Digital Image Processing HW2

電機四乙10828241 陳大荃

October 25, 2022

1. Problem Description

Use MATLAB to process/halftoning a grayscale image by a 2x2 dither matrix.

2. Program/Code

This is a part of the function I coded for this question.
The full function is available on GitHub: `https://github.com/belongtothenight/DIP_Code/blob/main/src/draw_halftoning_gimg_22.m` (click to open)

```
function img = md3(img, d22, dh, dw, ih, iw);
    % This function process the square of 2x2 pixels at a time.
    hl = round(ih / dh);
    wl = round(iw / dw);
    for i = 1:hl
        for j = 1:wl
            x1 = i*2-1;
            x2 = i*2;
            y1 = j*2-1;
            y2 = j*2;
            try
               img(x1:x2, y1:y2) = img(x1:x2, y1:y2) > d22;
            catch
                try
                    if img(i*2-1,j*2-1) > d22(1,1)
                        img(i*2-1,j*2-1) = 255;
                    else
                        img(i*2-1,j*2-1) = 0;
                    end
                end
                try
                    if img(i*2-1,j*2) > d22(1,2)
                        img(i*2-1,j*2) = 255;
                    else
                        img(i*2-1,j*2) = 0;
                    end
                end
                try
                    if img(i*2,j*2-1) > d22(2,1)
                        img(i*2,j*2-1) = 255;
                    else
                        img(i*2,j*2-1) = 0;
                    end
                end
                try
                    if img(i*2,j*2) > d22(2,2)
                        img(i*2,j*2) = 255;
                    else
                        img(i*2,j*2) = 0;
                    end
                end
            end
        end
    end
    img = img * 255;
```

```
end
```

This is the code in MATLAB to execute this function.

```
>> imgp = 'lena.tif';
>> fn = 'lena_hg.jpg';
>> imgsp = fn;
>> md = 3;
>> showimg = 1;
>> draw_halftoning_img_g22(imgp, fn, dm, imgsp, showimg)
```

Note that the picture to be processed needs to be stored in the same folder as where the function file is stored. If not, use the full path and store it in "imgp".
If you want to specify where the processed image is stored, use full path and store it in "imgsp".

3. Result and Discussion

3.1 Method

Since matrix calculation can do the same task with a lot less time, the best solution to quickly process the entire image is by expanding the original 2x2 dither matrix to the size of the original image. The processing time can be reduced by at least 6000 times.

The processing time and algorithm used are recorded in subfunctions in `https://github.com/belongtothenight/DIP_Code/blob/main/src/draw_halftoning_gimg_22.m` (click to open)

3.2 Result

(a) Original Image (b) Processed Image

Figure 1: Result of the required squares.

3.3 Discussion

In the following section, I used the generalized grayscale image processing function to try out dither matrixes of different sizes and used the generalized RGB image processing function to test how halftoning works on them.

Link to the generalized grayscale image algorithm: `https://github.com/belongtothenight/DIP_Code/blob/main/src/draw_halftoning_gimg.m`

Link to the generalized RGB image algorithm: `https://github.com/belongtothenight/DIP_Code/blob/main/src/draw_halftoning_cimg.m`

3.3.1 Dither Matrix of Different Sizes

(a) 2x2 (b) 4x4

(c) 8x8 (d) 16x16

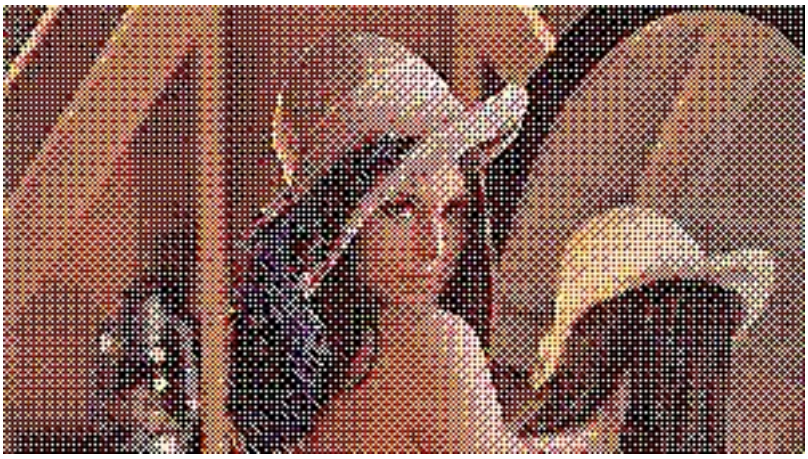
(e) 32x32

Figure 2: Dither matrix of different sizes.

As the size of the matrix grows, the obvious false contour affects more and more to the image, while darkening the image simultaneously.

3.3.2 Colored Image and Different Dither Matrix

(a) Original Image



(b) 2x2 Processed Image

(c) 4x4 Processed Image

Figure 3: Colored image and different dither matrix

My version of the algorithm works fine for a 2x2 dither matrix but starts to fail in the following circumstances.

$$mod(\text{image width}, \text{matrix width}) \neq mod(\text{image height}, \text{matrix height})$$

This happens in Figure 3(c)'s right-hand side, as you can see the same array of color is stretched till it reaches the original width.