# ECE637 Digital Image Processing I
# Laboratory work 8:
# Image Halftoning

Boris Diner

March 29, 2021

# 3 Thresholding

## 3.1 The original image and the result of thresholding

(a) The original image *house.tif*

(b) Threshold image for $T = 127$

Figure 1: The original image and the threshold image

## 3.2 The computed RMSE and fidelity values

$$\text{RMSE} = 87.3933$$
$$\text{Fidelity} = 77.3371$$

## 3.3 Matlab code for *fidelity* function

```
function fid = fidelity(f,b)
    % Un-gammacorrect f and b
    f = double(f);
    b = double(b);

    f_linear = 255*(f/255).^2.2;

    [M, N] = size(f_linear);

    % Low-pass filter
    sigma = 2; % variance
    num_pix = 7; % number of pixels used in the
       filter
    ind = -floor(num_pix/2):floor(num_pix/2);
    [I J] = meshgrid(ind, ind);
    h = exp(-(I.^2+J.^2)/(2*sigma));
    h = h / sum(h(:));

    % Convolution (applying the filter)
    f_conv = conv2(f_linear, h, 'same');
    b_conv = conv2(b, h, 'same');

    % Transformation for contrast sensitivity
    f_tilde = 255*(f_conv/255).^(1/3);
    b_tilde = 255*(b_conv/255).^(1/3);

```

```
26      fid = sqrt((sum(sum((f_tilde - b_tilde).^2)))/(M
            *N));
27  end
```

**Additional code**

```
1   f = imread('house.tif');
2   f = double(f);
3
4   [M, N] = size(f);
5
6   b = zeros(M,N); % binary image
7
8   T = 127; % threshold
9
10  for i=1:M
11      for j=1:N
12          if f(i,j) > T
13              b(i,j) = 255;
14          end
15      end
16  end
17
18  colormap(gray);
19  image(b);
20  truesize
21  imwrite(b, 'house_T_127.tif');
22
23  rmse = sqrt((sum(sum((f-b).^2)))/(M*N));
24  fid = fidelity(f,b);
```

# 4 Ordered Dithering

## 4.1 The three Bayer index matrices of sizes $2 \times 2$, $4 \times 4$, and $8 \times 8$

$$I_2 = \begin{bmatrix} 1 & 2 \\ 3 & 0 \end{bmatrix}$$

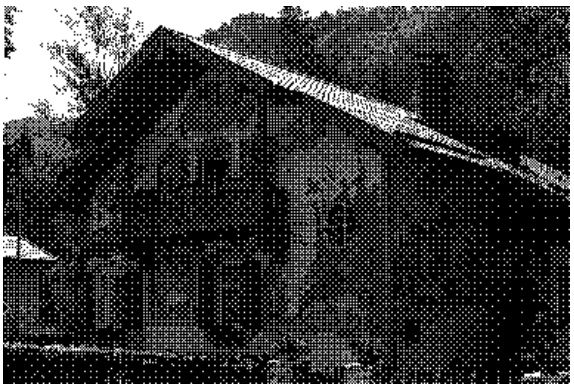
$$I_4 = \begin{bmatrix} 5 & 9 & 6 & 10 \\ 13 & 1 & 14 & 2 \\ 7 & 11 & 4 & 8 \\ 15 & 3 & 12 & 0 \end{bmatrix}$$

$$I_8 = \begin{bmatrix} 21 & 37 & 25 & 41 & 22 & 38 & 26 & 42 \\ 53 & 5 & 57 & 9 & 54 & 6 & 58 & 10 \\ 29 & 45 & 17 & 33 & 30 & 46 & 18 & 34 \\ 61 & 13 & 49 & 1 & 62 & 14 & 50 & 2 \\ 23 & 39 & 27 & 43 & 20 & 36 & 24 & 40 \\ 55 & 7 & 59 & 11 & 52 & 4 & 56 & 8 \\ 31 & 47 & 19 & 35 & 28 & 44 & 16 & 32 \\ 63 & 15 & 51 & 3 & 60 & 12 & 48 & 0 \end{bmatrix}$$

## 4.2 The three halftoned images produced by the three dither patterns

(a) The halftoned imaged produced by 2×2 dither pattern

(b) The halftoned imaged produced by 4×4 dither pattern



(c) The halftoned imaged produced by 8×8 dither pattern

Figure 2: The three halftoned images produced by the three dither pattern

### 4.3 The RMSE and fidelity for each of the three halftoned images

| Metrics / Bayer Matrix | RMSE | Fidelity |
|---|---|---|
| $2 \times 2$ | 97.6690 | 50.0569 |
| $4 \times 4$ | 101.0069 | 16.5583 |
| $8 \times 8$ | 100.9145 | 14.6918 |

## 5 Error Diffusion

### 5.1 The error diffusion Matlab code

```
function b = error_diffusion(f)

    f = double(f);
    f_lin = 255 * (f/255).^2.2;
    T = 127;
    [M, N] = size(f_lin);
    b = zeros(M,N);

    bord = 1;

    f_pad = zeros(M+2*bord,N+2*bord);
    for i=1+bord:M+bord
        for j=1+bord:N+bord
            f_pad(i,j)=f_lin(i-bord,j-bord);
        end
    end

    for i=1+bord:M+bord
        for j=1+bord:N+bord
```

```
                if f_pad(i,j) > 127
                    b(i-bord,j-bord) = 255;
                end

                e = f_pad(i,j) - b(i-bord,j-bord);

                f_pad(i+1,j-1) = f_pad(i+1,j-1) + e
                    *3/16;
                f_pad(i+1,j) = f_pad(i+1,j) + e*5/16;
                f_pad(i,j+1) = f_pad(i,j+1) + e*7/16;
                f_pad(i+1,j+1) = f_pad(i+1,j+1) + e
                    *1/16;

            end
        end
end
```

**Additional code**

```
f = imread('house.tif');
f = double(f);
[M,N] = size(f);

b = error_diffusion(f);

colormap(gray(256));
image(b);
truesize
imwrite(b,'bin_err_diffusion.tif')

rmse = sqrt((sum(sum((f-b).^2)))/(M*N));
fid = fidelity(f,b);
```

## 5.2 The error diffusion result

Figure 3: The Error Diffusion Result

## 5.3 The RMSE and fidelity of the error diffusion result

$$\text{RMSE} = 98.8471$$
$$\text{Fidelity} = 13.4273$$

## 5.4 The RMSE and fidelity for the simple thresholding, ordered dithering, and error diffusion results

| Halftoning \ Metrics | | RMSE | Fidelity |
|---|---|---|---|
| Thresholding | | 87.3933 | 77.3371 |
| Ordered Dithering | $2 \times 2$ | 97.6690 | 50.0569 |
| | $4 \times 4$ | 101.0069 | 16.5583 |
| | $8 \times 8$ | 100.9145 | 14.6918 |
| Error Diffusion | | 98.8471 | 13.4273 |

It can be concluded that RMSE metrics does not show significant difference between various halftoning techniques while fidelity does. However, the results obtained for fidelity metrics imply that thresholding produces the best match between the original image and the binary image while visually the binary image generated by means of the error diffusion technique matches the original image *house.tif* best.