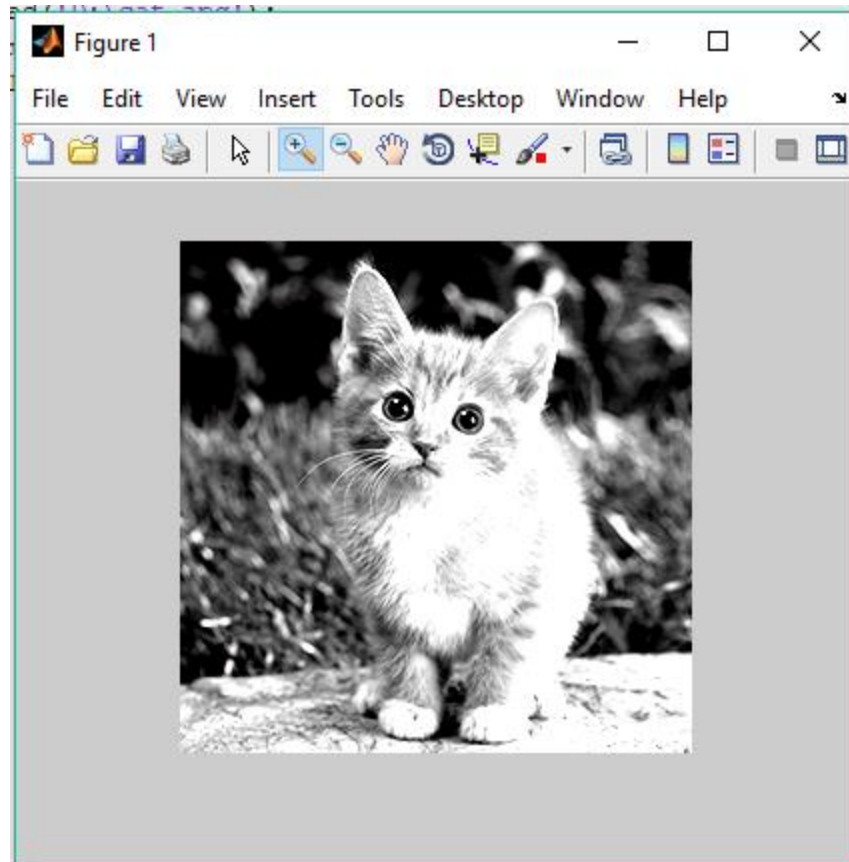
```
A1=rgb2gray(A);  
  
subplot(1,2,1); imshow(A1); title('Anh xam');  
  
subplot(1,2,2); imhist(A1); title('Histo gram anh xam');
```



**Tăng độ tương phản ảnh đầu ra:**

Code:

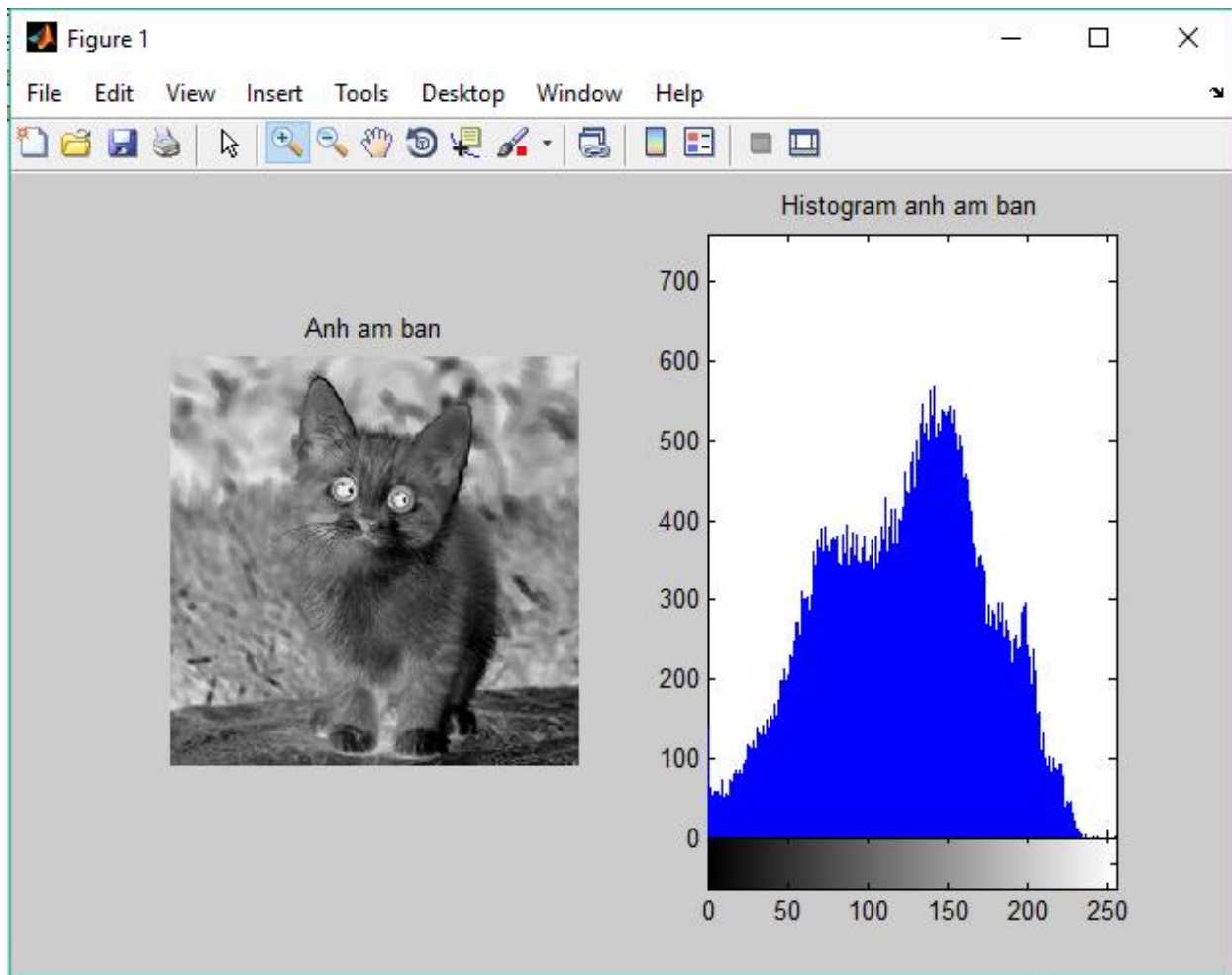
```
A1_adj=imadjust(A1,[0.3,0.7],[]);  
imshow(A1_adj);
```



#### 4. Hiển thị ảnh âm bản và vẽ histogram:

Code:

```
amban=255.-A1;  
  
subplot(121); imshow(amban); title('Anh am ban');  
  
subplot(122); imhist(amban); title('Histogram anh am ban');
```



## **BÀI 2: BIẾN ĐỔI DCT ẢNH KHỐI TRƯỚC KHI NÉN**

### **1. Công thức biến đổi cosine rời rạc thuận và ngược của ma trận 2 chiều 8x8.**

Giá trị  $x(n_1, n_2)$  biểu diễn các mức xám của ảnh trong miền không gian,  $X(k_1, k_2)$  là các hệ số sau biến đổi DCT trong miền tần số.

$$X(k_1, k_2) = \frac{\varepsilon_{k_1} \varepsilon_{k_2}}{4} \sum_{n_1=0}^7 \sum_{n_2=0}^7 x(n_1, n_2) \cos \frac{(2n_1+1)k_1\pi}{16} \cos \frac{(2n_2+1)k_2\pi}{16} \quad (1)$$

$$x(n_1, n_2) = \frac{\varepsilon_{k_1} \varepsilon_{k_2}}{4} \sum_{k_1=0}^7 \sum_{k_2=0}^7 X(k_1, k_2) \cos \frac{(2n_1+1)k_1\pi}{16} \cos \frac{(2n_2+1)k_2\pi}{16} \quad (2)$$

$$\text{Với } \varepsilon_{k_1} = \begin{cases} 1/\sqrt{2} & \text{khi } k_1 = 0 \\ 0 & \text{khi } 1 < k_1 < 8 \end{cases} \quad \text{và} \quad \varepsilon_{k_2} = \begin{cases} 1/\sqrt{2} & \text{khi } k_2 = 0 \\ 0 & \text{khi } 1 < k_2 < 8 \end{cases}$$

**Thực hiện DCT thuận và ngược cho một ma trận 8x8 ngẫu nhiên:**

Code và kết quả:

```
Command Window
>> rd=randint(8, 8, [0,255])
Warning: This is an obsolete function and may be removed in the future. Please use RANDI instead.
> In randint (line 40)

rd =

    228    208     90     97    145     42     58     27
    245     62    212    145    120    154    233    246
    140    237    149     19     3     67     39     1
     35     89    140     13     86    167    211    198
     38     50    234    135     41    176    137    209
     65     64     73    199    203    191    255    222
    215    157    193    239     79    115     20     21
     65    121    192     33    135     21    113    102

fx >> |
```



## Biến đổi DCT thuận:

```
Command Window
>> dct=dct2(rd)

dct =

    1.0e+03 *

    1.0024    0.0261    0.0487   -0.0639   -0.0651    0.0361    0.0511    0.0475
    0.0118    0.0675    0.1149    0.0665   -0.0058   -0.0086   -0.0081    0.0096
   -0.0103    0.2286   -0.0064    0.0507    0.0974   -0.0116   -0.0661    0.0538
    0.1185   -0.0800   -0.0764    0.0373    0.1155    0.0549   -0.0201   -0.0641
   -0.0934   -0.0599    0.0121   -0.0439   -0.0504   -0.0725    0.0027    0.0690
   -0.0876    0.1742   -0.0277    0.0757   -0.0136   -0.0492   -0.0614   -0.0639
   -0.1055    0.0129   -0.0048    0.0430   -0.0337   -0.0954   -0.1211    0.0122
   -0.1099    0.2579   -0.0292   -0.0404   -0.0417   -0.0151    0.0472   -0.0741

fx >> |
```

## Biến đổi DCT ngược:

```
Command Window

-0.1099    0.2579   -0.0292   -0.0404   -0.0417   -0.0151    0.0472   -0.0741

>> idct=idct2(rd)

idct =

    881.0650   -205.4284   272.9628   -136.3474    31.6475    -8.5806   100.1829    79.7583
   -228.0232   187.6999    32.3012   110.7241    11.4732    11.9765   -37.0192    32.9292
    285.1053    89.2474   -40.8235    55.3688   118.8778    73.5129    -2.8797    55.0488
    17.2520   -170.0222   -13.2853   -16.6374    97.9849    57.6098    44.0867   -23.8942
    39.7175   -26.3334    26.8179   -65.0958   -15.5238   -77.1432   -14.3321    72.2466
     8.4807    74.2659   -32.0590   108.0822    20.0742    38.3680   -10.0834   -50.2994
   -24.5265   -33.0040    33.8719    72.4359    27.8634   -56.4343   -156.9407   -15.6145
   -14.0781   277.8522   -97.4949    88.6009   -61.4884   -1.4542    -5.4379   -71.2075

fx >> |
```

## **2. Thực hiện DCT ma trận M bằng Matlab:**

```
A=[20 20 20 20 20 20 20 20];
```

```
B=2*A;
```

```
M= [A; A; A; A; B; B; B; B];
```

```
M1=dct2(M)
```

```

Command Window

    40    40    40    40    40    40    40    40
    40    40    40    40    40    40    40    40
    40    40    40    40    40    40    40    40
    40    40    40    40    40    40    40    40

>> b4

M1 =

240.0000     0     0     0     0     0     0     0
-72.4902     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0
 25.4552     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0
-17.0086     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0
 14.4192     0     0     0     0     0     0     0

fx >>

```

### 3. Thực hiện mô phỏng:

Code:

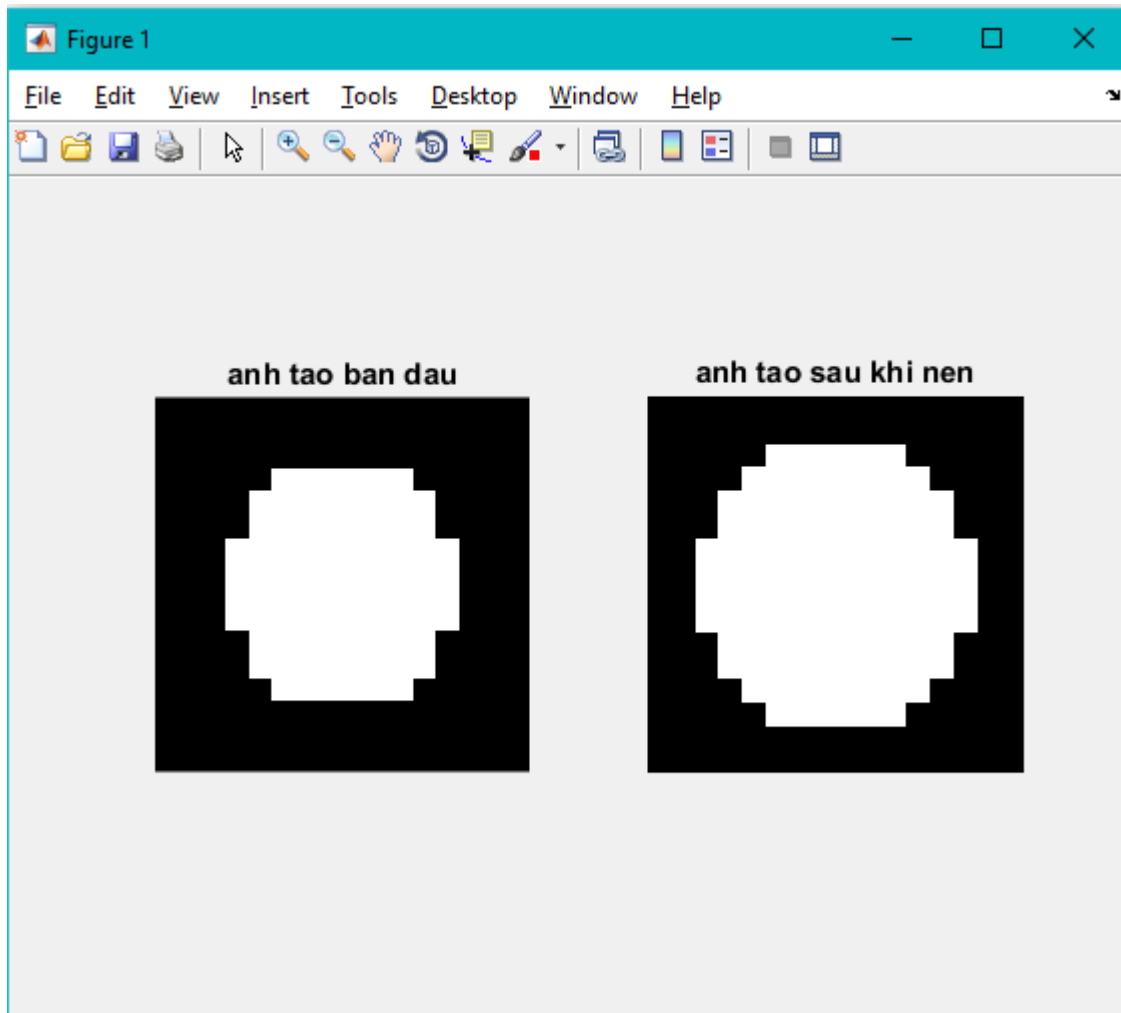
```

A=zeros(16); B=zeros(16); %tao ma tran 0 16x16
A(4,6:11)=255;
A(5:6,5:12)=255;
A(7:10,4:13)=255;
A(11:12,5:12)=255;
A(13,6:11)=255;
subplot(121); imshow(A); title('anh tao ban dau');
%chia A thanh 4 khoi 8x8
A1=A(1:8,1:8); A2=A(1:8,9:16); A3=A(9:16,1:8); A4=A(9:16,9:16);
%thuc hien DCT cac khoi anh
A1_dct=dct2(A1); A2_dct=dct2(A2); A3_dct=dct2(A3);
A4_dct=dct2(A4);
%bang trong so luong tu hoa
Q=[16 11 10 16 24 40 51 61;
    12 12 14 19 26 58 60 55;
    14 13 16 24 40 57 69 56;
    14 17 22 29 51 87 80 62;
    18 22 37 56 68 109 103 77;
    24 35 55 64 81 104 113 92;
    49 64 78 87 103 121 120 101;
    72 92 95 98 112 100 103 99];
%chia cho bang luong tu
A1_lt=A1_dct./Q; A2_lt=A2_dct./Q; A3_lt=A3_dct./Q;
A4_lt=A4_dct./Q;
%bien doi DCT nguoc
A1_I=idct2(A1_lt); A2_I=idct2(A2_lt); A3_I=idct2(A3_lt);
A4_I=idct2(A4_lt);

```

```
%ghép lại ảnh
B(1:8,1:8)=A1_I; B(1:8,9:16)=A2_I; B(9:16,1:8)=A3_I;
B(9:16,9:16)=A4_I;
subplot(122); imshow(B); title('anh tao sau khi nen');
```

*Ảnh trước và sau mô phỏng:*



## **BÀI 3: TIẾP CẬN XỬ LÝ AUDIO VÀ VIDEO**

- **AUDIO:**

### **1. Nạp một file audio và vẽ dạng sóng:**

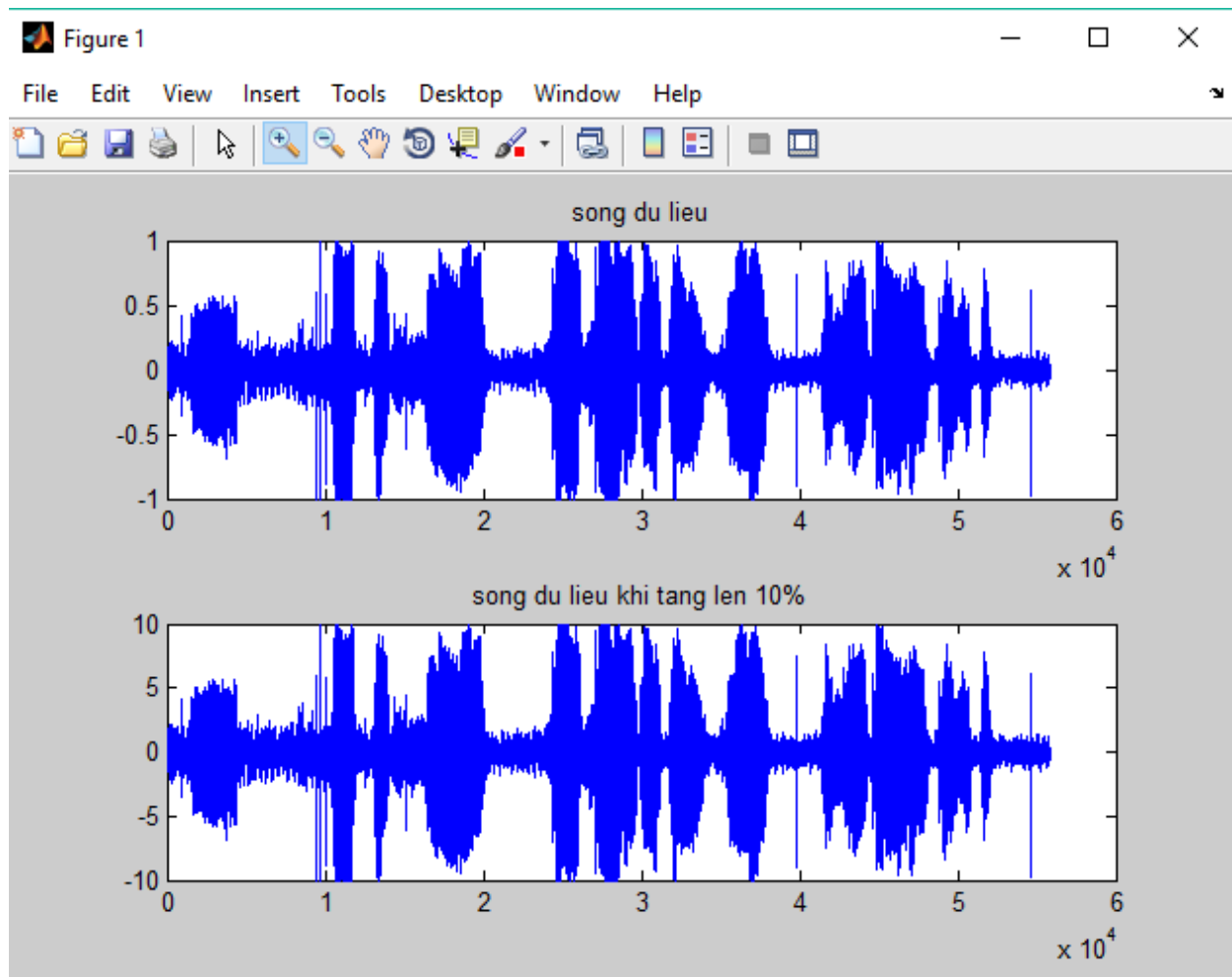
Code:

```
[y, fs]=wavread('D:\eagle.wav');  
  
sound(y, fs);
```

### **Dạng sóng dữ liệu và khi tăng kích thước sóng lên 10%**

Code:

```
[data, Fs]= audioread('D:\eagle.wav');  
subplot(2,1,1);  
plot(data);title ('song du lieu');  
A=data.*10;%tang kích thước dạng sóng lên 10%  
sound(A, Fs);  
subplot(2,1,2);  
plot(A);title ('song du lieu khi tang len 10%');
```



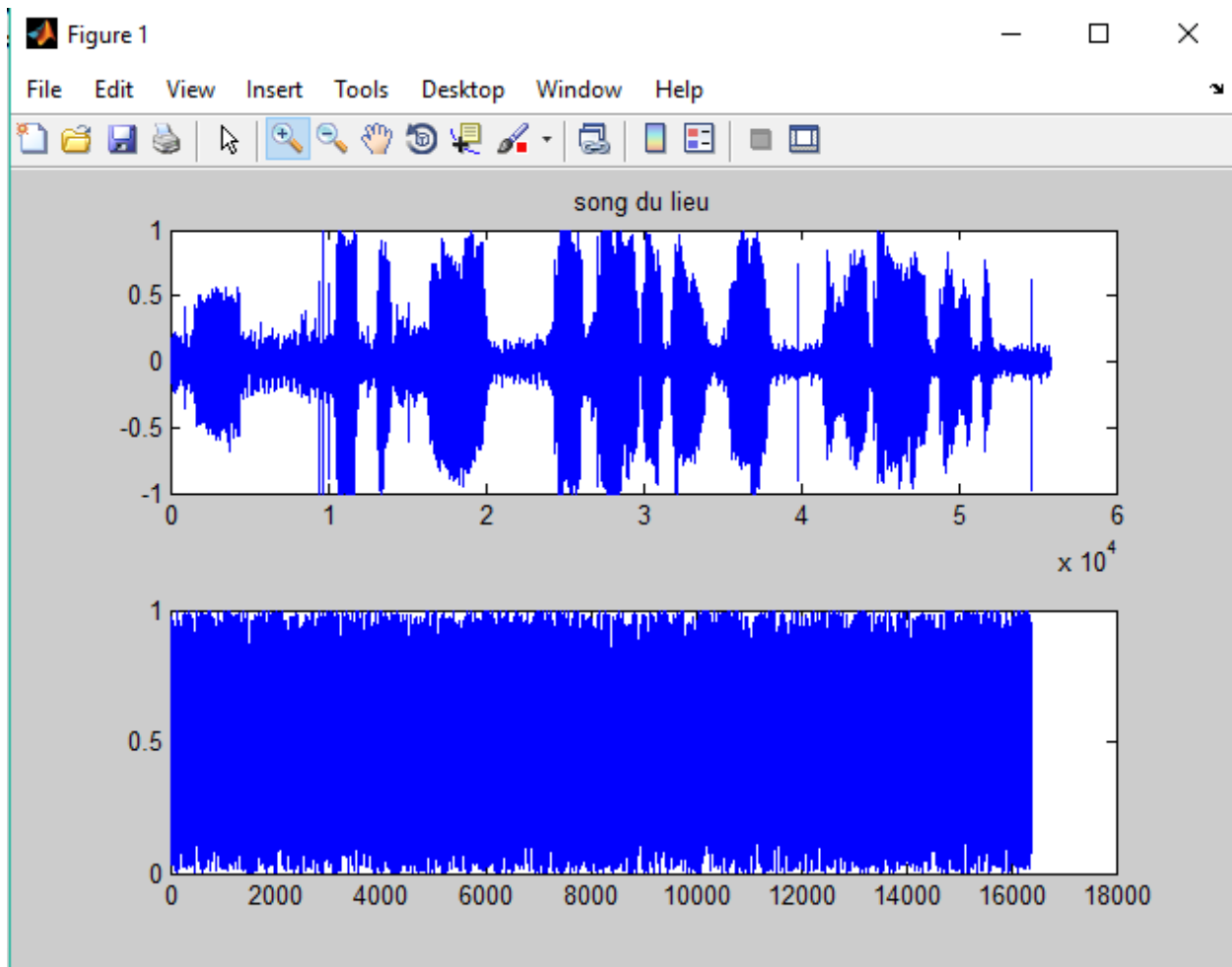
## 2. Tạo một vecto âm thanh và vẽ dạng sóng:

Code:

```
r=rand(1,2*8192);
```

```
sound(r);
```

```
plot(r);
```



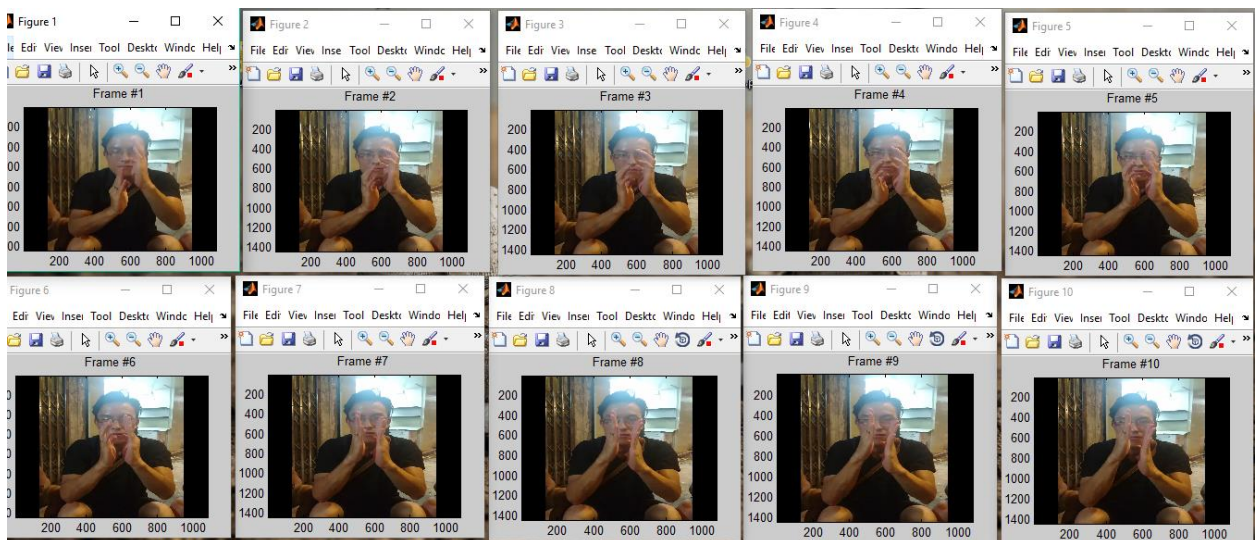
- **Video:**

## 1. Bật video trong matlab:

```
M=VideoReader('D:\Hung.avi');  
video=M.read();  
imshow(video);
```

## 2. Hiện thị từng khung ảnh(frame) như 1 ảnh Matlab:

```
file=aviinfo('D:\Hung.avi');  
nFrames=file.NumFrames;  
Rate=file.AudioRate;  
for k = 1:10  
    this_frame = read(M, k);  
    thisfig = figure();  
    image(this_frame);  
    title(sprintf('Frame #%d', k));  
end
```



### 3. Tạo ra 1 đoạn phim mới bằng cách hoán vị các frame gốc ban đầu:

```
newavi=VideoWriter('newvideo.avi'); %tao ra 1 file avi moi
newavi.FrameRate=Rate;
open(newvideo);
for k=200:(-1):1 %tao 200 khung anh
    this_frame = read(M, k); %doc tung khung mot
    writeVideo(newvideo,this_frame);
end
disp('Closing movie file'); %hien thi len dong lenh chinh
disp('display newvideo');
imshow('newvideo.avi');
```



## 4. Mã hóa và giải mã JPEG

### 4.1, Mã hóa

Code

```
clear all, close all;
imRGB = imread(D:\cat.jpg');
figure, imshow(imRGB), title('RGB Full Image');
imYIQ = rgb2ntsc(imRGB);

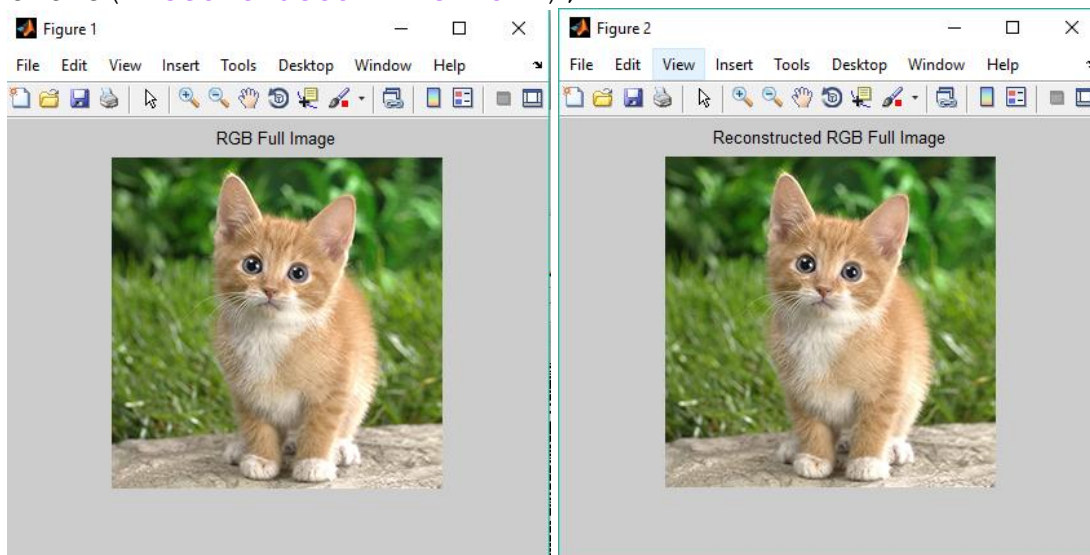
imYIQsubI = imresize(imYIQ(:, :, 2), 0.5, 'bilinear');
imYIQsubQ = imresize(imYIQ(:, :, 3), 0.5, 'bilinear');

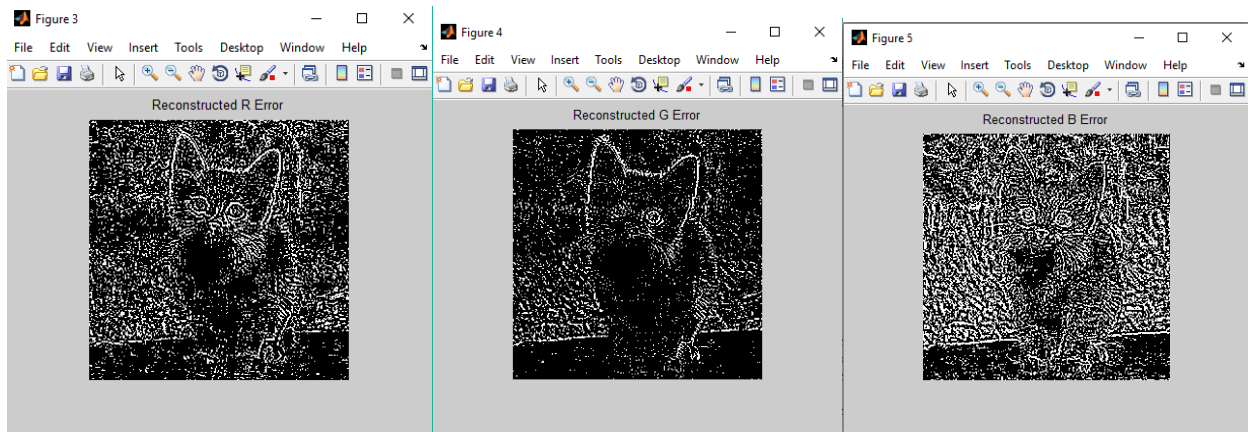
imYIQupsampI = imresize(imYIQsubI, 2);
imYIQupsampQ = imresize(imYIQsubQ, 2);

reconstruct_imYIQ = imYIQ;
reconstruct_imYIQ(:, :, 2) = imYIQupsampI;
reconstruct_imYIQ(:, :, 3) = imYIQupsampQ;

reconstruct_imRGB = uint8(256*ntsc2rgb(reconstruct_imYIQ));
figure, imshow(reconstruct_imRGB), title('Reconstructor RGB Full Image');

figure, imshow(256*abs(imRGB(:, :, 1)-reconstruct_imRGB(:, :, 1)));
title('Reconstructor R error');
figure, imshow(256*abs(imRGB(:, :, 2)-reconstruct_imRGB(:, :, 2)));
title('Reconstructor G error');
figure, imshow(256*abs(imRGB(:, :, 3)-reconstruct_imRGB(:, :, 3)));
title('Reconstructor B error');
```





## 4.2, Giải mã

Code

```
I = imread('D:\cat.bmp');
I1 = I;
I2 = double(I);
[row coln] = size(I);
I = double(I);

I = I - (128.);

QX = [16 11 10 16 24 40 51 61;
      12 12 14 19 26 58 60 55;
      14 13 16 24 40 57 69 56;
      14 17 22 29 51 87 80 62;
      18 22 37 56 68 109 103 77;
      24 35 55 64 81 104 113 92;
      49 64 78 87 103 121 120 101;
      72 92 95 98 112 100 103 99];

DCT_matrix8 = dct(eye(8));
iDCT_matrix8 = DCT_matrix8;

dct_restored = zeros(row, coln);
```

```

QX = double(QX);

for i1 = [1:8:row]
for i2 = [1:8:coln]
    zBLOCK = I(i1:i1+7, i2:i2+7);
    win1 = DCT_matrix8*zBLOCK*iDCT_matrix8;
    dct_domain(i1:i1+7, i2:i2+7) = win1;
end
end

for i1=[1:8:row]
for i2=[1:8:coln]
    win1 = dct_domain(i1:i1+7,i2:i2+7);
    win2=round(win1./QX);
    dct_quantized(i1:i1+7,i2:i2+7)=win2;
end
end

for i1=[1:8:row]
for i2=[1:8:coln]
    win2 = dct_quantized(i1:i1++7,i2:i2+7);
    win3 = win2.*QX;
    dct_dequantized(i1:i1+7,i2:i2+7)=win3;
end
end

for i1=[1:8:row]
for i2=[1:8:coln]
    win3 = dct_dequantized(i1:i1+7,i2:i2+7);
    win4=iDCT_matrix8*win3*DCT_matrix8;
    dct_restored(i1:i1+7,i2:i2+7)=win4;
end
end
I2=dct_restored;
K=mat2gray(I2);
figure(1);imshow(I1);
figure(2);imshow(K);title('Restored image from DCT');

```

