

# 第八章作业（5月14号）：

## 一、实验作业：

1、分别用区域编码和阈值编码方法实现图像压缩，用  $8 \times 8$  DCT 变换，保留 50% 的系数(区域编码保留前 50%个系数，阈值编码保留 50%的大系数)，并对解码图像进行比较。

(要求：DCT 要自己实现，不能用 matlab 中的 DCT 函数)

## 二、代码：

### 函数 dct.m

```
function dct_my = dct(f,m,n)
% 要求 f 为 8*8 块
% m=n=8;
% f = (rand(n,n)*100); % 生成 8x8 块
for i=0:m-1
    for j=0:n-1
        if i == 0
            c = sqrt(1/m);
        else
            c = sqrt(2/m);
        end
        A(i+1, j+1) = c * cos( (j + 0.5)* pi * i / m );
    end
end

dct_my = A*f*A';

dct_matlab = dct2(f);
```

### 函数 idct.m

```
function idct_my = idct(f,m,n)
% 要求 f 为 8*8 块
% m=n=8;
% f = (rand(n,n)*100); % 生成 8x8 块
for i=0:m-1
```

```

for j=0:n-1
    if i == 0
        c = sqrt(1/m);
    else
        c = sqrt(2/m);
    end
    A(i+1, j+1) = c * cos( (j + 0.5)* pi * i / m );
end
end

idct_my = A'*f*A;

idct_matlab = idct2(f);

```

## DCT 变换和反变换

```

% DCT 变换和反变换

clc;clear;
img = rgb2gray(imread('Esther Heesch. jpg'));
figure;imshow(img); xlabel('原图')

% 1, 使图像行列为 8 的倍数
n = 8;
[row,col] = size(img);
row = round(row/n) * n;
col = round(col/n) * n;
img = imresize(img, [row, col]);

% 2, 对图像块进行 dct 变换
img_dct = zeros(row, col); % 存放转换后的 dct 系数
for i=1:n:row-7
    for j=1:n:col-7
        img_block = im2double(img(i:i+7, j:j+7));
        dct_block = dct(img_block,n,n); % 也可用刚才实现的(定义成一个函数即可)
        % imshow(dct_block); % 显示 dct 块
        img_dct(i:i+7, j:j+7) = dct_block;
    end
end
figure;imshow(img_dct); xlabel('dct 变换图像')
% figure, imshow(img_dct); % 显示生成的 dct 系数

% 3, dct 反变换

```

```

new_img = zeros(row,col);
for i=1:n:row-7
    for j=1:n:col-7
        dct_block = img_dct(i:i+7, j:j+7);
        img_block = idct(dct_block,n,n); % 也可用刚才实现的(定义成一个函数即可)
        new_img(i:i+7, j:j+7) = img_block;
    end
end
figure;imshow(new_img); xlabel('dct 反变换图像')
% figure, imshow(mat2gray(new_img)); % 显示反变换回来的图像

```

## 阈值编码

```

% 阈值编码保留 50%的大系数的值

clc;clear;
img = rgb2gray(imread('Esther Heesch1.jpg'));
figure,subplot(221), imshow(img); xlabel('原图')
[m,n] = size(img);

% 1, 对整个图像 dct 变换
img = im2double(img);
% dct_img = dct2(img); % 系统自带函数
dct_img = dct(img,m,n); % 自己写的函数
subplot(222),imshow(1000*abs(dct_img), colormap(gray(5)));
colorbar;
xlabel('dct 变换图像')

% 2, 量化, 阈值编码保留 50%的大系数的值, 使得矩阵中小于中位数的点置为 0, 变得稀疏
arr=dct_img(:); % 矩阵变列向量
M = median(arr); % 求中位数
dct_img(abs(dct_img)<M)=0;
subplot(224),imshow(1000*abs(dct_img), colormap(gray(5)));
colorbar;
xlabel('阈值处理 dct 变换图像')

% 3, 反变换回来
% new_img = idct2(dct_img); % 系统自带函数
new_img = idct(dct_img,m,n); % 自己写的函数
new_img = mat2gray(new_img);
subplot(223),imshow(new_img);
xlabel('dct 反变换图像')

```

## 区域编码

```
% 区域编码保留前 50%个系数

clc;clear;
img = rgb2gray(imread('Esther Heesch1.jpg'));
figure,subplot(221), imshow(img); xlabel('原图')
[m,n] = size(img);

% 1, 对整个图像 dct 变换
img = im2double(img);
% dct_img = dct2(img); % 系统自带函数
dct_img = dct(img,m,n); % 自己写的函数
subplot(222), imshow(1000*abs(dct_img), colormap(gray(5)));
colorbar;
xlabel('dct 变换图像')

% 2, 量化, 区域编码保留前 50%个系数, 其他点置为 0, 变得稀疏
for i=1:round(sqrt(0.5)*m)
    for j=round(sqrt(0.5)*n):n
        dct_img(i,j)=0;
    end;
end;
for i=round(sqrt(0.5)*m):m
    for j=1:n
        dct_img(i,j)=0;
    end;
end;
subplot(224), imshow(1000*abs(dct_img), colormap(gray(5)));
colorbar;
xlabel('区域处理 dct 变换图像')

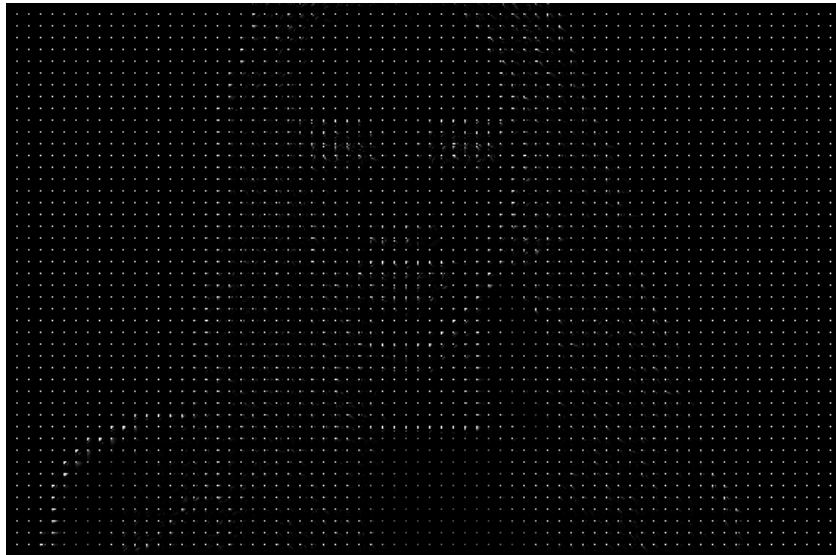
% 3, 反变换回来
% new_img = idct2(dct_img); % 系统自带函数
new_img = idct(dct_img,m,n); % 自己写的函数
new_img = mat2gray(new_img);
subplot(223), imshow(new_img);
xlabel('dct 反变换图像')
```

三、运行结果：

### DCT 变换和反变换



原图



dct变换图像

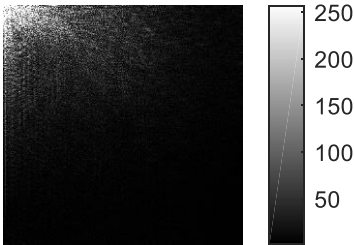


dct反变换图像

阈值编码



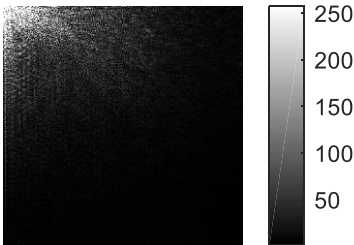
原图



dct变换图像



dct反变换图像

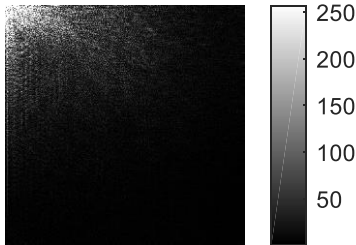


阈值处理dct变换图像

# 区域编码



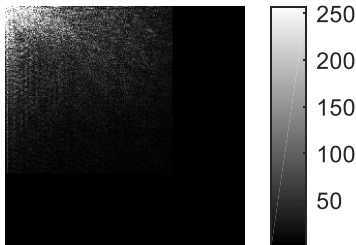
原图



dct变换图像



dct反变换图像



区域处理dct变换图像

## 四、结果分析：

阈值编码将大的 50%点保留，区域编码将面积的左上角 50%保留，显然前者恢复之后效果更好。

## 五、遇到的问题以及解决办法：

### 问题：

DCT 算法的实现

### 解决方法：

查找资料，找到算法，如下：

#### 一、背景介绍

DCT，即离散余弦变换，常用图像压缩算法，步骤如下

- 1) 分割，首先将图像分割成 8×8 或 16×16 的小块；
- 2) DCT 变换，对每个小块进行 DCT 变换；

3) 舍弃高频系数 (AC 系数), 保留低频信息 (DC 系数)。高频系数一般保存的是图像的边界、纹理信息, 低频信息主要是保存的图像中平坦区域信息。

4) 图像的低频和高频, 高频区域指的是空域图像中突变程度大的区域 (比如目标边界区域), 通常的纹理丰富区域。

## 二、图像

二维 DCT 变换就是将二维图像从空间域转换到频率域。形象的说, 就是计算出图像由哪些二维余弦波构成

$$F=A*f*A^T$$
$$A(i,j)=c(i)\cos[(j+0.5)\pi i/N]$$

其中  $F$  就是变换得到的系数,  $f$  是图像的像素值,  $A$  是转换矩阵, 其中  $i$  为二维波的水平方向频率,  $j$  为二维波的垂直方向频率, 取值范围都是  $0-(N-1)$ ,  $N$  是图像块的大小,

$$c(i)=\{\sqrt{1/N}, i=0; \sqrt{2/N}, i\neq 0\}$$

1) 求出转换矩阵  $A$ ;

2) 利用转换矩阵  $A$ , 转换到频域, 即由图像  $f$  得到系数矩阵  $F$ 。

原文: <https://blog.csdn.net/jizhidexiaoming/article/details/80826915>