5/19 画像信号処理特論

Schedule

- 5/12 “Hello World!” of image processing
- 5/19 Image filtering
- 5/26 Binarization
- 6/2 (Prof. Tehrani)
- 6/9 (Prof. Tehrani)
- 6/16 (Prof. Tehrani)
- 6/23 Histogram ← 1st report deadline
- 6/30 Discrete Cosine Transform
- 7/7 JPEG
- 7/14 (Prof. Fujii)
- 7/21 (Prof. Fujii) ← 2nd report deadline

myImageData class

- Get a pixel value

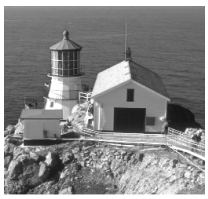
```
double v = img->get(x,y);     // for grey scale images
double v = img->get(x,y,1);  // for RGB color images
                    // last parameter = color channel
                    //  should be 0, 1, or 2
```

- Set a pixel value

```
img->set(x,y,value); // for grey scale images
img->set(x,y,2,value); // for RGB color images
```

Today’s Issue

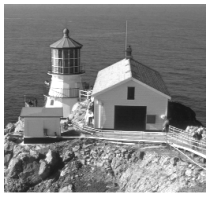
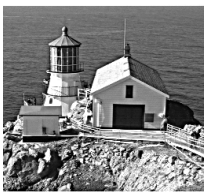
- Average filtering




Input Output

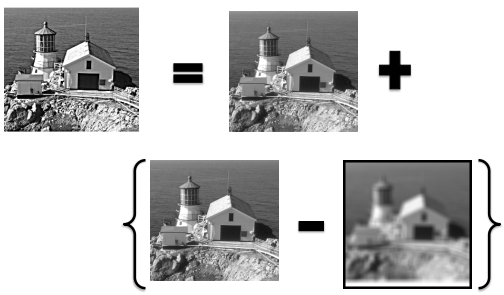
Today’s Issue

- Application of average filtering

Input Output

Detail Enhancement



Today's Issue

- Edge Detection

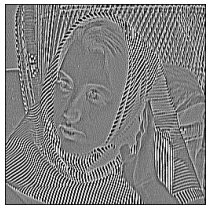
Input

Horizontal edge

Today's Issue

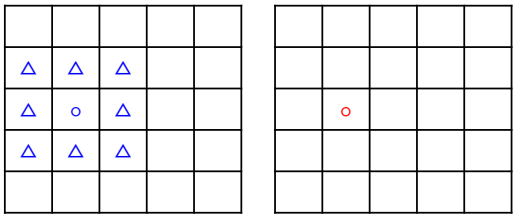
- Edge Detection

Vertical edge



Laplacian

Principle of Average Filtering



Input Output

Principle of Average Filtering

$$g(x,y) = \frac{1}{9}\sum_{i=-1}^{1}\sum_{j=-1}^{1} f(x+i, y+j)$$

Input Output

Implementation of image filtering

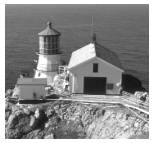
```
Loop for y (H times)
    Loop for x (W times)
        Value = 0
        Loop for i (M times)
            Loop for j (M times)
                Value += f(x+i, y+j)
        g(x,y) = Value
```

Take care when processing image boundaries

Representation of general filters

Filtering can be written as convolution

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9



$$g(x,y) = h(x,y) * f(x,y)$$

Representation of general filters

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Mean

-1	0	1
-2	0	2
-1	0	1

Sovel

-1	-2	-1
0	0	0
1	2	1

Sovel

1	1	1
1	-8	1
1	1	1

Laplacian

$$g(x,y) = h(x,y) * f(x,y)$$

Any filter kernel is applicable in the same manner

Edge detection with filters

-1	0	1
-2	0	2
-1	0	1

Sovel

$$h * \approx \frac{\partial}{\partial x}$$

-1	-2	-1
0	0	0
1	2	1

Sovel

$$h * \approx \frac{\partial}{\partial y}$$

1	1	1
1	-8	1
1	1	1

0	1	0
1	-4	1
0	1	0

Laplacian

$$h * \approx \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

Exercises

- Build and execute “sample2”
- Implement image filters
  - Average, horizontal/vertical edge detection, Laplacian
- Implement detail enhancement
  - (original) + (original) – (averaged)
  - (original) + (edge)