



# 计算机视觉

彭盛霖

西北大学信息学院

[pengshenglin@nwu.edu.cn](mailto:pengshenglin@nwu.edu.cn)

去噪去雾



## 2 去噪去雾-滤波

滤波：模板不含负数

```
1 import cv2 as cv
2 import numpy as np
3 kernel= np.array([
4     [3,2,3],
5     [2,1,2],
6     [3,2,3]])/21.#定义滤波核
7 im = cv.imread("eye.jpg")#读入图片
8 out = cv.filter2D(im, -1, kernel)
9 cv.imshow("origin", im) #显示原始图
10 cv.imshow("out", out) #显示滤波结果
11 cv.waitKey(0)
12 cv.destroyAllWindows()
```

**cv2.filter2D**



CV2





## 2 去噪去雾-滤波

滤波：模板含负数

```
1 import cv2 as cv
2 import numpy as np
3 kernel= np.array([
4     [-2,-1,0],
5     [-1,1,1],
6     [0,1,2]])#定义滤波核
7 im = cv.imread("eye.jpg")#读入图片
8 out = cv.filter2D(im, cv.CV_32F, kernel)
9 out = cv.convertScaleAbs(out)#转为0~255
10 cv.imshow("origin", im) #显示原始图
11 cv.imshow("out", out) #显示滤波结果
12 cv.waitKey(0)
13 cv.destroyAllWindows()
```

**cv2.filter2D**





## 2 去噪去雾-滤波

### 滤波：分通道再合成

```
1  import cv2 as cv
2  import numpy as np
3  kernel= np.array([[5,2,5],[2,0,2],[5,2,5]])/28. #定义滤波核
4  im = cv.imread("eye.jpg") #读入图片
5  b,g,r = cv.split(im) #通道分离
6  b = cv.filter2D(b, -1, kernel)
7  g = cv.filter2D(g, -1, kernel)
8  r = cv.filter2D(r, -1, kernel)
9  out=cv.merge([b,g,r]) #通道合并
10 cv.imshow("origin", im) #显示原始图
11 cv.imshow("out", out) #显示滤波结果
12 cv.waitKey(0)
13 cv.destroyAllWindows()
```

CV2

**cv2.filter2D**



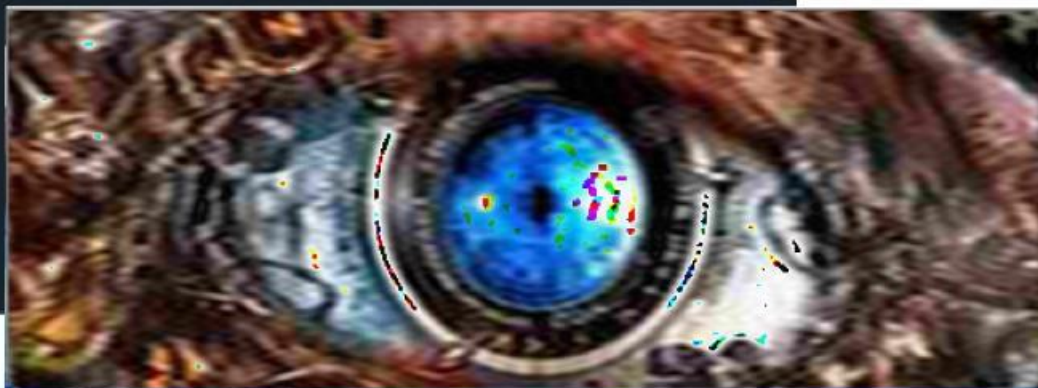
## 2 去噪去雾-滤波

### 滤波：频域滤波

CV2

np.fft

```
1 import cv2 as cv
2 import numpy as np
3 def cutBy_fft(im,k=0.5):
4     fft2=np.fft.fft2(im)
5     fft2=np.fft.fftshift(fft2)
6     n,m=fft2.shape
7     zeros=np.zeros((n,m),dtype=complex)
8     zeros[int((n-k*n)/2):int((n+k*n)/2),
9           int((m-k*n)/2):int((m+k*n)/2)]=fft2[int((n-k*n)/2):int((n+k*n)/2),
10                                                int((m-k*n)/2):int((m+k*n)/2)]
11     fft2=np.fft.ifftshift(zeros)
12     fft2=np.fft.ifft2(fft2)
13     return cv.convertScaleAbs(np.uint8(np.abs(fft2)))
14 im = cv.imread("eye.jpg")#读入图片
15 b,g,r = cv.split(im) #通道分离
16 b = cutBy_fft(b, 0.5)
17 g = cutBy_fft(g, 0.5)
18 r = cutBy_fft(r, 0.5)
19 out=cv.merge([b,g,r]) #通道合并
20 cv.imshow("origin", im) #显示原始图
21 cv.imshow("out", out) #显示滤波结果
22 cv.waitKey(0)
23 cv.destroyAllWindows()
```





### 常用形态学处理算法

- `cv2.MORPH_ERODE` 腐蚀
- `cv2.MORPH_DILATE` 膨胀
- `cv2.MORPH_OPEN` 开运算
- `cv2.MORPH_CLOSE` 闭运算
- `cv2.MORPH_GRADIENT` 形态学梯度
- `cv2.MORPH_TOPHAT` 礼帽运算
- `cv2.MORPH_BLACKHAT` 黑帽运算
- `cv2.MORPH_HITMISS` 击中不击中

### 常用形态学结构元素

- 0: MORPH\_RECT 矩形
- 1: MORPH\_CROSS 十字交叉形
- 2: MORPH\_ELLIPSE 椭圆



## 2 去噪去雾-形态学运算

### 形态学处理

CV2

```
1 import cv2
2 import numpy as np
3 kernel=cv2.getStructuringElement(cv2.MORPH_ELLIPSE,(5,5))#也可定义滤波核
4 im = cv2.imread("eye.jpg")#读入图片
5 b,g,r = cv2.split(im) #通道分离
6 b = cv2.morphologyEx(b, cv2.MORPH_HITMISS, kernel)
7 g = cv2.morphologyEx(g, cv2.MORPH_HITMISS, kernel)
8 r = cv2.morphologyEx(r, cv2.MORPH_HITMISS, kernel)
9 out=cv2.merge([b,g,r]) #通道合并
10 cv2.imshow("origin", im) #显示原始图
11 cv2.imshow("out", out) #显示滤波结果
12 cv2.waitKey(0)
13 cv2.destroyAllWindows()
```

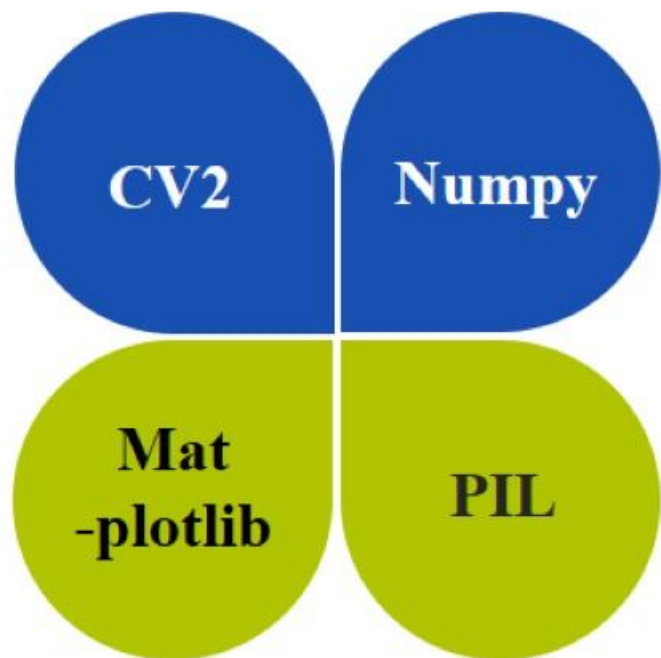
**cv2.morphologyEx**



## 2 去噪去雾-实战部分

### 去噪去雾

#### ◆ 思考&实践



➤ 善用百度、谷歌

- 自定义滤波模板、形态学元素结构能达到哪些效果
- 怎样提升去噪效果
- 怎样改进去雾效果



## 2 去噪去雾-实战部分

### ◆ 接下来的时间：上机实验并完成实验报告

#### 《计算机视觉》实验报告

#### · 实验 02：去噪去雾：滤波与形态学处理

姓名		学号	
实验地点		实验日期	

##### 一、实验内容

【1】熟悉基本的滤波与形态学处理技术，任选图片，通过 PIL 或 cv2 或自己编写代码，实现 3-5 种滤波操作、3-5 种形态学操作。

【2】任选带噪点的图片，通过 3-5 种方法去噪并简单比较分析其效果。

【3】任选有雾气的图片，通过 1-5 种方法去雾。