**Patchwork水印算法**

1. **嵌入算法**

随机选择N对像素点 (ai，bi) ，实验中按奇偶分类，然后将每个ai点的亮度值加 d ，每个bi点的亮度值减 d，这样整个图像的平均亮度保持不变。

1. **检测算法**

实验的Patchwork水印检测算法中，主要的判定式是：

其中，指对应像素点的亮度；M，N是图像的行列，Q为检测度。

1. **程序实现**

嵌入程序：

% input image

img\_rgb = imread('C:\Users\helin\Desktop\Fig0620(a).tif');

% convert to yuv color space

img\_yuv = rgb2ycbcr(img\_rgb);

% get the bright channal and its size and total elements

img\_y = img\_yuv(:,:,1);

[row, col] = size(img\_y);

total = row \* col;

% generate two set of coordinate

ai = 1:2:total;

bi = 2:2:total;

% modify the bright

degree = 1;

img\_y\_m = img\_y;

img\_y\_m(ai) = img\_y(ai) + degree;

img\_y\_m(bi) = img\_y(bi) - degree;

% original image

subplot(1,2,1);

imshow(img\_rgb);

title('original image');

% embeded image

subplot(1,2,2);

img\_yuv(:,:,1) = img\_y\_m;

img\_rgb = ycbcr2rgb(img\_yuv);

imshow(img\_rgb);

title('embeded image');

imwrite(img\_rgb,'embeded image.tif');

检测程序：

% input image

img\_rgb = imread('embeded image.tif');

% convert to yuv color space

img\_yuv = rgb2ycbcr(img\_rgb);

% get the bright channal and its size and total elements

img\_y = img\_yuv(:,:,1);

[row, col] = size(img\_y);

total = row \* col;

% generate two set of coordinate

ai = 1:2:total;

bi = 2:2:total;

% algin the dimension

if(length(ai) > length(bi))

ai(end) = [];

elseif(length(ai) < length(bi))

bi(end) = [];

end

% detect value

flag = sum( img\_y(ai) - img\_y(bi) ) / total;

% output

detect\_degree = 0.4;

if(flag > detect\_degree)

disp 'YES have watermark';

else

disp 'NO watermark';

end

1. **实验结果**

嵌入结果：





在亮度变化d较小时，图像肉眼感觉无变化。当d > 5时，变化肉眼可感知。

检测结果正确