
Computer Vision HW5

National Taiwan University

r02944002 - 王瀚宇



參數設定

```
int headerLength = 172;  
int imageWidth = 512;  
int imageHeight = 512;  
  
Kernel octagonKernel = new Kernel(new int[][]{  
    { 0,1,1,1,0},  
    { 1,1,1,1,1},  
    { 1,1,1,1,1},  
    { 1,1,1,1,1},  
    { 0,1,1,1,0}  
},2,2);
```

原圖(lena.im) :



Dilation

Concept Code

```
public static ArrayList<Integer> Dilation(ArrayList<Integer> origin,int
headerLength, int width, int height,Kernel kernel)
{ArrayList<Integer> results =
InitWhite(origin,headerLength,width,height);
    for(int y = 0 ; y < height ; y++)
    {for(int x = 0 ; x<width ; x++)
        {int index = headerLength+width*y+x;
        int max = 0;
        for(int y2 = 0 ; y2 < kernel.GetHeight() ; y2++)
        {for( int x2 = 0 ; x2 < kernel.GetWidth() ; x2++)
            {int localX = x2 - kernel.OriginX;
            int localY = y2 - kernel.OriginY;
            int globalX = x + localX;
            int globalY = y + localY;
            if(globalX<0)continue;
            if(globalX>=width)continue;
            if(globalY<0)continue;
            if(globalY>=height)continue;
            int globalIndex = headerLength + globalY*height + globalX;
            if(kernel.Data[y2][x2]==1)
            {if(max < origin.get(globalIndex))
                {max = origin.get(globalIndex);}}}
            results.set(index, max);}}}
    return results;}
```

Result



Erosion

Concept Code

```
public static ArrayList<Integer> Erosion(ArrayList<Integer> origin,int
headerLength, int width, int height,Kernel kernel)
{ArrayList<Integer> results =
InitWhite(origin,headerLength,width,height);
    for(int y = 0 ; y < height - (kernel.GetHeight()-1) ; y++)
    {for(int x = 0 ; x<width - (kernel.GetWidth()-1); x++)
        {boolean validate = true;
        int min = 255;
            for(int y2 = 0 ; y2 < kernel.GetHeight() ; y2 ++){
            for(int x2 =0 ; x2 < kernel.GetWidth() ;x2 ++){
                {if(kernel.Data[y2][x2]==1)
                    {int globalX = x + x2;
                    int globalY = y + y2;
                    int globalIndex = headerLength+globalY * width + globalX;
                    if(min>origin.get(globalIndex))min = origin.get(globalIndex);
                }}}
            int globalX = x + kernel.OriginX;
            int globalY = y + kernel.OriginY;
            int globalIndex = headerLength+globalY * width + globalX;
            results.set(globalIndex, min);}}
        return results;}
```

Result



Opening

Concept Code

```
ArrayList<Integer> erosion =  
Erosion(bytes,headerLength,imageWidth,imageHeight,octogonKernel);  
ArrayList<Integer> opening =  
Dilation(erosion,headerLength,imageWidth,imageHeight,octogonKernel);
```

Result



Closing

Concept Code

```
ArrayList<Integer> dialation =  
Dilation(bytes,headerLength,imageWidth,imageHeight,octogonKernel);  
ArrayList<Integer> closing =  
Erosion(dialation,headerLength,imageWidth,imageHeight,octogonKernel);
```

Result

